Computational and Statistical Advances in Testing and Learning

Aaditya Ramdas

July 2015
CMU-ML-15-101

Machine Learning Department
School of Computer Science
& Department of Statistics
Dietrich College of Humanities and Social Sciences
Carnegie Mellon University
Pittsburgh, PA, USA

Thesis Committee:
Larry Wasserman, Co-Chair
Aarti Singh, Co-chair
Ryan Tibshirani
Michael Jordan
Arthur Gretton

Submitted in fulfillment of the requirements
for the degree of joint Doctor of Philosophy
in Statistics and Machine Learning.

This research was sponsored by: National Science Foundation grant numbers IIS0644225 and IIS1247658; National Institutes of Health grant number R01GM085022; Office of Naval Research grant number N000140911052; and the Pittsburgh Life Sciences Greenhouse.

Keywords: active learning, convex optimization, hypothesis testing, computation-statistics tradeoffs, sequential analysis, stochastic oracles, randomized algorithms.

To my pampering parents, C.P. and Nalini.

Abstract

This thesis makes fundamental computational and statistical advances in testing and estimation, making critical progress in theory and application of classical statistical methods like classification, regression and hypothesis testing, and understanding the relationships between them. Our work connects multiple fields in often counter-intuitive and surprising ways, leading to new theory, new algorithms, and new insights, and ultimately to a cross-fertilization of varied fields like optimization, statistics and machine learning.

The first of three thrusts has to do with active learning, a form of sequential learning from feedback-driven queries that often has a provable statistical advantage over passive learning. We unify concepts from two seemingly different areas — active learning and stochastic first-order optimization. We use this unified view to develop new lower bounds for stochastic optimization using tools from active learning and new algorithms for active learning using ideas from optimization. We also study the effect of feature noise, or errors-in-variables, on the ability to actively learn.

The second thrust deals with the development and analysis of new convex optimization algorithms for classification and regression problems. We provide geometrical and convex analytical insights into the role of the margin in margin-based classification, and develop new greedy primal-dual algorithms for non-linear classification. We also develop a unified proof for convergence rates of randomized algorithms for the ordinary least squares and ridge regression problems in a variety of settings, with the purpose of investigating which algorithm should be utilized in different settings. Lastly, we develop fast state-of-the-art numerically stable algorithms for an important univariate regression problem called trend filtering with a wide variety of practical extensions.

The last thrust involves a series of practical and theoretical advances in nonparametric hypothesis testing. We show that a smoothed Wasserstein distance allows us to connect many vast families of univariate and multivariate two sample tests. We clearly demonstrate the decreasing power of the families of kernel-based and distance-based two-sample tests and independence tests with increasing dimensionality, challenging existing folklore that they work well in high dimensions. Surprisingly, we show that these tests are automatically adaptive to simple alternatives and achieve the same power as other direct tests for detecting mean differences. We discover a computation-statistics tradeoff, where computationally more expensive two-sample tests have a provable statistical advantage over cheaper tests. We also demonstrate the practical advantage of using Stein shrinkage for kernel independence testing at small sample sizes. Lastly, we develop a novel algorithmic scheme for performing sequential multivariate nonparametric hypothesis testing using the martingale law of the iterated logarithm to near-optimally control both type-1 and type-2 errors.

One perspective connecting everything in this thesis involves the closely related and fundamental problems of linear regression and classification. Every contribution in this thesis, from active learning to optimization algorithms, to the role of the margin, to nonparametric testing fits in this picture. An underlying theme that repeats itself in this thesis, is the computational and/or statistical advantages of sequential schemes with feedback. This arises in our work through comparing active with passive learning, through iterative algorithms for solving linear systems instead of direct matrix inversions, and through comparing the power of sequential and batch hypothesis tests.

Acknowledgments

The work presented in this document would simply not have been possible without the help of my excellent advisors Aarti Singh and Larry Wasserman; I learnt a lot from both of them and they gave me the freedom to pursue my varied interests with them and many other collaborators. Their extremely positive attitude towards problem solving has had an effect on me, and their lack of frustration over years of research has kept me inspired and thinking in a constructive manner. The friendly smiles that greet progress and roadblocks alike and the jokes that kept every meeting alive, shall be fondly remembered and replicated in my future career.

I have had the extreme fortune of collaborating with a huge variety of other brilliant professors, each with varying advising and collaboration styles. My very respectful list of directly interacting coauthors includes Ryan Tibshirani, Javier Peña, Barnabás Póczos, Cosma Shalizi, Deanna Needell, Marco Cuturi, Arthur Gretton, Rebecca Steorts and Dino Sejdinovic. They have been both and inspiring to work with!

Other professors whom I have had insightful discussions with include Avrim Blum, Geoff Gordon, Gary Miller, Rob Kass, Alessandro Rinaldo, Dave Choi, Nina Balcan, Dejan Slepcev, Suvrit Sra, Fatma Kilinc-Karzan, Alex Smola and Tom Mitchell. I am extremely thankful for their time and willingness to share their wisdom with me. A huge thanks also goes out to the many staff members who helped me out a huge deal, but the awesome Diane Stidle was usually my first and last stop for everything!

At the same time, I was lucky enough to be in the humbling company of many other students who were smarter than me. The extremely fun list of my student coauthors includes Leila Wehbe, Sashank Reddi, Ahmed Hefny, Nicolas Garcia, Akshay Balsubramani, Anna Ma and Kacper Chwialkowski.

Other students whom I have had enlightening discussions with include the Statistical ML theory group members Sivaraman Balakrishnan, James Sharpnack, Mladen Kolar, Martin Azizyan, Min Xu, Akshay Krishnamurthy, Gautam Dasarathy, Jisu Kim, Will Bishop, Yining Wang and Kirthevasan Kandasamy. I learnt a lot through the wonderful talks and discussions we had over the years.

I was also fortunate to attend the Select Lab reading group, where I learnt about rather different topics from Shiva Kaul, Erik Zawadzki, Carlton Downey, Adona Iosif, and others. I had many interesting discussions with my officemates Sashank Reddi, Avinava Dubey, Yifei Ma, Martin Azizyan, Ahmed Hefny and Tzu-Kuo Huang. They must definitely be thanked both for asking great questions and for answering many of mine!

A subset of MLD friends I learnt a lot from include Yisong Yue, Seth Flaxman, Yuxiang Wang, Yaoliang Yu, Ina Fiterau, Nicole Rafidi, Junier Oliva, Anthony Platanios, Ankur Parikh, Jing Xiang, Mrinmaya Sachan, Gus Xia, Saket Navlakha, Xi Chen, Willie Neiswanger and Partha Talukdar. I couldn't have asked for a smarter bunch to be around, and my chats with them over the years have surely been beneficial!

Some non-MLD friends who shared their expertise with me include Srivatsan Narayanan, John Wright, Jeremiah Blocki, Nisarg Shah, Jamie Morgenstern, Yair Movshovitz-Attias, Abhimanu Kumar, Adams Wei Yu, Negar Soheili, Satyajith Amaran, Vinay Prabhu and Sriram Somanchi. They come from a variety of departments, and enabled me to learn about a wide variety of interesting problems.

I wish to thank all my friends, my roommates over the years, Subhodeep Moitra, Saurabh Basu, Saurabh Shintre and Tanmay Mudholkar, for all the help and support they gave me

from the moment I stepped into Pittsburgh, to the moment I will leave.

Other people I must thank are my running buddies Rob Hall, Min Xu, Saurabh Basu, Leila Wehbe, biking buddies Po Ling Loh, Min Xu, Dar and Dan Oshop, Mary and Mike Rago, hiking buddies Benjamin Cowley, Siddharth Gopal, Angie Jasper and Amanda Clark, climbing buddies Bill McDowell, Dallas Card, Leila Wehbe, Pete Lund, Benjamin Cowley, Kartik Goyal, Kevin Waugh, Jayant Krishnamurthy.

A special thanks goes out to people who helped me in the weeks after my crazy car/bike accident - many of my friends like Leila Wehbe, Amanda Cowley, Benjamin Cowley, Will Bishop, Sudarshan Wadkar, and Angie Jasper spent days at my place and also helped take me to many of my doctor appointments!

I will always cherish the great times I had with my CSD buddies Dana and Yair Movshovitz-Attias, John Wright, Diana and Joao Martins, David Henriques, Sarah and Jeremy Karnowski, Kate Taralova, Anvesh Komuravelli, Catherine and Gabe Weisz, and many others — the boardgames, the rock band, the dinners, all of it was very memorable during those happy carefree times.

A lot of credit goes to my dearest Indian friends all over the world, ones I have known from my times in Muscat, Chennai, IIT Bombay, CMU, and elsewhere, for adding spice to my life, and making sure I had enough Indian food and culture around at all times to make me not too far from home.

There are many other people who have graciously opened their doors for my random visits through the years — a large but probably incomplete list of friends I have visited includes Harshad Khadilkar, Harsh Shah, Neha Rambhia, Swarna Ramineni, Anup Rao, Akshay Balsubramani, Manish Shetty, Sangram Raje, Aditi Dalmia, Abhay Anand, Yash Puranik, Bruno Vavala, Varun Jog, Aditya Gudipati, Himanshu Asnani, Sarah and Jeremy Karnowski, Ina Fiterau, Leila Wehbe, Min Xu, Paidimarri Arun, Ameya Muley, Palash Kusumwal, Viraj Kamat, Ankit Goyal, Parakram Mazumdar, Zubin Mehta, Navneet Agrawal, Amod Jog, Divya Surana, Annelot Krediet and Pranav Kashyap.

I also wish to thank the several other wonderful people with whom I have had intellectual or personal conversations during the last five years — the list is certainly too long and I risk missing many names, but if you are reading this, you are probably one of them!

Last, but certainly not the least, I'd like to thank my family — my parents, brother, cousins, aunts and uncles — who have supported me and pampered me through this long and arduous endeavor. All of their homes have always been welcome for me to visit, and they have never stopped short of helping me in any way I ever needed. I am very lucky to have ended up with such good blood!

Contents

List of Figures

List of Tables

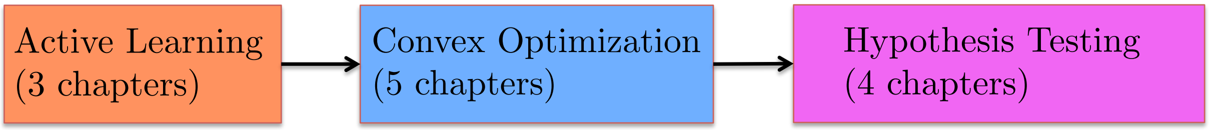
Chapter 1

Introduction

The past decade has seen an explosion of data types, models and algorithms, opening up new fields of research within machine learning and statistics. However, much is still left to be said about even the most classical settings in statistical learning like binary classification, linear regression and hypothesis testing, which one might imagine have been extremely well studied over the past century. In this thesis we will develop an understanding of the power that modern ideas can bring to such classical problems. In the process, we will draw connections between several fields, providing insights in all of them. The broad areas we will explore in this thesis are

1. Active learning
2. Convex optimization
3. Nonparametric hypothesis testing

These topics span the disciplines of Statistics, Machine Learning (ML) and Operations Research (OR). The following figure outlines the flow of chapters in this thesis, and we expand on this outline in the second half of the introduction section.



We will begin with 3 chapters on active learning, connecting it to stochastic convex optimization, and proving lower and upper bounds for both fields, and also examine the role of feature noise in active learning. Then, we move on to 5 chapters on optimization covering primal-dual algorithms for margin-based classification, randomized algorithms for least squares and ridge regression, and finally fast and flexible algorithms for trend filtering. We end with 4 chapters covering a suite of contributions in nonparametric testing, from a proof of adaptivity of kernel tests to simple mean-difference alternatives, to computation-statistics tradeoffs in testing, to sequential algorithms for testing in streaming settings.

There are different ways to draw connections between these chapters, some formal and some informal. In the following section, we intuitively describe some of the underlying these in this thesis.

1.1 The Intersections of Statistics, ML and OR

For the sake of context and familiarity, let us first briefly and informally introduce each of the aforementioned areas before summarizing our contributions, and outlining the rest of this document.

1.1.1 From Passive to Active Learning

Most machine learning and statistics courses often initially introduce learning as being completely passive, in the sense that one receives independent and identically distributed (iid) data from some source, using which the statistician or learner needs to make inferences about the underlying data distribution. In active learning, the learner gets to participate in the process of procuring the data, in such a manner that he usually minimizes the amount of data he/she needs to see in order to make certain inferences. In this feedback-driven situation, the data is no longer iid, and the inherently sequential nature of the problem involves intelligently using past data to procure future data. This often makes active/adaptive learning algorithms more sample-efficient than passive algorithms, but also sometimes more challenging to analyze.

1.1.2 The Role of Stochastic Convex Optimization

The process of making inferences from data can frequently be posed as an optimization problem, like finding parameters that maximize the likelihood of seeing the data, or minimizing some measure of discrepancy between predictions and observations. For computational tractability and other reasons, we often prefer to model our inference problems using the framework of convex optimization. Even so, in the era of big data, iteratively solving such convex problems is computationally demanding when done naively, going through gigabytes or more of data for each step. This is where stochastic methods vastly help — instead of processing the whole dataset at each step, we often process just a random subset of the data, sometimes even a single point, making each step very cheap. However, the randomization along with the sequential nature of these algorithms can sometimes lead to some tricky analysis.

To make things more concrete, two very common settings involving convex optimization for passive learning are linear regression and binary linear classification. One very interesting way of looking at linear classification is that it attempts to (approximately) solve a system of linear *inequalities*. On the other hand, linear regression can be viewed as attempting to solve a system of linear *equations*. This similarity/difference makes it interesting to compare incremental algorithms for the two problems. These are often stochastic/randomized or greedy algorithms, and their convergence rates depend on "similar but different" quantities that determine the computational and statistical difficulties of these related problems.

1.1.3 Two-Sample and Independence Testing

While linear regression and classification can be seen as *estimation* problems (estimating the best hyperplane fit, or the separating hyperplane), they have a related testing counterparts. Regression is related to independence testing, which instead of estimating a relationship between two random variables, just asks if the they are independent or not (i.e. whether the underlying joint distribution is a product of marginals or not). Similarly, classification is related to two sample testing, which instead of separating two variables just asks if it is possible to differentiate the two samples (i.e. whether they have different underlying marginal distributions or not).

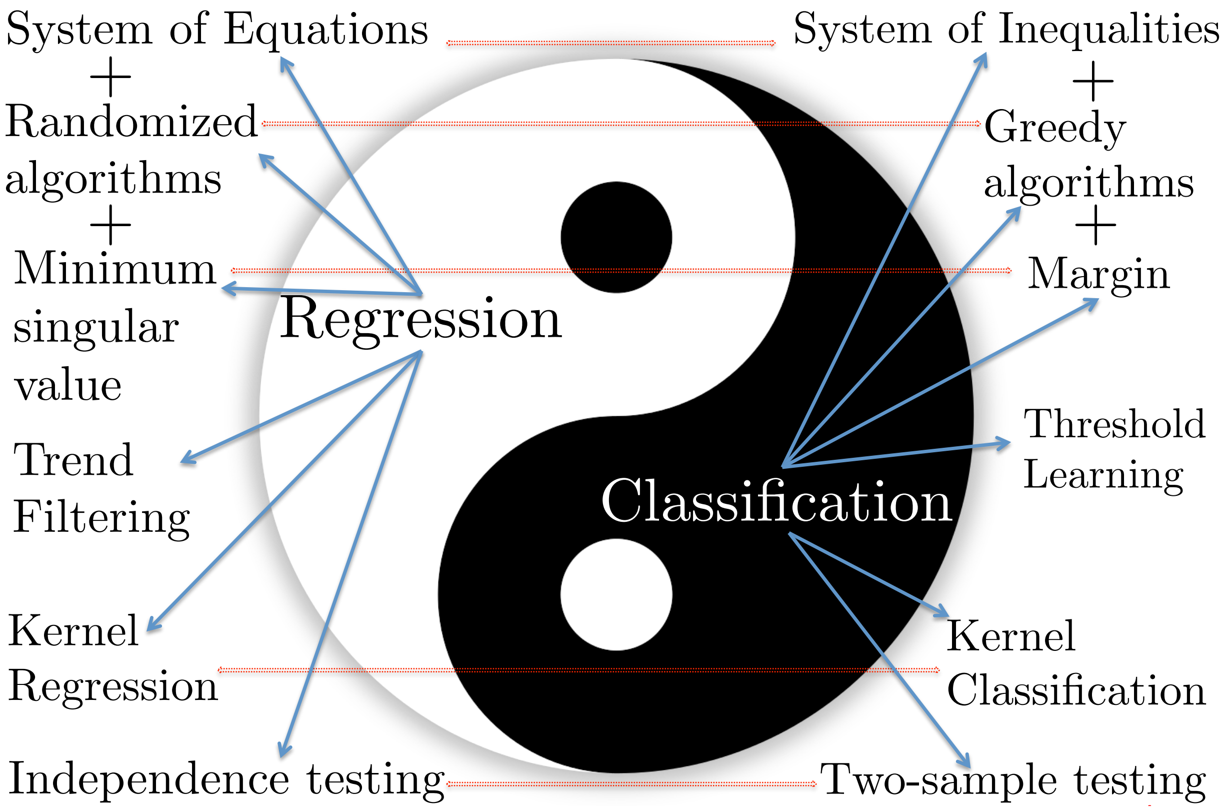


Figure 1.1: A unifying perspective on this thesis involves the closely related and fundamental problems of regression and classification. Every contribution in this thesis fits in this picture — from active learning (or trend filtering) to optimization algorithms for classification (or regression), to the role of the margin (or minimum singular value) in determining the computational difficulty of the classification (or regression) problems, to nonparametric two-sample (or independence) testing.

Specifically, we can view regression as solving a system of equations and classification as a system of inequalities, an interesting dual perspective that carries forth into the algorithms as well. We analyze randomized algorithms for regression, and greedy algorithms for classification. We show that while the minimum singular value determines the rate of convergence of the algorithms for regression, the "margin" determines the rate of convergence of our greedy classification algorithms, and interestingly we show that there is a tight relationship between these two quantities as expressed by *radius* theorems.

We also explore algorithms for kernel regression and kernel classification. In our last chapter, we provide algorithms and theoretical guarantees for the independence testing and two-sample testing problems, which are the "testing" versions of the estimation problems of regression and classification respectively. Due to the strong connection between kernel tests for independence and two sample testing, there is also such a relationship between kernel regression and classification (which can be used to perform nonparametric hypothesis testing, albeit indirectly).

Lastly, one can view trend filtering and threshold learning as regression and classification versions of changepoint detection problems. Together, this figure encompasses one way that the author views the various contributions in this thesis.

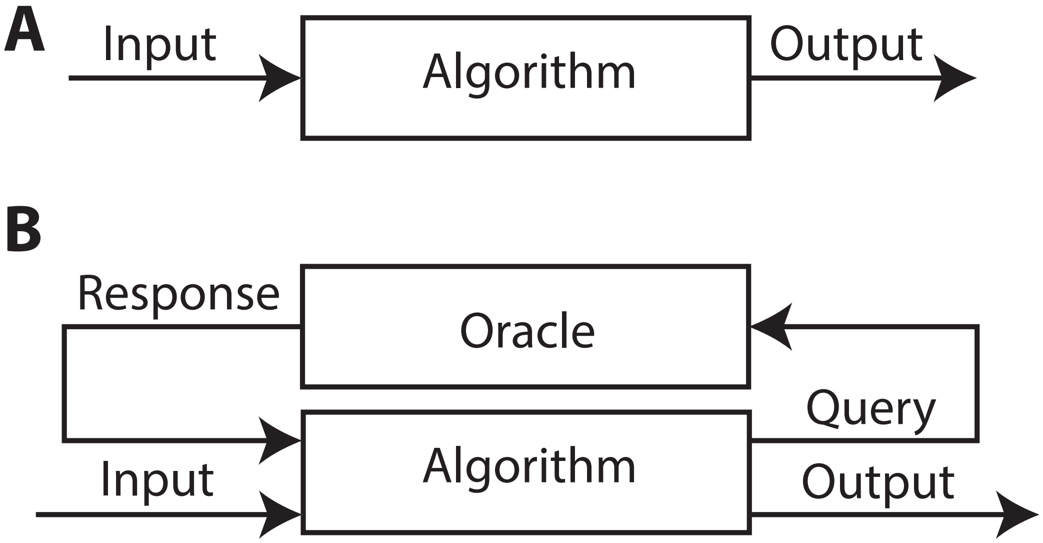
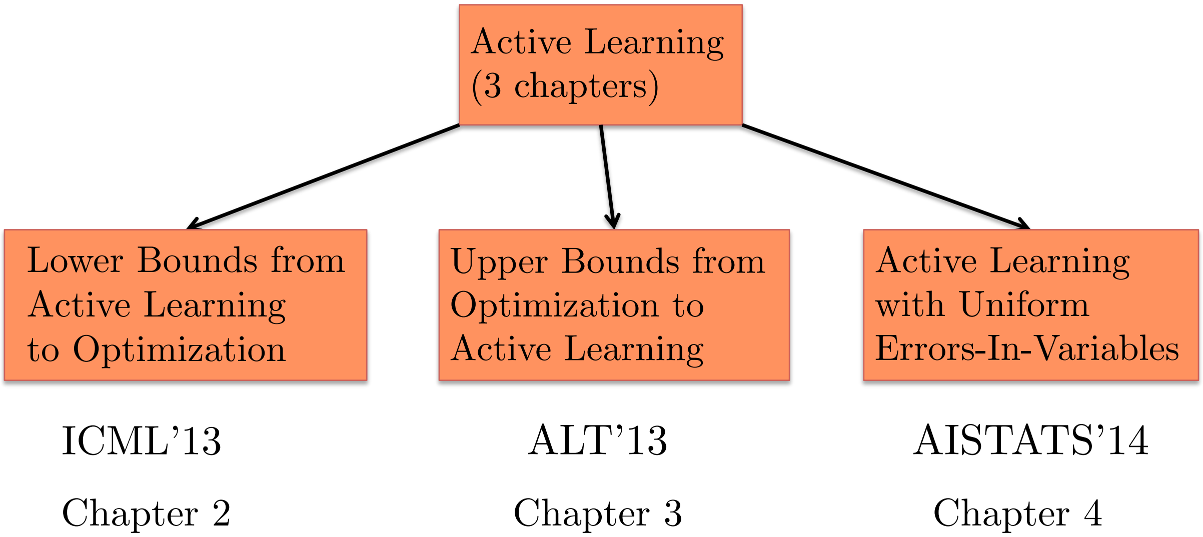


Figure 1.2: Another underlying theme of this thesis is that of sequential schemes with feedback (B), and their computational and/or statistical advantages over batch learning (A). This arises in our work through the statistical advantages of active over passive learning, through computational gains provided by iterative algorithms for solving linear systems (as opposed to direct matrix inversions), and through statistical and computational advantages of sequential over batch hypothesis tests. The "oracles" either provide labels, or gradients, or compare a test statistic to a varying threshold.

1.2 Summary of Contributions in Active learning


Active Learning (3 chapters)
Lower Bounds from Active Learning to Optimization
Upper Bounds from Optimization to Active Learning
Active Learning with Uniform Errors-In-Variables
ICML'13
ALT'13
AISTATS'14
Chapter 2
Chapter 3
Chapter 4

Active learning and stochastic convex optimization

In Chapters[1] 2 and 3, we show useful connections between active classification of thresholds and black-box stochastic convex optimization. One direct connection between the fields is that both rely on feedback-driven queries — in optimization, we use previous function and gradient values to obtain a new point at which we calculate the function and gradient value; in active learning, we use previous labels to obtain the new point at which we would like the next label — and the role of feedback, i.e. incorporating past information into future queries, is essential in both fields.

In Chapter 2, we focus on the problem of minimizing a convex function over a convex set given a budget of queries to a stochastic first order oracle. The main observation that we exploit is a previously unnoticed strong similarity between the role of the exponent (and constant) in Tsybakov's Noise/Margin Condition in active learning, and the role of the exponent (and constant) in uniform/strong convexity conditions for optimization. This results in the point errors in both settings (i.e. identifying the minimum, and identifying the Bayes optimal classifier) having the *exact same dependence* on number of queries (labels for active learning, gradients for optimization) and the exponents. It also results in the exact same dependence also holding not just for point errors, but also for function optimization error and excess classification risk. We use lower bound techniques from the active learning literature to get a new lower bounds in zeroth and first order stochastic optimization. We argue that the complexity of convex minimization is only determined by the rate of growth of the function around its minimizer, as quantified by a Tsybakov-like noise condition. Our results create strong formal connections between the two fields.

In Chapter 3, we continue this thread in two parts by exploiting these relations for the first time to yield novel algorithms in both fields, further motivating the study of their intersection. First, inspired by a recent optimization algorithm that was adaptive to unknown uniform convexity parameters, we present a new active learning algorithm for one-dimensional thresholds that can yield minimax rates by adapting to unknown noise parameters. Next, we show that one can perform d-dimensional stochastic minimization of smooth uniformly convex functions when only granted oracle access to noisy gradient signs along any coordinate instead of real-valued gradients, by using a simple randomized coordinate descent procedure where each line search can be solved by 1-dimensional active learning, provably achieving the same error convergence rate as having the entire real-valued gradient. Combining these two parts yields an algorithm that solves stochastic convex optimization of uniformly convex and smooth functions using only noisy gradient signs by repeatedly performing active learning, achieves optimal rates and is adaptive to all unknown convexity and smoothness parameters.

Active learning with uniform feature noise

In Chapter[2] 4, we consider the effect of feature noise in active learning, which could arise either because the feature itself is being measured, or is corrupted in transmission to the oracle/labeler, or the oracle/labeler returns the label of a noisy version of the query point. In statistics, feature noise is known as "errors in variables" and has been studied extensively in non-active settings. However, the effect of feature noise in active learning has not been studied before. We consider the well-known Berkson errors-in-variables model with additive uniform noise of width σ.

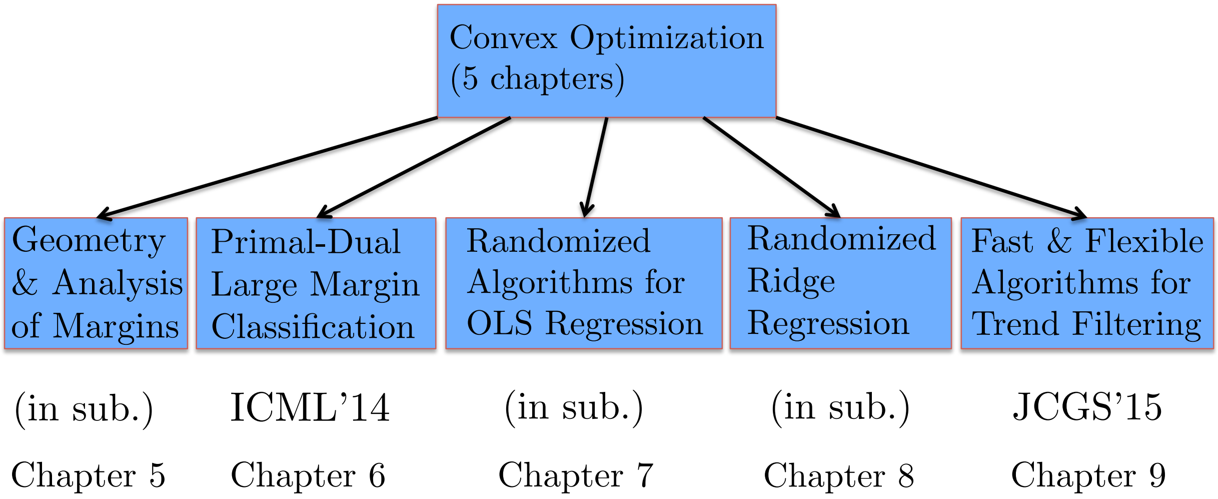
Our simple but revealing setting is that of one-dimensional binary classification setting where the goal is to learn a threshold (point where the probability of a $+$ label crosses half). We deal with regression functions that are antisymmetric in a region of size σ around the threshold and also satisfy Tsybakov's margin condition around the threshold. We prove minimax lower and upper bounds which demonstrate

[1] published in Ramdas and Singh [160] at ICML'13 and Ramdas and Singh [161] at ALT'13

[2] published in Ramdas et al. [164] at AISTATS'14

that when σ is smaller than the minimax active/passive noiseless error, then noise has no effect on the rates and one achieves the same noiseless rates. For larger σ, the *unflattening* of the regression function on convolution with uniform noise, along with its local antisymmetry around the threshold, together yield a behavior where noise *appears* to be beneficial. Our key result is that active learning can buy significant improvement over a passive strategy even in the presence of feature noise.

1.3 Summary of Contributions in Convex Optimization


Convex Optimization (5 chapters)
Geometry & Analysis of Margins
Primal-Dual Large Margin Classification
Randomized Algorithms for OLS Regression
Randomized Ridge Regression
Fast & Flexible Algorithms for Trend Filtering
(in sub.)
ICML'14
(in sub.)
(in sub.)
JCGS'15
Chapter 5
Chapter 6
Chapter 7
Chapter 8
Chapter 9

Linear classification: geometry and analysis of margins

Given a data matrix A, a linear feasibility problem (of which linear classification is a special case) aims to find a solution to a primal problem, "does there exist a vector making an acute angle with all data vectors?", or a certificate for the dual problem which is a probability distribution, "is there a convex combination of points that results in the origin?". Inspired by the continued importance of large-margin classifiers in machine learning, Chapter[3] 5 studies a condition measure of A called its *margin* that determines the difficulty of both the above problems. To aid geometrical intuition, we first establish new characterizations of the margin in terms of relevant balls, cones and hulls. Our second contribution is analytical, where we present generalizations of Gordan's theorem, and beautiful variants of Hoffman's theorems, both using margins. We end by proving some new results on a classical iterative scheme, the Perceptron, whose convergence rates famously depends on the margin. Our results are relevant for a deeper understanding of margin-based learning and proving convergence rates of iterative schemes, apart from providing a unifying perspective on this vast topic.

Linear classification: greedy primal-dual margin-based algorithms

In Chapter[4] 6, we focus on the problem of finding a non-linear classification function that lies in a Reproducing Kernel Hilbert Space (RKHS) both from the primal point of view (finding a perfect separator when

[3] in submission, ArXiv preprint Ramdas and Pena [159]

[4] published in Ramdas and Peña [158] at ICML'14

one exists) and the dual point of view (giving a certificate of non-existence), with special focus on generalizations of two classical schemes - the Perceptron (primal) and Von-Neumann (dual) algorithms. We cast our problem as one of maximizing the regularized normalized hard-margin (ρ) in an RKHS and rephrase it in terms of a Mahalanobis dot-product/semi-norm associated with the kernel's (normalized and signed) Gram matrix. We derive an accelerated smoothed algorithm with a convergence rate of $\frac{\sqrt{\log n}}{\rho}$ given n separable points, which is strikingly similar to the classical kernelized Perceptron algorithm whose rate is $\frac{1}{\rho^2}$. When no such classifier exists, we prove a version of Gordan's separation theorem for RKHSs, and give a reinterpretation of negative margins. This allows us to give guarantees for a primal-dual algorithm that halts in $\min\{\frac{\sqrt{n}}{|\rho|}, \frac{\sqrt{n}}{\epsilon}\}$ iterations with a perfect separator in the RKHS if the primal is feasible or a dual ϵ-certificate of near-infeasibility.

Linear regression: randomized algorithms for OLS and ridge regression

The Kaczmarz and Gauss-Seidel methods both solve a linear system of equations by iteratively refining the solution estimate. Recent interest in these methods has been sparked by a proof of Strohmer and Vershynin which shows the *randomized* Kaczmarz method converges linearly in expectation to the solution. Lewis and Leventhal then proved a similar result for the randomized Gauss-Seidel algorithm. However, the behavior of both methods depends heavily on whether the system is under or overdetermined, and whether it is consistent or not. In Chapter[5] 7, we provide a unified theory of both methods, their variants for these different settings, and draw connections between both approaches. In doing so, we also provide a proof that an extended version of randomized Gauss-Seidel converges linearly to the least norm solution in the underdetermined case (where the usual randomized Gauss Seidel fails to converge). We detail analytically and empirically the convergence properties of both methods and their extended variants in all possible system settings. With this result, a complete and rigorous theory of both methods is furnished.

In Chapter[6] 8 we extend this analysis to randomized ridge regression. We prove that when the number of points dominates the number of features, working with random features is preferable (RGS), and when the number of features dominates the number of points, then working with random data points is preferable (RK), and as a side result that an earlier algorithm that alternately uses both rows *and* columns is suboptimal. In both chapters, the conditioning of the linear systems naturally appears in the convergence rates through the singular values of the data matrix (just as the margin does for classification).

Fast and flexible ADMM algorithms for trend filtering

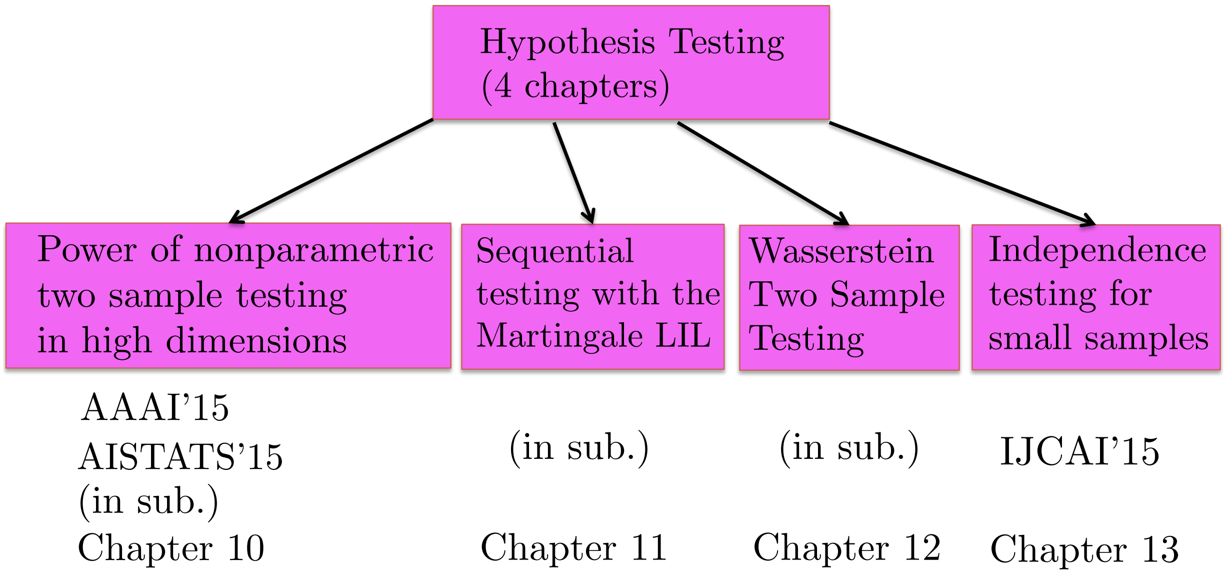
Chapter[7] 9 presents a fast and robust algorithm for trend filtering, a recently developed nonparametric regression tool. It has been shown that, for estimating functions whose derivatives are of bounded variation, trend filtering achieves the minimax optimal error rate, while other popular methods like smoothing splines and kernels do not. Standing in the way of a more widespread practical adoption, however, is a lack of scalable and numerically stable algorithms for fitting trend filtering estimates. We present a highly efficient, specialized ADMM routine for trend filtering. Our algorithm is competitive with the specialized interior point methods that are currently in use, and yet is far more numerically robust. Furthermore, the proposed ADMM implementation is very simple, and importantly, it is flexible enough to extend to many interesting related problems, such as sparse trend filtering and isotonic trend filtering. Software for our method is freely available, in both the C and R languages (see the `trendfilter` function in the R package `genlasso`).

[5] in submission, ArXiv preprint Ma* et al. [134]

[6] in submission, arXiv preprint Ramdas et al. [166]

[7] published in Ramdas and Tibshirani [162] at JCGS'15

1.4 Summary of Contributions in Nonparametric Testing


Hypothesis Testing
(4 chapters)
Power of nonparametric two sample testing in high dimensions
Sequential testing with the Martingale LIL
Wasserstein Two Sample Testing
Independence testing for small samples
AAAI'15
AISTATS'15
(in sub.)
Chapter 10
(in sub.)
Chapter 11
(in sub.)
Chapter 12
IJCAI'15
Chapter 13

Kernel and distance based Nonparametric two sample testing

In Chapter[8] 10, we show that kernelized methods (as well as distance-based methods) for two-sample testing or independence testing do suffer from the curse of dimensionality, challenging much existing folklore, by demonstrating a decrease of power in high dimensions (and explore the role that the median heuristic plays here).

We refer to the most common settings as mean difference alternatives (MDA), for testing differences only in first moments, and general difference alternatives (GDA), which is about testing for any difference in distributions. A large number of test statistics have been proposed for both these settings. We connect three classes of statistics - high dimensional variants of Hotelling's t-test, statistics based on Reproducing Kernel Hilbert Spaces, and energy statistics based on pairwise distances. We ask the following question - *how much statistical power do popular kernel and distance based tests for GDA have, compared against specialized tests for MDA, when the unknown distributions do actually differ in their means?*

To answer this, we characterize the power of popular tests for GDA like the Maximum Mean Discrepancy with the Gaussian kernel (gMMD) and Energy Distance with the Euclidean norm (eED) in the high-dimensional MDA regime. We prove several interesting properties relating these classes of tests under MDA, which include

(a) eED and gMMD have equal power; furthermore they also enjoy a free lunch, because (while they are additionally consistent for GDA) they have the same power as specialized high-dimensional t-tests for MDA, all of which are optimal for MDA according to our lower bounds.

(b) the power of gMMD is independent of the kernel bandwidth, as long as it is larger than the choice made by the median heuristic.

[8] Earlier work was published in Ramdas et al. [167] at AAAI'15, [168] at AISTATS'15; this chapter is in submission, with a preprint at Ramdas et al. [169].

(c) there is a clear and smooth computation-statistics tradeoff for linear-time, subquadratic-time and quadratic-time versions of these tests, with more computation resulting in higher power.

All three observations are practically important, since (a) implies that eED and gMMD while being consistent against all alternatives, are also automatically adaptive to simpler alternatives (and are optimal, for free, for MDA), (b) suggests that the median "heuristic" has some theoretical justification for being a default bandwidth choice, and (c) implies that, unlike previous analysis suggests, expending more computation yields direct statistical benefit by *orders of magnitude*. Point (a) has been previously observed in practice, but we provide the first theoretical explanation and further practical support for this phenomenon.

Sequential Two Sample Testing using the Martingale LIL

In Chapter[9] 11, we address a class of problems in multivariate nonparametric hypothesis testing, in one of the following two settings relevant to modern big-data analysis. In the first setting, the dataset is available offline, but it is prohibitively large; hence batch tests on the full data are computationally infeasible, and subsampling methods are impractical due to the impossibility of knowing the problem's hardness and hence how much subsampled data would suffice to decide between the hypotheses. In the second setting, data is arriving as a possibly infinite stream, but we are allowed minimal storage, and processing time linear in the dimensionality of the samples; however, most sequential hypothesis testing literature deals with either unidimensional or parametric or simple alternatives, and alternately running many smaller batch tests has the problem of having to aggressively correct for multiple testing.

We propose a new sequential algorithmic framework that has desirable computational and statistical properties, addressing shortcomings of the literature in both settings. Its analysis is based on a new finite-sample uniform empirical Bernstein version of the martingale law of the iterated logarithm (LIL), which may be of independent interest. As an example, we consider nonparametric two-sample mean testing, where one tests whether two (arbitrary) multivariate random variables have the same mean or not, when given access to a stream of i.i.d. data from the two distributions with unknown mean difference. We prove that (a) when the means are the same, the LIL allows proper type-1 error control, and (b) when the means are different, the expected stopping time and power of our sequential test are both essentially optimal, compared to the corresponding "oracle" batch test with the same computational resources. We also demonstrate how to extend this idea to nonparametric homogeneity testing and independence testing, and believe that many of the introduced ideas are more broadly applicable.

Wasserstein Two Sample Testing

Nonparametric two sample or homogeneity testing is a decision theoretic problem that involves identifying differences between two random variables without making parametric assumptions about their underlying distributions. The literature is old and rich, with a wide variety of statistics having being intelligently designed and analyzed, both for the unidimensional and the multivariate setting. In Chapter[10] 12, our contribution is to tie together many of these tests, drawing connections between seemingly very different statistics. Specifically, we form a chain of connections from univariate methods like the Kolmogorov-Smirnov test, QQ plots and ROC curves, to multivariate tests involving the Wasserstein distance, energy statistics and kernel based maximum mean discrepancy, that proceeds through the construction of a *smoothed* Wasserstein distance. Some observations in this chain are implicit in the literature, while others seem to have not been noticed thus far. We hope this will be a useful resource for theorists and practitioners familiar with one subset of methods but not with others.

[9] in submission, preprint at Ramdas and Balsubramani [157]

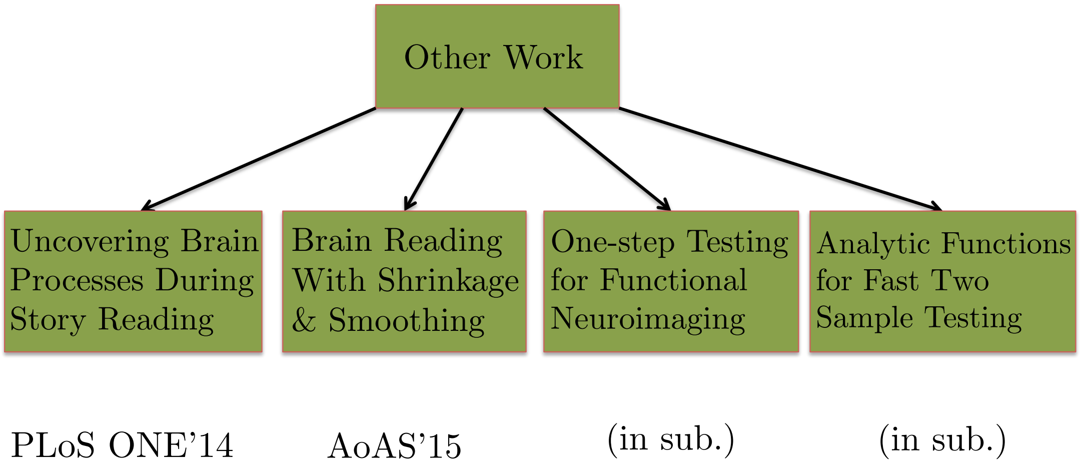
[10] in submission, preprint at Ramdas* et al. [165]

Nonparametric independence testing for small sample sizes

Chapter[11] 13 deals with the problem of nonparametric independence testing, a fundamental decision-theoretic problem that asks if two arbitrary (possibly multivariate) random variables X, Y are independent or not, a question that comes up in many fields like causality and neuroscience. While quantities like correlation of X, Y only test for (univariate) linear independence, natural alternatives like mutual information of X, Y are hard to estimate due to a serious curse of dimensionality. A recent approach, avoiding both issues, estimates norms of an *operator* in Reproducing Kernel Hilbert Spaces (RKHSs). Our main contribution is strong empirical evidence that by employing *shrunk* operators when the sample size is small, one can attain an improvement in power at low false positive rates. We analyze the effects of Stein shrinkage on a popular test statistic called HSIC (Hilbert-Schmidt Independence Criterion). Our observations provide insights into two recently proposed shrinkage estimators, SCOSE and FCOSE - we prove that SCOSE is (essentially) the optimal linear shrinkage method for *estimating* the true operator; however, the non-linearly shrunk FCOSE usually achieves greater improvements in *test power*. This work is important for more powerful nonparametric detection of subtle nonlinear dependencies for small samples.

1.5 Other Work

Here we summarize other work done that the author was actively involved with during the PhD, which unfortunately do not have their place in this thesis as the author did not lead these works. These are included here as a summary of the author's other interests (like neuroscience).


Other Work
Uncovering Brain Processes During Story Reading
Brain Reading With Shrinkage & Smoothing
One-step Testing for Functional Neuroimaging
Analytic Functions for Fast Two Sample Testing
PLoS ONE'14
AoAS'15
(in sub.)
(in sub.)

Simultaneously uncovering patterns of brain regions involved in story reading subprocesses

Story understanding involves many perceptual and cognitive subprocesses, from perceiving individual words, to parsing sentences, to understanding the relationships among the story characters. In Wehbe et al. [228], we present an integrated computational model of reading that incorporates these and additional subprocesses, simultaneously discovering their fMRI signatures. Our model predicts the fMRI activity associated with reading arbitrary text passages, well enough to distinguish which of two story segments is being read with 74% accuracy. This approach is the first to simultaneously track diverse reading subprocesses during complex story processing and predict the detailed neural representation of diverse

[11] Published in Ramdas* and Wehbe* [163] at IJCAI'15

story features, ranging from visual word properties to the mention of different story characters and different actions they perform. We construct brain representation maps that replicate many results from a wide range of classical studies that focus each on one aspect of language processing, and offers new insights on which type of information is processed by different areas involved in language processing. Additionally, this approach is promising for studying individual differences: it can be used to create single subject maps that may potentially be used to measure reading comprehension and diagnose reading disorders.

Regularized brain reading with smoothing and shrinkage

Functional neuroimaging measures how the brain responds to complex stimuli. However, sample sizes are modest, noise is substantial, and stimuli are high-dimensional. Hence, direct estimates are inherently imprecise and call for regularization. In Wehbe et al. [230], we compare a suite of approaches which regularize via *shrinkage*: ridge regression, the elastic net (a generalization of ridge regression and the lasso), and a hierarchical Bayesian model based on small-area estimation (SAE) ideas. We contrast regularization by shrinkage with regularization by *spatial smoothing*, and combinations of smoothing and shrinkage. All methods are tested on FMRI data from a reading comprehension experiment, for both predicting neural response to stimuli and decoding stimuli from responses. Surprisingly, *all* the regularization methods work equally well, suggesting that improvements will take not just *clever* methods, but *careful* modeling.

One step hypothesis testing for functional neuroimaging

A large part of functional imaging revolves around the localization of brain processes to specific regions. In Wehbe et al. [229], we frame many of the most common experimental approaches, including the recent adoption of classifiers as a way to decode brain processes, as indirect hypothesis tests. These tests fall in the categories of two-sample tests and independence tests. We advocate the direct use of more appropriate two-sample tests and independence tests that are theoretically sound and do not rely on the intermediate use of modeling procedures which might suffer from model misspecification and introduce arbitrary bias to the experimental results. Furthermore, we explore the problem of independence testing of non-IID random-processes data such as time series. We discuss available methods in the field and how to adapt them to the small sample setting of typical experiments. We illustrate this with a functional Magnetic Resonance Imaging (fMRI) experiment of subjects reading a chapter of a book, in which we show how to identify dependencies between the properties of the text and the brain activity of different regions at different latencies. The insights provided here are relevant beyond the realm of functional neuroimaging: they might be useful for any application field in which knowledge is to be concluded from data, such as medical or financial problems.

Analytic functions for fast two sample testing

In Chwialkowski et al. [41], we propose a nonparametric two-sample test with cost linear in the number of samples. Our test statistic uses differences in smoothed characteristic functions: these are able to distinguish a larger class of alternatives than the non-smoothed characteristic functions used in previous linear-time tests, while being much faster than the current state-of-the-art tests based on kernels or distances, which are quadratic in the sample size. Experiments on artificial benchmarks and on challenging real life testing problems demonstrate that our test gives a better time/power tradeoff than competing approaches, including sub-quadratic-time variants of the kernel tests. This performance advantage is retained even in high dimensions, and in cases where the difference in distributions is not observable in low order statistics.

Part I

Active Learning

Chapter 2

Active Learning : Lower bounds for Optimization

We focus on the problem of minimizing a convex function f over a convex set S given T queries to a stochastic first order oracle. We argue that the complexity of convex minimization is only determined by the rate of growth of the function around its minimizer $x^*_{f,S}$, as quantified by a Tsybakov-like noise condition. Specifically, we prove that if f grows at least as fast as $\|x - x^*_{f,S}\|^\kappa$ around its minimum, for some $\kappa > 1$, then the optimal rate of learning $f(x^*_{f,S})$ is $\Theta(T^{-\frac{\kappa}{2\kappa-2}})$. The classic rate $\Theta(1/\sqrt{T})$ for convex functions and $\Theta(1/T)$ for strongly convex functions are special cases of our result for $\kappa \to \infty$ and $\kappa = 2$, and even faster rates are attained for $\kappa < 2$. We also derive tight bounds for the complexity of learning $x^*_{f,S}$, where the optimal rate is $\Theta(T^{-\frac{1}{2\kappa-2}})$. Interestingly, these precise rates for convex optimization also characterize the complexity of active learning and our results further strengthen the connections between the two fields, both of which rely on feedback-driven queries.

2.1 Introduction and problem setup

Stochastic convex optimization in the first order oracle model is the task of approximately minimizing a convex function over a convex set, given oracle access to unbiased estimates of the function and gradient at any point, by using as few queries as possible [146].

A function f is convex on S if, for all $x, y \in S$, $t \in [0, 1]$,

$$f(tx + (1-t)y) \le tf(x) + (1-t)f(y)$$

f is Lipschitz with constant L if for all $x, y \in S$,

$$|f(x) - f(y)| \le L\|x - y\|$$

Equivalently, for subgradients $g_x \in \partial f(x)$, $\|g_x\|_* \le L$.

Without loss of generality, everywhere in this chapter we shall always assume $\|.\| = \|.\|_* = \|.\|_2$, and we shall always deal with convex functions with $L = 1$. Furthermore, we will consider the set $S \subseteq \mathbb{R}^d$ to be closed bounded convex sets with diameter $D = \max_{x,y \in S} \|x - y\| \le 1$. Let the collection of all such sets be $\mathbb{S}$. Given $S \in \mathbb{S}$, let the set of all such convex functions on S be $\mathcal{F}^C$ (with S implicit).

A stochastic first order oracle is a function that accepts $x \in S$ as input, and returns $(\hat{f}(x), \hat{g}(x))$ where $\mathbb{E}[\hat{f}(x)] = f(x)$, $\mathbb{E}[\hat{g}(x)] = g(x)$ (and furthermore, they have unit variance) where $g(x) \in \partial f(x)$ and the expectation is over any internal randomness of the oracle. Let the set of all such oracles be $\mathcal{O}$. As we refer to it later in the chapter, we note that a stochastic zeroth order oracle is defined analogously but only returns unbiased function values and no gradient information.

An optimization algorithm is a method M that repeatedly queries the oracle at points in S and returns $\hat{x}_T$ as an estimate of the optimum of f after T queries. Let the set of all such procedures be $\mathcal{M}$. A central question of the field is *"How close can we get to the optimum of a convex function given a budget of T queries?"*.

Let $x^*_{f,S} = \arg\min_{x \in S} f(x)$. Distance of an estimate $\hat{x}_T$ to the optimum $x^*_{f,S}$ can be measured in two ways. We define the *function-error* and *point-error* of M as:

$$\epsilon_T(M, f, S, O) = f(\hat{x}_T) - f(x^*_{f,S})$$
$$\rho_T(M, f, S, O) = \|\hat{x}_T - x^*_{f,S}\|$$

There has been a lot of past work on worst-case bounds for ϵ_T for common function classes. Formally, let

$$\epsilon^*_T(\mathcal{F}) = \sup_{O \in \mathcal{O}} \sup_{S \in \mathcal{S}} \inf_{M \in \mathcal{M}} \sup_{f \in \mathcal{F}} \mathbb{E}_O[\epsilon_T(M, f, S, O)]$$
$$\rho^*_T(\mathcal{F}) = \sup_{O \in \mathcal{O}} \sup_{S \in \mathcal{S}} \inf_{M \in \mathcal{M}} \sup_{f \in \mathcal{F}} \mathbb{E}_O[\rho_T(M, f, S, O)]$$

It is well known [146] that for the set of all convex functions, $\epsilon^*_T(\mathcal{F}^C) = \Theta(1/\sqrt{T})$. However, better rates are possible for smaller classes, like that of strongly convex functions, $\mathcal{F}^{SC}$.

A function f is strongly convex on S with parameter $\lambda > 0$ if for all $x, y \in S$ and for all $t \in [0, 1]$,

$$f(tx + (1-t)y) \leq tf(x) + (1-t)f(y) - \frac{1}{2}\lambda t(1-t)\|x - y\|^2$$

Intuitively, this condition means that f is lower bounded by a quadratic everywhere (in contrast, convex functions are lower bounded by a hyperplane everywhere). Again, it is well known [1, 95, 146] that that for the set of all strongly convex functions, $\epsilon^*_T(\mathcal{F}^{SC}) = \Theta(1/T)$. An immediate geometric question arises - what property of strongly convex functions allows them to be minimized quicker?

In this work, we answer the above question by characterizing precisely what determines the optimal rate and we derive what exactly that rate is for more general classes. We intuitively describe why such a characterization holds true and what it means by connecting it to a central concept in active learning. These bounds are shown to be tight for both function-error $f(x) - f(x^*_{f,S})$ and the less used, but possibly equally important, point-error $\|x - x^*_{f,S}\|$.

We claim that the sole determining factor for minimax rates is a condition about the growth of the function only around its optimum, and not a global condition about the strength of its convexity everywhere in space. For strongly convex functions, we get the well-known result that for optimal rates it is sufficient for the function to be lower bounded by a quadratic only around its optimum (not everywhere).

As we shall see later, any $f \in \mathcal{F}^{SC}$ satisfies

$$f(x) - f(x^*_{f,S}) \geq \frac{\lambda}{2}\|x - x^*_{f,S}\|^2 \tag{2.1}$$

On the same note, given a set $S \in \mathbb{S}$, let $\mathcal{F}^\kappa$ represent the set of all convex functions such that for all $x \in S$

$$f(x) - f(x^*_{f,S}) \geq \frac{\lambda}{2}\|x - x^*_{f,S}\|^\kappa \tag{2.2}$$

for some $\kappa \geq 1$. This forms a nested hierarchy of classes of $\mathcal{F}^C$, with $\mathcal{F}^{\kappa_1} \subset \mathcal{F}^{\kappa_2}$ whenever $\kappa_1 < \kappa_2$. Also notice that $\mathcal{F}^2 \supseteq \mathcal{F}^{SC}$ and $\bigcup_\kappa \mathcal{F}^\kappa \subseteq \mathcal{F}^C$. For any finite $\kappa < \infty$, this condition automatically ensures that the function is strictly convex and hence the minimizer is well-defined and unique.

Then we can state our main result as:

Theorem 1. *Let $\mathcal{F}^\kappa$ ($\kappa > 1$) be the set of all 1-Lipschitz convex functions on $S \in \mathbb{S}$ satisfying $f(x) - f(x^*_{f,S}) \geq \frac{\lambda}{2}\|x - x^*_{f,S}\|^\kappa$ for all $x \in S$ for some $\lambda > 0$. Then, for first order oracles, we have $\epsilon^*_T(\mathcal{F}^\kappa) = \Theta(T^{-\frac{\kappa}{2\kappa-2}})$ and $\rho^*_T(\mathcal{F}^\kappa) = \Theta(T^{-\frac{1}{2\kappa-2}})$. Also, for zeroth order oracles, we have $\epsilon^*_T(\mathcal{F}^\kappa) = \Omega(1/\sqrt{T})$ and $\rho^*_T(\mathcal{F}^\kappa) = \Omega(T^{-\frac{1}{2\kappa}})$.*

Note that for ϵ^*_T we get faster rates than $1/T$ for $\kappa < 2$. For example, if we choose $\kappa = 3/2$, then we surprisingly get $\epsilon^*_T(\mathcal{F}^{3/2}) = \Theta(T^{-3/2})$.

The proof idea in the lower bound arises from recognizing that the growth condition in equation (2.2) closely resembles the Tsybakov noise condition (TNC) [1] from statistical learning literature, which is known to determine minimax rates for passive and active classification [34, 218] and level set estimation [192, 216].

Specifically, we modify a proof from [34] that was originally used to find the minimax lower bound for active classification where the TNC was satisfied at the decision boundary. We translate this to our setting to get a lower bound on the optimization rate, where the function satisfies a convexity strength condition at its optimum. One can think of the rate of growth of the function around its minimum as determining how much the oracle's noise will drown out the true gradient information, thus measuring the signal to noise ratio near the optimum.

[155] notice that stochastic convex optimization and active learning have similar flavors because of the role of feedback and sequential dependence of queries. Our results make this connection more precise by demonstrating that the complexity of convex optimization in d-dimensions is precisely the same as the complexity of active learning in 1 dimension. Specifically, the rates we derive for function error and point error in first-order stochastic convex optimization of a d-dimensional function are precisely the same as the rates for classification error and error in localizing the decision boundary, respectively, in 1-dimensional active learning [34].

This result agrees with intuition since in 1 dimension, finding the decision boundary and the minimizer are equivalent to finding the zero-crossing of the regression function, $P(Y|X = x) - 1/2$, or the zero-point of the gradient, respectively (see Section 2.2.1 for details). Thus in 1D, it requires the same number of samples or time steps to find the decision boundary or the minimizer, respectively, using feedback-driven queries. In higher dimensions, the decision boundary becomes a multi-dimensional set whereas, for a convex function, the minimizer continues to be the point of zero-crossing of the gradient. Thus, rates for active learning degrade exponentially in dimension, whereas rates for first-order stochastic convex optimization don't.

For upper bounds, we slightly alter a recent variant of gradient descent from [95] and prove that it achieves the lower bound. While there exist algorithms in passive (non-active) learning that achieve the minimax rate without knowing the true behaviour at the decision boundary, unfortunately our upper bounds depend on knowing the optimal κ.

Summary of contributions

- We provide an interesting connection between strong convexity (more generally, uniform convexity) and the Tsybakov Noise Condition which is popular in statistical learning theory [218]. Both can be interpreted as the amount by which the signal to noise ratio decays on approaching the minimum in optimization or the decision boundary in classification.
- We use the above connection to strengthen the relationship between the fields of active learning and convex optimization, the seeds of which were sown in [155] by showing that the rates for first-order

[1] Sometimes goes by Tsybakov margin/regularity condition [118, 218]

stochastic convex optimization of a d-dimensional function are precisely the rates for 1-dimensional active learning.

- Using proof techniques from active learning [34], we get lower bounds for a hierarchy of function classes $\mathcal{F}^\kappa$, generalising known results for convex, strongly convex [146], [1] and uniformly convex classes [199].
- We show that the above rates are tight (all $\kappa > 1$) by generalising an algorithm from [95] that was known to be optimal for strongly convex functions, and also reproduce the optimal rates for κ-uniformly convex functions (only defined for $\kappa \geq 2$) [107].
- Our lower bounding proof technique also gets us, for free, lower bounds for the derivative free stochastic zeroth-order oracle setting, a generalization of those derived in [109].

2.2 From Uniform Convexity to TNC

A function f is said to be κ-uniformly convex ($\kappa \geq 2$) on $S \in \mathbb{S}$ if, for all $x, y \in S$ and all $t \in [0, 1]$,

$$f(tx + (1-t)y) \leq tf(x) + (1-t)f(y) - \frac{1}{2}\lambda t(1-t)\|x - y\|^\kappa$$

for some $\lambda > 0$ [107].

An equivalent first-order condition, is that for any subgradient $g_x \in \partial f(x)$, we have for all $x, y \in S$,

$$f(y) \geq f(x) + g_x^\top (y - x) + \frac{\lambda}{2}\|y - x\|^\kappa \tag{2.3}$$

When $\kappa = 2$, this is well known as strong convexity. It is well known that since $0 \in \partial f(x^*_{f,S})$, we have for all $x \in S$,

$$f(x) \geq f(x^*_{f,S}) + \frac{\lambda}{2}\|x - x^*_{f,S}\|^\kappa \tag{2.4}$$

This local condition is strictly weaker than (2.3) and it only States that the function grows at least as fast as $\|x - x^*_{f,S}\|^\kappa$ around its optimum. This bears a striking resemblance to the Tsybakov Noise Condition (also called the regularity or margin condition) from the statistical learning literature.

Tysbakov's Noise Condition We reproduce a relevant version of the condition from [34]. Define $\eta(x) := P(\ell(x) = 1|x)$, where $\ell(x)$ is the label of point x. Let x^* be the closest point to x such that $\eta(x^*) = 1/2$, ie on the decision boundary. η is said to satisfy the TNC with exponent $\kappa \geq 1$ if

$$|\eta(x) - \eta(x^*)| \geq \lambda\|x - x^*\|^\kappa \tag{2.5}$$

for all x in such that $|\eta(x) - 1/2| \leq \delta$ with $\delta > 0$.

It is natural to conjecture that the strength of convexity and the TNC play similar roles in determining minimax rates, and that rates of optimizing functions should really *only* depend on a TNC-like condition around their minima, motivating the definition of $\mathcal{F}^\kappa$ in equation 2.2. We emphasize that though uniform convexity is not defined for $\kappa < 2$, $\mathcal{F}^\kappa$ is well-defined for $\kappa \geq 1$ (see Appendix, Lemma 1).

The connection of the strength of convexity around the optimum to TNC is very direct in one-dimension, and we shall now see that it enables us to use an active classification algorithm to do stochastic convex optimization.

2.2.1 Making it transparent in 1-D

We show how to reduce the task of stochastically optimizing a one-dimensional convex function to that of active classification of signs of a monotone gradient. For simplicity of exposition, we assume that the set S of interest is $[0,1]$, and f achieves a unique minimizer x^* inside the set $(0,1)$.

Since f is convex, its true gradient g is an increasing function of x that is negative before x^* and positive after x^*. Assume that the oracle returns gradient values corrupted by unit variance gaussian noise [2]. Hence, one can think of $sign(g(x))$ as being the true label of point x, $sign(g(x)+z)$ as being the observed label, and finding x^* as learning the decision boundary (the point where labels switch signs). If we think of $\eta(x) = P(sign(g(x)+z) = 1|x)$, then minimizing f corresponds to identifying the Bayes classifier $[x^*, 1]$ because the point at which $\eta(x) = 0.5$ is where $g(x) = 0$, which is x^*.

If $f(x) - f(x^*) \geq \lambda \|x - x^*\|^\kappa$, then $|g_x| \geq \lambda \|x - x^*\|^{\kappa-1}$(see Appendix, Lemma 2). Let us consider a point x which is a distance $t > 0$ to the right of x^* and hence has label 1 (similar argument for $x < x^*$).

So, for all $g_x \in \partial f(x),\ g_x \geq \lambda t^{\kappa-1}$. In the presence of gaussian noise z, the probability of seeing label 1 is the probability that we draw z in $(-g_x, \infty)$ so that the sign of $g_x + z$ is still positive. This yields:

$$\eta(x) \;=\; P(g_x + z > 0) \;=\; 0.5 + P(-g_x < z < 0)$$

Note that the probability mass of a gaussian grows linearly around its mean (Appendix, Lemma 3); ie, for all $t < \sigma$ there exist constants a_1, a_2 such that $a_1 t \leq P(0 \leq z \leq t) \leq a_2 t$. So, we get

$$\begin{aligned} \eta(x) &\;\geq\; 0.5 + a_1 \lambda t^{\kappa-1} \\ \implies \quad & |\eta(x) - 1/2| \geq a_1 \lambda |x - x^*|^{\kappa-1} \end{aligned} \tag{2.6}$$

Hence, $\eta(x)$ satisfies TNC with exponent $\kappa - 1$.

[34] provide an analysis of the Burnashev-Zigangirov (BZ) algorithm, which is a noise-tolerant variant of binary bisection, when the regression function $\eta(x)$ obeys a TNC like in equation 2.6. The BZ algorithm solves the one-dimensional active classification problem such that after making T queries for a noisy label, it returns a confidence interval $\hat{I}_T$ which contains x^* with high probability, and $\hat{x}_T$ is chosen to be the midpoint of $\hat{I}_T$. They bound the excess risk $\int_{[x,1]\Delta[x^*,1]} |2\eta(x) - 1| dx$ where Δ is the symmetric difference operator over sets but small modifications to their proofs (see Appendix, Lemma 4) yield a bound on $\mathbb{E}|\hat{x}_T - x^*|$.

The setting of $\kappa = 1$ is easy because the regression function is bounded away from half (the true gradient doesn't approach zero, so the noisy gradient is still probably the correct sign) and we can show an exponential convergence of $\mathbb{E}(|\hat{x}_T - x^*|) = O(e^{-T\lambda^2/2})$. The unbounded noise setting of $\kappa > 1$ is harder and using a variant of BZ analysed in [34], we can show (see Appendix, Lemma 5) that $\mathbb{E}(|\hat{x}_T - x^*|) = \tilde{O}\left(\frac{1}{T}\right)^{\frac{1}{2\kappa-2}}$ and $\mathbb{E}(|\hat{x}_T - x^*|^\kappa) = \tilde{O}\left(\frac{1}{T}\right)^{\frac{\kappa}{2\kappa-2}}$. [3]

Interestingly, in the next section on lower bounds, we show that for any dimension, $\Omega\left(\frac{1}{T}\right)^{\frac{1}{2\kappa-2}}$ is the minimax convergence rate for $\mathbb{E}(\|\hat{x}_T - x^*\|)$.

2.3 Lower bounds using TNC

We prove lower bounds for $\epsilon_T^*(\mathcal{F}^\kappa), \rho_T^*(\mathcal{F}^\kappa)$ using a technique that was originally for proving lower bounds for active classification under the TNC [34], providing a nice connection between active learning and stochastic convex optimization.

[2] The gaussian assumption is only for this subsection

[3] We use $\tilde{O}$ to hide polylogarithmic factors.

Theorem 2. *Let $\mathcal{F}^\kappa$ ($\kappa > 1$) be the set of all 1-Lipschitz convex functions on $S \in \mathbb{S}$ satisfying $f(x) - f(x^*_{f,S}) \geq \frac{\lambda}{2}\|x - x^*_{f,S}\|^\kappa$ for all $x \in S$ for some $\lambda > 0$. Then, we have $\epsilon^*_T(\mathcal{F}^\kappa) = \Omega(T^{-\frac{\kappa}{2\kappa-2}})$ and $\rho^*_T(\mathcal{F}^\kappa) = \Omega(T^{-\frac{1}{2\kappa-2}})$.*

The proof technique is summarised below. We demonstrate an oracle O^* and set S^* over which we prove a lower bound for $\inf_{M\in\mathcal{M}} \sup_{f\in\mathcal{F}^\kappa} \mathbb{E}_O[\epsilon_T(M, f, S, O)]$. Specifically, let S^* be $[0,1]^d \cap \{\|x\| \leq 1\}$ and O^* just adds standard normal noise to the true function and gradient values. We then pick two similar functions in the class $\mathcal{F}^\kappa$ and show that they are hard to differentiate with only T queries to O^*.

We go about this by defining a semi-distance between any two elements of $\mathcal{F}^\kappa$ as the distance between their minima. We then choose two very similar functions f_0, f_1 whose minima are $2a$ apart (we shall fix a later). The oracle chooses one of these two functions and the learner gets to query at points x in domain S^*, receiving noisy gradient and function values $y \in \mathbb{R}^d, z \in \mathbb{R}$. We then define distributions corresponding to the two functions P^0_T, P^1_T and choose a so that these distributions are at most a constant KL-distance γ apart. We then use Fano's inequality which, using a and γ, lower bounds the probability of identifying the wrong function by any estimator (and hence optimizing the wrong function) given a finite time horizon of length T.

The use of Fano's inequality is not new to convex optimization, but proofs that lower-bound the probability of error under a sequential, feedback-driven querying strategy are prominent in active learning, and we show such proofs also apply to convex optimization thanks to the relation of uniform convexity around the minimum to the Tysbakov Noise Condition. We State Fano's inequality for completeness:

Theorem 3. *[218] Let $\mathcal{F}$ be a model class with an associated semi-distance $\delta(\cdot,\cdot) : \mathcal{F} \times \mathcal{F} \to \mathbb{R}$ and each $f \in \mathcal{F}$ having an associated measure P^f on a common probability space. Let $f_0, f_1 \in \mathcal{F}$ be such that $\delta(f_0, f_1) \geq 2a > 0$ and $KL(P^0||P^1) \leq \gamma$. Then,*

$$\inf_{\hat{f}} \sup_{f\in\mathcal{F}} P^f\left(\delta(\hat{f}, f) \geq a\right) \geq \max\left(\frac{\exp(-\gamma)}{4}, \frac{1-\sqrt{\gamma/2}}{2}\right)$$

2.3.1 Proof of Theorem 2

For technical reasons, we choose a subclass $\mathcal{U}^\kappa \subset \mathcal{F}^\kappa$ which is chosen such that every point in S^* is the unique minimizer of exactly one function in $\mathcal{U}^\kappa$. By construction of $\mathcal{U}^\kappa$, returning an estimate $\hat{x}_T \in S^*$ is equivalent to identifying the function $\hat{f}_T \in \mathcal{U}^\kappa$ whose minimizer is at $\hat{x}_T$. So we now proceed to bound $\inf_{\hat{f}_T} \sup_{f\in\mathcal{U}^\kappa} \mathbb{E}\|\widehat{x_T} - x^*_{f,S^*}\|$.

Recall that we chose $S^* = [0,1]^d \cap \{\|x\| \leq 1\}$. Define the semi-distance $\delta(f_a, f_b) = \|x^*_a - x^*_b\|$ and let [4]

$$f_0(x) = c_1 \sum_{i=1}^{d} |x_i|^\kappa = c_1\|x\|^\kappa_\kappa$$

$$g_0(x) = \kappa c_1(x_1^{\kappa-1}, ..., x_d^{\kappa-1})$$

so that $x^*_{0,S^*} = \vec{0}$. Now define $\vec{a_1} = (a, 0, ..., 0)$ and let

$$f_1(x) = \begin{cases} c_1\left(\|x - 2\vec{a_1}\|^\kappa_\kappa + c_2\right) & x_1 \leq 4a \\ f_0(x) & \text{o.w.} \end{cases}$$

$$g_1(x) = \begin{cases} \kappa c_1\left(\frac{|x_1-2a|^\kappa}{(x_1-2a)}, x_2^{\kappa-1}, ..., x_d^{\kappa-1}\right) & x_1 \leq 4a \\ g_0(x) & \text{o.w.} \end{cases}$$

[4] For $\kappa = 2$, note that $f_0, f_1 \in \mathcal{F}^{SC}$ (strongly convex)

so that $x^*_{1,S^*} = 2\vec{a}$ and hence $\delta(f_0, f_1) = 2a$. Notice that these two functions and their gradients differ only on a set of size $4a$. Here, $c_2 = (4a)^\kappa - (2a)^\kappa$ is a constant ensuring that f_2 is continuous at $x_1 = 4a$, and c_1 is a constant depending on κ, d ensuring that the functions are 1-Lipschitz on S^*. Both parts of f_1 are convex and the gradient of f_1 increases from $x_1 = 4a^-$ to $x_1 = 4a^+$, maintaining convexity. Hence we conclude that both functions are indeed convex and both are in $\mathcal{F}^\kappa$ for appropriate c_1 (Appendix, Lemma 6). Our interest here is the dependence on T, so we ignore these constants to enhance readability.

On querying at point $X = x$, the oracle returns $Z \sim \mathcal{N}(f(x), \sigma^2))$ and $Y \sim \mathcal{N}(g(x), \sigma^2 I_d)$. In other words, for $i = 0, 1$, we have $P^i(Z_t, Y_t | X = x_t) = \mathcal{N}\big((f_i(x_t), g_i(x_t)), \sigma^2 I_{d+1}\big)$. Let $S_1^T = (X_1^T, Y_1^T, Z_1^T)$ be the set of random variables corresponding to the whole sequence of T query points and responses. Define a probability distribution corresponding to every $f \in \mathcal{U}^\kappa$ as the joint distribution of S_1^T if the true function was f, and so

$$P_T^0 := P^0(X_1^T, Y_1^T, Z_1^T), \;\; P_T^1 := P^1(X_1^T, Y_1^T, Z_1^T)$$

We show that the KL-divergence of these distributions is $\mathrm{KL}(P_T^0, P_T^1) = O(Ta^{2\kappa-2})$ and choose $a = T^{-\frac{1}{2\kappa-2}}$ so that $\mathrm{KL}(P_T^0, P_T^1) \le \gamma$ for some constant $\gamma > 0$.

Lemma 1. $\mathrm{KL}(P_T^0, P_T^1) = O(Ta^{2\kappa-2})$

Proof:

$$\begin{aligned}
\mathrm{KL}(P_T^0, P_T^1) &= \mathbb{E}^0\left[\log \frac{P^0(X_1^T, Y_1^T, Z_1^T)}{P^1(X_1^T, Y_1^T, Z_1^T)}\right] \\
&= \mathbb{E}^0\left[\log \frac{\prod_t P^0(Y_t, Z_t|X_t) P(X_t|X_1^{t-1}, Y_1^{t-1}, Z_1^{t-1})}{\prod_t P^1(Y_t, Z_t|X_t) P(X_t|X_1^{t-1}, Y_1^{t-1}, Z_1^{t-1})}\right] \qquad (2.7) \\
&= \mathbb{E}^0\left[\log \frac{\prod_{t=1}^T P^0(Y_t, Z_t|X_t)}{\prod_{t=1}^T P^1(Y_t, Z_t|X_t)}\right] \\
&= \sum_{t=1}^T E^0\left[\mathbb{E}^0\left[\log \frac{P^0(Y_t, Z_t|X_t)}{P^1(Y_t, Z_t|X_t)}\middle| X_1, ..., X_T\right]\right] \\
&\le T \max_{x\in[0,1]^d} \mathbb{E}^0\left[\log \frac{P^0(Y_1, Z_1|X_1)}{P^1(Y_1, Z_1|X_1)}\middle| X_1 = x\right] \\
&= T \max_{x\in[0,1]^d} \mathbb{E}^0\left[\log \frac{P^0(Y_1|X_1)P^0(Z_1|X_1)}{P^1(Y_1|X_1)P^1(Z_1|X_1)}\middle| X_1 = x\right] \qquad (2.8) \\
&\le T\left(\max_{x\in[0,1]^d} \mathbb{E}^0\left[\log \frac{P^0(Y_1|X_1)}{P^1(Y_1|X_1)}\middle| X_1 = x\right]\right) + T\left(\max_{x\in[0,1]^d} \mathbb{E}^0\left[\log \frac{P^0(Z_1|X_1)}{P^1(Z_1|X_1)}\middle| X_1 = x\right]\right) \\
&= \frac{T}{2}\left(\max_{x\in[0,1]^d} \|g_0(x) - g_1(x)\|^2\right) + \frac{T}{2}\left(\max_{x\in[0,1]^d} (f_0(x) - f_1(x))^2\right) \qquad (2.9) \\
&= \frac{c_1^2 T}{2}\left(\kappa^2 \max_{x_1\in[0,4a]}\left(\frac{|x_1 - 2a|^\kappa}{(x_1 - 2a)} - x_1^{\kappa-1}\right)^2\right) \frac{c_1^2 T}{2}\left(\max_{x_1\in[0,4a]} (|x_1 - 2a|^\kappa - x_1^\kappa)^2\right) \qquad (2.10) \\
&= O(Ta^{2\kappa-2}) + O(Ta^{2\kappa}) = O(Ta^{2\kappa-2})
\end{aligned}$$

(2.7) follows because the distribution of X_t conditional on $X_1^{t-1}, Y_1^{t-1}, Z_1^{t-1}$ depends only on the algorithm M and does not change with the underlying distribution. (2.8) follows because $Y_t \perp Z_t$ when conditioned on X_t. We also used $(Y_i, Z_i|X_i) \perp (Y_j, Z_j|X_j)$ for $i \neq j$. (2.9) follows because the KL-divergence between two identity-covariance gaussians is just half the squared euclidean distance between their means. (2.10) follows by simply substituting the gradient/function values which differ only on $x_1 \in [0, 4a]$.

Using Theorem 3 with $a = T^{-\frac{1}{2\kappa-2}}$, for some $C > 0$ we get $\inf_{\hat{f}_T} \sup_{f \in \mathcal{U}^\kappa} P_f(\delta(\hat{f}_T, f) \geq a) \geq C$. Hence,

$$\begin{aligned}\inf_{\hat{f}_T} \sup_{f \in \mathcal{U}^\kappa} \mathbb{E}\|\hat{x}_T - x_f^*\| &\geq a \cdot \inf_{\hat{f}_T} \sup_{f \in \mathcal{U}^\kappa} P_f(\delta(\hat{f}_T, f) \geq a) \\ &\geq \quad a \cdot C \quad = \quad CT^{-\frac{1}{2\kappa-2}}\end{aligned}$$

where we used Markov's inequality, Fano's inequality and finally the aforementioned choice of a.

This gives us our required bound on $\rho_T^*(\mathcal{U}^\kappa)$, and correspondingly also for $\epsilon_T^*(\mathcal{U}^\kappa)$ because

$$\begin{aligned}\inf_{M} \sup_{f \in \mathcal{U}^\kappa} \mathbb{E}[f(\hat{x_T}) - f(x_f^*)] &\geq \inf_{M} \sup_{f \in \mathcal{U}^\kappa} \lambda[\mathbb{E}\|\hat{x_T} - x_f^*\|^\kappa] \\ &\geq \quad \inf_{\hat{f}_T} \sup_{f \in \mathcal{U}^\kappa} \lambda[\mathbb{E}\|\hat{x_T} - x^*\|]^\kappa\end{aligned}$$

where the first inequality follows because $f \in \mathcal{F}^\kappa$, and the second follows by applying Jensen's for $\kappa > 1$. Finally, we get the bounds on $\rho_T^*(\mathcal{F}^\kappa)$ and $\epsilon_T^*(\mathcal{F}^\kappa)$ because we are now taking sup over the larger class $\mathcal{F}^\kappa \supset \mathcal{U}^\kappa$. This concludes the proof of Theorem 2.

This is a generalisation of known lower bounds, because we can recover existing lower bounds for the convex and strongly convex settings by choosing $\kappa \to \infty$ and $\kappa = 2$ respectively. Furthermore, we will show that these bounds are tight for all $\kappa > 1$. These bounds also immediately yield lower bounds for uniformly convex functions, since $\|x\|_\kappa^\kappa$ is κ-uniformly convex (Appendix, Lemma 8) which can also be arrived from the results of [199] using an online-to-batch conversion.

2.3.2 Derivative-Free Lower Bounds

The above proof immediately gives us a generalization of recent tight lower bounds for derivative free optimization [109], in which the authors consider zeroth-order oracles (no gradient information) and find that $\epsilon_T^*(\mathcal{F}^C) = \Theta(1/\sqrt{T}) = \epsilon_T^*(\mathcal{F}^{SC})$ [5] concluding that strong convexity does not help in this setting. Here, we show

Theorem 4. *Let $\mathcal{F}^\kappa$ ($\kappa > 1$) be the set of all 1-Lipschitz convex functions on $S \in \mathbb{S}$ satisfying $f(x) - f(x_{f,S}^*) \geq \frac{\lambda}{2}\|x - x_{f,S}^*\|^\kappa$ for all $x \in S$ for some $\lambda > 0$. Then, in the derivative-free zeroth-order oracle setting, we have $\epsilon_T^*(\mathcal{F}^\kappa) = \Omega(1/\sqrt{T})$ and $\rho_T^*(\mathcal{F}^\kappa) = \Omega(T^{-\frac{1}{2\kappa}})$.*

Ignoring y, Y_1^T, define $P_T^0 := P^0(X_1^T, Z_1^T), P_T^1 := P^1(X_1^T, Z_1^T)$ to get $\mathrm{KL}(P_T^0, P_T^1) = O(Ta^{2\kappa})$. Choose $a = T^{-\frac{1}{2\kappa}}$ so that $\mathrm{KL}(P_T^0, P_T^1) \leq \gamma$ for some $\gamma > 0$, and apply Fano's to get $\inf_{\hat{f}_T} \sup_{f \in \mathcal{U}^\kappa} \mathbb{E}\|\hat{x}_T - x_f^*\| = CT^{-\frac{1}{2\kappa}}$ for some $C > 0$.

2.4 Matching Upper Bounds using Epoch-GD

We show that the bounds from Section 2.3 are tight by presenting an algorithm achieving the same rate.

Theorem 5. *Algorithm EpochGD$(S, \kappa, T, \delta, G, \lambda)$ returns $\hat{x}_T \in S$ after T queries to any oracle $O \in \mathcal{O}$, such that for any $f \in \mathcal{F}^\kappa, \kappa > 1$ on any $S \in \mathbb{S}$, $f(\hat{x}_T) - f(x_f^*) = \widetilde{O}(T^{-\frac{\kappa}{2\kappa-2}})$ and $\|\hat{x}_T - x_f^*\| = \widetilde{O}(T^{-\frac{1}{2\kappa-2}})$ hold with probability at least $1 - \delta$ for any $\delta > 0$.* [6]

[5] The κ in [109] should not be confused with our TNC exponent $\kappa = 2$ for $\mathcal{F}^{SC}$

[6] $\widetilde{O}$ hides $\log\log T$ and $\log(1/\delta)$ factors

Algorithm 1 EpochGD (domain S, exponent $\kappa > 0$, convexity parameter $\lambda > 0$, confidence $\delta > 0$, oracle budget T, subgradient bound G)

```
Initialize $x_1^1 \in S$ arbitrarily, $e = 1$
Initialize $T_1 = 2C_0$, $\eta_1 = C_1\, 2^{-\frac{\kappa}{2\kappa-2}}$, $R_1 = \left(\frac{C_2\eta_1}{\lambda}\right)^{1/\kappa}$

while $\sum_{i=1}^{e} T_i \le T$ do
    for $t = 1$ to $T_e$ do
        Query the oracle at $x_t^e$ to obtain $\hat{g}_t$
        $$x_{t+1}^e = \prod_{S \cap B(x_1^e, R_e)} (x_t^e - \eta_e \hat{g}_t)$$
    end for
    Set $x_1^{e+1} = \frac{1}{T_e}\sum_{t=1}^{T_e} x_t^e$
    Set $T_{e+1} = 2T_e$, $\eta_{e+1} = \eta_e \cdot 2^{-\frac{\kappa}{2\kappa-2}}$
    Set $R_{e+1} = \left(\frac{C_2\eta_{e+1}}{\lambda}\right)^{1/\kappa}$, $e \leftarrow e+1$
end while
Output: $x_1^e$
```

Recall that for $f \in \mathcal{F}^\kappa$, $\|g_x\| \le 1$ for any subgradient at any $x \in S$. Since the oracle may introduce bounded variance noise, we have $\|\hat{g}_x\| \le 1 + c\sigma^2$ with high probability. Here, to keep a parallel with [95], we use $\|\hat{g}_x\| \le G$ for convenience. Also, in algorithm 1 $B(x, R)$ refers to the ball around x of radius R i.e. $B(x, R) = \{y \mid \|x - y\| \le R\}$.

We note that for uniformly convex functions ($\kappa \ge 2$), [107] derive the same upper bounds. Our rates are valid for $1 < \kappa < 2$ and hold more generally as we have a weaker condition on $\mathcal{F}^\kappa$.

2.4.1 Proof of Theorem 5

We generalize the proof in [95] for strongly convex functions ($\kappa = 2$) and derive values for C_0, C_1 and C_2 for which Theorem 5 holds. We begin by showing that f having a bounded subgradient corresponds to a bound on the diameter of S, and hence on the maximum achievable function value.

Lemma 2. *If $f \in \mathcal{F}^\kappa$ and $\|g_x\| \le G$, then for all $x \in S$, we have $\|x - x_f^*\| \le (G\lambda^{-1})^{\frac{1}{\kappa-1}} =: D$ and $f(x) - f(x_f^*) \le (G^\kappa\lambda^{-1})^{\frac{1}{\kappa-1}} =: M$*

Proof: By convexity, $f(x) - f(x_f^*) \le g_x^\top(x - x_f^*) \le \|g_x\| \cdot \|x - x_f^*\|$ (Holder's inequality), implying that $G\|x - x_f^*\| \ge f(x) - f(x_f^*) \ge \lambda\|x - x^*\|^\kappa$.

Hence, $\|x - x_f^*\|^{\kappa-1} \le G/\lambda$ or $\|x - x_f^*\| \le G^{\frac{1}{\kappa-1}}/\lambda^{\frac{1}{\kappa-1}}$. Finally $f(x) - f(x_f^*) \le G\|x - x_f^*\| \le G^{\frac{\kappa}{\kappa-1}}/\lambda^{\frac{1}{\kappa-1}}$.

Lemma 3. *Let $\|x_1 - x_f^*\| \le R$. Apply T iterations of the update $x_{t+1} = \Pi_{S\cap B(x_1,R)}(x_t - \eta\hat{g}_t)$, where $\hat{g}_t$ is an unbiased estimator for the subgradient of f at x_t satisfying $\|\hat{g}_t\| \le G$. Then for $\bar{x} = \frac{1}{T}\sum_t x_t$ and any $\delta > 0$, with probability at least $1 - \delta$, we have*

$$f(\bar{x}) - f(x_f^*) \le \frac{\eta G^2}{2} + \frac{\|x_1 - x_f^*\|^2}{2\eta T} + \frac{4GR\sqrt{2\log(1/\delta)}}{\sqrt{T}}$$

Proof: Lemma 10 in [95].

Lemma 4. *For any epoch e and any $\delta > 0$, $T_e = C_0 2^e$, $E = \lfloor \log(\frac{T}{C_0} + 1) \rfloor$, $\eta_e = C_1 2^{-e\frac{\kappa}{2\kappa-2}}$, for appropriate C_0, C_1, C_2, we have with probability at least* $(1 - \frac{\delta}{E})^{e-1}$

$$\Delta_e := f(x_1^e) - f(x_f^*) \leq C_2 \eta_e$$

Proof: We let $\widetilde{\delta} = \frac{\delta}{E}$ and use proof by induction on e.

The first step of induction, $e = 1$, requires

$$\Delta_1 \leq C_2 \eta_1 = C_2 C_1 2^{-\frac{\kappa}{2\kappa-2}} \quad \textbf{[R1]}$$

Assume that $\Delta_e \leq C_2 \eta_e$ for some $e \geq 1$, with probability at least $(1 - \widetilde{\delta})^{e-1}$ and we now prove it correspondingly for epoch $e + 1$. We condition on the event $\Delta_e \leq C_2 \eta_e$ which happens with the above probability. By the TNC, $\Delta_e \geq \lambda \|x_1^e - x^*\|^\kappa$, and the conditioning implies that $\|x_1^e - x^*\| \leq (C_2 \eta_e / \lambda)^{1/\kappa}$, which is the radius R_e of the ball for the EpochGD projection step.

Lemma 3 applies with $R = R_e = (\frac{C_2 \eta_e}{\lambda})^{\frac{1}{\kappa}}$ and so with probability at least $1 - \widetilde{\delta}$, we have

$$\Delta_{e+1} \leq \frac{\eta_e G^2}{2} + \frac{\|x_1^e - x^*\|^2}{2\eta_e T_e} + \frac{4G(\frac{C_2\eta_e}{\lambda})^{\frac{1}{\kappa}} \sqrt{2\log(\frac{1}{\widetilde{\delta}})}}{\sqrt{T}_e}$$

$$\leq \frac{\eta_e G^2}{2} + \frac{C_2^{\frac{2}{\kappa}} \eta_e^{\frac{2}{\kappa}}}{2\eta_e T_e \lambda^{\frac{2}{\kappa}}} + \frac{4G(\frac{C_2\eta_e}{\lambda})^{\frac{1}{\kappa}} \sqrt{2\log(\frac{1}{\widetilde{\delta}})}}{\sqrt{T}_e}$$

For the induction, we would like $RHS \leq \eta_e G^2 \leq C_2 \eta_{e+1}$ which can be achieved by

$$\frac{C_2^{\frac{2}{\kappa}} \eta_e^{\frac{2}{\kappa}}}{2\eta_e T_e \lambda^{\frac{2}{\kappa}}} \leq \frac{\eta_e G^2}{6} \quad \textbf{[R2]}$$

$$\frac{4G(\frac{C_2\eta_e}{\lambda})^{\frac{1}{\kappa}} \sqrt{2\log(\frac{1}{\widetilde{\delta}})}}{\sqrt{T}_e} \leq \frac{\eta_e G^2}{3} \quad \textbf{[R3]}$$

$$\eta_e G^2 \leq C_2 \eta_{e+1} \quad \textbf{[R4]}$$

Then, factoring in the conditioned event which happens with probability at least $(1 - \widetilde{\delta})^{e-1}$ we would get $\Delta_{e+1} \leq C_2 \eta_{e+1}$ with probability at least $(1 - \widetilde{\delta})^e$.

We set C_0, C_1, C_2 such that the four conditions hold.

[R4] $\implies C_2 \geq G^2 2^{\frac{\kappa}{2\kappa-2}}$, a lower bound for C_2.

[R2] $\implies C_1 \geq \left(\frac{3}{G^2 C_0}\right)^{\frac{\kappa}{2\kappa-2}} \left(\frac{C_2}{\lambda}\right)^{\frac{1}{\kappa-1}}$

[R3] $\implies C_1 \geq \left(\frac{3(96\log(1/\widetilde{\delta}))}{G^2 C_0}\right)^{\frac{\kappa}{2\kappa-2}} \left(\frac{C_2}{\lambda}\right)^{\frac{1}{\kappa-1}}$

This is the stronger condition on C_1.

Observe that if $C_0 = 288 \log(1/\widetilde{\delta})$, by substitution we get the inequality $C_2 \eta_1 = C_1 C_2 2^{-\frac{\kappa}{2\kappa-2}} \geq M 2^{\frac{\kappa}{2(\kappa-1)^2}}$

[R1] is trivially true for the above choices of C_0, C_1, C_2, because $\Delta_1 \leq M \leq M 2^{\frac{\kappa}{2(\kappa-1)^2}} \leq C_2 \eta_1$

Hence, $C_0 = 288 \log(E/\delta)$, $C_1 = \frac{G^{\frac{2-\kappa}{\kappa-1}} 2^{\frac{\kappa}{2(\kappa-1)^2}}}{\lambda^{\frac{1}{\kappa-1}}}$ and $C_2 = G^2 2^{\frac{\kappa}{2\kappa-2}}$ satisfy the lemma. As a sanity check, [95] choose $C_0 = 288 \log(E/\delta), C_1 = 2/\lambda, C_2 = 2G^2$ for strongly convex functions.

The algorithm runs for $E = \lfloor \log(\frac{T}{C_0} + 1) \rfloor$ rounds so that the total number of queries is at most T. [7] The bound for Δ_{E+1} yields the bounds on function error immediately by noting that $(1 - \frac{\delta}{E})^E \geq 1 - \delta$ and since $f \in \mathcal{F}^\kappa$, we can bound the point error

$$\|\hat{x}_T - x^*\| \leq \lambda^{-1/\kappa}[f(\hat{x}_T) - f(x^*)]^{1/\kappa}$$

2.5 Discussion

The most common assumptions in the literature for proving convergence results for optimization algorithms are those of convexity and strong convexity, and [107] recently prove upper bounds using dual averaging for κ-uniformly convex functions when $\kappa \geq 2$. These classes impose a condition on the behaviour of the function, the strength of its convexity, everywhere in the domain. The TNC condition we consider for our smooth hierarchy of classes is natural and strictly weaker because it is immediately implied by uniform convexity or strong convexity in the realm of $\kappa \geq 2$, and has no corresponding notion when $1 < \kappa < 2$.

The lower bound $\Omega(T^{-\frac{\kappa}{2\kappa-2}})$ for ϵ^* that we prove immediately gives us the $\Omega(1/T)$ lower bound for strongly convex functions and the classic $\Omega(1/\sqrt{T})$ bound when $\kappa \to \infty$. The lower bound $\Omega(T^{-\frac{1}{2\kappa-2}})$ for ρ^* is interesting because the optimization literature does not often focus on point-error estimates. We demonstrate how to use an active learning proof technique that is novel in its application to optimization, having the additional benefit that it gives tight rates for derivative free optimization with no additional work. It is useful to have a unified proof generalising rates for convex, strongly convex, uniformly convex and more in both the first and zeroth order stochastic oracle settings.

We note that the rates for both ϵ^* and ρ^* are strongly supported by intuition as we can note by the rate's behaviour at the extremes of κ, near 1 and ∞. If the function has $\kappa \to 1$, then this is the best case because of large signal to noise ratio, as the gradient jumps signs rapidly without spending much time around zero where it can be corrupted by noise, and we should be able to identify the optimum extremely fast (function error rates even better than $1/T$), as supported by our result for the bounded noise setting in 1-D, and also by the tight upper bounds for using Epoch-GD. However, when $\kappa \to \infty$, the function starts to look extremely flat around its minimum, and while we can optimize function-error very well (because a wide range of points have function value close to the minimum value), it is hard to get close to the true optimum with noisy samples.

The reduction from stochastic optimization to active learning in 1D is interesting but we are uncertain if this can be extended to higher dimensions to give a generic reduction from one setting to the other (given an algorithm for active learning, can it solve an instance of stochastic optimization). It is an open problem to prove a positive or negative result of this type in the first or zeroth-order oracle setting.

Our upper bounds on ϵ and ρ involve a generalisation of Epoch Gradient Descent [95] and they demonstrate that the lower bounds achieved in terms of κ are indeed correct and tight. We make the same kind of assumptions as [107] and [95] - the number of time steps T, a bound on noisy subgradients G and the convexity parameter λ. Substituting $\kappa = 2$ in our algorithm yields the rate of $O(1/T)$ for strongly convex functions and $\kappa \to \infty$ recovers the $O(1/\sqrt{T})$ rate for convex functions as well.

Our lower bound proof bounds ϵ^* and ρ^* simultaneously, by bounding the point-error and using the class definition to bound the function-error (in both first order and zeroth order oracle settings). The upper-bound proofs proceed in the opposite direction by bounding the function-error and then using the TNC condition to bound the point-error.

[7] We lose $\log \log T$ factors here, like [95]. Alternatively, using $E = \lfloor \log(\frac{T}{288} + 1) \rfloor$, we could run for $T \log \log T$ steps and get error bound $O(T^{-\frac{\kappa}{2\kappa-2}})$.

In practice, one may not know the degree of convexity of the function at hand, but every function has a unique smallest κ for which it is in $\mathcal{F}^\kappa$, and using a larger κ will still maintain convergence (but at slower rates). If we only know that f is convex then we can use any gradient descent algorithm, and if we know it is strongly convex then we can use $\kappa = 2$, so our algorithm is not any weaker than existing ones, but it is certainly stronger if we know κ exactly.

We do not know if a variant of our algorithm can be designed which is adaptive to unknown κ. Function and gradient values should enable you to characterize the function around that region of space, but a function may have different smoothness is different parts of the space and old gradient information could be misleading. For example, consider a function on $[-0.5, 0.5]$ which is $2x^2$ between $[-0.25, 0.25]$, and grows linearly with gradient ± 1 in the rest of the space. This function is not strongly convex, but it is in $\mathcal{F}^2$, and it changes behaviour at $x = \pm 0.25$.

Hints of connections to active learning have been lingering in the literature, as noted by [155] but our borrowed lower bound proof from active learning and the one-dimensional upper bound reduction gives hope of a much more fertile intersection. While many active learning methods degrade exponentially with dimension d, the rates in optimization degrade polynomially since active learning is trying to solve harder problem like learning a $(d-1)$-dimensional decision boundary or level set, while optimization problems are just interested in getting to a single good point (for any d). This still leaves open the possibility of using a one dimensional active learning algorithm as a subroutine for a d-dimensional convex optimization problem. We feel that this is the start of stronger conceptual ties between these fields.

2.6 Supporting Proofs

Lemma 5. *No function can satisfy Uniform Convexity for $\kappa < 2$, but they can be in $\mathcal{F}^\kappa$ for $\kappa < 2$.*

Proof: If uniform convexity could be satisfied for (say) $\kappa = 1.5$, then we have for all $x, y \in S$

$$f(y) - f(x) - g_x^\top (y - x) \geq \frac{\lambda}{2} \|x - y\|_2^{1.5}$$

Take x, y both on the positive $\mathbf{x}$-axis. The Taylor expansion would require, for some $c \in [x, y]$,

$$\begin{aligned} f(y) - f(x) - g_x^\top (y - x) &= \frac{1}{2}(x - y)^\top H(c)(x - y) \\ &\leq \frac{\|H(c)\|_F}{2} \|x - y\|_2^2 \end{aligned}$$

Now, taking $\|x - y\|_2 = \epsilon \to 0$ by choosing x closer to y, the Taylor condition requires the residual to grow like ϵ^2 (going to zero fast), but the UC condition requires the residual to grow at least as fast as $\epsilon^{1.5}$ (going to zero slow). At some small enough value of ϵ, this would not be possible. Since the definition of UC needs to hold for all $x, y \in S$, this gives us a contradiction. So, no f can be uniformly convex for any $\kappa < 2$

However, one can note that for $f(x) = \|x\|_{1.5}^{1.5} = \sum_i |x_i|^{1.5}$, we have $x_f^* = 0$, and $f(x) - f(x_f^*) = \|x\|_{1.5}^{1.5} \geq \|x - x_f^*\|_2^{1.5}$, hence $f \in \mathcal{F}^{1.5}$.

Lemma 6. *If $f \in \mathcal{F}^\kappa$, then for any subgradient $g_x \in \partial f(x)$, we have $\|g_x\|_2 \geq \lambda \|x - x^*\|_2^{\kappa - 1}$.*

Proof: By convexity, we have

$$f(x^*) \geq f(x) + g_x^\top (x^* - x)$$

Rearranging terms and since $f \in \mathcal{F}^\kappa$, we get

$$g_x^\top(x - x^*) \geq f(x) - f(x^*) \geq \lambda \|x - x^*\|_2^\kappa$$

By Holder's inequality,

$$\|g_x\|_2 \|x - x^*\|_2 \geq g_x^\top(x - x^*)$$

Putting them together, we have

$$\|g_x\|_2 \|x - x^*\|_2 \geq \lambda \|x - x^*\|_2^\kappa$$

giving us our result.

Lemma 7. *For a gaussian random variable z, $\forall t < \sigma, \quad \exists a_1, a_2, \quad a_1 t \leq P(0 \leq z \leq t) \leq a_2 t$*

Proof: We wish to characterize how the probability mass of a gaussian random variable grows just around its mean. Our claim is that it grows linearly with the distance from the mean, and the following simple argument argues this neatly.

Consider a $X \sim N(0, \sigma^2)$ random variable at a distance t from the mean 0. We want to bound $\int_{-t}^{t} d\mu(X)$ for very small t. The key idea in bounding this integral is to approximate it by a smaller and larger rectangle, each of the rectangles having a width $2t$ (from $-t$ to t).

The first one has a height equal to $\frac{e^{-t^2/2\sigma^2}}{\sigma\sqrt{2\pi}}$, the smallest value taken by the gaussian in $[-t, t]$ achieved at t, and the other with a height equal to the $\frac{1}{\sigma\sqrt{2\pi}}$, the largest value of the gaussian in $[-t, t]$ achieved at 1.

The smaller rectangle has area $2t\frac{e^{-t^2/2\sigma^2}}{\sigma\sqrt{2\pi}} \geq 2t\frac{e^{-1/2}}{\sigma\sqrt{2\pi}}$ when $t < \sigma$. The larger rectangle clearly has an area of $2t\frac{1}{\sigma\sqrt{2\pi}}$.

Hence we have $A_1 t = 2t\frac{1}{\sigma\sqrt{2\pi e}} \leq P(|x| < t) \leq 2t\frac{1}{\sigma\sqrt{2\pi}} = A_2 t$ for $t < \sigma$. Similarly, for a one-sided inequality, we have $a_1 t = t\frac{1}{\sigma\sqrt{2\pi e}} \leq P(0 < X < t) \leq t\frac{1}{\sigma\sqrt{2\pi}} = a_2 t$ for $t < \sigma$.

We note that the gaussian tail inequality $P(X > t) \leq \frac{1}{t}e^{-t^2/2\sigma^2}$ really makes sense for large $t > \sigma$ and we are interested in $t < \sigma$. There are tighter inequalities, but for our purpose, this will suffice.

Lemma 8 ([34])**.** *If $|\eta(x) - 1/2| \geq \lambda$, the midpoint $\hat{x}_T$ of the high-probability interval returned by BZ satisfies $\mathbb{E}|\hat{x}_T - x^*| = O(e^{-T\lambda^2/2})$.*

Proof: The BZ algorithm works by dividing $[0, 1]$ into a grid of m points (interval size $1/m$) and makes T queries (only at gridpoints) to return an interval $\hat{I}_T$ such that $\Pr(x^* \notin \hat{I}_T) \leq me^{-T\lambda^2}$ [34]. We choose $\hat{x}_T$ to be the midpoint of this interval, and hence get

$$\begin{aligned}
\mathbb{E}|\hat{x}_T - x^*| &= \int_0^1 \Pr(|\hat{x}_T - x^*| > u) du \\
&= \int_0^{1/2m} \Pr(|\hat{x}_T - x^*| > u) du \\
&\quad + \int_{1/2m}^{1} \Pr(|\hat{x}_T - x^*| > u) du \\
&\leq \frac{1}{2m} + \left(1 - \frac{1}{2m}\right) \Pr\left(|\hat{x}_T - x^*| > \frac{1}{2m}\right) \\
&\leq \frac{1}{2m} + me^{-T\lambda^2} = O\left(e^{-T\lambda^2/2}\right)
\end{aligned}$$

for the choice of the number of gridpoints as $m = e^{T\lambda^2/2}$.

Lemma 9. *If $|\eta(x) - 1/2| \geq \lambda|x - x^*|^\kappa$, the point $\hat{x}_T$ obtained from a modified version of BZ satisfies $\mathbb{E}|\hat{x}_T - x^*| = O\left((\frac{\log T}{T})^{\frac{1}{2\kappa-2}}\right)$ and $\mathbb{E}[|\hat{x}_T - x^*|^\kappa] = O\left((\frac{\log T}{T})^{\frac{\kappa}{2\kappa-2}}\right)$.*

Proof: We again follow the same proof as in [34]. Initially, they assume that the grid points are not aligned with x^*, ie $\forall k \in \{0, ..., m\}, \quad |x^* - k/m| \geq 1/3m$. This implies that for all gridpoints x, $|\eta(x) - 1/2| \geq \lambda(1/3m)^{\kappa-1}$. Following the exact same proof above,

$$\mathbb{E}[|\hat{x}_T - x^*|^\kappa] = \int_0^1 \Pr(|\hat{x}_T - x^*|^\kappa > u)du$$

$$\begin{aligned} &= \int_0^{(1/2m)^\kappa} \Pr(|\hat{x}_T - x^*| > u^{1/\kappa})du \\ &\quad + \int_{(1/2m)^\kappa}^1 \Pr(|\hat{x}_T - x^*| > u^{1/\kappa})du \\ &\leq \left(\frac{1}{2m}\right)^\kappa + \left(1 - \left(\frac{1}{2m}\right)^\kappa\right) \Pr\left(|\hat{x}_T - x^*| > \frac{1}{2m}\right) \\ &\leq \left(\frac{1}{2m}\right)^\kappa + m\exp(-T\lambda^2(1/3m)^{2\kappa-2}) \\ &= O\left(\left(\frac{T}{\log T}\right)^{\frac{1}{2\kappa-2}}\right) \end{aligned}$$

on choosing m proportional to $\left(\frac{T}{\log T}\right)^{\frac{1}{2\kappa-2}}$.

[34] elaborate in detail how to avoid the assumption that the grid points don't align with x^*. They use a more complicated variant of BZ with three interlocked grids, and gets the same rate as above without that assumption. The reader is directed to their exposition for clarification.

Lemma 10. *$c_\kappa\|x\|_\kappa^\kappa = c_\kappa \sum_{i=1}^d |x_i|^\kappa =: f_0(x) \in \mathcal{F}^\kappa$, for all $\kappa > 1$. Also, $f_1(x)$ as defined in Section 2.3 is also in $\mathcal{F}^\kappa$.*

Proof: Firstly, this is clearly convex for $\kappa > 1$. Also, $f_0(x^*_{f_0}) = 0$ at $x^*_{f_0} = 0$. So, all we need to show is that for appropriate choice of c_κ, f is indeed 1-Lipschitz and that $f_0(x) - f_0(x^*_{f_0}) \geq \lambda\|x - x^*_{f_0}\|_2^\kappa$ for some $\lambda > 0$, ie

$$c_\kappa\|x\|_\kappa^\kappa \geq \lambda\|x\|_2^\kappa \quad , \quad c_\kappa(\|x\|_\kappa^\kappa - \|y\|_\kappa^\kappa) \leq \|x - y\|_2$$

Let us consider two cases, $\kappa \geq 2$ and $\kappa < 2$. Note that all norms are uniformly bounded with respect to each other, upto constants depending on d. Precisely, if $\kappa < 2$, then $\|x\|_\kappa > \|x\|_2$ and if $\kappa \geq 2$, then $\|x\|_\kappa \geq d^{1/\kappa - 1/2}\|x\|_2$.

When $\kappa \geq 2$, consider $c_\kappa = 1$. Then

$$(\|x\|_\kappa^\kappa - \|y\|_\kappa^\kappa) \leq \|x - y\|_\kappa^\kappa \leq \|x - y\|_2^\kappa \leq \|x - y\|_2$$

because $\|z\|_\kappa \leq \|z\|_2$ and $\|x - y\| \leq 1$. Also, $\|x\|_\kappa^\kappa \geq d^{1-\frac{\kappa}{2}}\|x\|_2^\kappa$, so $\lambda = d^{1-\frac{\kappa}{2}}$ works.

When $\kappa < 2$, consider $c_\kappa = \frac{1}{\sqrt{d}^\kappa}$. Similarly

$$c_\kappa(\|x\|_\kappa^\kappa - \|y\|_\kappa^\kappa) \leq \left(\frac{\|x - y\|_\kappa}{\sqrt{d}}\right)^\kappa \leq \|x - y\|_2^\kappa \leq \|x - y\|_2$$

Also $c_\kappa\|x\|_\kappa^\kappa \geq c_\kappa\|x\|_2^\kappa$, so $\lambda = c_\kappa$ works.

Hence $f_0(x)$ is 1-Lipschitz and in $\mathcal{F}^\kappa$ for appropriate c_κ.

Now, look at $f_1(x)$ for $x_1 \leq 4a$. It is actually just $f_0(x)$, but translated by $2a$ in direction x_1, with a constant added, and hence has the same growth around its minimum. Now, the part with $x_1 > 4a$ is just $f_0(x)$ itself, which have the same growth parameters as the part with $x_1 \leq 4a$. So $f_1(x) \in \mathcal{F}^\kappa$ also.

Lemma 11. *For all $i = 1...d$, let $f_i(x)$ be any one-dimensional κ-uniformly convex function ($\kappa \geq 2$) with constant λ_i. For a $d-$dimensional function $f(x) = \sum_{i=1}^d f_i(x_i)$ that decomposes over dimensions, $f(x)$ is also κ-uniformly convex with constant $\lambda = \frac{\min_i \lambda_i}{d^{1/2-1/\kappa}}$.*

Proof:

$$\begin{aligned}
f(x+h) &= \sum_i f_i(x_i + h_i) \\
\geq &\quad \sum_i (f_i(x_i) + g_{x_i}h_i + \lambda_i|h_i|^\kappa) \\
\geq &\quad f(x) + g_x^\top h + (\min_i \lambda_i)\|h\|_\kappa^\kappa \\
\geq &\quad f(x) + g_x^\top h + \frac{(\min_i \lambda_i)}{d^{1/2-1/\kappa}}\|h\|_2^\kappa
\end{aligned}$$

(one can use $h = y - x$ for the usual first-order definition)

Lemma 12. *$f(x) = |x|^k$ is κ-uniformly convex i.e.*

$$tf(x) + (1-t)f(y) \geq f(tx + (1-t)y) + \frac{\lambda}{2}t(1-t)|x-y|^k$$

for $\lambda = 4/2^k$. Lemma 11 implies $\|x\|_\kappa^\kappa$ is also κ-uniformly convex with $\lambda = \frac{4/2^k}{d^{1/2-1/\kappa}}$.

Proof: First we will show this for the special case of $t = 1/2$. We need to argue that:

$$\frac{1}{2}|x|^k + \frac{1}{2}|y|^k \geq |\frac{x+y}{2}|^k + \lambda\frac{1}{8}|x-y|^k$$

Let $\lambda = 4/2^k$. We will prove a stronger claim -

$$\frac{1}{2}|x|^k + \frac{1}{2}|y|^k \geq |\frac{x+y}{2}|^k + 2\lambda\frac{1}{8}|x-y|^k$$

Since $k \geq 2$

$$\begin{aligned}
RHS^{1/k} &= (|\frac{x+y}{2}|^k + |\frac{x-y}{2}|^k)^{1/k} \\
&\leq (|\frac{x+y}{2}|^2 + |\frac{x-y}{2}|^2)^{1/2} \\
&\leq (|x|^2/2 + |y|^2/2)^{1/2} \\
&\leq \frac{1}{\sqrt{2}}2^{1/2-1/k}(|x|^k + |y|^k)^{1/k} \\
&\leq \left(\frac{1}{2}|x|^k + \frac{1}{2}|y|^k\right)^{1/k} = LHS^{1/k}
\end{aligned}$$

Now, for the general case. We will argue that just proving the above for $t = 1/2$ is actually sufficient.

$$\begin{aligned} f(tx + (1-t)y) = f\left(2t\left(\frac{x+y}{2}\right) + (1-2t)y\right) & \\ \leq \quad & 2tf\left(\frac{x+y}{2}\right) + (1-2t)f(y) \\ \leq \quad & tf(x) + tf(y) - 2t\frac{2\lambda}{8}|x-y|^k + (1-2t)f(y) \\ \leq \quad & tf(x) + (1-t)f(y) - t(1-t)\frac{\lambda}{2}|x-y|^k \end{aligned}$$

Chapter 3

Active Learning : Algorithms from Optimization

Interesting theoretical associations have been established by recent papers between the fields of active learning and stochastic convex optimization due to the common role of feedback in sequential querying mechanisms. In this chapter, we continue this thread in two parts by exploiting these relations for the first time to yield novel algorithms in both fields, further motivating the study of their intersection. First, inspired by a recent optimization algorithm that was adaptive to unknown uniform convexity parameters, we present a new active learning algorithm for one-dimensional thresholds that can yield minimax rates by adapting to unknown noise parameters. Next, we show that one can perform d-dimensional stochastic minimization of smooth uniformly convex functions when only granted oracle access to noisy gradient signs along any coordinate instead of real-valued gradients, by using a simple randomized coordinate descent procedure where each line search can be solved by 1-dimensional active learning, provably achieving the same error convergence rate as having the entire real-valued gradient. Combining these two parts yields an algorithm that solves stochastic convex optimization of uniformly convex and smooth functions using only noisy gradient signs by repeatedly performing active learning, achieves optimal rates and is adaptive to all unknown convexity and smoothness parameters.

3.1 Introduction

The two fields of convex optimization and active learning seem to have evolved quite independently of each other. Recently, [155] pointed out their relatedness due to the inherent sequential nature of both fields and the complex role of feedback in taking future actions. Following that, [160] made the connections more explicit by tying together the exponent used in noise conditions in active learning and the exponent used in uniform convexity (UC) in optimization. They used this to establish lower bounds (and tight upper bounds) in stochastic optimization of UC functions based on proof techniques from active learning. However, it was unclear if there were concrete algorithmic ideas in common between the fields.

Here, we provide a positive answer by exploiting the aforementioned connections to form new and interesting algorithms that clearly demonstrate that the complexity of d-dimensional stochastic optimization is precisely the complexity of 1-dimensional active learning. Inspired by an optimization algorithm that was adaptive to unknown uniform convexity parameters, we design an interesting one-dimensional active learner that is also adaptive to unknown noise parameters. This algorithm is simpler than the adaptive active learning algorithm proposed recently in [92] which handles the pool based active learning setting.

Given access to this active learner as a subroutine for line search, we show that a simple randomized coordinate descent procedure can minimize uniformly convex functions with a much simpler stochastic oracle that returns only a Bernoulli random variable representing a noisy sign of the gradient in a single coordinate direction, rather than a full-dimensional real-valued gradient vector. The resulting algorithm is adaptive to all unknown UC and smoothness parameters and achieve minimax optimal convergence rates.

We spend the first two sections describing the problem setup and preliminary insights, before describing our algorithms.

3.1.1 Setup of First-Order Stochastic Convex Optimization

First-order stochastic convex optimization is the task of approximately minimizing a convex function over a convex set, given oracle access to unbiased estimates of the function and gradient at any point, using as few queries as possible ([146]).

We will assume that we are given an arbitrary set $S \subset \mathbb{R}^d$ of known diameter bound $R = \max_{x,y \in S} \|x - y\|$. A convex function f with $x^* = \arg\min_{x \in S} f(x)$ is said to be k-uniformly convex if, for some $\lambda > 0, k \geq 2$, we have for all $x, y \in S$

$$f(y) \geq f(x) + \nabla f(x)^\top (y - x) + \frac{\lambda}{2}\|x - y\|^k$$

(strong convexity arises when $k = 2$). f is L-Lipschitz for some $L > 0$ if $\|\nabla f(x)\|_* \leq L$ (where $\|.\|_*$ is the dual norm of $\|.\|$); equivalently for all $x, y \in S$

$$|f(x) - f(y)| \leq L\|x - y\|$$

A differentiable f is H-strongly smooth (or has a H-Lipschitz gradient) for some $H > \lambda$ if for all $x, y \in S$, we have $\|\nabla f(x) - \nabla f(y)\|_* \leq H\|x - y\|$, or equivalently

$$f(y) \leq f(x) + \nabla f(x)^\top (y - x) + \frac{H}{2}\|x - y\|^2$$

In this chapter we shall always assume $\|.\| = \|.\|_* = \|.\|_2$ and deal with strongly smooth and uniformly convex functions with parameters $\lambda > 0, k \geq 2, L, H > 0$.
A stochastic first order oracle is a function that accepts $x \in S$, and returns

$$\Big(\hat{f}(x), \hat{g}(x)\Big) \in \mathbb{R}^{d+1} \text{ where } \mathbb{E}\big[\hat{f}(x)\big] = f(x), \mathbb{E}\big[\hat{g}(x)\big] = \nabla f(x)$$

(these unbiased estimates also have bounded variance) and the expectation is over any internal randomness of the oracle.
An optimization algorithm is a method that sequentially queries an oracle at points in S and returns $\hat{x}_T$ as an estimate of the optimum of f after T queries (or alternatively tries to achieve an error of ϵ) and their performance can be measured by either function error $f(\hat{x}_T) - f(x^*)$ or point error $\|\hat{x}_T - x^*\|$.

3.1.2 Stochastic Gradient-Sign Oracles

Define a stochastic sign oracle to be a function of $x \in S, j \in \{1...d\}$, that returns

$$\hat{s}_j(x) \in \{+, -\} \text{ where}^1 \ \big|\eta(x) - 0.5\big| = \Theta\Big([\nabla f(x)]_j\Big) \text{ and } \eta(x) = \Pr\big(\hat{s}_j(x) = + | x\big)$$

[1] $f = \Theta(g)$ means $f = \Omega(g)$ and $f = O(g)$ (rate of growth)

where $\hat{s}_j(x)$ is a noisy $\mathrm{sign}\big([\nabla f(x)]_j\big)$ and $[\nabla f(x)]_j$ is the j-th coordinate of ∇f, and the probability is over any internal randomness of the oracle. This behavior of $\eta(x)$ actually needs to hold only when $\big|[\nabla f(x)]_j\big|$ is small.

In this chapter, we consider coordinate descent algorithms that are motivated by applications where computing the overall gradient, or even a function value, can be expensive due to high dimensionality or huge amounts of data, but computing the gradient in any one coordinate can be cheap. [147] mentions the example of $\min_x \frac{1}{2}\|Ax-b\|^2 + \frac{1}{2}\|x\|^2$ for some $n \times d$ matrix A (or any other regularization that decomposes over dimensions). Computing the gradient $A^\top(Ax-b)+x$ is expensive, because of the matrix-vector multiply. However, its j-th coordinate is $2A^{j\top}(Ax-b)+x_j$ and requires an expense of only n if the residual vector $Ax-b$ is kept track of (this is easy to do, since on a single coordinate update of x, the residual change is proportional to A^j, an additional expense of n).

A sign oracle is weaker than a first order oracle, and can actually be obtained by returning the sign of the first order oracle's noisy gradient if the mass of the noise distribution grows linearly around its zero mean (argued in next section). At the optimum along coordinate j, the oracle returns a ± 1 with equal probability, and otherwise returns the correct sign with a probability proportional to the value of the directional derivative at that point (this is reflective of the fact that the larger the derivative's absolute value, the easier it would be for the oracle to approximate its sign, hence the smaller the probability of error). It is not unreasonable that there may be other circumstances where even calculating the (real value) gradient in the i-th direction could be expensive, but estimating its sign could be a much easier task as it only requires estimating whether function values are expected to increase or decrease along a coordinate (in a similar spirit of function comparison oracles [109], but with slightly more power).

We will also see that the rates for optimization crucially depend on whether the gradient noise is sign-preserving or not. For instance, with rounding errors or storing floats with small precision, one can get deterministic rates as if we had the exact gradient since the rounding or lower precision doesn't flip signs.

3.1.3 Setup of Active Threshold Learning

The problem of one-dimensional threshold estimation assumes you have an interval of length R, say $[0, R]$. Given a point x, it has a label $y \in \{+,-\}$ that is drawn from an unknown conditional distribution $\eta(x) = \Pr\big(Y = +|X = x\big)$ and the threshold t is the unique point where $\eta(x) = 1/2$, with it being larger than half on one side of t and smaller than half on the other (hence it is more likely to draw a $+$ on one side of t and a $-$ on the other side).

The task of active learning of threshold classifiers allows the learner to sequentially query T (possibly dependent) points, observing labels drawn from the unknown conditional distribution after each query, with the goal of returning a guess $\hat{x}_T$ as close to t as possible. In the formal study of classification (cf. [217]), it is common to study minimax rates when the regression function $\eta(x)$ satisfies Tsybakov's noise or margin condition (TNC) with exponent k at the threshold t. Different versions of this boundary noise condition are used in regression, density or level-set estimation and lead to an improvement in minimax optimal rates (for classification, also cf. [10], [92]). Here, we present the version of TNC used in [34] :

$$M|x-t|^{k-1} \geq |\eta(x) - 1/2| \geq \mu|x-t|^{k-1} \text{ whenever}^2\ |\eta(x)-1/2| \leq \epsilon_0$$

for some constants $M > \mu > 0, \epsilon_0 > 0, k \geq 1$.

A standard measure for how well a classifier h performs is given by its risk, which is simply the probability of classification error (expectation under $0-1$ loss), $\mathcal{R}(h) = \Pr\big[h(x) \neq y\big]$. The performance

[2] Note that $|x-t| \leq \delta_0 := \big(\frac{\epsilon_0}{M}\big)^{\frac{1}{k-1}} \implies |\eta(x)-1/2| \leq \epsilon_0 \implies |x-t| \leq \left(\frac{\epsilon_0}{\mu}\right)^{\frac{1}{k-1}}$

of threshold learning strategies can be measured by the excess classification risk of the resultant threshold classifier at $\hat{x}_T$ compared to the Bayes optimal classifier at t as given by [3]

$$\mathcal{R}(\hat{x}_T) - \mathcal{R}(t) = \int_{\hat{x}_T \wedge t}^{\hat{x}_T \vee t} |2\eta(x) - 1| dx \tag{3.1}$$

In the above expression, akin to [34], we use a uniform marginal distribution for active learning since there is no underlying distribution over x. Alternatively, one can simply measure the one-dimensional point error $|\hat{x}_T - t|$ in estimation of the threshold. Minimax rates for estimation of risk and point error in active learning under TNC were provided in [34] and are summarized in the next section.

3.1.4 Summary of Contributions

Now that we have introduced the notation used in our chapter and some relevant previous work (more in the next section), we can clearly state our contributions.

- We generalize an idea from [107] to present a simple epoch-based active learning algorithm with a passive learning subroutine that can optimally learn one-dimensional thresholds and is adaptive to unknown noise parameters.
- We show that noisy gradient signs suffice for minimization of uniformly convex functions by proving that a random coordinate descent algorithm with an active learning line-search subroutine achieves minimax convergence rates.
- Due to the connection between the relevant exponents in the two fields, we can combine the above two methods to get an algorithm that achieves minimax optimal rates and is adaptive to unknown convexity parameters.
- As a corollary, we argue that with access to possibly noisy non-exact gradients that don't switch any signs (rounding errors or low-precision storage are sign-preserving), we can still achieve exponentially fast deterministic rates.

3.2 Preliminary Insights

3.2.1 Connections Between Exponents

Taking one point as x^* in the definition of UC, we see that

$$|f(x) - f(x^*)| \geq \frac{\lambda}{2} \|x - x^*\|^k$$

Since $\|\nabla f(x)\| \|x - x^*\| \geq \nabla f(x)^\top (x - x^*) \geq f(x) - f(x^*)$ (by convexity),

$$\|\nabla f(x) - 0\| \geq \frac{\lambda}{2} \|x - x^*\|^{k-1}$$

Another relevant fact for us will be that uniformly convex functions in d dimensions are uniformly convex along any one direction, or in other words, for every fixed $x \in S$ and fixed unit vector $u \in \mathbb{R}^d$, the

[3] $a \vee b := \max(a, b)$ and $a \wedge b := \min(a, b)$

univariate function of α defined by $f_{x,u}(\alpha) := f(x+\alpha u)$ is also UC with the same parameters[4]. For $u = e_j$,

$$\big|[\nabla f(x)]_j - 0\big| \geq \frac{\lambda}{2}\|x - x_j^*\|^{k-1}$$

where $x_j^* = x + \alpha_j^* e_j$ and $\alpha_j^* = \arg\min_{\{\alpha | x+\alpha e_j \in S\}} f(x+\alpha e_j)$. This uncanny similarity to the TNC (since $\nabla f(x^*) = 0$) was mathematically exploited in [160] where the authors used a lower bounding proof technique for one-dimensional active threshold learning from [34] to provide a new lower bounding proof technique for the d-dimensional stochastic convex optimization of UC functions. In particular, they showed that the minimax rate for 1-dimensional active learning excess risk and the d-dimensional optimization function error both scaled like[5] $\tilde{\Theta}\left(T^{-\frac{k}{2k-2}}\right)$, and that the point error in both settings scaled like $\tilde{\Theta}\left(T^{-\frac{1}{2k-2}}\right)$, where k is either the TNC exponent or the UC exponent, depending on the setting. The importance of this connection cannot be emphasized enough and we will see this being useful throughout this chapter.

As mentioned earlier [34] require a two-sided TNC condition (upper and lower growth condition to provide exact tight rate of growth) in order to prove risk upper bounds. On a similar note, for uniformly convex functions, we will assume such a Local k-Strong Smoothness condition around directional minima

$$\textbf{Assumption LkSS}: \quad \text{for all } j \in \{1...d\} \quad \big|[\nabla f(x)]_j - 0\big| \leq \Lambda\|x - x_j^*\|^{k-1}$$

for some constant $\Lambda > \lambda/2$, so we can tightly characterize the rate of growth as

$$\big|[\nabla f(x)]_j - 0\big| = \Theta\Big(\|x - x_j^*\|^{k-1}\Big)$$

This condition is implied by strong smoothness or Lipschitz smooth gradients when $k = 2$ (for strongly convex and strongly smooth functions), but is a slightly stronger assumption otherwise.

3.2.2 The One-Dimensional Argument

The basic argument for relating optimization to active learning was made in [160] in the context of stochastic first order oracles when the noise distribution $\mathrm{P}(z)$ is unbiased and grows linearly around its zero mean, i.e.

$$\int_0^\infty d\mathrm{P}(z) = \tfrac{1}{2} \text{ and } \int_0^t d\mathrm{P}(z) = \Theta(t)$$

for all $0 < t < t_0$, for constants t_0 (similarly for $-t_0 < t < 0$). This is satisfied for gaussian, uniform and many other distributions. We reproduce the argument for clarity and then sketch it for stochastic signed oracles as well.

For any $x \in S$, it is clear that $f_{x,j}(\alpha) := f(x+\alpha e_j)$ is convex; its gradient $\nabla f_{x,j}(\alpha) := [\nabla f(x+\alpha e_j)]_j$ is an increasing function of α that switches signs at $\alpha_j^* := \arg\min_{\{\alpha | x+\alpha e_j \in S\}} f_{x,j}(\alpha)$, or equivalently at directional minimum $x_j^* := x + \alpha_j^* e_j$. One can think of sign$([\nabla f(x)]_j)$ as being the true label of x, sign$([\nabla f(x)]_j + z)$ as being the observed label, and finding x_j^* as learning the decision boundary (point where labels switch signs). Define regression function

$$\eta(x) := \Pr\Big(\text{sign}([\nabla f(x)]_j + z) = + | x\Big)$$

[4] Since f is UC, $f_{x,u}(\alpha) \geq f_{x,u}(0) + \alpha\nabla f_{x,u}(0) + \frac{\lambda}{2}|\alpha|^k$

[5] we use $\tilde{\mathrm{O}}, \tilde{\Theta}$ to hide constants and polylogarithmic factors

and note that minimizing $f_{x_0,j}$ corresponds to identifying the Bayes threshold classifier as x_j^* because the point at which $\eta(x) = 0.5$ or $[\nabla f(x)]_j = 0$ is x_j^*. Consider a point $x = x_j^* + te_j$ for $t > 0$ with $[\nabla f(x)]_j > 0$ and hence has true label $+$ (a similar argument can be made for $t < 0$). As discussed earlier, $\big|[\nabla f(x)]_j\big| = \Theta\Big(\|x - x_j^*\|^{k-1}\Big) = \Theta(t^{k-1})$. The probability of seeing label $+$ is the probability that we draw z in $\big(-[\nabla f(x)]_j, \infty\big)$ so that the sign of $[\nabla f(x)]_j + z$ is still positive. Hence, the regression function can be written as

$$\begin{aligned}\eta(x) &= \Pr\Big([\nabla f(x)]_j + z > 0\Big)\\ &= \Pr(z > 0) + \Pr\Big(-[\nabla f(x)]_j < z < 0\Big) \;=\; 0.5 + \Theta\Big([\nabla f(x)]_j\Big)\end{aligned}$$

$$\implies \big|\eta(x) - \tfrac{1}{2}\big| \;=\; \Theta\Big([\nabla f(x)]_j\Big) \;=\; \Theta\big(t^{k-1}\big) \;=\; \Theta\Big(|x - x_j^*|^{k-1}\Big)$$

Hence, $\eta(x)$ satisfies the TNC with exponent k, and an active learning algorithm (next subsection) can be used to obtain a point $\hat{x}_T$ with small point-error and excess risk. Note that function error in convex optimization is bounded above by excess risk of the corresponding active learner using eq (3.1) because

$$\begin{aligned}f_j(\hat{x}_T) - f_j(x_j^*) &= \left|\int_{\hat{x}_T \wedge x_j^*}^{\hat{x}_T \vee x_j^*} [\nabla f(x)]_j \mathrm{dx}\right| \qquad = \Theta\left(\int_{\hat{x}_T \wedge x_j^*}^{\hat{x}_T \vee x_j^*} |2\eta(x) - 1| \mathrm{dx}\right)\\ &= \Theta\Big(\mathcal{R}(\hat{x}_T)\Big)\end{aligned}$$

Similarly, for stochastic sign oracles (Sec. 3.1.2), using $\eta(x) = \Pr\big(\hat{s}_j(x) = +\big)$,

$$\big|\eta(x) - \tfrac{1}{2}\big| \;=\; \Theta\Big([\nabla f(x)]_j\Big) \;=\; \Theta\Big(\|x - x_j^*\|^{k-1}\Big)$$

3.2.3 A Non-adaptive Active Threshold Learning Algorithm

One can use a grid-based probabilistic variant of binary search called the BZ algorithm [28] to approximately learn the threshold efficiently in the active setting, in the setting that $\eta(x)$ satisfies the TNC for known k, μ, M (it is not adaptive to the parameters of the problem - one needs to know these constants beforehand). The analysis of BZ and the proof of the following lemma are discussed in detail in Theorem 1 of [33], Theorem 2 of [34] and the Appendix of [160].

Lemma 13. *Given a 1-dimensional regression function that satisfies the TNC with known parameters μ, k, then after T queries, the BZ algorithm returns a point $\hat{t}$ such that $|\hat{t} - t| = \tilde{\Theta}(T^{-\frac{1}{2k-2}})$ and the excess risk is $\tilde{\Theta}(T^{-\frac{k}{2k-2}})$.*

Due to the described connection between exponents, one can use BZ to approximately optimize a one dimensional uniformly convex function f_j with known uniform convexity parameters λ, k. Hence, the BZ algorithm can be used to find a point with low function error by searching for a point with low risk. This, when combined with Lemma 13, yields the following important result.

Lemma 14. *Given a 1-dimensional k-UC and LkSS function f_j, a line search to find $\hat{x}_T$ close to x_j^* up to accuracy $|\hat{x}_T - x_j^*| \leq \eta$ in point-error can be performed in $\tilde{\Theta}(1/\eta^{2k-2})$ steps using the BZ algorithm. Alternatively, in T steps we can find $\hat{x}_T$ such that $f(\hat{x}_T) - f(x_j^*) = \tilde{\Theta}(T^{-\frac{k}{2k-2}})$.*

3.3 A 1-D Adaptive Active Threshold Learning Algorithm

We now describe an algorithm for active learning of one-dimensional thresholds that is adaptive, meaning it can achieve the minimax optimal rate even if the TNC parameters M, μ, k are unknown. It is quite different from the non-adaptive BZ algorithm in its flavour, though it can be regarded as a robust binary search procedure, and its design and proof are inspired from an optimization procedure from [107] that is adaptive to unknown UC parameters λ, k.

Even though [107] considers a specific optimization algorithm (dual averaging), we observe that their algorithm that adapts to unknown UC parameters can use any optimal convex optimization algorithm as a subroutine within each epoch. Similarly, our adaptive active learning algorithm is epoch-based and can use any optimal passive learning subroutine in each epoch. We note that [92] also developed an adaptive algorithm based on disagreement coefficient and VC-dimension arguments, but it is in a pool-based setting where one has access to a large pool of unlabeled data, and is much more complicated.

3.3.1 An Optimal Passive Learning Subroutine

The excess risk of passive learning procedures for 1-d thresholds can be bounded by $\mathrm{O}(T^{-1/2})$ (e.g. see Alexander's inequality in [56] to avoid $\sqrt{\log T}$ factors from ERM/VC arguments) and can be achieved by ignoring the TNC parameters.

Consider such a passive learning procedure under a uniform distribution of samples (mimicked by active learning by querying the domain uniformly) in a ball[6] $B(x_0, R)$ around an arbitrary point x_0 of radius R that is known to contain the true threshold t. Then without knowledge of M, μ, k, in T steps we can get a point $\hat{x}_T$ close to the true threshold t such that with probability at least $1 - \delta$

$$\mathcal{R}(\hat{x}) - \mathcal{R}(t) = \int\limits_{\hat{x}_T \vee t}^{\hat{x}_T \wedge t} |2\eta(x) - 1| dx \leq \frac{C_\delta R}{\sqrt{T}}$$

for some constant C_δ. Assuming $\hat{x}_T$ lies inside the TNC region,

$$\mu \int\limits_{\hat{x}_T \vee t}^{\hat{x}_T \wedge t} |x - t|^{k-1} dx \leq \int\limits_{\hat{x}_T \vee t}^{\hat{x}_T \wedge t} |2\eta(x) - 1| dx$$

Hence $\frac{\mu |\hat{x}_T - t|^k}{k} \leq \frac{C_\delta R}{\sqrt{T}}$. Since $k^{1/k} \leq 2$, w.p. at least $1 - \delta$ we get a point-error

$$|\hat{x}_T - t| \leq 2 \left[\frac{C_\delta R}{\mu \sqrt{T}} \right]^{1/k} \tag{3.2}$$

We assume that $\hat{x}_T$ lies within the TNC region since the interval $|\eta(x) - \frac{1}{2}| \leq \epsilon_0$ has at least constant width $|x - t| \leq \delta_0 = (\epsilon_0 / M)^{1/(k-1)}$, it will only take a constant number of iterations to find a point within it. A formal way to argue this would be to see that if the overall risk goes to zero like $\frac{C_\delta R}{\sqrt{T}}$, then the point cannot stay outside this constant sized region of width δ_0 where $|\eta(x) - 1/2| \leq \epsilon_0$, since it would accumulate a large constant risk of at least $\int\limits_t^{t+\delta_0} \mu |x - t|^{k-1} = \frac{\mu \delta_0^k}{k}$. So as long as T is larger than a constant $T_0 := \frac{C_\delta^2 R^2 k^2}{\mu^2 \delta_0^{2k}}$, our bound in eq 3.2 holds with high probability (we can even assume we waste a constant number of queries to just get into the TNC region before using this algorithm).

[6] Define $B(x, R) := [x - R, x + R]$

3.3.2 Adaptive One-Dimensional Active Threshold Learner

Algorithm 2 Adaptive Threshold Learner

```
Input: Domain $S$ of diameter $R$, oracle budget $T$, confidence $\delta$
Black Box: Any optimal passive learning procedure $P(x, R, N)$ that outputs an estimated threshold in $B(x, R)$ using $N$ queries
Choose any $x_0 \in S$, $R_1 = R, E = \log \sqrt{\frac{2T}{C_{\tilde{\delta}}^2 \log T}}, N = \frac{T}{E}$
while $1 \leq e \leq E$ do
    $x_e \leftarrow P(x_{e-1}, R_e, N)$
    $R_{e+1} \leftarrow \frac{R_e}{2}, e \leftarrow e + 1$
end while
Output: $x_E$
```

Algorithm 3.3.2 is a generalized epoch-based binary search, and we repeatedly perform passive learning in a halving search radius. Let the number of epochs be $E := \log \sqrt{\frac{2T}{C_{\tilde{\delta}}^2 \log T}} \leq \frac{\log T}{2}$ (if[7] constant $C_{\tilde{\delta}}^2 > 2$) and $\tilde{\delta} := 2\delta / \log T \leq \delta / E$. Let the time budget per epoch be $N := T/E$ (the same for every epoch) and the search radius in epoch $e \in \{1, ..., E\}$ shrink as $R_e := 2^{-e+1}R$.

Let us define the minimizer of the risk within the ball of radius R_e centered around x_{e-1} at epoch e as

$$x_e^* = \arg\min \left\{ \mathcal{R}(x) : x \in S \cap B(x_{e-1}, R_e) \right\}$$

Note that $x_e^* = t$ iff $t \in B(x_{e-1}, R_e)$ and will be one end of the interval otherwise.

Theorem 1. *In the setting of one-dimensional active learning of thresholds, Algorithm 1 adaptively achieves $\mathcal{R}(x_E) - \mathcal{R}(t) = \tilde{\mathrm{O}}\left(T^{-\frac{k}{2k-2}}\right)$ with probability at least $1 - \delta$ in T queries when the unknown regression function $\eta(x)$ has unknown TNC parameters μ, k.*

Proof: Since we use an optimal passive learning subroutine at every epoch, we know that after each epoch e we have with probability at least $1 - \tilde{\delta}$ [7]

$$\mathcal{R}(x_e) - \mathcal{R}(x_e^*) \leq \frac{C_{\tilde{\delta}} R_e}{\sqrt{T/E}} \leq C_{\tilde{\delta}} R_e \sqrt{\frac{\log T}{2T}} \tag{3.3}$$

Since $\eta(x)$ satisfies the TNC (and is bounded above by 1), we have for all x

$$\mu |x - t|^{k-1} \leq |\eta(x) - 1/2| \leq 1$$

If the set has diameter R, one of the endpoints must be at least $R/2$ away from t, and hence we get a limitation on the maximum value of μ as $\mu \leq \frac{1}{(R/2)^{k-1}}$. Since $k \geq 2$ and $E \geq 2$, and $2^{-E} = C_{\tilde{\delta}} \sqrt{\frac{\log T}{2T}}$, using simple algebra we get

$$\mu \leq \frac{2^{(k-2)E+2}}{(R/2)^{k-1}} = \frac{4.2^{-E} 2^{(k-1)E} 2^{(k-1)}}{R^{k-1}} = \frac{4.2^{-E} 2^{(k-1)}}{(2^{-E}R)^{k-1}} = \frac{4 C_{\tilde{\delta}} 2^{k-1}}{R_{E+1}^{k-1}} \sqrt{\frac{\log T}{2T}}$$

[7] By VC theory for threshold classifiers or similar arguments in [56], $C_{\tilde{\delta}}^2 \sim \log(1/\tilde{\delta}) \sim \log \log T$ since $\tilde{\delta} \sim \delta / \log T$. We treat it as constant for clarity of exposition, but actually lose $\log \log T$ factors like the high probability arguments in [95] and [160]

We prove that we will be appropriately close to t after some epoch e^* by doing case analysis on μ. When the true unknown μ is sufficiently small, i.e.

$$\mu \leq \frac{4C_{\tilde{\delta}}2^{k-1}}{R_2^{k-1}}\sqrt{\frac{\log T}{2T}} \tag{3.4}$$

then we show that we'll be done after $e^* = 1$. Otherwise, we will be done after epoch $2 \leq e^* \leq E$ if the true μ lies in the range

$$\frac{4C_{\tilde{\delta}}2^{k-1}}{R_{e^*}^{k-1}}\sqrt{\frac{\log T}{2T}} \leq \mu \leq \frac{4C_{\tilde{\delta}}2^{k-1}}{R_{e^*+1}^{k-1}}\sqrt{\frac{\log T}{2T}} \tag{3.5}$$

To see why we'll be done, equations (3.4) and (3.5) imply $R_{e^*+1} \leq 2\left(\frac{8C_{\tilde{\delta}}^2 \log T}{\mu^2 T}\right)^{\frac{1}{2k-2}}$ after epoch e^* and plugging this into equation (3.3) with $R_{e^*} = 2R_{e^*+1}$, we get

$$\mathcal{R}(x_{e^*}) - \mathcal{R}(x_{e^*}^*) \leq C_{\tilde{\delta}} R_{e^*}\left(\frac{\log T}{2T}\right)^{\frac{1}{2}} = \mathrm{O}\left(\left(\frac{\log T}{T}\right)^{\frac{k}{2k-2}}\right) \tag{3.6}$$

There are two issues hindering the completion of our proof. The first is that even though $x_1^* = t$ to start off with, it might be the case that $x_{e^*}^*$ is far away from t since we are chopping the radius by half at every epoch. Interestingly, in lemma 15 we will prove that round e^* is the last round up to which $x_e^* = t$. This would imply from eq (3.6) that

$$\mathcal{R}(x_{e^*}) - \mathcal{R}(t) = \tilde{\mathrm{O}}\left(T^{-\frac{k}{2k-2}}\right) \tag{3.7}$$

Secondly we might be concerned that after the round e^*, we may move further away from t in later epochs. However, we will show that since the radii are decreasing geometrically by half at every epoch, we cannot really wander too far away from x_{e^*}. This will give us a bound (see lemma 16) like

$$\mathcal{R}(x_E) - \mathcal{R}(x_{e^*}) = \tilde{\mathrm{O}}\left(T^{-\frac{k}{2k-2}}\right) \tag{3.8}$$

We will essentially prove that the final point x_{e^*} of epoch e^* is sufficiently close to the true optimum t, and the final point of the algorithm x_E is sufficiently close to x_{e^*}. Summing eq (3.7) and eq (3.8) yields our desired result.

Lemma 15. *For all $e \leq e^*$, conditioned on having $x_{e-1}^* = t$, with probability $1 - \tilde{\delta}$ we have $x_e^* = t$. In other words, up to epoch e^*, the optimal classifier in the domain of each epoch is the true threshold with high probability.*

Proof: $x_e^* = t$ will hold in epoch e if the distance between the first point x_{e-1} in the epoch e is such that the ball of radius R_e around it actually contains t, or mathematically if $|x_{e-1} - t| \leq R_e$. This is trivially satified for $e = 1$, and assuming that it is true for epoch $e-1$ we will show show by induction that it holds true for epoch $e \leq e^*$ w.p. $1 - \tilde{\delta}$. Notice that using equation (3.2), conditioned on the induction going through in previous rounds (t being within the search radius), after the completion of round $e-1$ we have with probability $1 - \tilde{\delta}$

$$|x_{e-1} - t| \leq 2\left[\frac{C_{\tilde{\delta}} R_{e-1}}{\mu\sqrt{T/E}}\right]^{1/k}$$

If this was upper bounded by R_e, then the induction would go through. So what we would really like to show is that $2\left[\frac{C_{\tilde{\delta}} R_{e-1}}{\mu\sqrt{T/E}}\right]^{\frac{1}{k}} \leq R_e$. Since $R_{e-1} = 2R_e$, we effectively want to show $\frac{2^k C_{\tilde{\delta}} 2R_e}{\mu}\sqrt{\frac{E}{T}} \leq R_e^k$ or equivalently that for all $e \leq e^*$ we would like to have $\frac{4C_{\tilde{\delta}}2^{k-1}}{R_e^{k-1}}\sqrt{\frac{E}{T}} \leq \mu$. Since $E \leq \frac{\log T}{2}$, we would be achieving something stronger if we showed

$$\frac{4C_{\tilde{\delta}}2^{k-1}}{R_e^{k-1}}\sqrt{\frac{\log T}{2T}} \leq \mu$$

which is known to be true for every epoch up to e^* by equation (3.5).

Lemma 16. *For all* $e^* < e \leq E$, $\mathcal{R}(x_e) - \mathcal{R}(x_{e^*}) \leq \frac{C_{\tilde{\delta}}R_{e^*}}{\sqrt{T/E}} = \tilde{\mathrm{O}}\left(T^{-\frac{k}{2k-2}}\right)$ *w.p.* $1 - \tilde{\delta}$*, ie after epoch* e^**, we cannot deviate much from where we ended epoch* e^**.*

Proof: For $e > e^*$, we have with probability at least $1 - \tilde{\delta}$

$$\mathcal{R}(x_e) - \mathcal{R}(x_{e-1}) \leq \mathcal{R}(x_e) - \mathcal{R}(x_e^*) \leq \frac{C_{\tilde{\delta}}R_e}{\sqrt{T/E}}$$

and hence even for the final epoch E, we have with probability $(1 - \tilde{\delta})^{E-e^*}$

$$\mathcal{R}(x_E) - \mathcal{R}(x_{e^*}) = \sum_{e=e^*+1}^{E} [\mathcal{R}(x_e) - \mathcal{R}(x_{e-1})] \leq \sum_{e=e^*+1}^{E} \frac{C_{\tilde{\delta}}R_e}{\sqrt{T/E}}$$

Since the radii are halving in size, this is upper bounded (like equation (3.6)) by

$$\frac{C_{\tilde{\delta}}R_{e^*}}{\sqrt{T/E}}[1/2 + 1/4 + 1/8 + ...] \leq \frac{C_{\tilde{\delta}}R_{e^*}}{\sqrt{T/E}} = \tilde{\mathrm{O}}\left(T^{-\frac{k}{2k-2}}\right)$$

These lemmas justify the use of equations (3.7) and (3.8), whose sum yields our desired result. Notice that the overall probability of success is at least $(1 - \tilde{\delta})^E \geq 1 - \delta$, hence concluding the proof of the theorem.

3.4 Randomized Stochastic-Sign Coordinate Descent

We now describe an algorithm that can do stochastic optimization of k-UC and LkSS functions in $d > 1$ dimensions when given access to a stochastic sign oracle and a black-box 1-D active learning algorithm, such as our adaptive scheme from the previous section as a subroutine. The procedure is well-known in the literature, but the idea that one only needs noisy gradient signs to perform minimization optimally, and that one can use active learning as a line-search procedure, is novel to the best of our knowledge.

The idea is to simply perform random coordinate-wise descent with approximate line search, where the subroutine for line search is an optimal active threshold learning algorithm that is used to approach the minimum of the function along the chosen direction. Let the gradient at epoch e be called $\nabla_{e-1} = \nabla f(x_{e-1})$, the unit vector direction of descent d_e be a unit coordinate vector chosen randomly from $\{1...d\}$, and our step size from x_{e-1} be α_e (determined by active learning) so that our next point is $x_e := x_{e-1} + \alpha_e d_e$.

Assume, for analysis, that the optimum of $f_e(\alpha) := f(x_{e-1} + \alpha d_e)$ is

$$\alpha_e^* := \arg\min_{\alpha} f(x_{e-1} + \alpha d_e) \text{ and } x_e^* := x_{e-1} + \alpha_e^* d_e$$

where (due to optimality) the derivative is

$$\nabla f_e(\alpha_e^*) = 0 = \nabla f(x_e^*)^\top d_e \tag{3.9}$$

The line search to find α_e and x_e that approximates the minimum x_e^* can be accomplished by any optimal active learning algorithm algorithm, once we fix the number of time steps per line search.

3.4.1 Analysis of Algorithm RSSCD

Algorithm 3 Randomized Stochastic-Sign Coordinate Descent (RSSCD)

```
Input: set $S$ of diameter $R$, query budget $T$
Oracle: stochastic sign oracle $O_f(x,j)$ returning noisy sign$\big([\nabla f(x)]_j\big)$
BlackBox: algorithm $LS(x,d,n)$ : line search from $x$, direction $d$, for $n$ steps
Choose any $x_0 \in S$, $E = d(\log T)^2$
while $1 \le e \le E$ do
    Choose a unit coordinate vector $d_e$ from $\{1...d\}$ uniformly at random
    $x_e \leftarrow LS(x_{e-1}, d_e, T/E)$ using $O_f$
    $e \leftarrow e+1$
end while
Output: $x_E$
```

Let the number of epochs be $E = d(\log T)^2$, and the number of time steps per epoch is T/E. We can do a line search from x_{e-1}, to get x_e that approximates x_e^* well in function error in $T/E = \tilde{\mathrm{O}}(T)$ steps using an active learning subroutine and let the resulting function-error be denoted by $\epsilon' = \tilde{\mathrm{O}}\Big(T^{-\frac{k}{2k-2}}\Big)$.

$$f(x_e) \le f(x_e^*) + \epsilon'$$

Also, LkSS and UC allow us to infer (for $k^* = \frac{k}{k-1}$, i.e. $1/k + 1/k^* = 1$)

$$f(x_{e-1}) - f(x_e^*) \;\ge\; \frac{\lambda}{2}\|x_{e-1} - x_e^*\|^k \;\ge\; \frac{\lambda}{2\Lambda^{k^*}}\left|\nabla_{e-1}^\top d_e\right|^{k^*}$$

Eliminating $f(x_e^*)$ from the above equations, subtracting $f(x^*)$ from both sides, denoting $\Delta_e := f(x_e) - f(x^*)$ and taking expectations

$$\mathbb{E}[\Delta_e] \le \mathbb{E}[\Delta_{e-1}] - \frac{\lambda}{2\Lambda^{k^*}}\mathbb{E}\Big[\left|\nabla_{e-1}^\top d_e\right|^{k^*}\Big] + \epsilon'$$

Since[8] $\mathbb{E}\Big[|\nabla_{e-1}^\top d_e|^{k^*}|d_1,...,d_{e-1}\Big] = \frac{1}{d}\|\nabla_{e-1}\|_{k^*}^{k^*} \ge \frac{1}{d}\|\nabla_{e-1}\|^{k^*}$ we get

$$\mathbb{E}[\Delta_e] \le \mathbb{E}[\Delta_{e-1}] - \frac{\lambda}{2d\Lambda^{k^*}}\mathbb{E}\Big[\|\nabla_{e-1}\|^{k^*}\Big] + \epsilon'$$

By convexity, Cauchy-Schwartz and UC[9], $\|\nabla_{e-1}\|^{k^*} \ge \left(\frac{\lambda}{2}\right)^{1/k-1}\Delta_{e-1}$, we get

$$\mathbb{E}[\Delta_e] \le \mathbb{E}[\Delta_{e-1}]\left(1 - \frac{1}{d}\left(\frac{\lambda}{2\Lambda}\right)^{k^*}\right) + \epsilon'$$

Defining[10] $C := \frac{1}{d}\left(\frac{\lambda}{2\Lambda}\right)^{k^*} < 1$, we get the recurrence

$$\mathbb{E}[\Delta_e] - \frac{\epsilon'}{C} \le (1-C)\left(\mathbb{E}[\Delta_{e-1}] - \frac{\epsilon'}{C}\right)$$

[8] $k \ge 2 \implies 1 \le k^* \le 2 \implies \|.\|_{k^*} \ge \|.\|_2$

[9] $\Delta_{e-1}^k \le [\nabla_{e-1}^\top(x_{e-1} - x^*)]^k \le \|\nabla_{e-1}\|^k\|x_{e-1} - x^*\|^k \le \|\nabla_{e-1}\|^\kappa \frac{2}{\lambda}\Delta_{e-1}$

[10] Since $1 < k^* \le 2$ and $\Lambda > \lambda/2$, we have $C < 1$

Since $E = d(\log T)^2$ and $\Delta_0 \le L\|x_0 - x^*\| \le LR$, after the last epoch, we have

$$\begin{aligned}\mathbb{E}[\Delta_E] - \frac{\epsilon'}{C} \;&\le\; (1-C)^E\left(\Delta_0 - \frac{\epsilon'}{C}\right) \;\le\; \exp\big\{-Cd(\log T)^2\big\}\Delta_0 \\ &\le\; LRT^{-Cd\log T}\end{aligned}$$

As long as $T > \exp\left\{(2\Lambda/\lambda)^{k^*}\right\}$, a constant, we have $Cd\log T \ge 1$ and

$$\mathbb{E}[\Delta_E] = \mathrm{O}(\epsilon') + \mathrm{o}(T^{-1}) = \tilde{\mathrm{O}}\Big(T^{-\frac{k}{2k-2}}\Big)$$

which is the desired result. Notice that in this section we didn't need to know λ, Λ, k, because we simply run randomized coordinate descent for $E = d(\log T)^2$ epochs with T/E steps per subroutine, and the active learning subroutine was also adaptive to the appropriately calculated TNC parameters. In summary,

Theorem 2. *Given access to only noisy gradient sign information from a stochastic sign oracle, Randomized Stochastic-Sign Coordinate Descent can minimize UC and LkSS functions at the minimax optimal convergence rate for expected function error of* $\tilde{\mathrm{O}}(T^{-\frac{k}{2k-2}})$ *adaptive to all unknown convexity and smoothness parameters. As a special case for* $k = 2$*, strongly convex and strongly smooth functions can be minimized in* $\tilde{\mathrm{O}}(1/T)$ *steps.*

3.4.2 Gradient Sign-Preserving Computations

A practical concern for implementing optimization algorithms is machine precision, the number of decimals to which real numbers are stored. Finite space may limit the accuracy with which every gradient can be stored, and one may ask how much these inaccuracies may affect the final convergence rate - how is the query complexity of optimization affected if the true gradients were rounded to one or two decimal points? If the gradients were randomly rounded (to remain unbiased), then one might guess that we could easily achieve stochastic first-order optimization rates.

However, our results give a surprising answer to that question, as a similar argument reveals that for UC and LkSS functions (with strongly convex and strongly smooth being a special case), our algorithm achieves exponential rates. Since rounding errors do not flip any sign in the gradient, even if the gradient was rounded or decimal points were dropped as much as possible and we were to return only a single bit per coordinate having the true signs, then one can still achieve the exponentially fast convergence rate observed in non-stochastic settings - our algorithm needs only a logarithmic number of epochs, and in each epoch active learning will approach the directional minimum exponentially fast with noiseless gradient signs using a perfect binary search. In fact, our algorithm is the natural generalization for a higher-dimensional binary search, both in the deterministic and stochastic settings.

We can summarize this in the following theorem:

Theorem 3. *Given access to gradient signs in the presence of sign-preserving noise (such as deterministic or random rounding of gradients, dropping decimal places for lower precision, etc), Randomized Stochastic-Sign Coordinate Descent can minimize UC and LkSS functions exponentially fast, with a function error convergence rate of* $\tilde{\mathrm{O}}(\exp\{-T\})$.

3.5 Discussion

While the assumption of smoothness is natural for strongly convex functions, our assumption of LkSS might appear strong in general. It is possible to relax this assumption and require the LkSS exponent to differ from the UC exponent, or to only assume strong smoothness - this still yields consistency for our algorithm, but the rate achieved is worse. [107] and [160] both have epoch based algorithms that achieve the minimax rates under just Lipschitz assumptions with access to a full-gradient stochastic first order oracle, but it is hard to prove the same rates for a coordinate descent procedure without smoothness assumptions.

Given a target function accuracy ϵ instead of query budget T, a similar randomized coordinate descent procedure to ours achieves the minimax rate with a similar proof, but it is non-adaptive since we presently don't have an adaptive active learning procedure when given ϵ. As of now, we know no adaptive UC optimization procedure when given ϵ.

Recently, [11] analysed stochastic gradient descent with averaging, and show that for smooth functions, it is possible for an algorithm to automatically adapt between convexity and strong convexity, and in comparision we show how to adapt to unknown uniform convexity (strong convexity being a special case of $\kappa = 2$). It may be possible to combine the ideas from this chapter and [11] to get a universally adaptive algorithm from convex to all degrees of uniform convexity. It would also be interesting to see if these ideas extend to connections between convex optimization and learning linear threshold functions.

In this chapter, we exploit recently discovered theoretical connections by providing explicit algorithms that take advantage of them. We show how these could lead to cross-fertilization of fields in both directions and hope that this is just the beginning of a flourishing interaction where these insights may lead to many new algorithms if we leverage the theoretical relations in more innovative ways.

Chapter 4

Active Learning : The effect of uniform feature noise

In active learning, the user sequentially chooses values for feature X and an oracle returns the corresponding label Y. In this chapter, we consider the effect of feature noise in active learning, which could arise either because X itself is being measured, or it is corrupted in transmission to the oracle, or the oracle returns the label of a noisy version of the query point. In statistics, feature noise is known as "errors in variables" and has been studied extensively in non-active settings. However, the effect of feature noise in active learning has not been studied before. We consider the well-known Berkson errors-in-variables model with additive uniform noise of width σ.

Our simple but revealing setting is that of one-dimensional binary classification setting where the goal is to learn a threshold (point where the probability of a $+$ label crosses half). We deal with regression functions that are antisymmetric in a region of size σ around the threshold and also satisfy Tsybakov's margin condition around the threshold. We prove minimax lower and upper bounds which demonstrate that when σ is smaller than the minimiax active/passive noiseless error derived in [34], then noise has no effect on the rates and one achieves the same noiseless rates. For larger σ, the *unflattening* of the regression function on convolution with uniform noise, along with its local antisymmetry around the threshold, together yield a behaviour where noise *appears* to be beneficial. Our key result is that active learning can buy significant improvement over a passive strategy even in the presence of feature noise.

4.1 Introduction

Active learning is a machine learning paradigm where the algorithm interacts with a label-providing oracle in a feedback driven loop where past training data (features queried and corresponding labels) are used to guide the design of subsequent queries. Typically, the oracle is queried with an exact feature value and the oracle returns the label corresponding precisely to that feature value. However, in many scenarios, the feature value being queried can be noisy and it helps to analyze what would happen in such a setting. Such situations include noisy sensor measurements of features, corrupted transmission of data from source to storage, or just access to a limited noisy oracle.

The errors-in-variables model has been well studied in the statistical literature and their effect can be profound. In density estimation, Gaussian error causes the minimax rate to become logarithmic in sample size instead of polynomial, see [67]. For results in passive regression, refer to [31, 68, 75], and for passive classification, see [132]. However, classification has not been studied in the *Berkson* model introduced below. Also, deconvolution estimators require the noise fourier transform to be bounded away from zero, ruling out uniform noise. Finally, to the best of our knowledge, feature noise has not been studied for active learning in any setting.

The *classical errors in variables model* has the graphical form $W \leftarrow X \rightarrow Y$, representing

$$\begin{aligned} W &= X + \delta \ , \\ Y &= m(X) + \epsilon \ . \end{aligned}$$

Here, the label Y depends on the feature X but we do not observe X; rather we observe the noisy feature W. The *Berkson errors in variables model* is

$$\begin{aligned} X &= W + \delta \ , \\ Y &= m(X) + \epsilon \ . \end{aligned}$$

The difference is that we start with an observed feature W and then noise is added to determine X. Graphically, this model is $W \rightarrow X \rightarrow Y$.

In this chapter, we focus on the Berkson error model since it intuitively makes more sense for active learning - it captures the idea that we request a label for feature W, but the oracle returns the label for X which is a corrupted version generated from W, i.e. the noise occurs between the label request and the oracle output. We use uniform noise since it yields insightful behavior and also has not been addressed in the literature. We conjecture that qualitatively similar results hold for other symmetric error models.

4.1.1 Setup

Threshold Classification. Let $\mathcal{X} = [-1, 1]$, $\mathcal{Y} = \{+, -\}$, and $f : \mathcal{X} \rightarrow \mathcal{Y}$ denote a classification rule. Assuming $0/1$ loss, the risk of the classification rule f is $R(f) = \mathbb{E}[1_{\{f(X) \neq Y\}}] = \mathbb{P}(f(X) \neq Y)$. It is known that the Bayes optimal classifier, the best measurable classifier that minimizes the risk $f^* = \arg\min_f R(f)$, has the following form

$$f^*(x) = \begin{cases} + & \text{if } m(x) \geq 1/2 \ , \\ - & \text{if } m(x) < 1/2 \ , \end{cases}$$

where $m(x) = \mathbb{P}(Y = +|X = x)$ is the unknown regression function. In what follows, we will consider the case where the f^* is a threshold classifier, i.e. there exists a unique $t \in [-1, 1]$ with $m(t) = 1/2$ such that $m(x) < 1/2$ if $x < t$, and $m(x) > 1/2$ if $x > t$.

Berkson Error Model. The model is:

1. User chooses W and requests label.
2. Oracle receives a noisy W namely $X = W + U$.
3. Oracle returns Y where $\mathbb{P}(Y = +|X = x) = m(x)$.

We take the noise to be uniform: $U \sim \text{Unif}[-\sigma, \sigma]$, where the noise width σ is known for simplicity.

Sampling Strategies. In *passive sampling*, assume that we are given a batch of $w_i \sim \text{Unif}[-1, 1]$ and corresponding labels y_i sampled independently of $\{w_j\}_{j \neq i}$ and $\{y_j\}_{j \neq i}$. In this case, a strategy S is just an estimator $S_n : (W \times Y)^n \rightarrow [-1, 1]$ that returns a guess $\widehat{t}$ of the threshold t on seeing $\{w_i, y_i\}_{i=1}^n$.

In *active sampling* we are allowed to sequentially choose $w_i = S_i(w_1, \ldots, w_{i-1}, y_1, \ldots, y_{i-1})$, where S_i is a possibly random function of past queries and labels, where the randomness is independent of queries and labels. In this case, a strategy A is a sequence of functions $S_i : (W \times Y)^{i-1} \to [-1, 1]$ returning query points and an estimator $S_n : (W \times Y)^n \to [-1, 1]$ that returns a guess $\widehat{t}$ at the end.

Let $\mathcal{S}_n^P, \mathcal{S}_n^A$ be the set of all passive or active strategies (and estimators) with a total budget of n labels.

To avoid the issue of noise resulting in a point outside the domain, we make a (Q)uerying assumption:

(Q). Querying within σ of the boundary is disallowed.

Loss Measure. Let $\widehat{t} = \widehat{t}(W_1^n, Y_1^n)$ denote an estimator of t using n samples from a passive or active strategy. Our task will be to estimate the location of t, where we measure accuracy of an estimator $\widehat{t}$ by a loss function which is the point error $|\widehat{t} - t|$.

Function Class. In the analysis of rates for classification (among others), it is common to use the *Tsybakov Noise/Margin Condition* (see [217]), to characterize the behavior of $m(x)$ around the threshold t. Given constants c, C with $C \geq c$, $k \geq 1$, and noise level σ, let $\mathcal{P}(c, C, k, \sigma)$ be the set of regression functions $m(x)$ that satisfy the following conditions (T,M,B) for some threshold t:

(T). $|x - t|^{k-1} \geq |m(x) - 1/2| \geq c|x - t|^{k-1}$ whenever $|m(x) - 1/2| \leq \epsilon_0$ for some constant ϵ_0

(M). $m(t + \delta) - 1/2 = 1/2 - m(t - \delta)$ for all $\delta \leq \sigma$.

(B). t is at least σ away from the boundary.

On adding noise U, the point where $m \star U$ ($\star$ means convolution) crosses half may differ from t, the point where m crosses half. However, the antisymmetry assumption (M) and boundary assumption (B) together imply that the two thresholds are the same. Getting rid of (M,B) seems substantially difficult.

When $\sigma = 0$, (Q), (M) and (B) are vacuously satisfied, and this is exactly the class of functions and strategies considered in [34]. Smaller k means that the regression function is steeper, which makes it easier to estimate the threshold and classify future labels (cf. [204]). $k = 1$ captures a discontinuous $m(x)$ jumping at t.

Minimax Risk. We are interested in the minimax risk under the point error loss :

$$\mathcal{R}_n(\mathcal{P}(c, C, k, \sigma)) = \inf_{S \in \mathcal{S}_n} \sup_{P \in \mathcal{P}(c,C,k,\sigma)} \mathbb{E}|\widehat{t} - t| \tag{4.1}$$

where $\mathcal{S}_n$ is the set of strategies accessing n samples. For brevity, $\mathcal{R}_n^P(k, \sigma)$ or $\mathcal{R}_n^A(k, \sigma)$ denotes risk for (P)assive/(A)ctive sampling stratgies $\mathcal{S}_n^P, \mathcal{S}_n^A$.

Notation $\prec, \succ, \asymp, \preceq, \succeq$. We analyse minimax point error rates in different regimes of σ as a function of n (or equivalently, for a given point error, we can analyse how the sample size n depends on σ) and we write σ_n for emphasis. In this chapter, $f_n \prec g_n$ means $f_n/g_n \to 0$, $f_n \asymp g_n$ means $c_1 g_n \leq f_n \leq c_2 g_n$ where c_1, c_2 are constants, $f_n \preceq g_n$ means $f_n \prec g_n$ or $f_n \asymp g_n$, $f_n \succeq g_n$ means $g_n \preceq f_n$ and $f_n \succ g_n$ means $g_n \prec f_n$.

4.2 Main Result and Comparisions

The main result of this chapter is as follows.

Theorem 4. *Under the Berkson error model, when given n labels sampled actively or passively with assumption (Q), and when the true underlying regression function lies in $\mathcal{P}(c, C, k, \sigma_n)$ for known k, σ_n, the minimax risk under the point error loss is:*

1. $\mathcal{R}_n^P(\mathcal{P}(k,\sigma)) \asymp \begin{cases} n^{-\frac{1}{2k-1}} & \textit{if } \sigma_n \prec n^{-\frac{1}{2k-1}} \\ \sigma_n^{-(k-\frac{3}{2})}\sqrt{\frac{1}{n}} & \textit{otherwise} \end{cases}$

2. $\mathcal{R}_n^A(\mathcal{P}(k,\sigma)) \asymp \begin{cases} n^{-\frac{1}{2k-2}} & \textit{if } \sigma_n \prec n^{-\frac{1}{2k-2}} \\ \sigma_n^{-(k-2)}\sqrt{\frac{1}{n}} & \textit{otherwise} \end{cases}$

When $k = 1$, $m(x)$ jumps at the threshold, and we interpret the quantity $n^{-\frac{1}{2k-2}}$ as being exponentially small, i.e. being smaller than n^{-p} for any p. We also suppress logarithmic factors in n, σ_n. If the domain was $[-R, R]$, the corresponding passive rates are obtained by substituting n by n/R, but active rates remain the same upto logarithmic factors in R.

Remark. In this chapter, we focus on learning the threshold t. This is relevant because the threshold maybe of intrinsic interest, and also of interest for prediction if, for example, future queries could be made with a different noise model or can be obtained (with some cost) noise-free. Similar results can be derived for 0/1-risk.

Zero Noise. When $\sigma = 0$, the assumptions (Q,B,M) are vacuously true, and our class $\mathcal{P}(c, C, k, 0)$ matches the class $\mathcal{P}(c, C, k)$ considered in [34], and our rates for $\sigma = 0$ i.e. $n^{-\frac{1}{2k-1}}$ and $n^{-\frac{1}{2k-2}}$ are precisely the passive and active minimax point error rates in [34].

Small Noise. When the noise is small, we get what we expect - the risk does not change with noise as long as the noise itself is smaller than the noiseless error. In other words, as long as the noise is smaller than the noiseless error rate of $n^{-\frac{1}{2k-1}}$ for passive learning, passive learners will not really be able to notice this tiny noise, and the minimax rate remains $n^{-\frac{1}{2k-1}}$. Similarly, as long as the noise is smaller than the noiseless error rate of $n^{-\frac{1}{2k-2}}$ for active learning, active learners will not really be able to notice this tiny noise, and the minimax rate remains $n^{-\frac{1}{2k-1}}$. Also, the passive rates vary smoothly - at the point when $\sigma_n \asymp n^{-\frac{1}{2k-1}}$, the rates for small and large noise coincide. Similarly, at the point when $\sigma_n \asymp n^{-\frac{1}{2k-2}}$, the aforementioned active rates for small and large noise coincide.

Large Noise and Assumption (M). When the noise is large, we see a curious behaviour of the rates. When $k > 2$, the error rates seem to get smaller/better with larger noise for both active and passive learning, and furthermore the noisy rates can also be better than the noiseless rate! This might seem to violate both the information processing inequality, and our intuition that more noise shouldn't help estimation. Moreover, a noiseless active learner may be able to simulate a noisy situation by adding noise and querying at the resulting point, and get better rates, violating lower bounds in [34].

However, we make the following crucial but subtle observation. Our claimed rates are *not* about a fixed function class - due to assumption (M), the function class changes with σ, and in fact (M) requires the antisymmetry of the regression function to hold over a larger region for larger σ. This set of functions is actually getting smaller with larger σ. Even though the functions can behave quite arbitrarily outside $(t - \sigma, t + \sigma)$, this assumption (M) on a small region of size 2σ actually helps us significantly.

Given that there is no contradiction to the results of [34] or more fundamental information theoretic ideas, there is also an intuitive explanation of why assumption (M) helps when we have large noise. As we will see in a later figure, convolution with noise seems to "stretch/unflatten" the function around the threshold. Specifically, for larger $k > 2$, the regression function can be quite flat around the threshold - convolution with noise makes it less flat and more linear - in fact it behaves linearly over a large region of width nearly 2σ. This is true regardless of whether assumption (M) holds - however if (M) does not hold, then the convolved threshold, which is the point where the convolved function crosses half, need not be the original threshold t. While dropping assumption (M) will not hurt if we only want to find the convolved threshold, but given that our aim is to estimate t, the problem of figuring out how much the threshold shifted can be quite non-trivial.

Hence, large noise ensures a behaviour that is less flat and more linear around the threshold, and assumption (M) ensures that the threshold doesn't shift from t. Intuitively this is why (M) and large noise help, and technically there is no contradiction becasue the function class is getting progressively simpler because of more controlled growth around the threshold.

The main takeaway is that in all settings, active learning yields a gain over passive sampling. We now describe the upper and lower bounds that lead to Theorem 1. The case $k = 1$ is handled in detail for intuitionb but proofs for $k > 1$ are in the Appendix.

4.2.1 Simulation of Noise Convolution

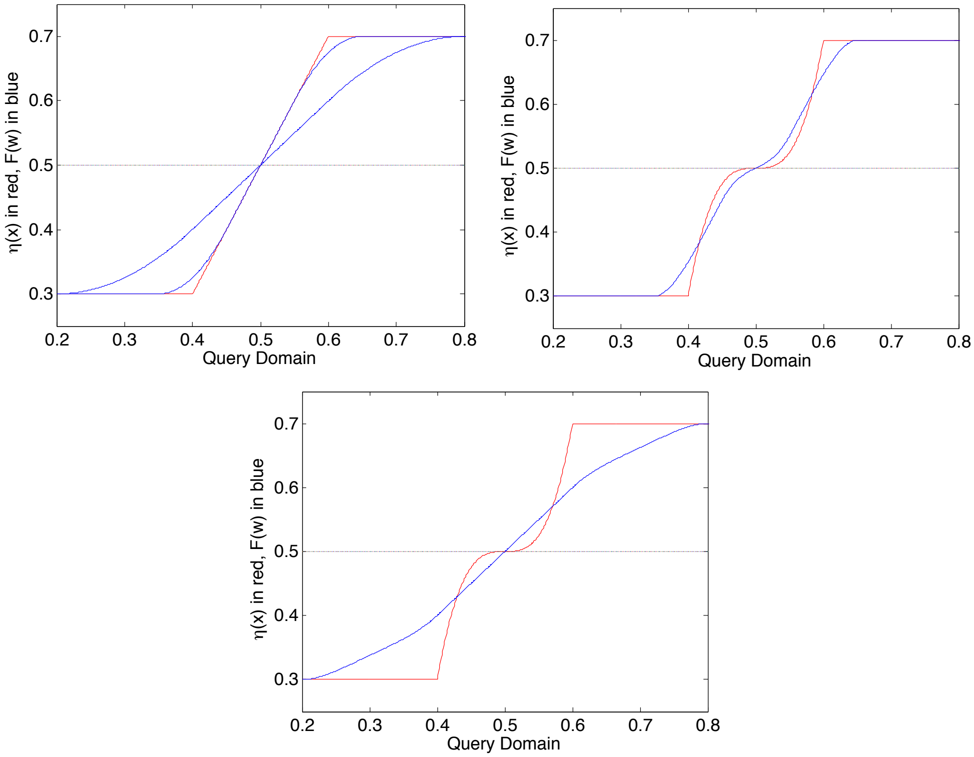


Figure 4.1: Regression function $\eta(x)$ (red) before and $F(w)$ (blue) after convolution with noise. In all 3 figures, Tsybakov's margin condition holds for $x \in [0.4, 0.6]$. The top plot has a linear regression function ($k = 2$), and its two blue curves are for $\sigma_n = 0.05\ (narrow), 0.2\ (wide)$, and they show that a linear growth around $t = 0.5$ remains linear. The middle and bottom figure are for a flatter regression function with $k = 4$, and $\sigma_n = 0.05, 0.2$ respectively, plotted separately for clarity. $k = 4$ is harder than for $k = 2$ because the red curve is flatter around t, making it harder to pinpoint the threshold. However, as one can see in both plots, noise actually *helps* by smoothing it out and making it more linear. However, note that the effect of assumption (M) cannot be understated, due to which in all plots the threshold before and after noise cross half at the same point. The effect of noise when $k = 1$ can be seen in the following section.

4.2.2 Chapter Roadmap

We devote the next two sections to proving the lower and upper bounds, in that order, that lead to Theorem 4. While the proofs will be self-contained, we leave some detailed calculations to the appendix.

For easier readability, we present lower bounds for $k = 1$ first to absorb the technique and then the lower bounds for $k > 1$. In Section 4.3 we will prove

Theorem 5 (Lower Bounds)**.** *Under the Berkson error model and assumption (Q),*

1. *For $k = 1$, the passive/active lower bounds are*

$$\inf_{S\in\mathcal{S}_n^P} \sup_{P\in\mathcal{P}(1,\sigma_n)} \mathbb{E}|\widehat{t}-t| \succeq \begin{cases} \frac{1}{n} & \textit{if } \sigma_n \prec \frac{1}{n} \\ \sqrt{\frac{\sigma_n}{n}} & \textit{otherwise} \end{cases}$$

$$\inf_{S\in\mathcal{S}_n^A} \sup_{P\in\mathcal{P}(1,\sigma_n)} \mathbb{E}|\widehat{t}-t| \succeq \begin{cases} e^{-n} & \textit{if } \sigma_n \prec e^{-n} \\ \frac{\sigma_n}{\sqrt{n}} & \textit{otherwise} \end{cases}$$

2. *For $k > 1$, the passive/active lower bounds are*

$$\inf_{S\in\mathcal{S}_n^P} \sup_{P\in\mathcal{P}(k,\sigma_n)} \mathbb{E}|\widehat{t}-t| \succeq \begin{cases} n^{-\frac{1}{2k-1}} & \textit{if } \sigma_n \prec n^{-\frac{1}{2k-1}} \\ \sigma_n^{-(k-\frac{3}{2})}\sqrt{\frac{1}{n}} & \textit{otherwise} \end{cases}$$

$$\inf_{S\in\mathcal{S}_n^A} \sup_{P\in\mathcal{P}(k,\sigma_n)} \mathbb{E}|\widehat{t}-t| \succeq \begin{cases} n^{-\frac{1}{2k-2}} & \textit{if } \sigma_n \prec n^{-\frac{1}{2k-2}} \\ \sigma_n^{-(k-2)}\sqrt{\frac{1}{n}} & \textit{otherwise} \end{cases}$$

Following that, we again present active and passive algorithms for $k = 1$ first to gather intuition and then generalize them for $k > 1$. In Section 4.4 we will prove

Theorem 6 (Upper Bounds)**.** *Under the Berkson error model and assumption (Q),*

1. *For $k = 1$, a passive algorithm (WIDEHIST) and an active algorithm (ACTPASS) return $\widehat{t}$ s.t.*

$$\sup_{P\in\mathcal{P}(1,\sigma_n)} \mathbb{E}|\widehat{t}-t| \preceq \begin{cases} \frac{1}{n} & \textit{if } \sigma_n \prec \frac{1}{n} \\ \sqrt{\frac{\sigma_n}{n}} & \textit{otherwise} \end{cases}$$

$$\sup_{P\in\mathcal{P}(1,\sigma_n)} \mathbb{E}|\widehat{t}-t| \preceq \begin{cases} e^{-n} & \textit{if } \sigma_n \prec e^{-n} \\ \frac{\sigma_n}{\sqrt{n}} & \textit{otherwise} \end{cases}$$

2. *For $k > 1$, a passive algorithm (WIDEHIST) and an active algorithm (ACTPASS) return $\widehat{t}$ s.t.*

$$\sup_{P\in\mathcal{P}(k,\sigma_n)} \mathbb{E}|\widehat{t}-t| \preceq \begin{cases} n^{-\frac{1}{2k-1}} & \textit{if } \sigma_n \prec n^{-\frac{1}{2k-1}} \\ \sigma_n^{-(k-\frac{3}{2})}\sqrt{\frac{1}{n}} & \textit{otherwise} \end{cases}$$

$$\sup_{P\in\mathcal{P}(k,\sigma_n)} \mathbb{E}|\widehat{t}-t| \preceq \begin{cases} n^{-\frac{1}{2k-2}} & \textit{if } \sigma_n \prec n^{-\frac{1}{2k-2}} \\ \sigma_n^{-(k-2)}\sqrt{\frac{1}{n}} & \textit{otherwise} \end{cases}$$

4.3 Lower Bounds

To derive lower bounds, we will follow the approach of [105, 218] which were exemplified in lower bounds for active learning problems without feature noise in [32, 34]. The standard methodology is to reduce the problem of classification in the class $P(c, C, k, \sigma)$ to one of hypothesis testing. Similar to [32, 34], it will suffice to consider two hypotheses and use the following version of Fano's lemma from [218] (Theorem 2.2).

Theorem 7 ([218])**.** *Let $\mathcal{F}$ be a class of models. Associated with each $f \in \mathcal{F}$ we have a probability measure P_f defined on a common probability space. Let $d(.,.) : \mathcal{F}, \mathcal{F} \to \mathbb{R}$ be a semi-distance. Let $f_0, f_1 \in \mathcal{F}$ be such that $d(f_0, f_1) \geq 2a$, with $a > 0$. Also assume that $KL(P_{f_0}, P_{f_1}) \leq \gamma$, where KL denotes the Kullback-Leibler divergence. Then, the following bound holds:*

$$\begin{aligned} \inf_{\widehat{f}} \sup_{f \in \mathcal{F}} P_f(d(\widehat{f}, f) \geq a) &\geq \inf_{\widehat{t}} \max_{j \in \{0,1\}} P_{f_j}(d(\widehat{f}, f_j) \geq a) \\ &\geq \max\left(\frac{e^{-\gamma}}{4}, \frac{1 - \sqrt{\frac{\gamma}{2}}}{2}\right) =: \rho \end{aligned}$$

where the $\inf$ *is taken with respect to the collection of all possible estimators of f based on a sample from P_f.*

Corollary 8. *If γ is a constant, then ρ is a constant, and by Markov's inequality, we would get*

$$\inf_{\widehat{f}} \sup_{f \in \mathcal{F}} \mathbb{E} d(\widehat{f}, f) \geq \rho a$$

and the minimax risk under loss d would be $\succeq a$.

Proof of Theorem 5, $k = 1$. Choose $\mathcal{F} = \mathcal{P}(1, \sigma_n)$. Let $P_t \in \mathcal{P}(1, \sigma_n)$ denote a regression function with threshold at t. We choose the semi-metric to be the distance between thresholds, i.e. $d(P_r, P_s) = |r - s|$. We now choose two such distributions with thresholds at least $2a_n$ apart (we use a_n to explicitly remind the reader that a will later be set to depend on n) - let them be denoted P_{t_0} and P_{t_1} with $t_0 = -a_n, t_1 = a_n$ and

$$P_t(Y = +|X = x) = \begin{cases} 0.5 - c & x < t\ , \\ 0.5 + c & x \geq t\ . \end{cases}$$

Due to addition of noise, we get convolved distributions $P^0 = P_{t_0}(Y|W)$ and $P^1 := P_{t_1}(Y|W)$.

As hinted by the above corollary, we will choose a_n so that $KL(P^0, P^1)$ is bounded by a constant, to get a lower bound on risk $\succeq a_n$. This follows by the following argument from [32].

The $KL(P^0, P^1)$ can be bounded as

$$\begin{aligned}
& \mathbb{E}^1_{W,Y}\left[\log \frac{P^1(W_1^n, Y_1^n)}{P^0(W_1^n, Y_1^n)}\right] && (4.2)\\
= \;& \mathbb{E}^1_{W,Y}\left[\log \frac{\prod_i P^1(Y_i|W_i)P(W_i|W_1^{i-1}, Y_1^{i-1})}{\prod_i P^0(Y_i|W_i)P(W_i|W_1^{i-1}, Y_1^{i-1})}\right] && \\
= \;& \mathbb{E}^1_{W,Y}\left[\log \frac{\prod_i P^1(Y_i|W_i)}{\prod_i P^0(Y_i|W_i)}\right] && (4.3)\\
= \;& \sum_i \mathbb{E}^1_W\left[\mathbb{E}^1_Y\left[\log \frac{P^1(Y_i|W_i)}{P^0(Y_i|W_i)} \;\middle|\; W_1, ..., W_n\right]\right] && (4.4)\\
\leq \;& n \max_{w\in[-1,1]} \mathbb{E}^1_Y\left[\log \frac{P^1(Y|W)}{P^0(Y|W)} \;\middle|\; W = w\right] && (4.5)\\
\preceq \;& n \max_{w\in[-1,1]} (P^1(Y|w) - P^0(Y|w))^2 && (4.6)
\end{aligned}$$

where (4.3) holds for active learning because the algorithm determines W_i when given $\{W_1^{i-1}, Y_1^{i-1}\}$ and is independent of the model, and follows by the independence of future from past for passive learning. (4.4) holds by law of iterated expectation. (4.5) is used for active learning but is not needed for passive learning. (4.6) follows by an approximation

$$KL(Ber(1/2 + p), Ber(1/2 + q)) \preceq (p - q)^2$$

for sufficiently small constants p, q.

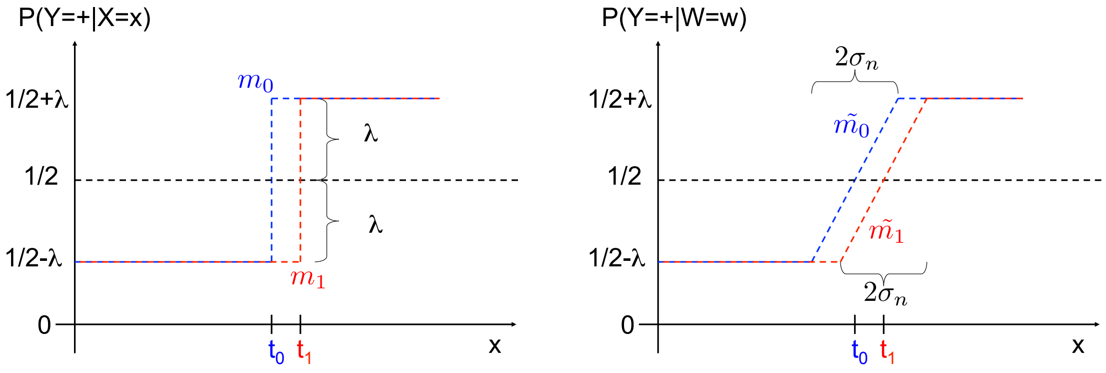


Figure 4.2: Regression functions before (top) and after (bottom) convolution with noise.

$F_t(w) := P_t(Y|W = w) = \int P_t(Y|X)P(X|W = w)dX$ and a straightforward calculation reveals that

$$F_t(w) = \begin{cases} 0.5 - c & w \leq t - \sigma_n\ , \\ 0.5 + \frac{c}{\sigma_n}(w - t) & w \in [t - \sigma_n, t + \sigma_n]\ , \\ 0.5 + c & w \geq t + \sigma_n\ . \end{cases} \qquad (4.7)$$

As depicted in Fig.4.2, note the behavior before and after convolution with noise: (i) $m(t) = F(t) = 1/2$, hence $F_1(a_n) = 1/2 = F_0(-a_n)$ (ii) Both convolved regression functions grow linearly for a region of width $2\sigma_n$, and differ only on a width of $2(\sigma_n + a_n)$; (iii) For a large region $[a_n - \sigma_n, -a_n + \sigma_n]$ of size $2(\sigma_n - a_n)$, we have $\big|F_1(w) - F_0(w)\big| = 2a_n c/\sigma_n$, a constant. Their gap varies when $\sigma_n \succeq a_n$ as $\big|F_0(w) - F_1(w)\big| =$

$$\begin{cases} \Big(w + a_n + \sigma_n\Big)\frac{c}{\sigma_n} & w \in [-a_n - \sigma_n, a_n - \sigma_n] \\ 2a_n\frac{c}{\sigma_n} & w \in [a_n - \sigma_n, -a_n + \sigma_n] \\ \Big((a_n + \sigma_n) - w\Big)\frac{c}{\sigma_n} & w \in [-a_n + \sigma_n, a_n + \sigma_n] \\ 0 & \text{otherwise.} \end{cases}$$

When $\sigma_n \prec a_n$, $\big|F_1(w) - F_0(w)\big| =$

$$\begin{cases} \Big(w + a_n + \sigma_n\Big)\frac{c}{\sigma_n} & w \in [-a_n - \sigma_n, -a_n + \sigma_n] \\ 2c & w \in [-a_n + \sigma_n, a_n - \sigma_n] \\ \Big((a_n + \sigma_n) - w\Big)\frac{c}{\sigma_n} & w \in [a_n - \sigma_n, a_n + \sigma_n] \\ 0 & \text{otherwise.} \end{cases}$$

For active learning, when $\sigma_n \succeq a_n$ we note

$$\max_{w\in[-1,1]} |P^1(Y|w) - P^0(Y|w)| = \frac{2a_n c}{\sigma_n}$$

and get $KL(P^0, P^1) \preceq n\frac{a_n^2}{\sigma_n^2}$ by Eq.(4.6). We choose $a_n \asymp \frac{\sigma_n}{\sqrt{n}}$, which becomes our active minimax error rate by Corollary 8 when $\sigma_n \succeq a_n$ i.e. $\sigma_n \succeq e^{-n}$.

Similarly, if $\sigma_n \prec \exp\{-n\}$, setting $a_n \asymp \exp\{-n\}$ easily gives us an exponentially small lower bound.

In the passive setting, Eq.(4.5) does not apply. Since the two convolved distributions differ only on an interval of size $2(\sigma_n + a_n)$, the effective number of points falling in this interval would be $\asymp n(\sigma_n + a_n)$.

When $\sigma_n \succeq a_n$, a simple calculation shows

$$KL(P^0, P^1) \;\preceq\; n(\sigma_n + a_n)\frac{a_n^2}{\sigma_n^2} \;\asymp\; n\frac{a_n^2}{\sigma_n},$$

giving rise to a choice of $a_n \asymp \sqrt{\frac{\sigma_n}{n}}$, which is the passive minimax rate when $\sigma_n \succeq a_n$ i.e. $\sigma_n \succeq \frac{1}{n}$.

When $\sigma_n \prec \frac{1}{n}$, a similar calculation shows

$$KL(P^0, P^1) \;\preceq\; n(\sigma_n + a_n)4c^2 \;\asymp\; na_n$$

giving rise to a choice of $a_n \asymp \frac{1}{n}$, which is the passive minimax rate when $\sigma_n \succeq a_n$ i.e. $\sigma_n \prec \frac{1}{n}$. ■

Proof of Theorem 5, $k > 1$ We follow a very similar setup to the case $k = 1$. The difference will lie in picking functions that are in $\mathcal{P}(c, C, k, \sigma_n)$ for general $k \neq 1$, and calculating the bounds on KL divergence appropriately. However, for notational convenience, we will assume that the domain is shifted to $[-\sigma_n, 2 - \sigma_n]$ instead of $[-1, 1]$ and that the distance between thresholds is a_n instead of $2a_n$. Define

$$P_0(Y|x) = \begin{cases} 1/2 - c|x|^{k-1} \text{ if } x \in [-\sigma_n, 0] \\ 1/2 + c|x|^{k-1} \text{ if } x > 0 \end{cases}$$

$$P_1(Y|x) = \begin{cases} 1/2 - c|x - a_n|^{k-1} \text{ if } x \in [-\sigma_n, a_n] \\ 1/2 + c|x - a_n|^{k-1} \text{ if } x \in [a_n, \beta a_n + \sigma_n] \\ 1/2 + c|x|^{k-1} \qquad \text{if } x > \beta a_n + \sigma_n \end{cases}$$

where $\beta = \frac{1}{1-(c/C)^{1/(k-1)}} \geq 1$ is a constant chosen such that $P_1 \in \mathcal{P}(c, C, k, \sigma_n)$ (this fact is verified explicitly in the Appendix). For ease of notation, P_0, P_1 are understood to actually saturate at $0, 1$ if need be (i.e. we are implicitly working with $\min\{P_{0/1}, 1\}$, etc). The two thresholds are clearly at $0, a_n$ respectively, and after the point $\beta a_n + \sigma_n$, the two functions are the same. Continuing the same notation as for $k = 1$, we let $P^i = P_i(Y|W) = F_i(w)$ for $i = 0, 1$.

The following claims hold true (Appendix).

1. When $\sigma_n \preceq a_n$, $\max_w |F_1(w) - F_2(w)| \asymp a_n^{k-1}$.
2. When $\sigma_n \succeq a_n$, $\max_w |F_1(w) - F_2(w)| \asymp \sigma_n^{k-2} a_n$.
3. As a subpart of the above cases, when $\sigma_n \asymp a_n$, $\max_w |F_1(w) - F_2(w)| \asymp \sigma_n^{k-2} a_n \asymp a_n^{k-1}$

If the above propositions are true, we can verify:

1. In the first case, $KL(P^0, P^1) \preceq n a_n^{2k-2}$, hence $a_n \asymp n^{-\frac{1}{2k-2}}$ is a lower bound when $\sigma_n \preceq n^{-\frac{1}{2k-2}}$.
2. Otherwise, $KL(P^0, P^1) \preceq n\sigma_n^{2k-4} a_n^2$, hence $a_n \asymp \frac{\sigma_n^{-(k-2)}}{\sqrt{n}}$ is a lower bound when $\sigma_n \succ n^{-\frac{1}{2k-2}}$.

The passive bounds follow by not just considering the maximum difference between $|F_1(w) - F_2(w)|$ but also the length of that difference, since it is directly proportional to the number of points that may randomly fall in that region. Following the same calculations,

1. When $\sigma_n \prec a_n$, $|F_1(w) - F_2(w)| \asymp a_n^{k-1}$ for all $w \in [0, \beta a_n + 2\sigma_n]$. Hence $KL(P^0, P^1) \preceq n(\beta a_n + 2\sigma_n) a_n^{2k-2} \asymp n a_n^{2k-1}$ and $a_n \asymp n^{-\frac{1}{2k-1}}$ is the minimax passive rate when $\sigma_n \prec n^{-\frac{1}{2k-1}}$.
2. When $\sigma_n \succ a_n$, $|F_1(w) - F_2(w)| \asymp \sigma_n^{k-2} a_n$ for all $w \in [0, \beta a_n + 2\sigma_n]$. Hence $KL(P^0, P^1) \preceq n(\beta a_n + 2\sigma_n)\sigma_n^{2k-4} a_n^2$ and $a_n \asymp \sigma_n^{-(k-\frac{3}{2})}\sqrt{\frac{1}{n}}$ is the minimax passive rate when $\sigma_n \succ n^{-\frac{1}{2k-1}}$.

as verified from the Appendix calculation. ■

4.4 Upper Bounds

For passive sampling, we present a modified histogram estimator, WIDEHIST, when the noise level σ_n is larger than the noiseless minimax rate of $1/n$. Assume for simplicity that the n sampled points on $[-1, 1]$ are equally spaced to mimic a uniform distribution, lying at $\frac{(2j-1)}{2n}$, $j = 1, ..., n$.

Algorithm WIDEHIST.

1. Divide $[-1, 1]$ into m bins of width $h > \frac{2}{n}$ so $m = \frac{2}{h} < n$. The i^{th} bin covers $[-1 + (i-1)h, -1 + ih]$, $i \in \{1, ..., m\}$ and hence each bin has $\frac{nh}{2}$ points. Let b_i be the average number of positive labels in bin i of these $\frac{nh}{2}$ points.
2. Let $\widehat{p}_i$ be the average of the b_i's over a all bins within $\pm\sigma_n/2$ of bin i. We "classify" regions with $\widehat{p}_i < 1/2$ as being $-$ and $\widehat{p}_i > 1/2$ as being $+$, and return $\widehat{t}$ as the center of the first bin from left to right where $\widehat{p}_i$ crosses half.

Observe that we need not operate on $[-1, 1]$ with n queries - WIDEHIST(D,B) could take as inputs any domain D and any query budget B. The argument below hinges on the fact that the convolved regression function behaves linearly around t.

Proof of Theorem 6, $k = 1$, (Passive). Let $i^* \in \{1, ..., m\}$ denote the true bin $[(i^* - 1)h, i^* h]$ that contains t. Let $\widehat{t}$ be from bin $\widehat{i}$, i.e. $\widehat{p}_{\widehat{i}} < 1/2$ and $\widehat{p}_{\widehat{i}+1} > 1/2$. We will argue that $\widehat{i}$ is very close to i^*, in which case the point error we suffer is $|\widehat{i} - i^*|h$. Specifically, we prove that all bins except $I^* = \{i^* - 1, i^*, i^* + 1\}$ will be "classified" correctly with high probability. In other words, we claim w.h.p. $\widehat{p}_i < 1/2$ if $i < i^* - 1$ and $\widehat{p}_i > 1/2$ if $i > i^* + 1$.

Indeed, we can show (Appendix)

$$\text{For } i > i^* + 2,\ \mathbb{E}[\widehat{p}_i] \geq \mathbb{E}[\widehat{p}_{i^*+2}] \geq 1/2 + \tfrac{c}{\sigma_n}h \tag{4.8}$$

$$\text{For } i < i^* - 2,\ \mathbb{E}[\widehat{p}_i] \leq \mathbb{E}[\widehat{p}_{i^*-2}] \leq 1/2 - \tfrac{c}{\sigma_n}h \tag{4.9}$$

Using Hoeffding's inequality, we get that for bin i, $\Pr(|\widehat{p}_i - p_i| > \epsilon) \ \leq\ 2\exp\left\{-2\frac{n\sigma_n}{2}\epsilon^2\right\}$ Taking union bound over all bins other than those in $i^* - 1, i^*, i^* + 1$ and setting $\epsilon = \frac{c}{\sigma_n}h$, we get

$$\Pr(\forall i \backslash I^*, |\widehat{p}_i - p_i| > \tfrac{c}{\sigma_n}h) \ \leq\ 2m\exp\left\{-2\frac{n\sigma_n}{2}\left(\frac{ch}{\sigma_n}\right)^2\right\}$$

So we get bins $i\backslash I^*$ correct and $\widehat{i} \in \{i^* \ - \ 1, i^*, i^* \ + \ 1\}$ with probability $\geq 1 - 2n\exp\left\{-n\sigma_n\left(\frac{ch}{\sigma_n}\right)^2\right\}$ since $m \ < \ n$. Setting $h = \frac{1}{c}\sqrt{\frac{\sigma_n}{n}\log(\frac{2n}{\delta})}$ makes this hold with probability $\geq 1 - \delta$ so the point error $|\widehat{i} - i^*|h < 2h$ behaves like $h \preceq \sqrt{\frac{\sigma_n}{n}}$. ■

For active sampling when the noise level σ_n is larger than the minimax noiseless rate e^{-n}, we present a algorithm ACTPASS which makes its n queries on the domain $[-1, 1]$ in E different epochs/rounds. As a subroutine, it uses any optimal passive learning algorithm, like WIDEHIST(D,B). In each round, ACTPASS runs WIDEHIST on progressively smaller domains D with a restricted budget B. Hence it "activizes" the WIDEHIST and achieves the optimal active rate in the process. This algorithm was inspired by a similar idea from [161].

Algorithm ACTPASS.

Let $E = \lceil\log(1/\sigma_n)\rceil$ be the number of epochs and $D_1 = [-1, 1]$ denote the domain of "radius" $R_1 = 1$ around $t_0 = 0$. The budget of every epoch is a constant $B = n/E$. For epochs $1 \leq e \leq E$, do:

1. Query for B labels uniformly on D_e.
2. Let $t_e = \text{WIDEHIST}(D_e, B)$ be the returned estimator using the most recent samples and labels.
3. Define $D_{e+1} = [t_e - 2^{-e}, t_e + 2^{-e}] \cap [-1, 1]$ with a radius of at most $R_{e+1} = 2^{-e}$ around t_e. Repeat.

Observe that ACTPASS runs while $R_e > \sigma_n$, since by design $E \geq \log(1/\sigma_n)$ so $\sigma_n \leq 2^{-E} = R_{E+1}$.

Proof of Theorem 6, $k = 2$, (Active). The analysis of ACTPASS proceeds in two stages depending on the value of σ_n. Initially, when R_e is large, it is possible that $\sigma_n \preceq R_e/n$ and in this phase, the passive algorithm WIDEHIST will behave as if it is in the noiseless setting since the noise is smaller than its noiseless rate. However, after some point, when R_e becomes quite small, $\sigma_n \succeq R_e/n$ is possible and then WIDEHIST will behave as if it is in the noisy setting since noise is larger than its noiseless rate. Observe that it cannot stay in the first phase till the end of the algorithm, since the first phase runs while $\sigma_n \preceq R_e/n$ but we know that $\sigma_n > R_{E+1}$ by construction, so there must be an epoch where it switches phases, and ends the algorithm in its second phase.

We prove (by a separate induction in each epoch) that with high probability, the true threshold t will always lie inside the domain at the start of every epoch (this is clearly true before the first epoch). We claim:

1. Before all e in phase one, $t \in D_e$ w.h.p.
2. Before all e in phase two, $t \in D_e$ w.h.p.

We prove these in the Appendix. If these are true, then in the second phase, WIDEHIST is in the large noise setting and it gets an error of $\sqrt{\frac{R_e\sigma_n}{B}}$. Hence the final error of the algorithm is $\sqrt{\frac{R_E\sigma_n}{n/E}} \asymp \frac{\sigma_n}{\sqrt{n}}$. ■

Proof of Theorem 6, $k > 1$. The proofs for $k > 1$ are simply generalizations of those for $k = 1$. Again, we present concise arguments here for the settings where the algorithm can actually detect noise, i.e. when the noise level is larger than the noiseless minimax rate (otherwise, one can argue that algorithms which worked for the noiseless case will suffice). In both cases, the algorithm remains unchanged.

1. We outline the proof for WIDEHIST when $\sigma_n \succeq n^{-\frac{1}{2k-1}}$. Using similar notation as before, we will again show that if t is in bin i^* of width $h < \sigma_n$, then except for bins $i^* - 1, i^*, i^* + 1$, we will "classify" all other bins correct with high probability, by averaging over the $n\sigma_n/2$ points to the left and right of that bin. Specifically, we claim

$$\text{For } i > i^* + 2, \mathbb{E}[\widehat{p}_i] \geq \mathbb{E}[\widehat{p}_{i^*+2}] \ \geq \ 1/2 + \lambda\sigma_n^{k-2}h \tag{4.10}$$

$$\text{For } i < i^* - 2, \mathbb{E}[\widehat{p}_i] \leq \mathbb{E}[\widehat{p}_{i^*-2}] \ \leq \ 1/2 - \lambda\sigma_n^{k-2}h \tag{4.11}$$

A similar use of Hoeffding's inequality gives

$$\begin{aligned}\Pr(\forall i \backslash I^*, |\widehat{p}_i - p_i| \ &> \ \lambda\sigma_n^{k-2}h) \ \leq \\ &2m\exp\left\{-2(\tfrac{n\sigma_n}{2R})h^2\lambda^2\sigma_n^{2k-4}\right\}.\end{aligned}$$

Arguing as before, w.h.p. we get a point error of $h \preceq \sqrt{\frac{R}{\sigma_n^{2k-3}n}} < \sigma_n$ when $\sigma_n \succ n^{-\frac{1}{2k-1}}$.

2. We outline the proof for ACTPASS when $\sigma_n \succeq n^{-\frac{1}{2k-2}}$. As before, the algorithm runs in two phases, and we will prove required properties within each phase by induction.

The first phase is when R_e is large and so σ_n may possibly be smaller than $(R_e/n)^{\frac{1}{2k-1}}$ and WIDEHIST will achieve noiseless rates within each epoch. In the second phase, after R_e has shrunk enough, σ_n will become larger than $(R_e/n)^{\frac{1}{2k-1}}$ and WIDEHIST will achieve noisy rates in these epochs.

One can verify, as before, that the second phase must occur, by design. Intuitively, the second phase must occur because we make a fixed number of queries $n/E \asymp n/\log n$ in a halving domain size (equivalently we make geometrically increasing queries on a rescaled domain), and so relatively in successive epochs this noiseless error shrinks, and at some point σ_n becomes larger than this shrinking noiseless error rate.

As before we make the following claims:

1. Before all e in phase one $t \in D_e$ w.h.p.
2. Before all e in phase two $t \in D_e$ w.h.p.

These are proved in the Appendix by induction.

The final point error is given by WIDEHIST in the last epoch as $\sqrt{\frac{R_E}{\sigma_n^{2k-3}n/E}} \asymp \frac{1}{\sigma_n^{k-2}}\sqrt{\frac{1}{n}}$ since $R_E \asymp \sigma_n$ and $E \asymp \log n$.

4.5 Conclusion

In this chapter, we propose a simple Berkson error model for one-dimensional threshold classification, inspired by the setup and model analysed in [32, 34], in which we can analyse active learning with additive uniform feature noise. To the best of our knowledge, this is the first attempt at jointly tackling feature noise and label noise in active learning.

This simple setting already yields interesting behaviour depending on the additive feature noise level and the label noise of the underlying regression function. For both passive and active learning, whenever the noise level is smaller than the minimax noiseless rate, the learner cannot notice that there is noise, and will continue to achieve the noiseless rate. As the noise gets larger, the rates do depend on the noise level. Importantly, one can achieve better rates than passive learning in most scenarios, and we propose unique algorithms/estimators to achieve tight rates. The idea of "activizing" passive algorithms, like algorithm ACTPASS did, seems especially powerful and could carry forward to other settings beyond our chapter and [161].

The immediate future work and most direct extension to this chapter concerns the main weakness of the chapter - the possibility of getting rid of Assumption (M), which is the only hurdle to a fair comparision with the noiseless setting. We would like to re-emphasize that at first glance, the rates may be misleading and counterintuitive because it "appears" as if larger noise could possibly help estimation due to the presence of σ_n in the denominator for larger k.

However, we point out once more that the class of functions is not constant over all σ_n - it depends on σ_n, and in fact it gets "smaller" in some sense with larger σ_n because the assumption (M) becomes more stringent. This observation about the non-constant function class, along with the fact that convolution with uniform noise seems to *unflatten* the regression function as shown in the figures, together cause the rates to seemingly improve with larger noise levels.

Analysing the case without (M) seems to be quite a challenging task since the noiseless and convolved thresholds can be different - we did attempt to formulate a few kernel-based estimators with additional assumptions, but do not presently have tight bounds, and leave those for a future work.

Acknowledgements

We thank Rui Castro for detailed conversations about our model and results. This work is supported in part by NSF Big Data grant IIS-1247658.

4.6 Appendix: Justifying Claims in the Lower Bounds

Approximations:

1. $(x+y)^k = x^k(1+y/x)^k \approx x^k + kx^{k-1}y$ when $y \prec x$. Even when $y \preceq x$, both terms are the same order.
2. $(x-y)^k = x^k(1-y/x)^k \approx x^k - kx^{k-1}y$ when $y \prec x$. Even when $y \preceq x$ both terms are the same order.
3. When $y < x$ but not $y \prec x$, by Taylor expansion of $(1+z)^k$ around $z = 0$, we have $(x+y)^k = x^k(1+y/x)^k = x^k[1+(1+c)^{k-1}y/x] = x^k + Cx^{k-1}y$ for some $0 < c < y/x < 1$ and some constant C. Similarly for $(x-y)^k$.

Let's assume the boundary is at $-\sigma$ for easier calculations. (we denote a_n, σ_n as a, σ here). Remember

$$m_1(x) = 1/2 + cx|x|^{k-2} \text{ if } x \geq -\sigma$$

$$m_2(x) = \begin{cases} 1/2 + c(x-a)|x-a|^{k-2} & \text{if } x < \beta a + \sigma \\ m_1(x) & \text{if } x \geq \beta a + \sigma \end{cases}$$

where $\beta = \frac{1}{1-(c/C)^{1/(k-1)}} \geq 1$ is such that $m_2 \in P(\kappa, c, C, \sigma)$. Clearly, when $x < \beta a + \sigma$, m_2 satisfies condition (T). So, we only need to verify that whenever $x \geq \beta a + \sigma$ we have

$$m_2(x) - 1/2 \; = \; cx^{k-1} \quad \leq \quad C(x-a)^{k-1} \tag{4.12}$$

This statement holds iff $(c/C)^{1/(k-1)} \leq 1 - a/x \Leftrightarrow a/x \leq 1 - (c/C)^{1/(k-1)} \Leftrightarrow x \geq \beta a$, which holds for all $\sigma \geq 0$, and hence m_2 satisfies condition (T).

Proposition 1. When $\sigma \prec a$, $\max_w |F_1(w) - F_2(w)| \asymp a^{k-1}$

Proposition 2. When $\sigma \succ a$ $\max_w |F_1(w) - F_2(w)| \asymp \sigma^{k-2}a$

Let us now prove these two propositions, with detailed calculations in each case (note that when $\sigma \asymp a$, then $\max_w |F_1(w) - F_2(w)| \asymp a^{k-1} \asymp \sigma^{k-2}a$, and can be checked using our approximations 1,2,3).

1. When $\sigma \prec a$, we will prove proposition 1. Remember that we can't query in $-\sigma \leq w \leq 0$.

   (a) When $0 \leq w \leq \sigma$, we have

$$\begin{aligned} F_1(w) = (m_1 \star U)(w) &= \int_{w-\sigma}^{0} (1/2 - cx|x|^{k-2})dx/2\sigma + \int_{0}^{w+\sigma} (1/2 + cx^{k-1})dx/2\sigma \\ &= 1/2 + \frac{c}{2\sigma k}[(w+\sigma)^k - (\sigma - w)^k] \\ &= 1/2 + \frac{c}{2\sigma k}\sigma^k[(1 + w/\sigma)^k - (1 - w/\sigma)^k] \\ &\approx 1/2 + c\sigma^{k-2}w \end{aligned}$$

$$\begin{aligned} F_2(w) = (m_2 \star U)(w) &= \int_{w-\sigma}^{w+\sigma} (1/2 - c(x-a)|x-a|^{k-2})dx/2\sigma \\ &= 1/2 - \frac{c}{2\sigma k}[(a - w - \sigma)^k - (a + \sigma - w)^k] \\ &\approx 1/2 - c(a-w)^{k-1} \end{aligned}$$

[Boundaries: $F_1(0) - \frac{1}{2} = 0, F_1(\sigma) - \frac{1}{2} \asymp \sigma^{k-1}, F_2(0) - \frac{1}{2} \asymp -a^{k-1}, F_2(\sigma) - \frac{1}{2} \asymp -a^{k-1}$].

$$F_1(w) - F_2(w) \preceq a^{k-1}$$

   (b) When $\sigma \leq w \leq a - \sigma$

$$\begin{aligned} F_1(w) = (m_1 \star U)(w) &= \int_{w-\sigma}^{w+\sigma} (1/2 + cx^{k-1})dx/2\sigma \\ &= 1/2 + \frac{c}{2\sigma k}[(w+\sigma)^k - (w-\sigma)^k] \\ &\approx 1/2 + cw^{k-1} \end{aligned}$$

$$\begin{aligned} F_2(w) = (m_2 \star U)(w) &= \int_{w-\sigma}^{w+\sigma} (1/2 - c(x-a)|x-a|^{k-2})dx/2\sigma \\ &= 1/2 - \frac{c}{2\sigma k}[(a - w - \sigma)^k - (a + \sigma - w)^k] \\ &\approx 1/2 - c(a-w)^{k-1} \end{aligned}$$

[Boundaries: $F_1(\sigma) - \frac{1}{2} \asymp \sigma^{k-1}, F_1(a-\sigma) - \frac{1}{2} \asymp a^{k-1}, F_2(\sigma) - \frac{1}{2} \asymp -a^{k-1}, F_2(a-\sigma) - \frac{1}{2} \asymp -\sigma^{k-1}$].

$$\begin{aligned} F_1(w) - F_2(w) &= cw^{k-1} + c(a-w)^{k-1} \\ &\leq c(a-\sigma)^{k-1} + c(a-\sigma)^{k-1} \\ &\preceq a^{k-1} \end{aligned}$$

(c) When $a-\sigma \le w \le a$

$$F_1(w) \approx 1/2 + cw^{k-1}$$

$$\begin{aligned} F_2(w) &= \int_{w-\sigma}^{a} (1/2 - c(x-a)|x-a|^{k-2})dx/2\sigma + \int_{a}^{w+\sigma} 1/2 + c(x-a)^{k-1}dx/2\sigma \\ &= 1/2 - \frac{c}{2\sigma k}[(a-w+\sigma)^k - (w+\sigma-a)^k] \\ &\approx 1/2 - c\sigma^{k-2}(a-w) \end{aligned}$$

[Boundaries: $F_1(a-\sigma)-\frac{1}{2} \asymp a^{k-1}, F_1(a)-\frac{1}{2} \asymp a^{k-1}, F_2(a-\sigma)-\frac{1}{2} \asymp -\sigma^{k-1}, F_2(a)-\frac{1}{2} = 0$]

$$\begin{aligned} F_1(w) - F_2(w) &\approx cw^{k-1} + c\sigma^{k-2}(a-w) \\ &\le ca^{k-1} + c\sigma^{k-2}\sigma \\ &\preceq a^{k-1} \end{aligned}$$

(d) When $a \le w \le a+\sigma$

$$F_1(w) \approx 1/2 + cw^{k-1}$$

$$F_2(w) \approx 1/2 + c\sigma^{k-2}(a-w)$$

[Boundaries: $F_1(a)-\frac{1}{2} \asymp a^{k-1}, F_1(a+\sigma)-\frac{1}{2} \asymp a^{k-1}, F_2(a)-\frac{1}{2} = 0, F_2(a+\sigma)-\frac{1}{2} \asymp \sigma^{k-1}$]

$$F_1(w) - F_2(w) \preceq a^{k-1}$$

(e) When $a+\sigma \le w \le \beta a - \sigma$

$$F_1(w) \approx 1/2 + cw^{k-1}$$

$$\begin{aligned} F_2(w) &= \int_{w-\sigma}^{w+\sigma} 1/2 + c(x-a)^{k-1}dx/2\sigma \\ &= 1/2 + \frac{c}{2\sigma k}[(w+\sigma-a)^k - (w-\sigma-a)^k] \\ &\approx 1/2 + c(w-a)^{k-1} \end{aligned}$$

[B: $F_1(a+\sigma)-\frac{1}{2} \asymp a^{k-1}, F_1(\beta a-\sigma)-\frac{1}{2} \asymp a^{k-1}, F_2(a+\sigma)-\frac{1}{2} \asymp \sigma^{k-1}, F_2(\beta a-\sigma)-\frac{1}{2} \asymp a^{k-1}$]

$$\begin{aligned} F_1(w) - F_2(w) &\approx cw^{k-1} - c(w-a)^{k-1} \\ &\le c(\beta a - \sigma)^{k-1} + c\sigma^{k-1} \\ &\le c(\beta^{k-1}+1)a^{k-1} \\ &\preceq a^{k-1} \end{aligned}$$

(f) When $\beta a - \sigma \le w \le \beta a + \sigma$

$$F_1(w) \quad \approx \quad 1/2 + cw^{k-1}$$

$$\begin{aligned} F_2(w) &= \int_{w-\sigma}^{\beta a} 1/2 + c(x-a)^{k-1}dx/2\sigma + \int_{\beta a}^{w+\sigma} 1/2 + x^{k-1}dx/2\sigma \\ &= 1/2 + \frac{c}{2\sigma k}[(\beta a - a)^k - (w-\sigma-a)^k + (w+\sigma)^k - (\beta a)^k] \end{aligned}$$

$[F_1(\beta a-\sigma)-\frac{1}{2} \asymp a^{k-1}, F_1(\beta a+\sigma)-\frac{1}{2} \asymp a^{k-1}, F_2(\beta a-\sigma)-\frac{1}{2} \asymp a^{k-1}, F_2(\beta a+\sigma)-\frac{1}{2} \asymp a^{k-1}]$

$$\begin{aligned} F_1(w) - F_2(w) &= cw^{k-1} + \frac{c}{2\sigma k}[(\beta^k - (\beta-1)^k)a^k + (w-\sigma-a)^k - (w-\sigma)^k] \\ &\le c(\beta+1)^{k-1}a^{k-1} + \frac{c}{2\sigma k}[(\beta a)^k - (\beta a - 2\sigma)^k] \\ &\quad -\frac{c}{2\sigma k}[(\beta-1)^k a^k - ((\beta-1)a-\sigma)^k] \\ &\approx c(\beta+1)^{k-1}a^{k-1} + \frac{c}{2\sigma k}[k(\beta a)^{k-1}2\sigma] - \frac{c}{2\sigma k}[k(\beta-1)^{k-1}a^{k-1}\sigma] \\ &= ca^{k-1}[(\beta+1)^{k-1} + \beta^{k-1} - \tfrac{1}{2}(\beta-1)^{k-1}] \\ &\asymp a^{k-1} \end{aligned}$$

(g) When $\beta a + \sigma \le w \le \beta a + 2\sigma$

$$F_1(w) \quad = \quad 1/2 + \frac{c}{2\sigma k}[(w+\sigma)^k - (w-\sigma)^k]$$

$$\begin{aligned} F_2(w) &= \int_{w-\sigma}^{\beta a+\sigma} 1/2 + c(x-a)^{k-1}dx/2\sigma + \int_{\beta a+\sigma}^{w+\sigma} 1/2 + cx^{k-1}dx/2\sigma \\ &= 1/2 + \frac{c}{2\sigma k}[(\beta a + \sigma - a)^k - (w-\sigma-a)^k + (w+\sigma)^k - (\beta a+\sigma)^k] \end{aligned}$$

$[F_1(\beta a+\sigma)-\frac{1}{2} \asymp a^{k-1}, F_1(\beta a+2\sigma)-\frac{1}{2} \asymp a^{k-1}, F_2(\beta a+\sigma)-\frac{1}{2} \asymp a^{k-1}, F_2(\beta a+2\sigma)-\frac{1}{2} \asymp a^{k-1}]$

$$\begin{aligned} F_1(w) - F_2(w) &= \frac{c}{2\sigma k}[(\beta a+\sigma)^k - (\beta a+\sigma-a)^k + (w-\sigma-a)^k - (w-\sigma)^k] \\ &\approx \frac{c}{2\sigma k}[(\beta a+\sigma)^{k-1}ka - (w-\sigma)^{k-1}ka] \\ &\le \frac{ca}{2\sigma}[(\beta a+\sigma)^{k-1} - (\beta a)^{k-1}] \\ &\approx \frac{ca}{2\sigma}[(\beta a)^{k-1}(1 + \frac{(k-1)\sigma}{\beta a}) - (\beta a)^{k-1}] \\ &= a^{k-1}[c\beta^{k-2}(k-1)/2] \\ &\asymp a^{k-1} \end{aligned}$$

(h) When $w \ge \beta a + 2\sigma$

$$F_1(w) = F_2(w)$$

That completes the proof of the first claim.

2. When $\sigma \succ a$, we will prove the second proposition.
   (a) When $-\sigma \leq w \leq 0$, we are not allowed to query here.
   (b) When $0 < w \leq \beta a$

$$\begin{aligned} F_1(w) = (m_1 \star U)(w) &= \int_{w-\sigma}^{0} (1/2 - cx|x|^{k-2})dx/2\sigma + \int_{0}^{w+\sigma} (1/2 + cx^{k-1})dx/2\sigma \\ &= 1/2 + \frac{c}{2\sigma k}[(w+\sigma)^k - (\sigma - w)^k] \\ &= 1/2 + \frac{c}{2\sigma k}\sigma^k[(1+w/\sigma)^k - (1-w/\sigma)^k] \\ &\approx 1/2 + c\sigma^{k-2}w \end{aligned}$$

Similarly $F_2(w) \approx 1/2 + c\sigma^{k-2}(w-a)$
[Boundaries: $F_1(0) - \frac{1}{2} = 0, F_1(\beta a) - \frac{1}{2} \asymp \sigma^{k-2}a, F_2(0) - \frac{1}{2} \asymp -\sigma^{k-2}a, F_2(\beta a) \asymp \sigma^{k-2}a$]

$$F_1(w) - F_2(w) \asymp \sigma_n^{k-2}a.$$

   (c) When $\beta a \leq w \leq \sigma$

$$\begin{aligned} F_1(w) = &= \int_{w-\sigma}^{0} (1/2 - cx|x|^{k-2})dx/2\sigma + \int_{0}^{w+\sigma} (1/2 + cx^{k-1})dx/2\sigma \\ &= 1/2 + \frac{c}{2\sigma k}[(w+\sigma)^k - (\sigma - w)^k] \\ &= 1/2 + \frac{c}{2\sigma k}\sigma^k[(1+w/\sigma)^k - (1-w/\sigma)^k] \\ &\approx 1/2 + c\sigma^{k-2}w \end{aligned}$$

$$\begin{aligned} F_2(w) &= \int_{w-\sigma}^{a} (1/2 - c(x-a)|x-a|^{k-2})\frac{dx}{2\sigma} + \int_{a}^{\beta a+\sigma} (1/2 + c(x-a)^{k-1})\frac{dx}{2\sigma} \\ &\quad + \int_{\beta a+\sigma}^{w+\sigma} 1/2 + cx^{k-1}\frac{dx}{2\sigma} \\ &= 1/2 + \frac{c}{2\sigma k}[-(\sigma + a - w)^k + (\beta a + \sigma - a)^k + (w+\sigma)^k - (\beta a + \sigma)^k] \\ &\approx 1/2 + \frac{c}{2\sigma k}[-\sigma^k(1 - \frac{k(w-a)}{\sigma}) + \sigma^k(1 + \frac{k(\beta-1)a}{\sigma}) + \sigma^k(1 + \frac{kw}{\sigma}) - \sigma^k(1 + \frac{k\beta a}{\sigma})] \\ &= 1/2 + \frac{c}{2}\sigma^{k-2}[w - a + (\beta - 1)a + w - \beta a] \\ &= 1/2 + c\sigma^{k-2}(w-a) \end{aligned}$$

[Boundaries: $F_1(\beta a) - \frac{1}{2} \asymp \sigma^{k-2}a, F_1(\sigma) - \frac{1}{2} \asymp \sigma^{k-1}, F_2(\beta a) \asymp \sigma^{k-2}a, F_2(\sigma) - \frac{1}{2} \asymp -\sigma^{k-2}a$]

$$F_1(w) - F_2(w) \asymp \sigma^{k-2}a$$

Specifically, verify the boundary at σ

$$
\begin{aligned}
F_1(\sigma) - F_2(\sigma) &= \frac{c}{2\sigma k}[a^k - (\beta a + \sigma - a)^k + (\beta a + \sigma)^k] \\
&= \frac{c}{2\sigma k}[a^k - \sigma^k(1 + k\frac{\beta a - a}{\sigma}) + \sigma^k(1 + k\frac{\beta a}{\sigma})] \\
&= \frac{c}{2\sigma k}[a^k + k\sigma^{k-1}a] \\
&\leq c\sigma^{k-2}a
\end{aligned}
$$

(d) When $\sigma \leq w \leq a + \sigma$

$$
\begin{aligned}
F_1(w) &= \int_{w-\sigma}^{w+\sigma} (1/2 + cx^{k-1})dx/2\sigma \\
&= 1/2 + \frac{c}{2\sigma k}[(w + \sigma)^k - (w - \sigma)^k]
\end{aligned}
$$

$$
\begin{aligned}
F_2(w) &= \int_{w-\sigma}^{a} (1/2 - c(x - a)|x - a|^{k-2})\frac{dx}{2\sigma} + \int_{a}^{\beta a+\sigma} (1/2 + c(x - a)^{k-1})\frac{dx}{2\sigma} \\
&\quad + \int_{\beta a+\sigma}^{w+\sigma} 1/2 + cx^{k-1}\frac{dx}{2\sigma} \\
&= 1/2 + \frac{c}{2\sigma k}[-(\sigma + a - w)^k + (\beta a + \sigma - a)^k + (w + \sigma)^k - (\beta a + \sigma)^k]
\end{aligned}
$$

$$
F_1(w) - F_2(w) = \frac{c}{2\sigma k}[(\sigma + a - w)^k - (\beta a + \sigma - a)^k - (w - \sigma)^k + (\beta a + \sigma)^k]
$$

Differentiating the above term with respect to w, gives $\frac{c}{2\sigma}[-(\sigma + a - w)^{k-1} - (w - \sigma)^{k-1}] \leq 0$ because $\sigma \leq w \leq a + \sigma$ and hence $F_1(w) - F_2(w)$ is decreasing with w. We already saw $F_1(\sigma) - F_2(\sigma) \leq c\sigma^{k-2}a$. We can also verify that at the other boundary,

$$
\begin{aligned}
F_1(a + \sigma) - F_2(a + \sigma) &= \frac{c}{2\sigma k}[-(\beta a + \sigma - a)^k - a^k + (\beta a + \sigma)^k] \\
&= \frac{c}{2\sigma k}[-a^k - \sigma^k(1 + k\frac{\beta a - a}{\sigma}) + \sigma^k(1 + k\frac{\beta a}{\sigma})] \\
&= \frac{c}{2\sigma k}[-a^k + k\sigma^{k-1}a] \\
&\leq \frac{c}{2}\sigma^{k-2}a
\end{aligned}
$$

(e) When $\sigma + a \leq w \leq \beta a + \sigma$

$$
\begin{aligned}
F_1(w) &= \int_{w-\sigma}^{w+\sigma} (1/2 + cx^{k-1})dx/2\sigma \\
&= 1/2 + \frac{c}{2\sigma k}[(w + \sigma)^k - (w - \sigma)^k]
\end{aligned}
$$

$$\begin{aligned} F_2(w) &= \int_{w-\sigma}^{\beta a+\sigma} (1/2 + c(x-a)^{k-1})\frac{dx}{2\sigma} + \int_{\beta a+\sigma}^{w+\sigma} 1/2 + cx^{k-1}\frac{dx}{2\sigma} \\ &= 1/2 + \frac{c}{2\sigma k}[(\beta a+\sigma-a)^k - (w-\sigma-a)^k + (w+\sigma)^k - (\beta a+\sigma)^k] \end{aligned}$$

$$F_1(w) - F_2(w) = \frac{c}{2\sigma k}[(w-\sigma-a)^k - (\beta a+\sigma-a)^k - (w-\sigma)^k + (\beta a+\sigma)^k]$$

Differentiating with respect to w gives $\frac{c}{2\sigma}[(w-\sigma-a)^{k-1} - (w-\sigma)^{k-1}] \leq 0$ because $w-\sigma-a \leq w-\sigma$ and so $F_1 - F_2$ is decreasing with w. We know $F_1(a+\sigma) - F_2(a+\sigma) \leq \frac{c}{2}\sigma^{k-2}a$, and we can verify at the other boundary that

$$\begin{aligned} F_1(\beta a+\sigma) - F_2(\beta a+\sigma) &= \frac{c}{2\sigma k}[(\beta a-a)^k - (\beta a+\sigma-a)^k - (\beta a)^k + (\beta a+\sigma)^k] \\ &\approx \frac{c}{2\sigma k}[(\beta a-a)^k - (\beta a)^k - \sigma^k(1+k\frac{\beta a-a}{\sigma}) + \sigma^k(1+k\frac{\beta a}{\sigma})] \\ &= \frac{c}{2\sigma k}[(\beta a-a)^k - (\beta a)^k + k\sigma^{k-1}a] \\ &\leq \frac{c}{2}\sigma^{k-2}a \end{aligned}$$

(f) When $\beta a+\sigma \leq w \leq \beta a+2\sigma$

$$F_1(w) = 1/2 + \frac{c}{2\sigma k}[(w+\sigma)^k - (w-\sigma)^k]$$

$$\begin{aligned} F_2(w) &= \int_{w-\sigma}^{\beta a+\sigma} 1/2 + c(x-a)^{k-1}dx/2\sigma + \int_{\beta a+\sigma}^{w+\sigma} 1/2 + cx^{k-1}dx/2\sigma \\ &= 1/2 + \frac{c}{2k\sigma}[(\beta a+\sigma-a)^k - (w-\sigma-a)^k + (w+\sigma)^k - (\beta a+\sigma)^k] \end{aligned}$$

Hence

$$\begin{aligned} F_1(w) - F_2(w) &= \frac{c}{2\sigma k}[(\beta a+\sigma)^k - (\beta a+\sigma-a)^k + (w-\sigma-a)^k - (w-\sigma)^k] \\ &\approx \frac{c}{2\sigma k}[(\beta a+\sigma)^{k-1}ka - (w-\sigma)^{k-1}ka] \\ &\leq \frac{ca}{2\sigma}[(\beta a+\sigma)^{k-1} - (\beta a)^{k-1}] \\ &\approx c/2\sigma^{k-2}a \\ &\asymp \sigma^{k-2}a \end{aligned}$$

Alternately, by the same argument as in the previous case, differentiating with respect to w gives $\frac{c}{2\sigma}[(w-\sigma-a)^{k-1} - (w-\sigma)^{k-1}] \leq 0$ because $w-\sigma-a \leq w-\sigma$ and so $F_1 - F_2$ is decreasing with w. We know $F_1(\beta a+\sigma) - F_2(\beta a+\sigma) \leq \frac{c}{2}\sigma^{k-2}a$, and we can verify at the other endpoint that

$$F_1(\beta a+2\sigma) - F_2(\beta a+2\sigma) = 0$$

(g) When $w \geq \beta a+2\sigma$, $F_1(w) = F_2(w)$

That completes the proof of the second proposition.

4.7 Appendix: Convolved Regression Function, Justifying Eqs.4.8-4.11

For ease of presentation, let us assume the threshold is at 0, and define $m \in \mathcal{P}(c, C, k, \sigma)$ as

$$m(x) = \begin{cases} 1/2 + f(x) + \Delta(x) & \text{if } x \geq 0 \\ 1/2 - f(x) \text{ if } x < 0 \end{cases}$$

Due to assumption (M), $\Delta(x)$ must be 0 when $0 \leq x \leq \sigma$. Hence, the Taylor expansion of $\Delta(x)$ around $x = \sigma$ looks like

$$\Delta(x) = (x - \sigma)\Delta'(\sigma) + (x - \sigma)^2 \Delta''(\sigma) + ...$$

If one represents, as before, $F(x) = m \star U$, then directly from the definitions, it follows for $\delta > 0$ that

$$F(\delta) - F(0) = \int_{\sigma}^{\sigma+\delta} (1/2 + f(z) + \Delta(z)) \frac{dz}{2\sigma} - \int_{-\sigma}^{-\sigma+\delta} (1/2 - f(z)) \frac{dz}{2\sigma}$$

In particular, due to the form (T) of m, let $f = c_1 |x|^{k-1}$ for some $c \leq c_1 \leq C$ (we could also break f into parts where it has different c_1s but this is a technicality and does not change the behaviour). Then

$$\begin{aligned} F(\delta) - F(0) &= \frac{c_1}{2k\sigma}[(x^k)_{\sigma}^{\sigma+\delta} - (x^k)_{-\sigma}^{-\sigma+\delta}] + \int_{\sigma}^{\delta+\sigma} [(z-\sigma)\Delta'(\sigma) + (z-\sigma)^2 \Delta''(\sigma) + ...] \frac{dz}{2\sigma} \\ &= \frac{c_1}{2k\sigma}[(\sigma+\delta)^k - \sigma^k + (-\sigma+\delta)^k - (-\sigma)^k] + \frac{[(z-\sigma)^2]_{\sigma}^{\sigma+\delta}}{4\sigma} \Delta'(\sigma) + ... \\ &\approx c_1 \sigma^{k-2} \delta + \frac{\delta^2}{4\sigma} \Delta'(\sigma) + o(\delta^2) \end{aligned}$$

Thus we get behaviour of the form

$$F(t + h) \geq 1/2 + c\sigma^{k-2} h$$

One can derive similar results when $\delta < 0$.

The claims about WIDEHIST immediately follow from the above, but we can make them a little more explicit. First note that $F(w) = 1/2 + \frac{c}{\sigma_n}(w - t)$ for w close to t (in fact for $w \in [t - \sigma, t + \sigma]$), as seen in the Appendix. Consider a bin just outside the bins $i^* - 1, i^*, i^* + 1$, for instance bin $i = i^* + 2$ centered at b_i (note $b_i \geq t + h$), and let J be the set of points j that fall within $b_i \pm \sigma/2$. Define

$$\widehat{p}_i = \frac{1}{n\sigma/2R} \sum_{j \in J} \mathbb{I}(Y_j = +)$$

where $Y_j \in \{\pm 1\}$ are observations at points $j \in J$. Now, we have, since $P(Y_j = +) = F(j)$

$$\begin{aligned} \mathbb{E}[\widehat{p}_i] &= \frac{1}{n\sigma/2R} \sum_{j \in J} F(j) \\ &= \frac{1}{n\sigma/2R} \left[\sum_{j \in J} 1/2 + \tfrac{c}{\sigma_n}(X_j - t) \right] \\ &\approx 1/2 + \frac{1}{\sigma} \int_{b_i - t - \sigma/2}^{b_i - t + \sigma/2} \tfrac{c}{\sigma_n} z dz \\ &= 1/2 + \frac{c}{2\sigma^2} \left[(b_i - t + \sigma/2)^2 - (b_i - t - \sigma/2)^2 \right] \\ &= 1/2 + \tfrac{c}{\sigma_n}(b_i - t) \\ &\geq 1/2 + \tfrac{c}{\sigma_n} h \end{aligned}$$

4.8 Appendix: Justifying Claims in the Active Upper Bounds

Phase 1 $(k = 1)$**.** In the first phase of the algorithm, it is possible that $\sigma \preceq R_e/n$ but $\succeq R_e e^{-n}$ - in other words the noise may be small enough that passive learning cannot make out that we are in the errors-in-variables setting, and then the passive estimator will get a point error of $\frac{C_1 R_e}{n/E}$ in each of those epochs (as if there is no feature noise). This point error is to the best point in epoch e, which we can prove by induction is the true threshold t with high probability. Since it trivially holds in the first epoch $(t \in D_1 = [-1, 1])$, we assume that it is true in epoch $e - 1$. Then, in epoch e, the true threshold t is still the best point if the estimator x_{e-1} of epoch $e-1$ was within R_e of t, or in other words if $|x_{e-1} - t| \leq R_e$. This would definitely hold if $\frac{C_1 R_{e-1}}{n/E} \leq R_e$ i.e. $n \geq 2C_1 E = 2C_1 \lceil \log(1/\sigma) \rceil$, which is true since $\sigma \succ \exp\{-n/2C_1\}$. However, the algorithm cannot stay in this phase of $\sigma \preceq R_e/n$ this until the last epoch since $\sigma > R_{E+1} = R_E/2$.

Phase 2 ($k = 1$**).** When $\sigma \succeq R_e/n$, WIDEHIST gets an estimation error of $C_2 \sqrt{\frac{R_e \sigma}{n/E}}$ in epoch e. This error is the distance to the best point in epoch e, which is t by the following similar induction. In epoch e, t is still the best point only if $|x_{e-1} - t| \leq R_e$, i.e. $C_2^2 \frac{R_{e-1}\sigma}{n/E} \leq R_e^2$ i.e. $nR_e \geq 2C_2^2 E\sigma$ which holds since $R_e > \sigma$ for all $e \leq E$ and since $n \geq 2C_2^2 E$ ($\sigma \succ \exp\{-n/2C_2^2\}$ implies $E \leq n/2C_2^2$).

The final error of the algorithm is is $\sqrt{\frac{R_E \sigma}{n/E}} = \tilde{O}(\frac{\sigma}{\sqrt{n}})$ since $R_E < 2\sigma$.

Explanation for $k > 1$ Assume $\sigma \succ n^{-\frac{1}{2k-2}}$, otherwise active learning won't notice the feature noise, and so $\log(1/\sigma) \leq \frac{\log n}{(2k-2)}$. Choose total epochs $E = \lceil \log(\frac{1}{\sigma}) \rceil \leq \frac{\log n}{(2k-2)} \leq C \log n$ for some C. In each epoch of length n/E in a region of radius $R_e = 2^{-e+1}$, we get a passive bound of $C_1 \sqrt{\frac{R_e}{\sigma^{2k-3} n/E}}$ whenever $\sigma > (\frac{R_e}{n})^{\frac{1}{2k-1}}$. (This must happen at some $e \leq E = \lceil \log(\frac{1}{\sigma}) \rceil$ because $R_E = 2^{-E+1} < 2\sigma < \sigma\sigma^{2k-2}n$ since $\sigma \succ n^{-\frac{1}{2k-2}}$ and hence in the last epoch $\sigma > (\frac{R_E}{n})^{\frac{1}{2k-1}}$.) By the same logic as for $k = 1$, we need to verify that $|x_{e-1} - t| \leq R_e$ so that if t was in the search space in epoch $e - 1$ then it remains the in the search space in epoch e, i.e. we want to verify $C_1^2 \frac{R_{e-1}}{\sigma^{2k-3} n/E} \leq R_e^2 \Leftrightarrow \sigma^{2k-2} R_e \geq \frac{2C_1^2 E}{n} \sigma$ which is true since $R_e \geq \sigma$ and $\sigma^{2k-2} > 2C_1^2 E/n$. (By choice of $E = \lceil \log(\frac{1}{\sigma}) \rceil$, $R_e \geq R_E \geq \sigma \geq R_{E+1}$. Since $\sigma \succ n^{-\frac{1}{2k-2}}$ we get $\sigma^{2k-2} > 2C_1^2 E/n$ since $E \leq C \log n$.)

The final point error is given by the passive algorithm in the last epoch as $\sqrt{\frac{R_E}{\sigma^{2k-3} n/E}}$; since $R_E < 2\sigma$ and $E \leq C \log n$, this becomes $\preceq \frac{1}{\sigma^{k-2}} \sqrt{\frac{1}{n}}$.

Part II

Convex Optimization

Chapter 5

Margin-based classification : geometry, analysis and greedy algorithms

In this chapter[1], we study computational aspects of linear classification, which can be reduced to linear feasibility problems, where (for a given a matrix A) one tries to find $w : A^T w > \mathbf{0}$ or a probability distribution $p : Ap = \mathbf{0}$. We aim to deepen our understanding of a condition measure of A called *margin* that determines the difficulty of these problems. Geometrically, we establish new characterizations of the margin in terms of balls, cones and hulls, and tie them to old ones. Analytically, we present generalizations of Gordan's theorem, and variants of Hoffman's theorems, both using margins. Algorithmically, we prove new properties of classical iterative schemes, the Perceptron and Von-Neumann or Gilbert algorithms, whose rates depend on the margin and provide a unifying perspective with known results.

5.1 Introduction

Assume that we have a $d \times n$ matrix A representing n points $a_1, ..., a_n$ in $\mathbb{R}^d$. In this chapter, we will be concerned with linear feasibility problems that ask if there exists a vector $w \in \mathbb{R}^d$ that makes positive dot-product with every a_i, i.e.

$$?\exists w \ : \ A^T w > \mathbf{0}, \tag{P}$$

where boldfaced $\mathbf{0}$ is a vector of zeros. The corresponding algorithmic question is "if (P) is feasible, how quickly can we find a w that demonstrates (P)'s feasibility?".

Such problems abound in optimization as well as machine learning. For example, consider *binary linear classification* - given n points $x_i \in \mathbb{R}^d$ with labels $y_i \in \{+1, -1\}$, a classifier w is said to separate the given points if $w^T x_i$ has the same sign as y_i or succinctly $y_i(w^T x_i) > 0$ for all i. Representing $a_i = y_i x_i$ shows that this problem is a specific instance of (P).

We call (P) the *primal* problem, and (we will later see why) we define the *dual* problem (D) as:

$$?\exists p \geq \mathbf{0} \ : \ Ap = \mathbf{0}, p \neq \mathbf{0}, \tag{D}$$

and the corresponding algorithmic question is "if (D) is feasible, how quickly can we find a certificate p that demonstrates feasibility of (D)?".

Our aim is to deepen the geometric, algebraic and algorithmic understanding of the problems (P) and (D), tied together by a concept called *margin*. Geometrically, we provide intuition about ways to interpret

[1] See Ramdas and Peña [158].

margin in the primal and dual settings relating to various balls, cones and hulls. Analytically, we prove new margin-based versions of classical results in convex analysis like Gordan's and Hoffman's theorems. Algorithmically, we give new insights into the classical Perceptron algorithm. We begin with a gentle introduction to some of these concepts, before getting into the details.

Notation We assume that the a_i's are unit length according to the ℓ_2 (Euclidean) norm represented by $\|.\|$. To distinguish surfaces and interiors of balls more obviously to the eye in mathematical equations, we choose to denote Euclidean balls in $\mathbb{R}^d$ by $\bigcirc := \{w \in \mathbb{R}^d : \|w\| = 1\}$, $\bullet := \{w \in \mathbb{R}^d : \|w\| \leq 1\}$ and the probability simplex $\mathbb{R}^n$ by $\triangle := \{p \in \mathbb{R}^n : p \geq \mathbf{0}, \|p\|_1 = 1\}$. We denote the linear subspace spanned by A as $\text{lin}(A)$, and convex hull of A by $\text{conv}(A)$. Lastly, define $\bullet_A := \bullet \cap \text{lin}(A)$ and $r\bullet$ is the ball of radius r ($\bigcirc_A, r\bigcirc$ are similarly defined).

5.1.1 Margin ρ

The margin of the problem instance A is classically defined as

$$\begin{aligned} \rho \;&:=\; \sup_{w \in \bigcirc} \inf_{p \in \triangle} \; w^T A p \qquad (5.1) \\ &=\; \sup_{w \in \bigcirc} \inf_{i} \; w^T a_i . \end{aligned}$$

If there is a w such that $A^T w > \mathbf{0}$, then $\rho > 0$. If for all w, there is a point at an obtuse angle to it, then $\rho < 0$. At the boundary ρ can be zero. The $w \in \bigcirc$ in the definition is important – if it were $w \in \bullet$, then ρ would be non-negative, since $w = 0$ would be allowed.

This definition of margin was introduced by [78] who gave several geometric interpretations. It has since been extensively studied (for example, [170, 171]) as a notion of complexity and conditioning of the problem instance. Broadly, the larger its magnitude, the better conditioned the pair of feasibility problems (P) and (D) are, and the easier it is to find a witnesses of their feasibility. Ever since [220], the margin-based algorithms have been extremely popular with a growing literature in machine learning which it is not relevant to presently summarize.

In Sec.(5.2), we define an important and "corrected" variant of the margin, which we call *affine-margin*, that turns out to be the actual quantity determining convergence rates of iterative algorithms.

Gordan's Theorem This is a classical *theorem of the alternative*, see [23, 39]. It implies that exactly one of (P) and (D) is feasible. Specifically, it states that exactly one of the following statements is true.

1. There exists a w such that $A^T w > \mathbf{0}$.
2. There exists a $p \in \triangle$ such that $Ap = \mathbf{0}$.

This, and other separation theorems like Farkas' Lemma (see above references), are widely applied in algorithm design and analysis. We will later prove generalizations of Gordan's theorem using affine-margins.

Hoffman's Theorem The classical version of the theorem from [99] characterizes how close a point is to the solution set of the feasibility problem $Ax \leq b$ in terms of the amount of violation in the inequalities and a problem dependent constant. In a nutshell, if $\mathbb{S} := \{x | Ax \leq b\} \neq \emptyset$ then

$$\text{dist}(x, \mathbb{S}) \;\leq\; \tau \big\| [Ax - b]_+ \big\| \qquad (5.2)$$

where τ is the "Hoffman constant" and it depends on A but is *independent of* b. This and similar theorems have found extensive use in convergence analysis of algorithms - examples include [100], [77], [196].

[87] generalize this bound to any norms on the left and right hand sides of the above inequality. We will later prove theorems of a similar flavor for (P) and (D), where τ^{-1} will almost magically turn out to be the affine-margin. Such theorems are useful for proving rates of convergence of algorithms, and having the constant explicitly in terms of a familiar quantity is extremely useful.

5.1.2 Summary of Contributions

- **Geometric**: In Sec.5.2, we define the *affine-margin*, and argue why a subtle difference from Eq.(5.1) makes it the "right" quantity to consider, especially for problem (D). We then establish geometrical characterizations of the affine-margin when (P) is feasible as well as when (D) is feasible and connect it to well-known *radius theorems*. This is the chapter's appetizer.
- **Analytic**: Using the preceding geometrical insights, in Sec.5.3 we prove two generalizations of Gordan's Theorem to deal with alternatives involving the affine-margin when either (P) or (D) is strictly feasible. Building on this intuition further, in Sec.5.4, we prove several interesting variants of Hoffman's Theorem, which explicitly involve the affine-margin when either (P) or (D) is strictly feasible. This is the chapter's main course.
- **Algorithmic**: In Sec.5.5, we prove new properties of the Normalized Perceptron, like its margin-maximizing and margin-approximating property for (P) and dual convergence for (D). This is the chapter's dessert.

We end with a historical discussion relating Von-Neumann's and Gilbert's algorithms, and their advantage over the Perceptron.

5.2 From Margins to *Affine*-Margins

An important but subtle point about margins that is that the quantity determining the difficulty of solving (P) and (D) is actually *not* the margin as defined classically in Eq.(5.1), but the affine-margin which is the margin when w is restricted to lin(A), i.e. $w = A\alpha$ for some coefficient vector $\alpha \in \mathbb{R}^n$. The affine-margin is defined as

$$\begin{aligned} \rho_A \;&:=\; \sup_{w \in \bigcirc_A} \inf_{p \in \triangle} w^T A p \\ &=\; \sup_{\|\alpha\|_G = 1} \inf_{p \in \triangle} \alpha^T G p \end{aligned} \tag{5.3}$$

where $G = A^T A$ is a key quantity called the Gram matrix, and $\|\alpha\|_G = \sqrt{\alpha^T G \alpha}$ is easily seen to be a self-dual semi-norm.

Intuitively, when the problem (P) is infeasible but A is not full rank, i.e. lin(A) is not $\mathbb{R}^d$, then ρ will never be negative (it will always be zero), because one can always pick w as a unit vector perpendicular to lin(A), leading to a zero dot-product with every a_i. Since no matter how easily inseparable A is, the margin is always zero if A is low rank, this definition does not capture the difficulty of verifying linear infeasibility.

Similarly, when the problem (P) is feasible, it is easy to see that searching for w in directions perpendicular to A is futile, and one can restrict attention to lin(A), again making this the right quantity in some sense. For clarity, we will refer to

$$\rho_A^+ \;:=\; \max\{0, \rho_A\} \;\;;\;\; \rho_A^- \;:=\; \min\{0, \rho_A\} \tag{5.4}$$

when the problem (P) is strictly feasible ($\rho_A > 0$) or strictly infeasible ($\rho_A < 0$) respectively.

We remark that when $\rho > 0$, we have $\rho_A^+ = \rho_A = \rho$, so the distinction really matters when $\rho_A < 0$, but it is still useful to make it explicit. One may think that if A is not full rank, performing PCA would get rid of the unnecessary dimensions. However, we often wish to only perform elementary operations on (possibly large matrices) A that are much simpler than eigenvector computations.

Instability of ρ_A^- compared to ρ

Unfortunately, the behaviour of ρ_A^- is quite finicky – unlike ρ_A^+ it is not stable to small perturbations when conv(A) is not full-dimensional. To be more specific, if (P) is strictly feasible and we perturb all the vectors by a small amount or add a vector that maintains feasibility, ρ_A^+ can only change by a small amount. However, if (P) is strictly *in*feasible and we perturb all the vectors by a small amount or add a vector that maintains infeasibility, ρ_A^- can change by a large amount.

For example, assume lin(A) is not full-dimensional, and $|\rho_A^-|$ is large. If we add a new vector v to A to form $A' = \{A \cup v\}$ where v has a even a tiny component $v^\perp$ orthogonal to lin(A), then $\rho_{A'}^-$ suddenly becomes zero. This is because it is now possible to choose a vector $w = v^\perp/\|v^\perp\|$ which is in lin(A'), and makes zero dot-product with A, and positive dot-product with v. Similarly, instead of adding a vector, if we perturb a given set of vectors so that lin(A) increases dimension, the negative margin can suddenly jump to zero.

Despite its instability and lack of "continuity", it is indeed this negative affine margin that determines rate of convergence of algorithms for (D).

5.2.1 Geometric Interpretations of ρ_A^+

The positive margin has many known geometric interpretations – it is the width of the feasibility cone, and also the largest ball centered on the unit sphere that can fit inside the dual cone ($w : A^Tw > \mathbf{0}$ is the dual cone of cone(A)) – see, for example [73] and [38]. Here, we provide a few more interpretations. Remember that $\rho_A^+ = \rho$ when Eq.(P) is feasible.

Proposition 9. *The distance of the origin to conv(A) is ρ_A^+.*

$$\rho_A^+ = \inf_{p\in\triangle} \|p\|_G = \inf_{p\in\triangle} \|Ap\| \tag{5.5}$$

Proof: When $\rho_A \le 0$, $\rho_A^+ = 0$ and Eq.(5.5) holds because (D) is feasible making the right hand side also zero. When $\rho_A > 0$,

$$\rho_A^+ = \sup_{w\in\bigcirc} \inf_{p\in\triangle} w^TAp = \sup_{w\in\bullet} \inf_{p\in\triangle} w^TAp = \inf_{p\in\triangle} \sup_{w\in\bullet} w^TAp = \inf_{p\in\triangle} \|Ap\|. \tag{5.6}$$

Note that the first two equalities holds when $\rho_A > 0$, the next by the minimax theorem, and the last by self-duality of $\|.\|$.

The quantity ρ_A^+ is also closely related to a particular instance of the Minimum Enclosing Ball (MEB) problem. While it is common knowledge that MEB is connected to margins (and support vector machines), it is possible to explicitly characterize this relationship, as we have done below.

Proposition 10. *The radius of the minimum enclosing ball of conv(A) is $\sqrt{1 - \rho_A^{+2}}$.*

Proof: It is a simple exercise to show that the following are the MEB problem, and its Lagrangian dual

$$\min_{c,r} \quad r^2 \ \text{ s.t. } \ \|c - a_i\|^2 \le r^2$$

$$\max_{p\in\triangle} \quad 1 - \|Ap\|^2.$$

The result then follows from Proposition 9.

Though we will not return to this point, one may note that the (Normalized) Perceptron and related algorithms that we introduce later yields a sequence of iterates that converge to the center of the MEB, and if the distance of the origin to conv(A) is zero (because $\rho_A < 0$), then the sequence of iterates coverges to the origin, and the MEB just ends up being the unit ball. Independently, note that the MEB is related to the concept of *coresets*, recently quite popular in machine learning (especially support vector machines), see [42, 150]. The margin is also closely related to a central quantity in convex geometry called the *support function* of a closed, convex set. The connection of margins with coresets and support functions is out of the scope of this chapter.

5.2.2 Geometric Interpretations of $|\rho_A^-|$

Proposition 11. *If $\rho_A \le 0$ then $|\rho_A^-|$ is the radius of the largest Euclidean ball centered at the origin that completely fits inside the relative interior of the convex hull of A. Mathematically,*

$$|\rho_A^-| \;\;=\;\; \sup\Big\{R \,\big|\, \|\alpha\|_G \le R \Rightarrow A\alpha \in \mathrm{conv}(A)\Big\}. \tag{5.7}$$

The proof is not particularly enlightening, and we leave it for Appendix 5.7. One might be tempted to deal with the usual margin and prove that

$$|\rho| \;=\; \sup\Big\{R \;\big|\; \|w\| \le R \Rightarrow w \in \mathrm{conv}(A)\Big\} \tag{5.8}$$

While the two definitions are equivalent for full-dimensional lin(A), they differ when lin(A) is not full-dimensional, which is especially relevant in the context of infinite dimensional reproducing kernel Hilbert spaces, but could even occur when A is low rank. In this case, Eq.(5.8) will always be zero since a full-dimensional ball cannot fit inside a finite-dimensional hull. The right thing to do is to only consider balls ($\|\alpha\|_G \le R$) in the linear subspace spanned by columns of A (or the relative interior of the convex hull of A) and not full-dimensional balls ($\|w\| \le R$). The reason it matters is that it is this altered $|\rho_A^-|$ that determines rates for algorithms and the complexity of problem (D), and not the classical margin in Eq.(5.1) as one might have expected.

"Radius Theorems"

Recall that $A\triangle = \{Ap : p \in \triangle\} = \mathrm{conv}(A)$, $\bullet_A = \bullet \cap \mathrm{lin}(A)$, and $R\bullet_A$ is just $\bullet_A$ of radius R. Since $\|\alpha\|_G \le R \;\Leftrightarrow\; \|A\alpha\| \le R \;\Leftrightarrow\; A\alpha \in R\bullet_A$, an enlightening restatement of Eq.(5.7) and Eq.(5.8) is

$$|\rho_A^-| = \sup\Big\{R \;\big|\; R\bullet_A \subseteq A\triangle\Big\}, \text{ and } |\rho| = \sup\Big\{R \;\big|\; R\bullet \subseteq A\triangle\Big\}.$$

It can be read as "largest radius (affine) ball that fits inside the convex hull". There is a nice parallel to the smallest (overall) and smallest positive singular values of a matrix. Using $A\bullet = \{Ax : x \in \bullet\}$ for brevity,

$$\sigma_{\min}^{+}(A) = \sup\Big\{R \;\big|\; R\bullet_A \subseteq A\bullet\Big\}, \text{ and } \sigma_{\min}(A) = \sup\Big\{R \;\big|\; R\bullet \subseteq A\bullet\Big\} \tag{5.9}$$

This highlights the role of the margin is a measure of conditioning of the linear feasibility systems (P) and (D). Indeed, there are a number of far-reaching extensions of the classical "radius theorem" of [59]. The latter states that the Euclidean distance from a square non-singular matrix $A \in \mathbb{R}^{n\times n}$ to the set of singular matrices in $\mathbb{R}^{n\times n}$ is precisely $\sigma_{\min}(A)$. In an analogous fashion, for the feasibility problems (P) and (D), the set Σ of *ill-posed* matrices A are those with $\rho = 0$. [38] show that for a given a matrix $A \in \mathbb{R}^{m \times n}$ with normalized columns, the margin is the largest perturbation of a row to get an ill-posed instance or the "distance to ill-posedness", i.e.

$$\min_{\tilde{A} \in \Sigma} \max_{i=1,\dots,n} \|a_i - \tilde{a}_i\| = |\rho|. \tag{5.10}$$

See [38, 171] for related discussions.

5.3 Gordan's Theorem with Margins

We would like to make quantitative statements about what happens when either of the alternatives is satisfied *easily* (with large positive or negative margin). There does not seem to be a similar result in the literature, though we did observe a technical report by [127] which derives an approximate Farkas' Lemma, which is mathematically different but in the same spirit as the theorem below.

Note that without our preceding geometrical intuition, it is extremely difficult to conjecture what the statement of the following alternatives might possibly be. The previous propositions also vastly simplify this theorem's proof, which if presented directly would seem unmotivated and unnatural. We hope that just as Gordan's theorem has found innumerable uses, one may also find our generalizations, as well as their geometrical interpretations, useful.

Theorem 12. *For any problem instance A and any constant $\gamma \geq 0$,*

1. *Either $\exists w \in \bigcirc_A$ s.t. $A^T w > \mathbf{0}$, or $\exists p \in \triangle$ s.t. $Ap = \mathbf{0}$.*
2. *Either $\exists w \in \bigcirc_A$ s.t. $A^T w > \gamma$, or $\exists p \in \triangle$ s.t. $\|Ap\| \leq \gamma$.*
3. *Either $\exists w \in \bigcirc_A$ s.t. $A^T w > -\gamma$, or $\forall v \in \gamma \bullet_A$ $\exists p_v \in \triangle$ s.t. $v = Ap_v$.*

Proof: The first statement is the usual form of Gordan's Theorem. It is also a particular case of the other two when $\gamma = 0$. Thus, we will prove the other two:

2. If the first alternative does not hold, then from the definition of ρ_A it follows that $\rho_A \leq \gamma$. In particular, $\rho_A^+ \leq \gamma$. To finish, observe that by Proposition 9 there exists $p \in \triangle$ such that

$$\|Ap\| = \rho_A^+ \ \leq \ \gamma. \tag{5.11}$$

3. Analogously to the previous case, if the first alternative does not hold, then $\rho_A \leq -\gamma$. In particular, it captures

$$|\rho_A^-| \geq \gamma. \tag{5.12}$$

Observe that by Proposition 11, every point $v \in \gamma \bullet_A$ must be inside $\mathrm{conv}(A)$, that is, $v = Ap_v$ for some distribution $p_v \in \triangle$.

One can similarly argue that in each case if the first alternative is true, then the second must be false.

In the spirit of radius theorems introduced in the previous section, the statements in Theorem 12 can be equivalently written in the following succinct forms:

1'. Either $\{w \in \bigcirc_A : A^T w > \mathbf{0}\} \neq \emptyset$, or $\mathbf{0} \in A\triangle$

2'. Either $\{w \in \bigcirc_A : A^T w > \gamma\} \neq \emptyset$, or $\gamma \bullet_A \cap A\triangle \neq \emptyset$

3'. Either $\{w \in \bigcirc_A : A^T w > -\gamma\} \neq \emptyset$, or $\gamma \bullet_A \subseteq A\triangle$

As noted in the proof of Theorem 12, the first statement is a special case of the other two when $\gamma = 0$. In case 2, we have at least one witness p close to the origin, and in 3, we have an entire ball of witnesses close to the origin.

5.4 Hoffman's Theorem with Margins

Hoffman-style theorems are often useful to prove the convergence rate of iterative algorithms by characterizing the distance of a current iterate from a target set. For example, a Hoffman-like theorem was also proved by [100] (Lemma 2.3), where they use it to prove the linear convergence rate of the alternating direction method of multipliers, and in [77] (Lemma 4), where they use it to prove the linear convergence of a first order algorithm for calculating ϵ-approximate equilibria in zero sum games.

It is worth pointing out that Hoffman, in whose honor the theorem is named and also an author of [87] whose proof strategy we follow in the alternate proof of Theorem 15, himself has not noticed the intimate connection of the "Hoffman constant" (τ in Eq.(5.2)) to the positive and negative margin, as we elegantly and surprisingly present in our theorems below.

5.4.1 Hoffman's theorem for (D) when $\rho_A^- \neq 0$

Theorem 13. *Assume $A \in \mathbb{R}^{m\times n}$ is such that $|\rho_A^-| > 0$. For $b \in \mathbb{R}^m$ define the "witness" set $W = \{x \geq \mathbf{0} | Ax = b\}$. If $W \neq \emptyset$ then for all $x \geq \mathbf{0}$,*

$$\operatorname{dist}_1(x, W) \ \leq \ \frac{\|Ax - b\|}{|\rho_A^-|} \tag{5.13}$$

where $\operatorname{dist}_1(x, W)$ is the distance from x to W measured by the ℓ_1 norm $\|\cdot\|_1$.

Proof: Given $x \geq \mathbf{0}$ with $Ax \neq b$, consider a point

$$v \ = \ |\rho_A^-| \frac{b - Ax}{\|Ax - b\|} \tag{5.14}$$

Note that $\|v\| = |\rho_A^-|$ and crucially $v \in \operatorname{lin}(A)$ (since $b \in \operatorname{lin}(A)$ since $W \neq \emptyset$). Hence, by Theorem 12, there exists a distribution p such that $v = Ap$. Define

$$\bar{x} \ = \ x + p\frac{\|Ax - b\|}{|\rho_A^-|} \tag{5.15}$$

Then, by substitution for p and v one can see that

$$A\bar{x} = Ax + v\frac{\|Ax - b\|}{|\rho_A^-|} = Ax + (b - Ax) = b \tag{5.16}$$

Hence $\bar{x} \in W$, and $\operatorname{dist}_1(x, W) \leq \|x - \bar{x}\|_1 = \frac{\|Ax - b\|}{\rho_A^-}$.

The following variation (using witnesses only in Δ) on the above theorem also holds, but we omit its proof since it is similar to that of the above theorem.

Proposition 14. *Assume $A \in \mathbb{R}^{m\times n}$ is such that $|\rho_A^-| > 0$. Define the set of witnesses $W = \{p \in \triangle | Ap = \mathbf{0}\}$. Then at any $p \in \triangle$,*

$$\operatorname{dist}_1(p, W) \ \leq \ \frac{2\|Ap\|}{|\rho_A^-|} = \frac{2\|p\|_G}{|\rho_A^-|}. \tag{5.17}$$

5.4.2 Hoffman's theorem for (P) when $\rho_A^+ \neq 0$

Theorem 15. *Define $S = \{y | A^T y \geq c\}$. Then, for all $w \in \mathbb{R}^d$,*

$$\mathrm{dist}(w, S) \leq \frac{\|[A^T w - c]^- \|_\infty}{\rho_A^+}$$

where $\mathrm{dist}(w, S)$ is the $\| \cdot \|_2$-distance from w to S and $(x^-)_i = \min\{x_i, 0\}$.

Proof: Since $\rho_A^+ > 0$, there exists $\bar{w} \in \bigcirc_A$ with $A^T \bar{w} > \rho_A^+ \mathbf{1}$. Suppose, $A^T w \not> c\mathbf{1}$. Then we can add a multiple of $\bar{w}$ to w as follows. Let $a = [A^T w - c]^-$ where $(x^-)_i = \min\{x_i, 0\} = \max\{-x_i, 0\}$. Then one can see that

$$A^T \Big(w + \frac{\|a\|_\infty}{\rho_A^+} \bar{w} \Big) \ > \ A^T w + \| \max\{c - A^T w, \mathbf{0}\} \|_\infty \mathbf{1} \ \geq \ c.$$

Hence, $w + \frac{a}{\rho_A^+} \bar{w} \in S$ whose distance from w is precisely $\frac{\|a\|_\infty}{\rho_A^+}$.

The interpretation of the preceding theorem is that the distance to feasibility for the problem (P) is governed by the magnitude of the largest mistake and the positive affine-margin of the problem instance A.

We also provide an alternative proof of the theorem above, since proving the same fact from completely different angles can often yield insights. We follow the techniques of [87], though we significantly simplify it. This is perhaps a more classical proof style, and possibly more amenable to other bounds not involving the margin, and hence it is instructive for those unfamiliar with proving these sorts of bounds.

Proof: [Alternate Proof of Theorem 15] For any given w, define $a = -(A^T w - c)^- = (-A^T w + c)^+$ and hence note that $a \geq -(A^T w - c)$.

$$\begin{aligned}
\min_{A^T u \geq c} \|w - u\| &= \min_{A^T(u-w) \geq -A^T w + c} \|w - u\| = \min_{A^T z \geq -A^T w + c} \|z\| \\
&= \sup_{\|\mu\| \leq 1} \Big(\min_{A^T z \geq -A^T w + c} \mu^T z \Big) && (5.18) \\
&= \sup_{\|\mu\| \leq 1} \Big(\sup_{p \geq \mathbf{0}, Ap = \mu} p^T (-A^T w + c) \Big) && (5.19) \\
&= \sup_{\|p\|_G \leq 1, p \geq \mathbf{0}} p^\top (-A^T w + c) && (5.20) \\
&\leq \sup_{\|p\|_G \leq 1, p \geq \mathbf{0}} p^T a \ \leq \sup_{\|p\|_G \leq 1, p \geq \mathbf{0}} \|p\|_1 \|a\|_\infty && (5.21) \\
&= \frac{\|a\|_\infty}{\rho_A^+}
\end{aligned}$$

We used the self-duality of $\|.\|$ in Eq.(5.18), LP duality for Eq.(5.19), $\|Ap\| = \|p\|_G$ by definition for Eq.(5.20), and Holder's inequality in Eq.(5.21). The last equality follows because $\frac{1}{\rho_A^+} = \max_{\|p\|_G = 1, p \geq \mathbf{0}} \|p\|_1$, since $\rho_A^+ = \inf_{p \geq \mathbf{0}, \|p\|_1 = 1} \|p\|_G$.

5.5 The Perceptron Algorithm : New Insights

This was introduced and analysed by [176], [152], [21] to solve the primal (P), with many variants in the machine learning literature. The classical algorithm starts with $w_0 := \mathbf{0}$, and in iteration t performs

(choose any mistake)
$$\begin{aligned} a_i &: w_{t-1}^T a_i < 0. \\ w_t &\leftarrow w_{t-1} + a_i. \end{aligned}$$

A variant called Normalized Perceptron, which is a subgradient method, only updates on the worst mistake, and tracks a normalized w that which is a convex combination of a_i's.

(choose the worst mistake)
$$\begin{aligned} a_i &= \arg\min_{a_i} \{w_{t-1}^T a_i\} \\ w_t &\leftarrow \Big(1 - \tfrac{1}{t}\Big) w_{t-1} + \Big(\tfrac{1}{t}\Big) a_i. \end{aligned}$$

The best known property of the unnormalized Perceptron or the Normalized Perceptron algorithm is that when (P) is strictly feasible with margin ρ_A^+, it finds such a solution w in $1/\rho_A^{+2}$ iterations, as proved by [21, 152]. What is less obvious is that the Perceptron is actually *primal-dual* in nature, and we have not found any published work with the following proposition.

Proposition 16. *If (D) is feasible, the Perceptron algorithm (when normalized) yields an ϵ-certificate α_t for (D) in $1/\epsilon^2$ steps.*

Proof: When normalized, it yields a sequence of iterates $w_t = A\alpha_t,\ \alpha_t \in \triangle$ with $\|w_t\| \to 0$. To see this, observe that throughout the algorithm for $t \geq 1$ the iterate w_t satisfies $w_t = A\alpha_t$ with $\alpha_t \in \triangle$ because of way the sequence w_t is constructed. Furthermore, observe that if $\min_i a_i^T w_{t-1} \leq 0$ then

$$\|tw_t\|^2 = \|(t-1)w_{t-1}\|^2 + (t-1)a_i^T w_{t-1} + 1 \leq \|(t-1)w_{t-1}\|^2 + 1.$$

Thus $\|tw_t\|^2 < t$ as long as the algorithm has not found a solution to (P). In particular, when (D) is feasible (and hence (P) is infeasible) the iterates $w_t = A\alpha_t,\ \alpha_t \in \triangle$ satisfy $\|w_t\| = \|\alpha_t\|_G \leq \frac{1}{\sqrt{t}}$ and so we get an ϵ-certificate α_t for (D) in $1/\epsilon^2$ steps.

We prove one more nontrivial fact about the Normalized Perceptron that we have not found in the published literature - not only does it produce a *feasible* w in $O(1/\rho_A^{+2})$ steps, but on continuing to run the algorithm, w_t will approach the *optimal* w that maximizes margin, i.e. achieves margin ρ_A^+. This is actually *not* true with the classical Perceptron. The normalization in the following theorem is needed because $\|w_t\| = \|\alpha_t\|_G \neq 1$.

Theorem 17. *Assume (P) is feasible. If $w_t = A\alpha_t$ is the sequence of NP iterates with margin $\rho_t = \inf_{p\in\triangle} \frac{\alpha_t}{\|\alpha_t\|_G}^T Gp$, and the optimal point $\alpha_* := \arg\sup_{\|\alpha\|_G=1} \inf_{p\in\triangle} \alpha^T Gp$ achieves the optimal margin $\rho_A^+ = \inf_{p\in\triangle} \alpha_*^T Gp$, then*

$$\rho_A^+ - \rho_t \ \leq\ \left\| \frac{\alpha_t}{\|\alpha_t\|_G} - \alpha_* \right\|_G \ \leq\ 8/\rho_A^+\sqrt{t}.$$

Proof: For any α, let $\tilde{\alpha} := \alpha/\|\alpha\|_G$. The first inequality follows because the function

$$\rho(\tilde{\alpha}) = \inf_{p\in\triangle} \tilde{\alpha}^T Gp$$

is 1-Lipschitz with respect to the $\|.\|_G$. We can then argue that

$$\begin{aligned}\Big\|\frac{\alpha_t}{\|\alpha_t\|_G} - \alpha_*\Big\|_G &= \frac{1}{\|\alpha_t\|_G}\Big\|\alpha_t - \rho_A^+\alpha_* + (\rho_A^+ - \|\alpha_t\|_G)\alpha_*\Big\|_G \\ &\leq \frac{1}{\|\alpha_t\|_G}\Big(\|\alpha_t - \rho_A^+\alpha_*\|_G + |\rho_A^+ - \|\alpha_t\|_G|\Big) \\ &\leq \frac{1}{\rho_A^+}\Big(\|\alpha_t - \rho_A^+\alpha_*\|_G + |\rho_A^+ - \|\alpha_t\|_G|\Big) \qquad (5.22)\end{aligned}$$

where the first inequality follows by triangle inequality, and because $\|\alpha_*\|_G = 1$, and the second inequality holds because $\rho_A^+ = \inf_{p\in\triangle} \|p\|_G$ and $\alpha_t \in \triangle$ implies that

$$\|\alpha_t\|_G \;\geq\; \rho_A^+. \qquad (5.23)$$

The rest of the proof hinges on the fact that NP can be interpreted as a subgradient algorithm for the following problem:

$$\begin{aligned}\min_{\alpha\in\mathbb{R}^n} L(\alpha) &:= \min_{\alpha\in\mathbb{R}^n} \max_p\{-\alpha^T Gp\} + \tfrac{1}{2}\|\alpha\|_G^2 \qquad (5.24) \\ &= \min_{w\in\mathbb{R}^d} \max_i\{-w^T a_i\} + \tfrac{1}{2}\|w\|^2 \;=:\; \min_{w\in\mathbb{R}^d} L(w).\end{aligned}$$

We reproduce a short argument from [158, 196] which shows that $L(\alpha)$ is minimized at $\rho_A^+\alpha_*$. Let $\arg\min_\alpha L(\alpha) = t\alpha'$ for some $\|\alpha'\|_G = 1$ and some $t \in \mathbb{R}$. Substituting this into Eq.(5.24), we see that

$$\min_{\alpha\in\mathbb{R}^n} L(\alpha) = \min_{t>0}\{-t\rho_A^+ + \tfrac{1}{2}t^2\} = -\tfrac{1}{2}\rho_A^{+2}$$

achieved at $t = \rho_A^+$ and $\alpha' = \alpha_*$. Hence $\arg\min_\alpha L(\alpha) = \rho_A^+\alpha_*$.

Note that NP is a (nonstochastic) subgradient method for $L(\alpha)$, which is 1-strongly convex with respect to $\|.\|_G$, and all its subgradients are bounded by 2 (since every iterate satisfies $\|w_t\| \leq 1$ and all $\|a_i\| \leq 1$). Hence, substituting $c = \lambda = 1$ and $G = 2$ in Lemma 1 from [156], we can infer that the rate of convergence of iterates α_t towards the optimum $\rho_A^+\alpha_*$ is

$$\begin{aligned}\|\alpha_t - \rho_A^+\alpha_*\|_G &\leq 4/\sqrt{t} \\ \Rightarrow \|\alpha_t\|_G - \rho_A^+ &\leq 4/\sqrt{t}. \qquad (5.25)\end{aligned}$$

This yields the required bound of $8/\rho_A^+\sqrt{t}$ when plugged into Eq.(5.22).

Proposition 18. *The Normalized Perceptron gives an ϵ-approximation to the value of the positive margin in $16/\epsilon^2$ steps. Specifically,*

$$\|w_{16/\epsilon^2}\| - \epsilon \;\leq\; \rho_A^+ \;\leq\; \|w_{16/\epsilon^2}\|$$

Proof: The proof follows from Eq.(5.25) and Eq.(5.23), which imply that w_t satisfies

$$\rho_A^+ \leq \|w_t\| \leq \rho_A^+ + 4/\sqrt{t}$$

whose rearrangement with $t = 16/\epsilon^2$ completes the proof.

Interestingly, the question of finding elementary algorithms to estimate $|\rho_A^-|$ is open, which is surprising since estimating the smallest non-negative singular value is not hard (and we have earlier noted the similarity between the two).

5.6 Discussion

Von-Neumann or Gilbert Algorithm for (D)

Von-Neumann described an iterative algorithm for solving dual (D) in a private communication with Dantzig in 1948, which was subsequently analyzed by the latter, but only published in [45], and goes by the name of Von-Neumann's algorithm in optimization circles. Independently, Gilbert described an essentially identical algorithm in [76], that goes by the name of Gilbert's algorithm in the computational geometry literature. We respect the independent findings in different literatures, and refer to it as the Von-Neumann-Gilbert (VNG) algorithm. It starts from a point in conv(A), say $w := a_1$ and loops:

(choose furthest point) $$a_i \;=\; \arg\max_{a_i}\{\|w_{t-1} - a_i\|\}$$

(line search, $\lambda \in [0,1]$) $$w_t \;\leftarrow\; \arg\min_{w_\lambda} \|w_\lambda\|; \;\; w_\lambda = \lambda w_{t-1} + (1-\lambda)a_i$$

Dantzig's paper showed that the Von-Neumann-Gilbert (VNG) algorithm can produce an ϵ-approximate solution (p such that $\|Ap\| \leq \epsilon$) to (D) in $1/\epsilon^2$ steps, establishing it as a dual algorithm as conjectured by Von-Neumann. Though designed for (D), [64] proved that when (P) is feasible, VNG also produces a feasible w in $1/\rho_A^{+2}$ steps and hence VNG is also primal-dual like the Perceptron (as proved in Proposition 16). We can prove results analagous to Theorem 17 and Proposition 18 for VNG as well.

Nesterov was the first to point out in a private note to [65] that VNG is a Frank-Wolfe algorithm for

$$\min_{p \in \triangle} \; \|Ap\| \tag{5.26}$$

Note that Eq.(5.24) is a relaxed version of Eq.(5.3), and also that Eq.(5.26) and Eq.(5.3) are Lagrangian duals of each other as seen in Eq.(5.6). In this light, it is not surprising that NP and VNG algorithms have such similar properties. Moreover, [12] recently pointed out the strong connection via duality between subgradient and Frank-Wolfe methods.

However, VNG possesses one additional property. Restating a result of [64] - if $|\rho_A^-| > 0$, then VNG has linear convergence, finding an ϵ-approximate solution to (D) in $O\Big(\frac{1}{|\rho_A^-|^2}\log\left(\frac{1}{\epsilon}\right)\Big)$ steps, and this fact has a simple geometrical proof, summarised in Fig.5.4 in Appendix 5.7). Hence, VNG can converge linearly with strict infeasibility of (P), but NP cannot. Nevertheless, NP and VNG can both be seen geometrically as trying to represent the center of circumscribing or inscribing balls (in (P) or (D)) of conv(A) as a convex combination of input points.

Summary

In this chapter, we advance and unify our understanding of margins through a slew of new results and connections to old ones. First, we point out the correctness of using the affine margin, deriving its relation to the smallest ball enclosing conv(A), and the largest ball within conv(A). We proved generalizations of Gordan's theorem, whose statements were conjectured using the preceding geometrical intuition. Using these tools, we then derived interesting variants of Hoffman's theorems that explicitly use affine margins. We ended by proving that the Perceptron algorithm turns out to be primal-dual, its iterates are margin-maximizers, and the norm of its iterates are margin-approximators.

Right from his seminal introductory paper in the 1950s, Hoffman-like theorems have been used to prove convergence rates and stability of algorithms. Our theorems and also their proof strategies can be very useful in this regard, since such Hoffman-like theorems can be very challenging to conjecture and prove (see [100] for example). Similarly, Gordan's theorem has been used in a wide array of settings in optimization, giving a precedent for the possible usefulness of our generalization. Lastly, large margin classification is now such an integral machine learning topic, that it seems fundamental that we unify our understanding of the geometrical, analytical and algorithmic ideas behind margins.

5.7 Figures

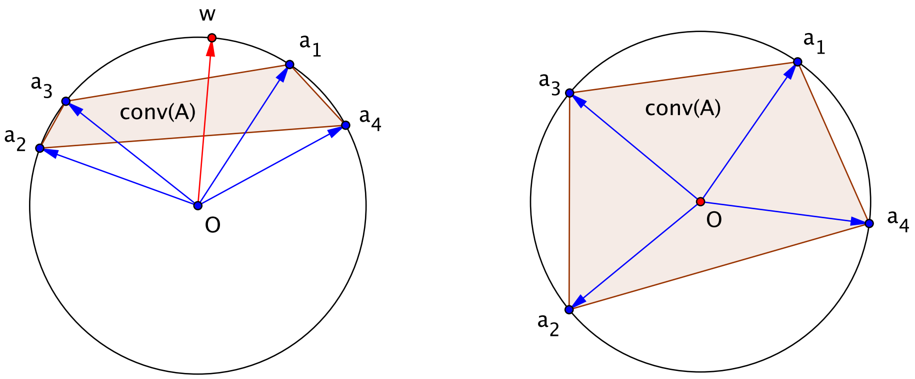


Figure 5.1: Gordan's Theorem: Either there is a w making an acute angle with all points, or the origin is in their convex hull. (note $\|a_i\| = 1$)

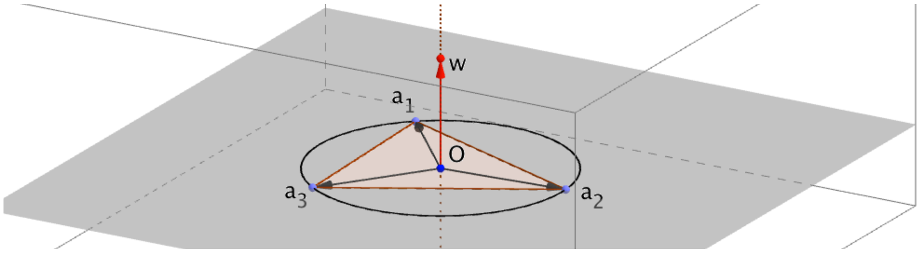


Figure 5.2: When restricted to lin(A), the margin is strictly negative. Otherwise, it would be possible to choose w perpendicular to lin(A), leading to a zero margin.

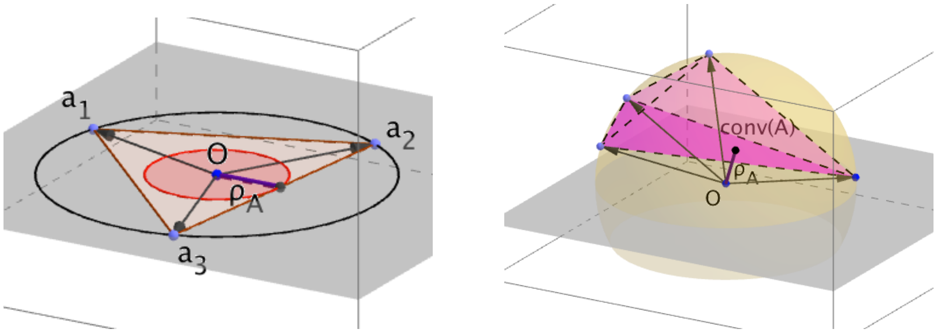


Figure 5.3: Left: ρ_A^- is the radius of the largest ball centered at origin, inside the relative interior of conv(A). Right: ρ_A^+ is the distance from origin to conv(A).

Supporting Proofs

Proof of Proposition 11

Proof: We split the proof into two parts, one for each inequality.

(1) For inequality $\geq$. Choose any R such that $A\alpha \in \mathrm{conv}(A)$ for any $\|\alpha\|_G \leq R$. Given an arbitrary $\|\alpha'\|_G = 1$, put $\tilde{\alpha} := -R\alpha'$.

By our assumption on R, since $\|\tilde{\alpha}\|_G = R$, we can infer that $A\tilde{\alpha} \in \mathrm{conv}(A)$ implying there exists a $\tilde{p} \in \triangle$ such that $A\tilde{\alpha} = A\tilde{p}$. Also

$$\begin{array}{lrcl} & \alpha'^T G\tilde{p} & = & \alpha'^T G\tilde{\alpha} \\ & & = & -R\|\alpha'\|_G^2 \ = \ -R. \\ \Rightarrow & \inf_{p\in\triangle} \alpha'^T Gp & \leq & -R. \\ \Rightarrow & \sup_{\|\alpha\|_G=1} \inf_{p\in\triangle} \alpha^T Gp & \leq & -R, \\ \text{(in other words)} & |\rho_A^-| & \geq & R. \end{array}$$

(2) For inequality $\leq$. It suffices to show $\|\alpha\|_G \leq \rho_A^- \Rightarrow A\alpha \in \mathrm{conv}(A)$. We will prove the contrapositive $A\alpha \notin \mathrm{conv}(A) \Rightarrow \|\alpha\|_G > |\rho_A^-|$. Since $\mathrm{conv}(A)$ is closed and convex, if $A\alpha \notin \mathrm{conv}(A)$, then there exists a hyperplane, say (β, b), with normal $\|\beta\|_G = 1$ (i.e. $\|A\beta\| \in \bigcirc$) in $\mathrm{lin}(A)$ and constant $b \in \mathbb{R}$ that separates $A\alpha$ and $\mathrm{conv}(A)$, i.e. for all $p \in \triangle$,

$$\begin{array}{rcl} \beta^T G\alpha & < & b \text{ and } \beta^T Gp \ \geq \ b, \\ \text{i.e. } \beta^T G\alpha & < & \inf_{p\in\triangle} \beta^T Gp \\ & \leq & \sup_{\|\beta\|_G=1} \inf_{p\in\triangle} \beta^T Gp = \rho_A^-. \\ \text{Since } \rho_A^- < 0, \quad |\rho_A^-| & < & |\beta^T G\alpha| \\ & \leq & \|\beta\|_G\|\alpha\|_G \ = \ \|\alpha\|_G. \end{array}$$

Proof of linear convergence of VNG ($|\rho_A^-| > 0$)

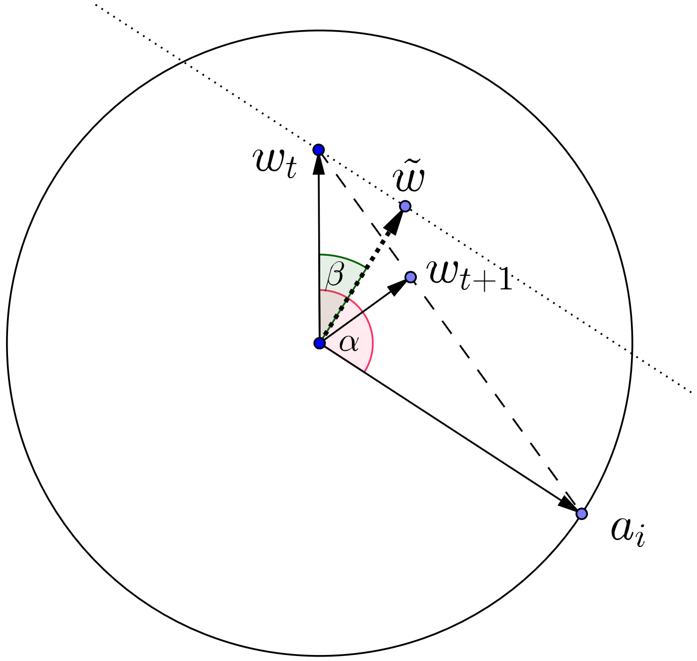


Figure 5.4: There is always a point a_i such that $\cos\alpha = \frac{w_t}{\|w_t\|} \cdot a_i \leq \rho_A^-$ or $|\cos\alpha| \geq |\rho_A^-|$. VNG sets w_{t+1} to be the nearest point to the origin on the (hyphenated) line joining w_t with a_i. Consider $\tilde{w}$ as the nearest point to the origin on a (dotted) line parallel to a_i through w_t. Note $(\pi/2 - \beta) + \alpha = \pi$ (internal angles of parallel lines). Then, $\|w_{t+1}\| \leq \|\tilde{w}\| = \|w_t\| \cos\beta = \|w_t\| \sin\alpha = \|w_t\|\sqrt{1 - \cos^2\alpha} \leq \|w_t\|\sqrt{1 - |\rho_A^-|^2}$. Hence $\|w_t\| \leq \epsilon$ in $O\Big(\frac{1}{|\rho_A^-|^2} \log\left(\frac{1}{\epsilon}\right)\Big)$ iterations.

Chapter 6

Margin-based classification : kernelized smoothed primal-dual algorithms

We focus on the problem of finding a non-linear classification function that lies in a Reproducing Kernel Hilbert Space (RKHS) both from the primal point of view (finding a perfect separator when one exists) and the dual point of view (giving a certificate of non-existence), with special focus on generalizations of two classical schemes - the Perceptron (primal) and Von-Neumann (dual) algorithms.

We cast our problem as one of maximizing the regularized normalized hard-margin (ρ) in an RKHS and rephrase it in terms of a Mahalanobis dot-product/semi-norm associated with the kernel's (normalized and signed) Gram matrix. We derive an accelerated smoothed algorithm with a convergence rate of $\frac{\sqrt{\log n}}{\rho}$ given n separable points, which is strikingly similar to the classical kernelized Perceptron algorithm whose rate is $\frac{1}{\rho^2}$. When no such classifier exists, we prove a version of Gordan's separation theorem for RKHSs, and give a reinterpretation of negative margins. This allows us to give guarantees for a primal-dual algorithm that halts in $\min\{\frac{\sqrt{n}}{|\rho|}, \frac{\sqrt{n}}{\epsilon}\}$ iterations with a perfect separator in the RKHS if the primal is feasible or a dual ϵ-certificate of near-infeasibility.

6.1 Introduction

We are interested in the problem of finding a non-linear separator for a given set of n points $x_1, ..., x_n \in \mathbb{R}^d$ with labels $y_1, ..., y_n \in \{\pm 1\}$. Finding a linear separator can be stated as the problem of finding a unit vector $w \in \mathbb{R}^d$ (if one exists) such that for all i

$$y_i(w^\top x_i) \geq 0 \quad \text{i.e.} \quad \text{sign}(w^\top x_i) = y_i. \tag{6.1}$$

This is called the primal problem. In the more interesting non-linear setting, we will be searching for functions f in a Reproducing Kernel Hilbert Space (RKHS) $\mathcal{F}_K$ associated with kernel K (to be defined later) such that for all i

$$y_i f(x_i) \geq 0. \tag{6.2}$$

We say that problems (6.1), (6.2) have an unnormalized margin $\rho > 0$, if there exists a unit vector w, such that for all i,

$$y_i(w^\top x_i) \geq \rho \quad \text{or} \quad y_i f(x_i) \geq \rho.$$

True to the chapter's title, margins of non-linear separators in an RKHS will be a central concept, and we will derive interesting *smoothed accelerated* variants of the Perceptron algorithm that have convergence rates (for the aforementioned primal and a dual problem introduced later) that are inversely proportional to the RKHS-margin as opposed to inverse squared margin for the Perceptron.

The linear setting is well known by the name of linear feasibility problems - we are asking if there exists any vector w which makes an acute angle with all the vectors $y_i x_i$, i.e.

$$(XY)^\top w > \mathbf{0}_n, \tag{6.3}$$

where $Y := diag(y), X := [x_1, ..., x_n]$. This can be seen as finding a vector w inside the dual cone of $cone\{y_i x_i\}$.

When normalized, as we will see in the next section, the margin is a well-studied notion of conditioning for these problems. It can be thought of as the width of the feasibility cone as in [73], a radius of well-posedness as in [38], and its inverse can be seen as a special case of a condition number defined by [171] for these systems.

Related Work

In this chapter we focus on the famous Perceptron algorithm [176] and the less-famous Von-Neumann algorithm [45] that we introduce in later sections. As mentioned in [64], in a technical report by the same name, Nesterov pointed out in a note to the authors that the latter is a special case of the now-popular Frank-Wolfe algorithm.

Our work builds on [195, 196] from the field of optimization - we generalize the setting to learning functions in RKHSs, extend the algorithms, simplify proofs, and simultaneously bring new perspectives to it. There is extensive literature around the Perceptron algorithm in the learning community; we restrict ourselves to discussing only a few directly related papers, in order to point out the several differences from existing work.

We provide a general unified proof in the Appendix which borrows ideas from accelerated smoothing methods developed by Nesterov [148] - while this algorithm and others by [145], [180] can achieve similar rates for the same problem, those algorithms do not possess the simplicity of the Perceptron or Von-Neumann algorithms and our variants, and also don't look at the infeasible setting or primal-dual algorithms.

Accelerated smoothing techniques have also been seen in the learning literature like in [215] and many others. However, most of these deal with convex-concave problems where both sets involved are the probability simplex (as in game theory, boosting, etc), while we deal with hard margins where one of the sets is a unit ℓ_2 ball. Hence, their algorithms/results are not extendable to ours trivially. This work is also connected to the idea of ϵ-coresets [42], though we will not explore that angle.

A related algorithm is called the Winnow [128] - this works on the ℓ_1 margin and is a saddle point problem over two simplices. One can ask whether such accelerated smoothed versions exist for the Winnow. The answer is in the affirmative - however such algorithms look completely different from the Winnow, while in our setting the new algorithms retain the simplicity of the Perceptron.

Chapter Outline

Section 6.2 will introduce the Perceptron and Normalized Perceptron algorithm and their convergence guarantees for linear separability, with specific emphasis on the unnormalized and normalized margins. Section 6.3 will then introduce RKHSs and the Normalized Kernel Perceptron algorithm, which we interpret as a subgradient algorithm for a regularized normalized hard-margin loss function. Section 6.4 describes the Smoothed Normalized Kernel Perceptron algorithm that works with a smooth approximation to the original loss function, and outlines the argument for its faster convergence rate. Section 6.5

discusses the non-separable case and the Von-Neumann algorithm, and we prove a version of Gordan's theorem in RKHSs. We finally give an algorithm in Section 6.6 which terminates with a separator if one exists, and with a dual certificate of near-infeasibility otherwise, in time inversely proportional to the margin. We end with a discussion and some open problems.

6.2 Linear Feasibility Problems

6.2.1 Perceptron

The classical perceptron algorithm can be stated in many ways, one is in the following form

Algorithm 4 Perceptron

```
Initialize $w_0 = 0$
for $k = 0, 1, 2, 3, ...$ do
    if sign$(w_k^\top x_i) \neq y_i$ for some $i$ then
        $w_{k+1} := w_k + y_i x_i$
    else
        Halt: Return $w_k$ as solution
    end if
end for
```

It comes with the following classic guarantee as proved by [21] and [152]: *If there exists a unit vector $u \in \mathbb{R}^d$ such that $YX^\top u \geq \rho > 0$, then a perfect separator will be found in $\frac{\max_i \|x_i\|_2^2}{\rho^2}$ iterations/mistakes.*

The algorithm works when updated with any arbitrary point (x_i, y_i) that is misclassified; it has the same guarantees when w is updated with the point that is misclassified by the largest amount, $\arg\min_i y_i w^\top x_i$. Alternately, one can define the probability distribution over examples

$$p(w) = \arg\min_{p \in \Delta_n} \langle YX^\top w, p \rangle, \tag{6.4}$$

where Δ_n is the n-dimensional probability simplex.

Intuitively, p picks the examples that have the lowest margin when classified by w. One can also normalize the updates so that we can maintain a probability distribution over examples used for updates from the start, as seen below:

Algorithm 5 Normalized Perceptron

```
Initialize $w_0 = 0, p_0 = 0$
for $k = 0, 1, 2, 3, ...$ do
    if $YX^\top w_k > 0$ then
        Exit, with $w_k$ as solution
    else
        $\theta_k := \frac{1}{k+1}$
        $w_{k+1} := (1 - \theta_k) w_k + \theta_k XYp(w_k)$
    end if
end for
```

Remark. Normalized Perceptron has the same guarantees as perceptron - the Perceptron can perform its update *online* on *any* misclassified point, while the Normalized Perceptron performs updates on the *most* misclassified point(s), and yet there does not seem to be any change in performance. However, we will soon see that the ability to see all the examples at once gives us much more power.

6.2.2 Normalized Margins

If we normalize the data points by the ℓ_2 norm, the resulting mistake bound of the perceptron algorithm is slightly different. Let X_2 represent the matrix with columns $x_i/\|x_i\|_2$. Define the unnormalized and normalized margins as

$$\begin{aligned} \rho &:= \sup_{\|w\|_2=1} \inf_{p\in\Delta_n} \langle YX^\top w, p\rangle, \\ \rho_2 &:= \sup_{\|w\|_2=1} \inf_{p\in\Delta_n} \langle YX_2^\top w, p\rangle. \end{aligned}$$

Remark. Note that we have $\sup_{\|w\|_2=1}$ in the definition, this is equivalent to $\sup_{\|w\|_2\leq 1}$ iff $\rho_2 > 0$.

Normalized Perceptron has the following guarantee on X_2: *If $\rho_2 > 0$, then it finds a perfect separator in $\frac{1}{\rho_2^2}$ iterations.*

Remark. Consider the max-margin separator u^* for X (which is also a valid perfect separator for X_2). Then

$$\begin{aligned} \frac{\rho}{\max_i \|x_i\|_2} &= \min_i \left(\frac{y_i x_i^\top u^*}{\max_i \|x_i\|_2} \right) \leq \min_i \left(\frac{y_i x_i^\top u^*}{\|x_i\|_2} \right) \\ &\leq \sup_{\|u\|_2=1} \min_i \left(\frac{y_i x_i^\top u}{\|x_i\|_2} \right) = \rho_2. \end{aligned}$$

Hence, it is always better to normalize the data as pointed out in [80]. This idea extends to RKHSs, motivating the normalized Gram matrix considered later.

Example Consider a simple example in $\mathbb{R}^2_+$. Assume that $+$ points are located along the line $6x_2 = 8x_1$, and the $-$ points along $8x_2 = 6x_1$, for $1/r \leq \|x\|_2 \leq r$, where $r > 1$. The max-margin linear separator will be $x_1 = x_2$. If all the data were normalized to have unit Euclidean norm, then all the $+$ points would all be at $(0.6, 0.8)$ and all the $-$ points at $(0.8, 0.6)$, giving us a normalized margin of $\rho_2 \approx 0.14$. Unnormalized, the margin is $\rho \approx 0.14/r$ and $\max_i \|x_i\|_2 = r$. Hence, in terms of bounds, we get a discrepancy of r^4, which can be arbitrarily large.

Winnow The question arises as to which norm we should normalize by. There is a now classic algorithm in machine learning, called Winnow [128] or Multiplicate Weights. It works on a slight transformation of the problem where we only need to search for $u \in \mathbb{R}^d_+$. It comes with some very well-known guarantees - *If there exists a $u \in \mathbb{R}^d_+$ such that $YX^\top u \geq \rho > 0$, then feasibility is guaranteed in $\|u\|_1^2 \max_i \|a_i\|_\infty^2 \log n/\rho^2$ iterations.* The appropriate notion of normalized margin here is

$$\rho_1 := \max_{w\in\Delta_d} \min_{p\in\Delta_n} \langle YX_\infty^\top w, p\rangle,$$

where X_∞ is a matrix with columns $x_i/\|x_i\|_\infty$. Then, the appropriate iteration bound is $\log n/\rho_1^2$. We will return to this ℓ_1-margin in the discussion section. In the next section, we will normalize by using the kernel appropriately.

6.3 Kernels and RKHSs

The theory of Reproducing Kernel Hilbert Spaces (RKHSs) has a rich history, and for a detailed introduction, refer to [184]. Let $K : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ be a symmetric positive definite kernel, giving rise to a Reproducing Kernel Hilbert Space $\mathcal{F}_K$ with an associated feature mapping at each point $x \in \mathbb{R}^d$ called $\phi_x : \mathbb{R}^d \to \mathcal{F}_K$ where $\phi_x(.) = K(x, .)$ i.e. $\phi_x(y) = K(x, y)$. $\mathcal{F}_K$ has an associated inner product $\langle \phi_u, \phi_v \rangle_K = K(u, v)$. For any $f \in \mathcal{F}_K$, we have $f(x) = \langle f, \phi_x \rangle_K$.

Define the normalized feature map

$$\tilde{\phi}_x = \frac{\phi_x}{\sqrt{K(x,x)}} \in \mathcal{F}_K \quad \text{and} \quad \tilde{\phi}_X := [\tilde{\phi}_{x_i}]_1^n.$$

For any function $f \in \mathcal{F}_K$, we use the following notation

$$Y\tilde{f}(X) := \langle f, Y\tilde{\phi}_X \rangle_K = [y_i \langle f, \tilde{\phi}_{x_i} \rangle_K]_1^n = \left[\frac{y_i f(x_i)}{\sqrt{K(x_i,x_i)}} \right]_1^n.$$

We analogously define the normalized margin here to be

$$\rho_K \;\; := \;\; \sup_{\|f\|_K = 1} \inf_{p \in \Delta_n} \left\langle Y\tilde{f}(X), p \right\rangle. \tag{6.5}$$

Consider the following regularized empirical loss function

$$L(f) = \left\{ \sup_{p \in \Delta_n} \left\langle -Y\tilde{f}(X), p \right\rangle \right\} + \tfrac{1}{2}\|f\|_K^2. \tag{6.6}$$

Denoting $t := \|f\|_K > 0$ and writing $f = t\left(\frac{f}{\|f\|_K}\right) = t\bar{f}$, let us calculate the minimum value of this function

$$\begin{aligned} \inf_{f \in \mathcal{F}_K} L(f) &= \inf_{t>0} \inf_{\|\bar{f}\|_K = 1} \sup_{p \in \Delta_n} \langle -\langle t\bar{f}, Y\tilde{\phi}_X \rangle_K, p \rangle + \tfrac{t^2}{2} \\ &= \inf_{t>0} \left\{ -t\rho_K + \tfrac{1}{2}t^2 \right\} \\ &= -\tfrac{1}{2}\rho_K^2 \quad \text{when } t = \rho_K > 0. \end{aligned} \tag{6.7}$$

Since $\max_{p \in \Delta_n} \left\langle -Y\tilde{f}(X), p \right\rangle$ is some empirical loss function on the data and $\frac{1}{2}\|f\|_K^2$ is an increasing function of $\|f\|_K$, the Representer Theorem [185] implies that the minimizer of the above function lies in the span of ϕ_{x_i}s (also the span of the $y_i\tilde{\phi}_{x_i}$s). Explicitly,

$$\arg\min_{f \in \mathcal{F}_K} L(f) = \sum_{i=1}^n \alpha_i y_i \tilde{\phi}_{x_i} = \langle Y\tilde{\phi}_X, \alpha \rangle. \tag{6.8}$$

Substituting this back into Eq.(6.6), we can define

$$L(\alpha) \;\; := \;\; \left\{ \sup_{p \in \Delta_n} \langle -\alpha, p \rangle_G \right\} + \tfrac{1}{2}\|\alpha\|_G^2, \tag{6.9}$$

where G is a normalized signed Gram matrix with $G_{ii} = 1$,

$$G_{ji} = G_{ij} := \frac{y_i y_j K(x_i,x_j)}{\sqrt{K(x_i,x_i)K(x_j,x_j)}} = \langle y_i\tilde{\phi}_{x_i}, y_j\tilde{\phi}_{x_j} \rangle_K,$$

and $\langle p, \alpha \rangle_G := p^\top G\alpha$, $\|\alpha\|_G := \sqrt{\alpha^\top G\alpha}$. One can verify that G is a PSD matrix and the G-norm $\|.\|_G$ is a semi-norm, whose properties are of great importance to us.

6.3.1 Some Interesting and Useful Lemmas

The first lemma justifies our algorithms' exit condition.

Lemma 17. $L(\alpha) < 0$ *implies* $G\alpha > 0$ *and there exists a perfect classifier iff* $G\alpha > 0$.

Proof: $L(\alpha) < 0 \Rightarrow \sup_{p\in\Delta_n} \langle -G\alpha, p\rangle < 0 \Leftrightarrow G\alpha > 0$. $G\alpha > 0 \Rightarrow f_\alpha := \langle \alpha, Y\tilde{\phi}_X\rangle$ is perfect since

$$\begin{aligned}\frac{y_j f_\alpha(x_j)}{\sqrt{K(x_j,x_j)}} &= \sum_{i=1}^{n} \alpha_i \frac{y_i y_j K(x_i,x_j)}{\sqrt{K(x_i,x_i)K(x_j,x_j)}} \\ &= G_j\alpha > 0.\end{aligned}$$

If a perfect classifier exists, then $\rho_K > 0$ by definition and

$$L(f^*) \;=\; L(\alpha^*) = -\tfrac{1}{2}\rho_K^2 < 0 \quad\Rightarrow\quad G\alpha > 0,$$

where f^*, α^* are the optimizers of $L(f), L(\alpha)$.

The second lemma bounds the G-norm of vectors.

Lemma 18. *For any* $\alpha \in \mathbb{R}^n$, $\|\alpha\|_G \le \|\alpha\|_1 \le \sqrt{n}\|\alpha\|_2$.

Proof: Using the triangle inequality of norms, we get

$$\begin{aligned}\sqrt{\alpha^\top G\alpha} &= \sqrt{\Big\langle \langle \alpha, Y\tilde{\phi}_X\rangle, \langle \alpha, Y\tilde{\phi}_X\rangle \Big\rangle_K} \\ &= \|\sum_i \alpha_i y_i \tilde{\phi}_{x_i}\|_K \le \sum_i \|\alpha_i y_i \tilde{\phi}_{x_i}\|_K \\ &\le \sum_i |\alpha_i| \left\| y_i \frac{\phi_{x_i}}{\sqrt{K(x_i,x_i)}} \right\|_K = \sum_i |\alpha_i|,\end{aligned}$$

where we used $\langle \phi_{x_i}, \phi_{x_i}\rangle_K = K(x_i,x_i)$.

The third lemma gives a new perspective on the margin.

Lemma 19. *When* $\rho_K > 0$, f *maximizes the margin iff* $\rho_K f$ *optimizes* $L(f)$. *Hence, the margin is equivalently*

$$\rho_K = \sup_{\|\alpha\|_G=1} \inf_{p\in\Delta_n} \langle \alpha, p\rangle_{\mathrm{G}} \le \|p\|_G \quad \textit{for all } p \in \Delta_n.$$

Proof: Let f_ρ be any function with $\|f_\rho\|_K = 1$ that achieves the max-margin $\rho_K > 0$. Then, it is easy to plug $\rho_K f_\rho$ into Eq. (6.6) and verify that $L(\rho_K f_\rho) = -\frac{1}{2}\rho_K^2$ and hence $\rho_K f_\rho$ minimizes $L(f)$.

Similarly, let f_L be any function that minimizes $L(f)$, i.e. achieves the value $L(f_L) = -\frac{1}{2}\rho_K^2$. Defining $t := \|f_L\|_K$, and examining Eq. (6.7), we see that $L(f_L)$ cannot achieve the value $-\frac{1}{2}\rho_K^2$ unless $t = \rho_K$ and $\sup_{p\in\Delta_n} \Big\langle -Y\tilde{f}_L(X), p\Big\rangle = -\rho_K^2$ which means that f_L/ρ_K must achieve the max-margin.

Hence considering only $f = \sum_i \alpha_i y_i \tilde{\phi}_{x_i}$ is acceptable for both. Plugging this into Eq. (6.5) gives the equality and

$$\begin{aligned}\rho_K &= \inf_{p\in\Delta_n} \sup_{\|\alpha\|_G=1} \langle \alpha, p\rangle_G \le \sup_{\|\alpha\|_G=1} \langle \alpha, p\rangle_G \\ &\le \|p\|_G \text{ by applying Cauchy-Schwartz}\end{aligned}$$

(can also be seen by going back to function space).

6.4 Smoothed Normalized Kernel Perceptron

Define the distribution over the worst-classified points

$$\begin{aligned} p(f) \quad &:= \quad \arg\min_{p\in\Delta_n} \left\langle Y\tilde{f}(X), p \right\rangle \\ \text{or} \quad p(\alpha) \quad &:= \quad \arg\min_{p\in\Delta_n} \langle \alpha, p \rangle_G. \end{aligned} \tag{6.10}$$

Algorithm 6 Normalized Kernel Perceptron (NKP)

```
Set $\alpha_0 := 0$
for $k = 0, 1, 2, 3, ...$ do
    if $G\alpha_k > \mathbf{0}_n$ then
        Exit, with $\alpha_k$ as solution
    else
        $\theta_k := \frac{1}{k+1}$
        $\alpha_{k+1} := (1-\theta_k)\alpha_k + \theta_k p(\alpha_k)$
    end if
end for
```

$$\begin{aligned} \text{Implicitly} \quad f_{k+1} \quad &= \quad (1-\theta_k)f_k + \theta_k \langle Y\tilde{\phi}_X, p(f_k)\rangle \\ &= \quad f_k - \theta_k \Big(f_k - \langle Y\tilde{\phi}_X, p(f_k)\rangle \Big) \\ &= \quad f_k - \theta_k \partial L(f_k) \end{aligned}$$

and hence the Normalized Kernel Perceptron (NKP) is a *subgradient algorithm* to minimize $L(f)$ from Eq. (6.6).

Remark. Lemma 19 yields deep insights. Since NKP can get arbitrarily close to the minimizer of strongly convex $L(f)$, it also gets arbitrarily close to a margin maximizer. It is known that it finds a perfect classifier in $1/\rho_K^2$ iterations - we now additionally infer that it will continue to improve to find an approximate max-margin classifier. While both classical and normalized Perceptrons find perfect classifiers in the same time, the latter is guaranteed to improve.

Remark. α_{k+1} is always a probability distribution. Curiously, a guarantee that the solution will lie in Δ_n is *not* made by the Representer Theorem in Eq. (6.8) - any $\alpha \in \mathbb{R}^n$ could satisfy Lemma 17. However, since NKP is a subgradient method for minimizing Eq. (6.6), we know that we will approach the optimum while only choosing $\alpha \in \Delta_n$.

Define the smooth minimizer analogous to Eq. (6.10) as

$$\begin{aligned} p_\mu(\alpha) \quad &:= \quad \arg\min_{p\in\Delta_n} \Big\{ \langle \alpha, p \rangle_G + \mu d(p) \Big\} && (6.11) \\ &= \quad \frac{e^{-G\alpha/\mu}}{\|e^{-G\alpha/\mu}\|_1}, \\ \text{where} \quad d(p) \quad &:= \quad \sum_i p_i \log p_i + \log n && (6.12) \end{aligned}$$

is 1-strongly convex with respect to the ℓ_1-norm [148]. Define a smoothened loss function as in Eq. (6.9)

Algorithm 7 Smoothed Normalized Kernel Perceptron

```
Set $\alpha_0 = \mathbf{1}_n/n,\ \mu_0 := 2,\ p_0 := p_{\mu_0}(\alpha_0)$
for $k = 0, 1, 2, 3, ...$ do
    if $G\alpha_k > \mathbf{0}_n$ then
        Halt: $\alpha_k$ is solution to Eq. (6.8)
    else
        $\theta_k := \frac{2}{k+3}$
        $\alpha_{k+1} := (1-\theta_k)(\alpha_k + \theta_k p_k) + \theta_k^2 p_{\mu_k}(\alpha_k)$
        $\mu_{k+1} = (1-\theta_k)\mu_k$
        $p_{k+1} := (1-\theta_k)p_k + \theta_k p_{\mu_{k+1}}(\alpha_{k+1})$
    end if
end for
```

$$L_\mu(\alpha) = \sup_{p \in \Delta_n} \Big\{ - \langle \alpha, p \rangle_G - \mu d(p) \Big\} + \tfrac{1}{2}\|\alpha\|_G^2.$$

Note that the maximizer above is precisely $p_\mu(\alpha)$.

Lemma 20 (Lower Bound)**.** *At any step k, we have*

$$L_{\mu_k}(\alpha_k) \geq L(\alpha_k) - \mu_k \log n.$$

Proof: First note that $\sup_{p\in\Delta_n} d(p) = \log n$. Also,

$$\begin{aligned} &\sup_{p\in\Delta_n} \big\{ - \langle \alpha, p \rangle_G - \mu d(p) \big\} \\ \geq\ &\sup_{p\in\Delta_n} \big\{ - \langle \alpha, p \rangle_G \big\} - \sup_{p\in\Delta_n} \big\{ \mu d(p) \big\}. \end{aligned}$$

Combining these two facts gives us the result.

Lemma 21 (Upper Bound)**.** *In any round k, SNKP satisfies*

$$L_{\mu_k}(\alpha_k) \leq -\tfrac{1}{2}\|p_k\|_G^2.$$

Proof: We provide a concise, self-contained and unified proof by induction in the Appendix for Lemma 21 and Lemma 24, borrowing ideas from Nesterov's excessive gap technique [148] for smooth minimization of structured non-smooth functions.

Finally, we combine the above lemmas to get the following theorem about the performance of SNKP.

Theorem 19. *The SNKP algorithm finds a perfect classifier $f \in \mathcal{F}_K$ when one exists in $O\left(\frac{\sqrt{\log n}}{\rho_K}\right)$ iterations.*

Proof: Lemma 20 gives us for any round k,

$$L_{\mu_k}(\alpha_k) \geq L(\alpha_k) - \mu_k \log n.$$

From Lemmas 19, 21 we get

$$L_{\mu_k}(\alpha_k) \leq -\tfrac{1}{2}p_k^\top G p_k \leq -\tfrac{1}{2}\rho_K^2.$$

Combining the two equations, we get that

$$L(\alpha_k) \leq \mu_k \log n - \tfrac{1}{2}\rho_K^2.$$

Noting that $\mu_k = \frac{4}{(k+1)(k+2)} < \frac{4}{(k+1)^2}$, we see that $L(\alpha_k) < 0$ (and hence we solve the problem by Lemma 17) after at most $k = 2\sqrt{2\log n}/\rho_K$ steps.

6.5 Infeasible Problems

What happens when the points are not separable by any function $f \in \mathcal{F}_K$? We would like an algorithm that terminates with a solution when there is one, and terminates with a certificate of non-separability if there isn't one. The idea is based on theorems of the alternative like Farkas' Lemma, specifically a version of Gordan's theorem [39]:

Lemma 22 (Gordan's Thm)**.** *Exactly one of the following two statements can be true*

1. *Either there exists a $w \in \mathbb{R}^d$ such that for all i,*

$$y_i(w^\top x_i) > 0,$$

2. *Or, there exists a $p \in \Delta_n$ such that*

$$\|XYp\|_2 = 0, \tag{6.13}$$

or equivalently $\sum_i p_i y_i x_i = 0$.

As mentioned in the introduction, the primal problem can be interpreted as finding a vector in the interior of the dual cone of $cone\{y_i x_i\}$, which is infeasible the dual cone is flat i.e. if $cone\{y_i x_i\}$ is not pointed, which happens when the origin is in the convex combination of $y_i x_i$s.

We will generalize the following algorithm for linear feasibility problems, that can be dated back to Von-Neumann, who mentioned it in a private communication with Dantzig, who later studied it himself [45].

Algorithm 8 Normalized Von-Neumann (NVN)

```
Initialize $p_0 = \mathbf{1}_n/n, w_0 = XYp_0$
for $k = 0, 1, 2, 3, ...$ do
    if $\|XYp_k\|_2 \leq \epsilon$ then
        Exit and return $p_k$ as an $\epsilon$-solution to (6.13)
    else
        $j := \arg\min_i y_i x_i^\top w_k$
        $\theta_k := \arg\min_{\lambda \in [0,1]} \|(1-\lambda)w_k + \lambda y_j x_j\|_2$
        $p_{k+1} := (1-\theta_k)p_k + \theta_k e_j$
        $w_{k+1} := XYp_{k+1} = (1-\theta_k)w_k + \theta_k y_j x_j$
    end if
end for
```

This algorithm comes with a guarantee: *If the problem (6.3) is infeasible, then the above algorithm will terminate with an ϵ-approximate solution to (6.13) in $1/\epsilon^2$ iterations.*

[64] proved an incomparable bound - Normalized Von-Neumann (NVN) can compute an ϵ-solution to (6.13) in $O\left(\frac{1}{\rho_2^2}\log\left(\frac{1}{\epsilon}\right)\right)$ and can also find a solution to the primal (using w_k) in $O\left(\frac{1}{\rho_2^2}\right)$ when it is feasible.

We derive a smoothed variant of NVN in the next section, after we prove some crucial lemmas in RKHSs.

6.5.1 A Separation Theorem for RKHSs

While finite dimensional Euclidean spaces come with strong separation guarantees that come under various names like the separating hyperplane theorem, Gordan's theorem, Farkas' lemma, etc, the story isn't always the same for infinite dimensional function spaces which can often be tricky to deal with. We will prove an appropriate version of such a theorem that will be useful in our setting.

What follows is an interesting version of the Hahn-Banach separation theorem, which looks a lot like Gordan's theorem in finite dimensional spaces. The conditions to note here are that either $G\alpha > 0$ or $\|p\|_G = 0$.

Theorem 20. *Exactly one of the following has a solution:*

1. *Either $\exists f \in \mathcal{F}_K$ such that for all i,*
$$\frac{y_i f(x_i)}{\sqrt{K(x_i,x_i)}} = \langle f, y_i \tilde{\phi}_{x_i} \rangle_K > 0 \quad \textit{i.e.} \quad G\alpha > 0,$$

2. *Or $\exists p \in \Delta_n$ such that*
$$\sum_i p_i y_i \tilde{\phi}_{x_i} = 0 \in \mathcal{F}_K \quad \textit{i.e.} \quad \|p\|_G = 0. \tag{6.14}$$

Proof: Consider the following set

$$\begin{aligned} Q &= \left\{ (f,t) = \left(\sum_i p_i y_i \tilde{\phi}_{x_i}, \sum_i p_i \right) : p \in \Delta_n \right\} \\ &= conv\left[(y_1\tilde{\phi}_{x_1}, 1), ..., (y_n \tilde{\phi}_{x_n}, 1) \right] \\ &\subseteq \mathcal{F}_K \times \mathbb{R}. \end{aligned}$$

If (2) does not hold, then it implies that $(0,1) \notin Q$. Since Q is closed and convex, we can find a separating hyperplane between Q and $(0,1)$, or in other words there exists $(f,t) \in \mathcal{F}_K \times \mathbb{R}$ such that

$$\begin{aligned} \Big\langle (f,t), (g,s) \Big\rangle &\geq 0 \ \forall (g,s) \in Q \\ \text{and } \Big\langle (f,t), (0,1) \Big\rangle &< 0. \end{aligned}$$

The second condition immediately yields $t < 0$. The first condition, when applied to $(g,s) = (y_i\tilde{\phi}_{x_i}, 1) \in Q$ yields

$$\begin{aligned} \langle f, y_i \tilde{\phi}_{x_i} \rangle_K + t &\geq 0 \\ \Leftrightarrow \quad \frac{y_i f(x_i)}{\sqrt{K(x_i,x_i)}} &> 0 \end{aligned}$$

since $t < 0$, which shows that (1) holds.

It is also immediate that if (2) holds, then (1) cannot.

Note that G is positive semi-definite - infeasibility requires both that it is not positive definite, and also that the witness to $p^\top G p = 0$ must be a probability vector. Similarly, while it suffices that $G\alpha > 0$ for some $\alpha \in \mathbb{R}^n$, but coincidentally in our case α will also lie in the probability simplex.

6.5.2 The infeasible margin ρ_K

Note that constraining $\|f\|_K = 1$ (or $\|\alpha\|_G = 1$) in Eq. (6.5) and Lemma 19 allows ρ_K to be negative in the infeasible case. If it was $\leq$, then ρ_K would have been non-negative because $f = 0$ (ie $\alpha = 0$) is always allowed.

So what is ρ_K when the problem is infeasible? Let

$$\operatorname{conv}(Y\tilde{\phi}_X) := \Big\{ \sum_i p_i y_i \tilde{\phi}_{x_i} | p \in \Delta_n \Big\} \subset \mathcal{F}_K$$

be the convex hull of the $y_i\tilde{\phi}_{x_i}$s.

Theorem 21. *When the primal is infeasible, the margin*[1] *is*

$$|\rho_K| = \delta_{\max} := \sup \Big\{ \delta \;\Big|\; \|f\|_K \leq \delta \Rightarrow f \in \operatorname{conv}(Y\tilde{\phi}_X) \Big\}$$

Proof: (1) For inequality $\geq$. Choose any δ such that $f \in \operatorname{conv}(Y\tilde{\phi}_X)$ for any $\|f\|_K \leq \delta$. Given an arbitrary $f' \in \mathcal{F}_K$ with $\|f'\|_K = 1$, put $\tilde{f} := -\delta f'$.

By our assumption on δ, we have $\tilde{f} \in \operatorname{conv}(Y\tilde{\phi}_X)$ implying there exists a $\tilde{p} \in \Delta_n$ such that $\tilde{f} = \langle Y\tilde{\phi}_X, \tilde{p} \rangle$. Also

$$\begin{aligned} \Big\langle f', \langle Y\tilde{\phi}_X, \tilde{p} \rangle \Big\rangle_K &= \langle f', \tilde{f} \rangle_K \\ &= -\delta \|f'\|_K^2 = -\delta. \end{aligned}$$

Since this holds for a particular $\tilde{p}$, we can infer

$$\inf_{p \in \Delta_n} \Big\langle f', \langle Y\tilde{\phi}_X, \tilde{p} \rangle \Big\rangle_K \leq -\delta.$$

Since this holds for any f' with $\|f'\|_G = 1$, we have

$$\sup_{\|f\|_K = 1} \inf_{p \in \Delta_n} \Big\langle f', \langle Y\tilde{\phi}_X, \tilde{p} \rangle \Big\rangle_K \leq -\delta \;\text{ i.e. }\; |\rho_K| \geq \delta.$$

(2) For inequality $\leq$. It suffices to show $\|f\|_K \leq |\rho_K| \Rightarrow f \in \operatorname{conv}(Y\tilde{\phi}_X)$. We will prove the contrapositive $f \notin \operatorname{conv}(Y\tilde{\phi}_X) \Rightarrow \|f\|_K > |\rho_K|$.

Since Δ_n is compact and convex, $\operatorname{conv}(Y\tilde{\phi}_X) \subset \mathcal{F}_K$ is closed and convex. Therefore if $f \notin \operatorname{conv}(Y\tilde{\phi}_X)$, then there exists $g \in \mathcal{F}_K$ with $\|g\|_K = 1$ that separates f and $\operatorname{conv}(Y\tilde{\phi}_X)$, i.e. for all $p \in \Delta_n$,

$$\begin{aligned} \langle g, f \rangle_K &< 0 \text{ and } \langle g, \langle Y\tilde{\phi}_X, p \rangle \rangle_K \geq 0 \\ \text{i.e. } \langle g, f \rangle_K &< \inf_{p \in \Delta_n} \langle g, \langle Y\tilde{\phi}_X, p \rangle \rangle_K \\ &\leq \sup_{\|f\|_K = 1} \inf_{p \in \Delta_n} \langle f, \langle Y\tilde{\phi}_X, p \rangle \rangle_K = \rho_K. \end{aligned}$$

$$\begin{aligned} \text{Since } \rho_K < 0 \quad |\rho_K| &< |\langle f, g \rangle_K| \\ &\leq \|f\|_K \|g\|_K = \|f\|_K. \end{aligned}$$

[1] We thank a reviewer for pointing out that by this definition, ρ_K might always be 0 for infinite dimensional RKHSs because there are always directions perpendicular to the finite-dimensional hull - we conjecture the definition can be altered to restrict attention to the relative interior of the hull, making it non-zero.

6.6 Kernelized Primal-Dual Algorithms

The preceding theorems allow us to write a variant of the Normalized VonNeumann algorithm from the previous section that is smoothed and works for RKHSs. Define

$$W := \Big\{p \in \Delta_n \Big| \sum_i p_i y_i \tilde{\phi}_{x_i} = 0\Big\} = \Big\{p \in \Delta_n \Big| \|p\|_G = 0\Big\}$$

as the set of witnesses to the infeasibility of the primal. The following lemma bounds the distance of any point in the simplex from the witness set by its $\|.\|_G$ norm.

Lemma 23. *For all $q \in \Delta_n$, the distance to the witness set*

$$\mathrm{dist}(q, W) := \min_{w\in W} \|q - w\|_2 \leq \min\left\{\sqrt{2}, \frac{\sqrt{2}\|q\|_G}{|\rho_K|}\right\}.$$

As a consequence, $\|p\|_G = 0$ iff $p \in W$.

Proof: This is trivial for $p \in W$. For arbitrary $p \in \Delta_n \backslash W$, let $\tilde{p} := -\frac{|\rho_K| p}{\|p\|_G}$ so that $\|\langle Y\tilde{\phi}_X, \tilde{p}\rangle\|_K = \|\tilde{p}\|_G \leq |\rho_K|$.

Hence by Theorem 21, there exists $\alpha \in \Delta_n$ such that

$$\langle Y\tilde{\phi}_X, \alpha\rangle = \langle Y\tilde{\phi}_X, \tilde{p}\rangle.$$

Let $\beta = \lambda\alpha + (1-\lambda)p$ where $\lambda = \frac{\|p\|_G}{\|p\|_G + |\rho_K|}$. Then

$$\begin{aligned}\langle Y\tilde{\phi}_X, \beta\rangle &= \frac{1}{\|p\|_G + |\rho_K|}\Big\langle Y\tilde{\phi}_X, \|p\|_G\alpha + |\rho_K|p\Big\rangle \\ &= \frac{1}{\|p\|_G + |\rho_K|}\langle Y\tilde{\phi}_X, \|p\|_G\tilde{p} + |\rho_K|p\rangle \\ &= 0,\end{aligned}$$

so $\beta \in W$ (by definition of what it means to be in W) and

$$\|p - \beta\|_2 = \lambda\|p - \alpha\|_2 \leq \lambda\sqrt{2} \leq \min\left\{\sqrt{2}, \frac{\sqrt{2}\|q\|_G}{|\rho_K|}\right\}.$$

We take $\min$ with $\sqrt{2}$ because ρ_K might be 0.

Hence for the primal or dual problem, points with small G-norm are revealing - either Lemma 19 shows that the margin $\rho_K \leq \|p\|_G$ will be small, or if it is infeasible then the above lemma shows that it is close to the witness set.

We need a small alteration to the smoothing entropy prox-function that we used earlier. We will now use

$$d_q(p) = \tfrac{1}{2}\|p - q\|_2^2$$

for some given $q \in \Delta_n$, which is strongly convex with respect to the ℓ_2 norm. This allows us to define

$$\begin{aligned}p_\mu^q(\alpha) &= \arg\min_{p\in\Delta_n} \langle G\alpha, p\rangle + \frac{\mu}{2}\|p - q\|_2^2, \\ L_\mu^q(\alpha) &= \sup_{p\in\Delta_n}\Big\{-\langle\alpha, p\rangle_G - \mu d_q(p)\Big\} + \tfrac{1}{2}\|\alpha\|_G^2,\end{aligned}$$

which can easily be found by sorting the entries of $q - \frac{G\alpha}{\mu}$.

When the primal is feasible, SNKPVN is similar to SNKP.

Algorithm 9 Smoothed Normalized Kernel Perceptron-VonNeumann ($SNKPVN(q, \delta)$)

```
INPUT: $q \in \Delta_n$, accuracy $\delta > 0$
  Set $\alpha_0 = q,\ \mu_0 := 2n,\ p_0 := p^q_{\mu_0}(\alpha_0)$
  for $k = 0, 1, 2, 3, ...$ do
      if $G\alpha_k > \mathbf{0}_n$ then
          Halt: $\alpha_k$ is solution to Eq. (6.8)
      else if $\|p_k\|_G < \delta$ then
          Return $p_k$
      else
          $\theta_k := \frac{2}{k+3}$
          $\alpha_{k+1} := (1-\theta_k)(\alpha_k + \theta_k p_k) + \theta_k^2\ p^q_{\mu_k}(\alpha_k)$
          $\mu_{k+1} = (1-\theta_k)\mu_k$
          $p_{k+1} := (1-\theta_k)p_k + \theta_k\ p^q_{\mu_{k+1}}(\alpha_{k+1})$
      end if
  end for
```

Lemma 24 (When $\rho_K > 0$ and $\delta < \rho_K$)**.** *For any $q \in \Delta_n$,*

$$-\tfrac{1}{2}\|p_k\|_G^2 \ \geq\ L^q_{\mu_k}(\alpha_k)\ \geq\ L(\alpha_k) -\ \mu_k.$$

Hence SNKPVN finds a separator f in $O\left(\frac{\sqrt{n}}{\rho_K}\right)$ iterations.

Proof: We give a unified proof for the first inequality and Lemma 21 in the Appendix. The second inequality mimics Lemma 20. The final statement mimics Theorem 19.

The following lemma captures the near-infeasible case.

Lemma 25 (When $\rho_K < 0$ or $\delta > \rho_K$)**.** *For any $q \in \Delta_n$,*

$$-\tfrac{1}{2}\|p_k\|_G^2 \ \geq\ L^q_{\mu_k}(\alpha_k)\ \geq\ -\ \tfrac{1}{2}\mu_k \mathrm{dist}(q, W)^2.$$

Hence SNKPVN finds a δ-solution in at most $O\left(\min\left\{\frac{\sqrt{n}}{\delta}, \frac{\sqrt{n}\|q\|_G}{\delta|\rho_K|}\right\}\right)$ iterations.

Proof: The first inequality is the same as in the above Lemma 24, and is proved in the Appendix.

$$\begin{aligned}
L^q_{\mu_k}(\alpha_k) &= \sup_{p\in\Delta_n}\left\{ -\langle \alpha, p\rangle_G - \mu_k d_q(p)\right\} + \tfrac{1}{2}\|\alpha\|_G^2 \\
&\geq \sup_{p\in W}\left\{ -\langle \alpha, p\rangle_G - \mu_k d_q(p)\right\} \\
&= \sup_{p\in W}\left\{ -\tfrac{1}{2}\mu_k\|p-q\|_2^2\right\} \\
&= -\tfrac{1}{2}\mu_k \mathrm{dist}(q,W)^2 \\
&\geq -\mu_k \min\left\{2, \tfrac{\|q\|_G^2}{|\rho_K|^2}\right\} \qquad \text{using Lemma 23.}
\end{aligned}$$

Since $\mu_k = \frac{4n}{(k+1)(k+2)} \leq \frac{4n}{(k+1)^2}$ we get

$$\|p_k\|_G \leq \frac{2\sqrt{n}}{(k+1)} \min\left\{\sqrt{2}, \frac{\|q\|_G}{\rho_K}\right\}.$$

Hence $\|p\|_G \leq \delta$ after $\frac{2\sqrt{n}}{\delta}\min\left\{\sqrt{2}, \frac{\|q\|_G}{\rho_K}\right\}$ steps.

Algorithm 10 Iterated Smoothed Normalized Kernel Perceptron-VonNeumann ($ISNKPVN(\gamma, \epsilon)$)

```
INPUT: Constant $\gamma > 1$, accuracy $\epsilon > 0$
  Set $q_0 := \mathbf{1}_n/n$
  for $t = 0, 1, 2, 3, ...$ do
      $\delta_t := \|q_t\|_G/\gamma$
      $q_{t+1} := SNKPVN(q_t, \delta_t)$
      if $\delta_t < \epsilon$ then
          Halt; $q_{t+1}$ is a solution to Eq. (6.14)
      end if
  end for
```

Using SNKPVN as a subroutine gives our final algorithm.

Theorem 22. *Algorithm ISNKPVN satisfies*

1. *If the primal (6.2) is feasible and $\epsilon < \rho_K$, then each call to SNKPVN halts in at most $\frac{2\sqrt{2n}}{\rho_K}$ iterations. Algorithm ISNKPVN finds a solution in at most $\frac{\log(1/\rho_K)}{\log(\gamma)}$ outer loops, bounding the total iterations by*
$$O\left(\frac{\sqrt{n}}{\rho_K}\log\left(\frac{1}{\rho_K}\right)\right).$$

2. *If the dual (6.14) is feasible or $\epsilon > \rho_K$, then each call to SNKPVN halts in at most $O\left(\min\left\{\frac{\sqrt{n}}{\epsilon}, \frac{\sqrt{n}}{|\rho_K|}\right\}\right)$ steps. Algorithm ISNKPVN finds an ϵ-solution in at most $\frac{\log(1/\epsilon)}{\log(\gamma)}$ outer loops, bounding the total iterations by*
$$O\left(\min\left\{\frac{\sqrt{n}}{\epsilon}, \frac{\sqrt{n}}{|\rho_K|}\right\}\log\left(\frac{1}{\epsilon}\right)\right).$$

Proof: First note that if ISNKPVN has not halted, then we know that after t outer iterations, q_{t+1} has small G-norm:
$$\|q_{t+1}\|_G \le \delta_t \le \frac{\|q_0\|_G}{\gamma^{t+1}}. \tag{6.15}$$

The first inequality holds because of the inner loop return condition, the second because of the update for δ_t.

1. Lemma 19 shows that for all p we have $\rho_K \le \|p\|_G$, so the inner loop will halt with a solution to the primal as soon as $\delta_t \le \rho_K$ (so that $\|p\|_G < \delta_t \le \rho_K$ cannot be satisfied for the inner loop to return). From Eq. (6.15), this will definitely happen when $\frac{\|q_0\|_G}{\gamma^{t+1}} \le \rho_K$, ie within $T = \frac{\log(\|q_o\|_G/\rho_K)}{\log(\gamma)}$ iterations. By Lemma 24, each iteration runs for at most $\frac{2\sqrt{2n}}{\rho_K}$ steps.

2. We halt with an ϵ-solution when $\delta_t < \epsilon$, which definitely happens when $\frac{\|q_0\|_G}{\gamma^{t+1}} < \epsilon$, ie within $T = \frac{\log(\|q_o\|_G/\epsilon)}{\log(\gamma)}$ iterations. Since $\frac{\|q_t\|_G}{\delta_t} = \gamma$, by Lemma 25, each iteration runs for at most $O\left(\min\left\{\frac{\sqrt{n}}{\epsilon}, \frac{\sqrt{n}}{|\rho_K|}\right\}\right)$ steps.

6.7 Discussion

The SNK-Perceptron algorithm presented in this chapter has a convergence rate of $\frac{\sqrt{\log n}}{\rho_K}$ and the Iterated SNK-Perceptron-Von-Neumann algorithm has a $\min\left\{\frac{\sqrt{n}}{\epsilon}, \frac{\sqrt{n}}{|\rho_K|}\right\}$ dependence on the number of points.

Note that both of these are independent of the underlying dimensionality of the problem. We conjecture that it is possible to reduce this dependence to $\sqrt{\log n}$ for the primal-dual algorithm also, without paying a price in terms of the dependence on margin $1/\rho$ (or the dependence on ϵ).

It is possible that tighter dependence on n is possible if we try other smoothing functions instead of the ℓ_2 norm used in the last section. Specifically, it might be tempting to smooth with the $\|.\|_G$ semi-norm and define:

$$p_\mu^q(\alpha) = \arg\min_{p \in \Delta_n} \langle \alpha, p \rangle_G + \frac{\mu}{2}\|p - q\|_G^2$$

One can actually see that the proofs in the Appendix go through with no dimension dependence on n at all! However, it is not possible to solve this in closed form - taking $\alpha = q$ and $\mu = 1$ reduces the problem to asking

$$p^q(q) = \arg\min_{p \in \Delta_n} \tfrac{1}{2}\|p\|_G^2$$

which is an oracle for our problem as seen by equation (6.14) - the solution's G-norm is 0 iff the problem is infeasible.

In the bigger picture, there are several interesting open questions. The ellipsoid algorithm for solving linear feasibility problems has a logarithmic dependence on $1/\epsilon$, and a polynomial dependence on dimension. Recent algorithms involving repeated rescaling of the space like [58] have logarithmic dependence on $1/\rho$ and polynomial in dimension. While both these algorithms are poly-time under the real number model of computation of [22], it is unknown whether there is any algorithm that can achieve a polylogarithmic dependence on the margin/accuracy, and a polylogarithmic dependence on dimension. This is strongly related to the open question of whether it is possible to learn a decision list polynomially in its binary description length.

One can nevertheless ask whether rescaled *smoothed* perceptron methods like [58] can be lifted to RKHSs, and whether using an iterated smoothed kernel perceptron would yield faster rates. The recent work [197] is a challenge to generalize - the proofs relying on geometry involve arguing about volumes of balls of functions in an RKHS - we conjecture that it is possible to do, but we leave it for a later work.

Chapter 7

Linear Regression : Randomized algorithms for ordinary least-squares

The Kaczmarz and Gauss-Seidel methods both solve a linear system $\boldsymbol{X}\boldsymbol{\beta} = \boldsymbol{y}$ by iteratively refining the solution estimate. Recent interest in these methods has been sparked by a proof of Strohmer and Vershynin which shows the *randomized* Kaczmarz method converges linearly in expectation to the solution. Lewis and Leventhal then proved a similar result for the randomized Gauss-Seidel algorithm. However, the behavior of both methods depends heavily on whether the system is under or overdetermined, and whether it is consistent or not. Here we provide a unified theory of both methods, their variants for these different settings, and draw connections between both approaches. In doing so, we also provide a proof that an extended version of randomized Gauss-Seidel converges linearly to the least norm solution in the underdetermined case (where the usual randomized Gauss Seidel fails to converge). We detail analytically and empirically the convergence properties of both methods and their extended variants in all possible system settings. With this result, a complete and rigorous theory of both methods is furnished.

7.1 Introduction

We consider solving a linear system of equations

$$\boldsymbol{X}\boldsymbol{\beta} = \boldsymbol{y}, \tag{7.1}$$

for a (real or complex) $m \times n$ matrix $\boldsymbol{X}$, in various problem settings. Recent interest in the topic was reignited when Strohmer and Vershynin [206] proved the linear[1] convergence rate of the Randomized Kaczmarz (RK) algorithm that works on the rows of $\boldsymbol{X}$ (data points). Following that, Leventhal and Lewis [126] proved the linear convergence of a Randomized Gauss-Seidel (RGS), i.e. Randomized Coordinate Descent, algorithm that works on the columns of $\boldsymbol{X}$ (features).

When the system of equations is inconsistent, as is typically the case when $m > n$ in real-world overconstrained systems, RK is known to not converge to the ordinary least squares solution

$$\boldsymbol{\beta_{LS}} := \arg\min_{\boldsymbol{\beta}} \tfrac{1}{2}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2 \tag{7.2}$$

as studied by Needell [141]. Zouzias and Freris [235] extended the RK method with the modified Randomized Extended Kaczmarz (REK) algorithm, which linearly converges to $\boldsymbol{\beta_{LS}}$. Interestingly, in this setting, we will argue in Section 7.3.3 that RGS does converge to $\boldsymbol{\beta_{LS}}$ without any special extensions.

[1] Mathematicians often refer to linear convergence as exponential convergence.

Motivation and contribution

The above introduction represents only half the story. When $m < n$, there are fewer constraints than variables, and the system has infinitely many solutions. In this case, especially if we have no prior reason to believe any additional sparsity in the signal structure, we are often interested in finding the least Euclidean norm solution:

$$\boldsymbol{\beta_{LN}} := \arg\min_{\boldsymbol{\beta}} \|\boldsymbol{\beta}\|_2 \text{ s.t. } \boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta}. \tag{7.3}$$

While RGS converges to $\boldsymbol{\beta_{LS}}$ in the overcomplete setting, we shall argue in Section 7.3.3 that in the undercomplete setting it does not converge to $\boldsymbol{\beta_{LN}}$. We will also argue that RK does converge to $\boldsymbol{\beta_{LN}}$ without any extensions in this setting.

The main contribution of this chapter is to provide a unified theory of these related iterative methods. We will also construct an extension to RGS, paralleling REK, which does converge to $\boldsymbol{\beta_{LN}}$ (like REK converges to $\boldsymbol{\beta_{LS}}$). Some desired properties for this algorithm include that it should *also* converge linearly, not require much extra computation, and work well in simulations. We shall see that our Randomized Extended Gauss-Seidel (REGS) method does indeed possess these desired properties. A summary of this unified theory is provided in Table 7.1.

Method	Overconstrained, consistent : convergence to $\boldsymbol{\beta}^\star$?	Overconstrained, inconsistent : convergence to $\boldsymbol{\beta_{LS}}$?	Underconstrained : convergence to $\boldsymbol{\beta_{LN}}$?
RK	Yes [206]	No [141]	Yes (see Sec. 7.3.3)
REK	Yes [235]	Yes [235]	Yes (see Sec. 7.3.3)
RGS	Yes [126]	Yes [126]	No (see Sec. 7.3.3)
REGS	Yes (this chapter)	Yes (this chapter)	Yes (this chapter)

Table 7.1: Summary of convergence properties for the overdetermined and consistent setting, overdetermined and inconsistent setting, and underdetermined settings. We write $\boldsymbol{\beta}^\star$ to denote the solution to (7.1) in the overdetermined consistent setting, with $\boldsymbol{\beta_{LS}}$ and $\boldsymbol{\beta_{LN}}$ being defined in (7.2) and (7.3) for the other two settings.

Chapter Outline

In Section 7.2 we recap the three main existing algorithms mentioned in the introduction (RK, RGS, REK). We discuss the performance of these algorithms in the three natural settings described in Table 7.1 in Section 7.3. Section 7.4 introduces our proposed algorithm (REGS) and proves its linear convergence to the least norm solution, completing the theoretical framework. Lastly, we end with some simulation experiments in Section 7.5 to demonstrate the tightness and usefulness of our theory, and conclude in Section 7.5.

7.2 Existing Algorithms and Related Work

In this section, we will summarize the algorithms mentioned in the introduction, i.e. RK, RGS and REK. We will describe their iterative update rules and mention their convergence guarantees, leaving the details of convergence to the next section. Throughout the chapter we will use the notation $\boldsymbol{X}^i$ to represent the ith row of $\boldsymbol{X}$ (or ith entry in the case of a vector) and $\boldsymbol{X}_{(j)}$ to denote the jth column of a matrix $\boldsymbol{X}$. We will write the estimation $\boldsymbol{\beta}$ as a column vector. We write vectors and matrices in boldface, and constants in standard font.

7.2.1 Randomized Kaczmarz (RK)

The Kaczmarz method was first introduced in the notable work of Kaczmarz [111]. It has gained recent interest in tomography research where it is known as the *Algebraic Reconstruction Technique* (ART) [29, 79, 97, 140]. Although in its original form the method selects rows in a deterministic fashion (often simply cyclically), it has been well observed that a random selection scheme reduces the possibility of a poor choice of row ordering [90, 98]. Earlier convergence analysis of the randomized variant were obtained (e.g. [232]), but yielded bounds with expressions that were difficult to evaluate. Strohmer and Vershynin [206] showed that the RK method described above has an expected linear convergence rate to the solution $\boldsymbol{\beta}^\star$ of (7.1), and are the first to provide an explicit convergence rate in expectation which depends only on the geometric properties of the system. This work was extended by Needell [141] to the inconsistent case, analyzed almost surely by Chen and Powell [37], accelerated in several ways [62, 63, 142, 143, 154], and extended to more general settings [126, 144, 172].

We describe here the randomized variant of the Kaczmarz method put forth by Strohmer and Vershynin [206]. Taking $\boldsymbol{X}, \boldsymbol{y}$ as input and starting from an arbitrary initial estimate for $\boldsymbol{\beta}$ (for example $\boldsymbol{\beta}_0 = \mathbf{0}$), RK repeats the following in each iteration. First, a random row $i \in \{1, ..., m\}$ is selected with probability proportional to its Euclidean norm, i.e.

$$\Pr(\text{row} = i) = \frac{\|\boldsymbol{X}^i\|_2^2}{\|\boldsymbol{X}\|_F^2},$$

where $\|\boldsymbol{X}\|_F$ denotes the Frobenius norm of $\boldsymbol{X}$. Then, project the current iterate onto that row, i.e.

$$\boldsymbol{\beta}_{t+1} := \boldsymbol{\beta}_t + \frac{(y^i - \boldsymbol{X}^i\boldsymbol{\beta}_t)}{\|\boldsymbol{X}^i\|_2^2}(\boldsymbol{X}^i)^*, \tag{7.4}$$

where here and throughout $\boldsymbol{X}^*$ denotes the (conjugate) transpose of $\boldsymbol{X}$.

Intuitively, this update can be seen as greedily satisfying the ith equation in the linear system. Indeed, it is easy to see that after the update,

$$\boldsymbol{X}^i\boldsymbol{\beta}_{t+1} = y^i. \tag{7.5}$$

Referring to (7.2) and defining

$$L(\boldsymbol{\beta}) = \tfrac{1}{2}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|^2 = \tfrac{1}{2}\sum_{i=1}^{m}(y^i - \boldsymbol{X}^i\boldsymbol{\beta})^2,$$

we can alternatively interpret this update as stochastic gradient descent (choosing a random data-point on which to update), where the stepsize is the inverse Lipschitz constant of the stochastic gradient

$$\nabla^2 \tfrac{1}{2}(y^i - \boldsymbol{X}^i\boldsymbol{\beta})^2 = \|\boldsymbol{X}^i\|_2^2.$$

7.2.2 Randomized Extended Kaczmarz (REK)

For inconsistent systems, the RK method does not converge to the least-squares solution as one might desire. This fact is clear since the method at each iteration projects completely onto a selected solution space, being unable to break the so-called *convergence horizon*. One approach to overcome this is to use relaxation parameters, so that the estimates are not projected completely onto the subspace at each iteration [35, 91, 209, 231]. Recently, Zouzias and Freris [235] proposed a variant of the RK method motivated by the work of Popa [153] which instead includes a random projection to iteratively reduce the component of $\boldsymbol{y}$ which is orthogonal to the range of $\boldsymbol{X}$. This method, named Randomized Extended Kaczmarz (REK) can be described by the following iteration updates, which can be initialized with $\boldsymbol{\beta}_0 = \boldsymbol{0}$ and $\boldsymbol{z}_0 = \boldsymbol{y}$:

$$\boldsymbol{\beta}_{t+1} := \boldsymbol{\beta}_t + \frac{(y^i - z_t^i - \boldsymbol{X}^i\boldsymbol{\beta}_t)}{\|\boldsymbol{X}^i\|_2^2}(\boldsymbol{X}^i)^*, \quad \boldsymbol{z}_{t+1} = \boldsymbol{z}_t - \frac{\langle \boldsymbol{X}_{(j)}, \boldsymbol{z}_t\rangle}{\|\boldsymbol{X}_{(j)}\|_2^2}\boldsymbol{X}_{(j)}. \tag{7.6}$$

Here, a column $j \in \{1, ..., n\}$ is also selected at random with probability proportional to its Euclidean norm:

$$\Pr(\text{column} = j) = \frac{\|\boldsymbol{X}_{(j)}\|_2^2}{\|\boldsymbol{X}\|_F^2}, \tag{7.7}$$

and again $\boldsymbol{X}_{(j)}$ denotes the jth column of $\boldsymbol{X}$. Here, $\boldsymbol{z}_t$ approximates the component of $\boldsymbol{y}$ which is orthogonal to the range of $\boldsymbol{X}$, allowing for the iterates $\boldsymbol{\beta}_t$ to converge to the true least-squares solution of the system. Zouzias and Freris [235] prove that REK converges linearly in expectation to this solution $\boldsymbol{\beta}_{\boldsymbol{LS}}$.

7.2.3 Randomized Gauss-Seidel (RGS)

Again taking $\boldsymbol{X}, \boldsymbol{y}$ as input and starting from an arbitrary $\boldsymbol{\beta}_0$, the Randomized Gauss-Seidel (RGS) method (or the Randomized Coordinate Descent method) repeats the following in each iteration. First, a random column $j \in \{1, ..., n\}$ is selected as in (7.7). We then minimize the objective $L(\boldsymbol{\beta}) = \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2$ with respect to this coordinate to get

$$\boldsymbol{\beta}_{t+1} := \boldsymbol{\beta}_t + \frac{\boldsymbol{X}_{(j)}^*(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}_t)}{\|\boldsymbol{X}_{(j)}\|_2^2}\boldsymbol{e}_{(j)} \tag{7.8}$$

where $\boldsymbol{e}_{(j)}$ is the jth coordinate basis column vector (all zeros with a 1 in the jth position). It can be seen as greedily minimizing the objective with respect to the jth coordinate. Indeed, letting $\boldsymbol{X}_{(-j)}, \boldsymbol{\beta}^{-j}$ represent $\boldsymbol{X}$ without its jth column and $\boldsymbol{\beta}$ without its jth coordinate,

$$\frac{\partial L}{\partial \boldsymbol{\beta}^j} = -\boldsymbol{X}_{(j)}^*(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}) = -\boldsymbol{X}_{(j)}^*(\boldsymbol{y} - \boldsymbol{X}_{(-j)}\boldsymbol{\beta}^{-j} - \boldsymbol{X}_{(j)}\beta^j). \tag{7.9}$$

Setting this equal to zero for the coordinate-wise minimization, we get the aforementioned update (7.8) for $\boldsymbol{\beta}^j$. Alternatively, since $[\nabla L(\boldsymbol{\beta})]^j = -\boldsymbol{X}_{(j)}^*(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})$, the above update can intuitively be seen as a univariate descent step where the stepsize is the inverse Lipschitz constant of the gradient along the jth coordinate, since the (j, j) entry of the Hessian is

$$[\nabla^2 L(\boldsymbol{\beta})]_{j,j} = (\boldsymbol{X}^*\boldsymbol{X})_{j,j} = \|\boldsymbol{X}_{(j)}\|_2^2.$$

Leventhal and Lewis [126] showed that this algorithm has an expected linear convergence rate. We will detail the convergence properties of this algorithm and the others in the next section.

7.3 Problem Variations

We first examine the differences in behavior of the two algorithms RGS and RK in three distinct but related settings. This will highlight the opposite behaviors of these two similar algorithms.

When the system of equations (7.1) has a unique solution, we represent this by $\boldsymbol{\beta}^\star$. This happens when $m \geq n$, and the system is consistent. Assuming that $\boldsymbol{X}$ has full column rank,

$$\boldsymbol{\beta}^\star = (\boldsymbol{X}^*\boldsymbol{X})^{-1}\boldsymbol{X}^*\boldsymbol{y}, \tag{7.10}$$

and then $\boldsymbol{X}\boldsymbol{\beta}^\star = \boldsymbol{y}$.

When (7.1) does not have any consistent solution, we refer to the least-squares solution of (7.2) as $\boldsymbol{\beta_{LS}}$. This could happen in the overconstrained case, when $m > n$. Again, assuming that $\boldsymbol{X}$ has full column rank, we have

$$\boldsymbol{\beta_{LS}} = (\boldsymbol{X}^*\boldsymbol{X})^{-1}\boldsymbol{X}^*\boldsymbol{y}, \tag{7.11}$$

and we can write $\boldsymbol{r} := \boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta_{LS}}$ as the residual vector.

When (7.1) has infinitely many solutions, we call the minimum Euclidean norm solution given by (7.3), $\boldsymbol{\beta_{LN}}$. This could happen in the underconstrained case, when $m < n$. Assuming that $\boldsymbol{X}$ has full row rank, we have

$$\boldsymbol{\beta_{LN}} = \boldsymbol{X}^*(\boldsymbol{X}\boldsymbol{X}^*)^{-1}\boldsymbol{y}. \tag{7.12}$$

In the above notation,

the LS stands for Least Squares and LN for Least Norm. We shall return to each of these three situations in that order in future sections.

One of our main contributions is to achieve a unified understanding of the behavior of RK and RGS in these different situations. The literature for RK deals mainly with the first two settings only (see [206], [141], [235]). In the third setting, one readily obtains convergence to an *arbitrary* solution (see e.g. (3) of [129]), but the convergence to the least norm solution is not often studied (likely for practical reasons). The literature for RGS typically focuses on more general setups than our specific quadratic least squares loss function $L(\beta)$ (see Nesterov [147] or Richtárik and Takáč [172]). However, for both the purposes of completeness, and for a more thorough understanding of the relationship between RK and RGS, it turns out to be crucial to analyze all three settings (for equations (7.1)-(7.3)).

1. When $\boldsymbol{\beta}^\star$ is a unique consistent solution, we present proofs of the linear convergence of both algorithms - the results are known from papers by [206] and [126] but are presented here in a novel manner so that their relationship becomes clearer and direct comparison is easily possible.
2. When $\boldsymbol{\beta_{LS}}$ is the (inconsistent) least squares solution, we show why RGS iterates converge linearly to $\boldsymbol{\beta_{LS}}$, but RK iterates do not - making RGS preferable. These facts are not hard to see, but we make it more intuitively and mathematically clear why this should be the case.
3. When $\boldsymbol{\beta_{LN}}$ is the minimum norm consistent solution, we explain why RK converges linearly to it, but RGS iterates do not (both so far seemingly undocumented observations) - making RK preferable.

Together, the above three points complete the picture (with solid accompanying intuition) of the opposing behavior of RK and RGS. Later, we will present our variant of the RGS method, the Randomized Extended Gauss-Seidel (REGS), and compare with the corresponding variant of RK (REK). This new analysis will complete the unified framework for these methods.

7.3.1 Overconstrained System, Consistent

Here we will assume that $m > n$, $\boldsymbol{X}$ has full column rank, and the system is consistent, so $\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta}^\star$. First, let us write the updates used by both algorithms in a revealing fashion. If RK and RGS select row i and column j at step $t + 1$, and $\boldsymbol{e}^i$ (resp. $\boldsymbol{e}_{(j)}$) is the ith coordinate basis row (resp. column) vector, then the updates can be rewritten as:

$$\text{(RK)} \qquad \boldsymbol{\beta}_{t+1} := \boldsymbol{\beta}_t + \frac{\boldsymbol{e}^i \boldsymbol{r}_t}{\|\boldsymbol{X}^i\|_2^2}(\boldsymbol{X}^i)^* \tag{7.13}$$

$$\text{(RGS)} \qquad \boldsymbol{\beta}_{t+1} := \boldsymbol{\beta}_t + \frac{\boldsymbol{X}_{(j)}^* \boldsymbol{r}_t}{\|\boldsymbol{X}_{(j)}\|_2^2}\boldsymbol{e}_j \tag{7.14}$$

where $\boldsymbol{r}_t = \boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}_t = \boldsymbol{X}\boldsymbol{\beta}^\star - \boldsymbol{X}\boldsymbol{\beta}_t$ is the residual vector at iteration t. Then multiplying both equations by $\boldsymbol{X}$ gives

$$\text{(RK)} \qquad \boldsymbol{X}\boldsymbol{\beta}_{t+1} := \boldsymbol{X}\boldsymbol{\beta}_t + \frac{\boldsymbol{X}^i(\boldsymbol{\beta}^\star - \boldsymbol{\beta}_t)}{\|\boldsymbol{X}^i\|_2^2}\boldsymbol{X}(\boldsymbol{X}^i)^* \tag{7.15}$$

$$\text{(RGS)} \qquad \boldsymbol{X}\boldsymbol{\beta}_{t+1} := \boldsymbol{X}\boldsymbol{\beta}_t + \frac{\boldsymbol{X}_{(j)}^*\boldsymbol{X}(\boldsymbol{\beta}^\star - \boldsymbol{\beta}_t)}{\|\boldsymbol{X}_{(j)}\|_2^2}\boldsymbol{X}_j. \tag{7.16}$$

We now come to an important difference, which is the key update equation for RK and RGS.

First, from the update (7.13) for RK, we have that $\boldsymbol{\beta}_{t+1} - \boldsymbol{\beta}_t$ is parallel to $\boldsymbol{X}^i$. Also, $\boldsymbol{\beta}_{t+1} - \boldsymbol{\beta}^\star$ is orthogonal to $\boldsymbol{X}^i$ (since $\boldsymbol{X}^i(\boldsymbol{\beta}_{t+1} - \boldsymbol{\beta}^\star) = y^i - y^i = 0$). Then by the Pythagorean theorem,

$$\|\boldsymbol{\beta}_{t+1} - \boldsymbol{\beta}^\star\|_2^2 = \|\boldsymbol{\beta}_t - \boldsymbol{\beta}^\star\|_2^2 - \|\boldsymbol{\beta}_{t+1} - \boldsymbol{\beta}_t\|_2^2. \tag{7.17}$$

Note that from the update (7.16), we have that $\boldsymbol{X}\boldsymbol{\beta}_{t+1} - \boldsymbol{X}\boldsymbol{\beta}_t$ is parallel to $\boldsymbol{X}_{(j)}$. Also, $\boldsymbol{X}\boldsymbol{\beta}_{t+1} - \boldsymbol{X}\boldsymbol{\beta}^\star$ is orthogonal to $\boldsymbol{X}_{(j)}$ (since $\boldsymbol{X}_{(j)}^*(\boldsymbol{X}\boldsymbol{\beta}_{t+1} - \boldsymbol{X}\boldsymbol{\beta}^\star) = \boldsymbol{X}_{(j)}^*(\boldsymbol{X}\boldsymbol{\beta}_{t+1} - \boldsymbol{y}) = 0$ by the optimality condition $\partial L/\partial \boldsymbol{\beta}^j = 0$). Then again by the Pythagorean theorem,

$$\|\boldsymbol{X}\boldsymbol{\beta}_{t+1} - \boldsymbol{X}\boldsymbol{\beta}^\star\|_2^2 = \|\boldsymbol{X}\boldsymbol{\beta}_t - \boldsymbol{X}\boldsymbol{\beta}^\star\|_2^2 - \|\boldsymbol{X}\boldsymbol{\beta}_{t+1} - \boldsymbol{X}\boldsymbol{\beta}_t\|_2^2. \tag{7.18}$$

The rest of the proof follows by simply substituting for the last term in the above two equations, and is presented in the following table for easy comparison. Note $\boldsymbol{\Sigma} = \boldsymbol{X}^*\boldsymbol{X}$ is the full-rank covariance matrix and we first take expectations with respect to the randomness at the $(t+1)$st step, conditioning on all randomness up to the tth step. We later iterate this expectation.

Randomized Kaczmarz: $\mathbb{E}_t\|\boldsymbol{\beta}_{t+1} - \boldsymbol{\beta}^\star\|_2^2$	Randomized Gauss-Seidel: $\mathbb{E}_t\|\boldsymbol{X}\boldsymbol{\beta}_{t+1} - \boldsymbol{X}\boldsymbol{\beta}^\star\|_2^2$
$= \|\boldsymbol{\beta}_t - \boldsymbol{\beta}^\star\|_2^2 - \mathbb{E}\|\boldsymbol{\beta}_{t+1} - \boldsymbol{\beta}_t\|_2^2$	$= \|\boldsymbol{X}\boldsymbol{\beta}_t - \boldsymbol{X}\boldsymbol{\beta}^\star\|_2^2 - \mathbb{E}\|\boldsymbol{X}\boldsymbol{\beta}_{t+1} - \boldsymbol{X}\boldsymbol{\beta}_t\|_2^2$
$= \|\boldsymbol{\beta}_t - \boldsymbol{\beta}^\star\|_2^2$	$= \|\boldsymbol{X}\boldsymbol{\beta}_t - \boldsymbol{X}\boldsymbol{\beta}^\star\|_2^2$
$-\sum_i \frac{\|\boldsymbol{X}^i\|_2^2}{\|\boldsymbol{X}\|_F^2}\frac{(\boldsymbol{X}^i(\boldsymbol{\beta}_t - \boldsymbol{\beta}^\star))_2^2}{(\|\boldsymbol{X}^i\|_2^2)^2}\|\boldsymbol{X}^i\|_2^2$	$-\sum_j \frac{\|\boldsymbol{X}_{(j)}\|_2^2}{\|\boldsymbol{X}\|_F^2}\frac{(\boldsymbol{X}_{(j)}^*\boldsymbol{X}(\boldsymbol{\beta}_t - \boldsymbol{\beta}^\star))^2}{(\|\boldsymbol{X}_{(j)}\|_2^2)^2}\|\boldsymbol{X}_{(j)}\|_2^2$
$= \|\boldsymbol{\beta}_t - \boldsymbol{\beta}^\star\|_2^2\left(1 - \frac{\|\boldsymbol{X}(\boldsymbol{\beta}_t - \boldsymbol{\beta}^\star)\|_2^2}{\|\boldsymbol{X}\|_F^2\|\boldsymbol{\beta}_t - \boldsymbol{\beta}^\star\|_2^2}\right)$	$= \|\boldsymbol{X}\boldsymbol{\beta}_t - \boldsymbol{X}\boldsymbol{\beta}^\star\|_2^2\left(1 - \frac{\|\boldsymbol{X}^*\boldsymbol{X}(\boldsymbol{\beta}_t - \boldsymbol{\beta}^\star)\|_2^2}{\|\boldsymbol{X}\|_F^2\|\boldsymbol{X}\boldsymbol{\beta}_t - \boldsymbol{X}\boldsymbol{\beta}^\star\|_2^2}\right)$
$\leq \|\boldsymbol{\beta}_t - \boldsymbol{\beta}^\star\|_2^2\left(1 - \frac{\lambda_{\min}(\boldsymbol{\Sigma})}{Tr(\boldsymbol{\Sigma})}\right)$	$\leq \|\boldsymbol{X}\boldsymbol{\beta}_t - \boldsymbol{X}\boldsymbol{\beta}^\star\|_2^2\left(1 - \frac{\lambda_{\min}(\boldsymbol{\Sigma})}{Tr(\boldsymbol{\Sigma})}\right)$

Here, $\lambda_{\min}(\boldsymbol{\Sigma})\|\boldsymbol{\beta}_t - \boldsymbol{\beta}^\star\|_2^2 \leq \|\boldsymbol{X}(\boldsymbol{\beta}_t - \boldsymbol{\beta}^\star)\|_2^2$ i.e. $\lambda_{\min}(\boldsymbol{\Sigma})$ is the smallest eigenvalue of $\boldsymbol{\Sigma}$ (singular value of $\boldsymbol{X}$). It follows that

$$\text{(RK)} \qquad \mathbb{E}\|\boldsymbol{\beta}_t - \boldsymbol{\beta}^\star\|_2^2 \leq \left(1 - \frac{\lambda_{\min}(\boldsymbol{\Sigma})}{Tr(\boldsymbol{\Sigma})}\right)^t \|\boldsymbol{\beta}_0 - \boldsymbol{\beta}^\star\|_2^2 \tag{7.19}$$

$$\text{(RGS)} \qquad \mathbb{E}\|\boldsymbol{\beta}_t - \boldsymbol{\beta}^\star\|_{\boldsymbol{\Sigma}}^2 \leq \left(1 - \frac{\lambda_{\min}(\boldsymbol{\Sigma})}{Tr(\boldsymbol{\Sigma})}\right)^t \|\boldsymbol{\beta}_0 - \boldsymbol{\beta}^\star\|_{\boldsymbol{\Sigma}}^2, \tag{7.20}$$

where $\|\boldsymbol{w}\|_{\boldsymbol{\Sigma}}^2 = \boldsymbol{w}^*\boldsymbol{\Sigma}\boldsymbol{w} = \|\boldsymbol{X}\boldsymbol{w}\|_2^2$ is the norm induced by $\boldsymbol{\Sigma}$. Since $\boldsymbol{\Sigma}$ is invertible when $m > n$ and $\boldsymbol{X}$ has full column rank, the last equation also implies linear convergence of $\mathbb{E}\|\boldsymbol{\beta}_t - \boldsymbol{\beta}^\star\|^2$. The final results exist in Leventhal and Lewis [126], Strohmer and Vershynin [206] but there is utility in seeing the two proofs in a form that differs from their original presentation, side by side. In this setting, both RK and RGS are essentially equivalent (without computational considerations).

7.3.2 Overconstrained System, Inconsistent

Here, we will assume that $m > n$, $\boldsymbol{X}$ is full column rank, and the system is inconsistent, so $\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta}_{\boldsymbol{LS}} + \boldsymbol{r}$, where $\boldsymbol{r}$ is such that $\boldsymbol{X}^*\boldsymbol{r} = 0$. It is easy to see this condition, because as mentioned earlier,

$$\boldsymbol{\beta}_{\boldsymbol{LS}} = (\boldsymbol{X}^*\boldsymbol{X})^{-1}\boldsymbol{X}^*\boldsymbol{y},$$

implying that $\boldsymbol{X}^*\boldsymbol{X}\boldsymbol{\beta}_{\boldsymbol{LS}} = \boldsymbol{X}^*\boldsymbol{y}$. Substituting $\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta}_{\boldsymbol{LS}} + \boldsymbol{r}$ gives that $\boldsymbol{X}^*\boldsymbol{r} = 0$.

In this setting, RK is known to not converge to the least squares solution, as is easily verified experimentally and geometrically. The tightest convergence upper bounds known are by [141] and [235] who show that

$$\begin{aligned}\mathbb{E}\|\boldsymbol{\beta}_t - \boldsymbol{\beta}_{\boldsymbol{LS}}\|_2^2 &\leq \left(1 - \frac{\lambda_{\min}(\boldsymbol{\Sigma})}{Tr(\boldsymbol{\Sigma})}\right)^t \|\boldsymbol{\beta}_0 - \boldsymbol{\beta}_{\boldsymbol{LS}}\|_2^2 + \frac{\|\boldsymbol{r}\|_2^2}{\lambda_{\min}(\boldsymbol{\Sigma})} \\ &= \left(1 - \frac{\sigma_{\min}^2(\boldsymbol{X})}{\|\boldsymbol{X}\|_F^2}\right)^t \|\boldsymbol{\beta}_0 - \boldsymbol{\beta}_{\boldsymbol{LS}}\|_2^2 + \frac{\|\boldsymbol{r}\|_2^2}{\sigma_{\min}(\boldsymbol{X})^2},\end{aligned}$$

where we write $\sigma_{\min}(\boldsymbol{X})$ to denote the smallest (non-zero) singular value of $\boldsymbol{X}$ and again $\|\boldsymbol{X}\|_F$ its Frobenius norm. Attempting the previous proof, (7.17) no longer holds – the Pythagorean theorem fails because $\boldsymbol{\beta}_{t+1} - \boldsymbol{\beta}_{\boldsymbol{LS}}$ is no longer orthogonal to $\boldsymbol{X}^i$ since $\boldsymbol{X}^i(\boldsymbol{\beta}_{t+1} - \boldsymbol{\beta}_{\boldsymbol{LS}}) = y^i - \boldsymbol{X}^i\boldsymbol{\beta}_{\boldsymbol{LS}} \neq 0$. Intuitively, the reason RK does not converge is that every update of RK (say of row i) is a projection onto the "wrong" hyperplane that has constant y^i (where the "right" hyperplane would involve projecting onto a parallel hyperplane with constant $y^i - r^i$ where $\boldsymbol{r}$ was defined above). An alternative intuition is that all RK updates are in the span of the rows, but $\boldsymbol{\beta}_{\boldsymbol{LS}}$ is not in the row span. These intuitive explanations are easily confirmed by experiments seen in [141, 235]. Zouzias and Freris [235] alleviate this issue with the REK algorithm, whose convergence obeys

$$\mathbb{E}\|\boldsymbol{\beta}_t - \boldsymbol{\beta}_{\boldsymbol{LS}}\|_2^2 \leq \left(1 - \frac{\sigma_{\min}^2(\boldsymbol{X})}{\|\boldsymbol{X}\|_F^2}\right)^{\lfloor t/2 \rfloor} \left(1 + 2\frac{\sigma_{\min}^2(\boldsymbol{X})}{\sigma_{\max}^2(\boldsymbol{X})}\|\boldsymbol{\beta}_{\boldsymbol{LS}}\|_2^2\right). \tag{7.21}$$

However, the fate of RK doesn't hold for RGS. Almost magically, in the previous proof, the Pythagorean theorem still holds in Eq.(7.18) because

$$\boldsymbol{X}_{(j)}^*(\boldsymbol{X}\boldsymbol{\beta}_{t+1} - \boldsymbol{X}\boldsymbol{\beta}_{\boldsymbol{LS}}) = \boldsymbol{X}_{(j)}^*(\boldsymbol{X}\boldsymbol{\beta}_{t+1} - \boldsymbol{y}) + \boldsymbol{X}_{(j)}^*(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}_{\boldsymbol{LS}}) = 0.$$

The first term is 0 by the optimality condition for $\boldsymbol{\beta}_{t+1}$ i.e. $\boldsymbol{X}^*_{(j)}(\boldsymbol{X}\boldsymbol{\beta}_{t+1} - \boldsymbol{y}) = \partial L/\partial \beta^j = 0$. The second term is zero by the global optimality of $\boldsymbol{\beta_{LS}}$, i.e. $\boldsymbol{X}^*(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta_{LS}}) = \nabla L = 0$. Also, $\boldsymbol{\Sigma}$ is full rank as before. Indeed, RGS works in the space of fitted values $\boldsymbol{X}\boldsymbol{\beta}$ and not the iterates $\boldsymbol{\beta}$.

In summary, RK does not converge to the LS solution, but RGS does at the same linear rate. This is what motivated the development of Randomized Extended Kaczmarz (REK) by Zouzias and Freris [235] which, as discussed earlier, is a modification of RK designed to converge to $\boldsymbol{\beta_{LS}}$ by randomly projecting out r. An independent paper by Dumitrescu [57] argues however that in this setting RGS is preferable to REK in terms of computational convergence.

7.3.3 Underconstrained System, Infinite Solutions

Here, we will assume that $m < n$, $\boldsymbol{X}$ is full row rank and the system is consistent with infinitely many solutions. As mentioned earlier, it is easy to show that

$$\boldsymbol{\beta_{LN}} = \boldsymbol{X}^*(\boldsymbol{X}\boldsymbol{X}^*)^{-1}\boldsymbol{y}$$

(which clearly satisfies $\boldsymbol{X}\boldsymbol{\beta_{LN}} = \boldsymbol{y}$). Every other consistent solution can be expressed as

$$\boldsymbol{\beta} = \boldsymbol{\beta_{LN}} + \boldsymbol{z} \quad \text{where} \quad \boldsymbol{X}\boldsymbol{z} = 0.$$

Clearly any such $\boldsymbol{\beta}$ would also satisfy $\boldsymbol{X}\boldsymbol{\beta} = \boldsymbol{X}\boldsymbol{\beta_{LN}} = 0$. Since $\boldsymbol{X}\boldsymbol{z} = 0$, $\boldsymbol{z} \perp \boldsymbol{\beta_{LN}}$ implying $\|\boldsymbol{\beta}\|^2 = \|\boldsymbol{\beta_{LN}}\|^2 + \|\boldsymbol{z}\|^2$, showing that $\boldsymbol{\beta_{LN}}$ is indeed the minimum norm solution as claimed.

In this case, RK has good behavior, and starting from $\boldsymbol{\beta}_0 = 0$, it does converge linearly to $\boldsymbol{\beta_{LN}}$. Intuitively, $\boldsymbol{\beta_{LN}} = \boldsymbol{X}^*\boldsymbol{\alpha}$ (for $\boldsymbol{\alpha} = (\boldsymbol{X}\boldsymbol{X}^*)^{-1}\boldsymbol{y}$) and hence is in the row span of $\boldsymbol{X}$. Starting from $\boldsymbol{\beta}_0 = 0$, RK only adds multiples of rows to its iterates, and hence will never have any component orthogonal to the row span of $\boldsymbol{X}$. There is exactly one solution with no component orthogonal to the row span of $\boldsymbol{X}$, and that is $\boldsymbol{\beta_{LN}}$, and hence RK converges linearly to the required point, where the rate can be bounded in exactly the same way as (7.20). It is important not to start from an arbitrary $\boldsymbol{\beta}_0$ since the RK updates can never eliminate any component of $\boldsymbol{\beta}_0$ that is perpendicular to the row span of $\boldsymbol{X}$. Of course, the same properties are shared by REK for this case as well.

Mathematically, the previous earlier proof works because the Pythagorean theorem holds since it is a consistent system. Now, $\boldsymbol{\Sigma}$ is not full rank but note that since both $\boldsymbol{\beta_{LN}}$ and $\boldsymbol{\beta}_t$ are in the row span, $\boldsymbol{\beta}_t - \boldsymbol{\beta_{LN}}$ has no component orthogonal to $\boldsymbol{X}$ (unless it equals zero in which case the algorithm has already converged). Hence $\lambda_{\min}(\boldsymbol{\Sigma})\|\boldsymbol{\beta}_t - \boldsymbol{\beta_{LN}}\|^2 \leq \|\boldsymbol{X}(\boldsymbol{\beta}_t - \boldsymbol{\beta_{LN}})\|^2$ holds, where $\lambda_{\min}(\boldsymbol{\Sigma})$ is now understood to be the smallest positive eigenvalue of $\boldsymbol{\Sigma}$.

RGS unfortunately suffers the opposite fate. The iterates do not converge to $\boldsymbol{\beta_{LN}}$, even though $\boldsymbol{X}\boldsymbol{\beta}_t$ does converge to $\boldsymbol{X}\boldsymbol{\beta_{LN}}$. Mathematically, the convergence proof still carries forward as before until (7.20), but in the last step when $\boldsymbol{X}^*\boldsymbol{X}$ cannot be inverted because it is not full rank. Hence we get convergence of the residual to zero, without getting convergence of the iterates to the least norm solution.

Unfortunately, when each update is cheaper for RK than RGS (due to matrix size), RGS is preferred for reasons of convergence and when it is cheaper for RGS than RK, RK is preferred.

7.4 REGS

We next introduce an extension of RGS, analogous to the extension REK of RK. The purpose of extending RK was to allow for convergence to the least squares solution. Now, the purpose of extending RGS is to allow for convergence to the least norm solution. We view this method as a completion to the unified analysis of these approaches, and it may also possess advantages in its own right.

The algorithm

Consider the linear system (7.1) with $m < n$. Let $\boldsymbol{\beta_{LN}}$ denote the least norm solution of the underdetermined system as described in (7.3). The REGS algorithm is described by the following pseudo-code. Analogous to the role $\boldsymbol{z}$ plays in REK, $\boldsymbol{z}$ iteratively approximates the component in $\boldsymbol{\beta}$ orthogonal to the row-span of $\boldsymbol{X}$. By iteratively removing this component, we converge to the least norm solution.

Algorithm 11 Randomized Extended Gauss-Seidel (REGS)

```
procedure ($\boldsymbol{X}$, $\boldsymbol{y}$, maxIter)    ▷ $m \times n$ matrix $\boldsymbol{X}$, $\boldsymbol{y} \in \mathbb{C}^m$, maximum iterations $T$
    Initialize $\boldsymbol{\beta}_0 = \boldsymbol{0}$, $\boldsymbol{z}_0 = \boldsymbol{0}$
    for $t = 1, 2, \ldots, T$ do
        Choose column $\boldsymbol{X}_{(j)}$ with probability $\frac{\|\boldsymbol{X}_{(j)}\|_2^2}{\|\boldsymbol{X}\|_F^2}$
        Choose row $\boldsymbol{X}^i$ with probability $\frac{\|\boldsymbol{X}^i\|_2^2}{\|\boldsymbol{X}\|_F^2}$
        Set $\boldsymbol{\gamma}_t = \frac{\boldsymbol{X}^*_{(j)}(\boldsymbol{X}\boldsymbol{\beta}_{(t-1)} - \boldsymbol{y})\boldsymbol{e}_{(j)}}{\|\boldsymbol{X}_{(j)}\|_2^2}$
        Set $\boldsymbol{\beta}_t = \boldsymbol{\beta}_{t-1} + \boldsymbol{\gamma}_t$
        Set $\boldsymbol{P}_i = \boldsymbol{Id}_n - \frac{(\boldsymbol{X}^i)^*\boldsymbol{X}^i}{\|\boldsymbol{X}^i\|_2^2}$    ▷ $\boldsymbol{Id}_n$ denotes the $n \times n$ identity matrix
        Update $\boldsymbol{z}_t = \boldsymbol{P}_i(\boldsymbol{z}_{t-1} + \boldsymbol{\gamma}_t)$
        Update $\boldsymbol{\beta}_t^{LN} = \boldsymbol{\beta}_t - \boldsymbol{z}_t$
    end for
    Output $\boldsymbol{\beta}_t^{LN}$
end procedure
```

Main result

Our main result for the REGS method shows linear convergence to the least norm solution.

Theorem 23. *The REGS algorithm outputs an estimate $\boldsymbol{\beta}_T^{LN}$ such that*

$$\mathbb{E}\|\boldsymbol{\beta}_T^{LN} - \boldsymbol{\beta_{LN}}\|_2^2 \le \alpha^T \|\boldsymbol{\beta}^{LN}\|_2^2 + 2\alpha^{\lfloor T/2 \rfloor}\frac{B}{1-\alpha} \tag{7.22}$$

where $B = \frac{\|\boldsymbol{X}\boldsymbol{\beta_{LN}}\|_2^2}{\|\boldsymbol{X}\|_F^2}$ and $\alpha = \left(1 - \frac{\sigma^2_{min}(\boldsymbol{X})}{\|\boldsymbol{X}\|_F^2}\right)$.

Proof: We devote the remainder of this section to the proof of Theorem 23.

Let $\mathbb{E}_{t-1}$ denote the expected value conditional on the first $t-1$ iterations, and instate the notation of the theorem. That is, $\mathbb{E}_{t-1}[\cdot] = \mathbb{E}[\cdot \mid i_1, j_1, i_2, j_2, ... i_{t-1}, j_{t-1}]$ where i_{t^*} is the t^{*th} row chosen and j_{t^*} is the t^{*th} column chosen. We denote conditional expectation with respect to the choice of column as $\mathbb{E}^j_{t-1}[\cdot] = \mathbb{E}[\cdot \mid i_1, j_1, ... i_{t-1}, j_{t-1}, i_t]$. Similarly, we denote conditional expectation with respect to the choice of row as $\mathbb{E}^i_{t-1}[\cdot] = \mathbb{E}[\cdot \mid i_1, j_1, ... i_{t-1}, j_{t-1}, j_t]$. Then note by the law of total expectation we have that $\mathbb{E}_{t-1}[\cdot] = \mathbb{E}^i_{t-1}[\mathbb{E}^j_{t-1}[\cdot]]$. We will use the following elementary facts and lemmas.

Fact 1. *([235, Fact 3]) For any $\boldsymbol{P}_i$ as in the algorithm, $\mathbb{E}\|\boldsymbol{P}_i\boldsymbol{w}\|_2^2 \le \alpha\|\boldsymbol{w}\|_2^2$ for any $\boldsymbol{w}$.*

Lemma 26. *([126, Thm. 3.6]) We have that*

$$\mathbb{E}_{t-1}\|\boldsymbol{X}\boldsymbol{\beta}_t - \boldsymbol{X}\boldsymbol{\beta_{LN}}\|_2^2 \le \alpha\|\boldsymbol{X}\boldsymbol{\beta}_{t-1} - \boldsymbol{X}\boldsymbol{\beta_{LN}}\|_2^2$$

and that

$$\mathbb{E}\|\boldsymbol{X}\boldsymbol{\beta}_t - \boldsymbol{X}\boldsymbol{\beta_{LN}}\|_2^2 \le \alpha^t\|\boldsymbol{X}\boldsymbol{\beta}_0 - \boldsymbol{X}\boldsymbol{\beta_{LN}}\|_2^2.$$

Now we first consider $\|\boldsymbol{\beta}_t^{LN} - \boldsymbol{\beta_{LN}}\|_2^2$:

$$\begin{aligned}
\|\boldsymbol{\beta}_t^{LN} - \boldsymbol{\beta_{LN}}\|_2^2 &= \|\boldsymbol{\beta}_t - \boldsymbol{z}_t - \boldsymbol{\beta_{LN}}\|_2^2 \\
&= \|\boldsymbol{\beta}_t - \boldsymbol{P}_i(\boldsymbol{z}_{t-1} + \boldsymbol{\gamma}_t) - \boldsymbol{P}_i\boldsymbol{\beta_{LN}} - (\boldsymbol{Id}_n - \boldsymbol{P}_i)\boldsymbol{\beta_{LN}}\|_2^2 \\
&= \|\boldsymbol{\beta}_t - \boldsymbol{P}_i(\boldsymbol{z}_{t-1} + \boldsymbol{\beta}_t - \boldsymbol{\beta}_{t-1}) - \boldsymbol{P}_i\boldsymbol{\beta_{LN}} - (\boldsymbol{Id}_n - \boldsymbol{P}_i)\boldsymbol{\beta_{LN}}\|_2^2 \\
&= \|(\boldsymbol{Id}_n - \boldsymbol{P}_i)\boldsymbol{\beta}_t + \boldsymbol{P}_i(\boldsymbol{\beta}_{t-1} - \boldsymbol{z}_{t-1}) - \boldsymbol{P}_i\boldsymbol{\beta_{LN}} - (\boldsymbol{Id}_n - \boldsymbol{P}_i)\boldsymbol{\beta_{LN}}\|_2^2 \\
&= \|(\boldsymbol{Id}_n - \boldsymbol{P}_i)\boldsymbol{\beta}_t + \boldsymbol{P}_i\boldsymbol{\beta}_{t-1}^{LN} - \boldsymbol{P}_i\boldsymbol{\beta_{LN}} - (\boldsymbol{Id}_n - \boldsymbol{P}_i)\boldsymbol{\beta_{LN}}\|_2^2 \\
&= \|\boldsymbol{P}_i(\boldsymbol{\beta}_{t-1}^{LN} - \boldsymbol{\beta_{LN}}) + (\boldsymbol{Id}_n - \boldsymbol{P}_i)(\boldsymbol{\beta}_t - \boldsymbol{\beta_{LN}})\|_2^2 \\
&= \|\boldsymbol{P}_i(\boldsymbol{\beta}_{t-1}^{LN} - \boldsymbol{\beta_{LN}})\|_2^2 + \|(\boldsymbol{Id}_n - \boldsymbol{P}_i)(\boldsymbol{\beta}_t - \boldsymbol{\beta_{LN}})\|_2^2. \qquad (7.23)
\end{aligned}$$

So far, we have only used substitution of variables as defined for the algorithm and that $\boldsymbol{\beta_{LN}} = \boldsymbol{P}_i\boldsymbol{\beta_{LN}} + (\boldsymbol{Id}_n - \boldsymbol{P}_i)\boldsymbol{\beta_{LN}}$ is an orthogonal decomposition. We first focus on the expected value of the second term.

Lemma 27. *We also have that*

$$\mathbb{E}_{t-1}\|(\boldsymbol{Id}_n - \boldsymbol{P}_i)(\boldsymbol{\beta}_t - \boldsymbol{\beta_{LN}})\|_2^2 \leq \frac{\alpha\|\boldsymbol{X}(\boldsymbol{\beta}_{t-1} - \boldsymbol{\beta_{LN}})\|_2^2}{\|\boldsymbol{X}\|_F^2}.$$

Proof:

$$\begin{aligned}
&\mathbb{E}_{t-1}\|(\boldsymbol{Id}_n - \boldsymbol{P}_i)(\boldsymbol{\beta}_t - \boldsymbol{\beta_{LN}})\|_2^2 \\
&\qquad = \mathbb{E}_{t-1}[(\boldsymbol{\beta}_t - \boldsymbol{\beta_{LN}})^*(\boldsymbol{Id}_n - \boldsymbol{P}_i)^*(\boldsymbol{Id}_n - \boldsymbol{P}_i)(\boldsymbol{\beta}_t - \boldsymbol{\beta_{LN}})] \\
&\qquad = \mathbb{E}_{t-1}[(\boldsymbol{\beta}_t - \boldsymbol{\beta_{LN}})^*(\boldsymbol{Id}_n - \boldsymbol{P}_i)(\boldsymbol{\beta}_t - \boldsymbol{\beta_{LN}})] \\
&\qquad = \mathbb{E}_{t-1}\left[(\boldsymbol{\beta}_t - \boldsymbol{\beta_{LN}})^*\left(\frac{(\boldsymbol{X}^i)^*\boldsymbol{X}^i}{\|\boldsymbol{X}^i\|_2^2}\right)(\boldsymbol{\beta}_t - \boldsymbol{\beta_{LN}})\right] \\
&\qquad = \mathbb{E}_{t-1}\left[\frac{\|\boldsymbol{X}^i(\boldsymbol{\beta}_t - \boldsymbol{\beta_{LN}})\|_2^2}{\|\boldsymbol{X}^i\|_2^2}\right] \\
&\qquad = \mathbb{E}_{t-1}^j\left[\mathbb{E}_{t-1}^i\frac{\|\boldsymbol{X}^i(\boldsymbol{\beta}_t - \boldsymbol{\beta_{LN}})\|_2^2}{\|\boldsymbol{X}^i\|_2^2}\right] \\
&\qquad = \mathbb{E}_{t-1}^j\left[\sum_{i=1}^m \frac{\|\boldsymbol{X}^i(\boldsymbol{\beta}_t - \boldsymbol{\beta_{LN}})\|_2^2}{\|\boldsymbol{X}^i\|_2^2}\cdot\frac{\|\boldsymbol{X}^i\|_2^2}{\|\boldsymbol{X}\|_F^2}\right] \\
&\qquad = \mathbb{E}_{t-1}^j\left[\frac{\|\boldsymbol{X}(\boldsymbol{\beta}_t - \boldsymbol{\beta_{LN}})\|_2^2}{\|\boldsymbol{X}\|_F^2}\right] \\
&\qquad \leq \frac{\alpha\|\boldsymbol{X}(\boldsymbol{\beta}_{t-1} - \boldsymbol{\beta_{LN}})\|_2^2}{\|\boldsymbol{X}\|_F^2}.
\end{aligned}$$

The first line follows by expanding the norm, the second line since $(\boldsymbol{Id}_n - \boldsymbol{P}_i)$ is a projection matrix, the third line from the definition of $\boldsymbol{P}_i$, the fourth line is computation, the fifth line follows from the law of total expectation, the next two lines are computation, and finally the last line follows by Lemma 26. Notice that in the seventh line, $\mathbb{E}_{t-1}^j = \mathbb{E}_{t-1}$ because the random variable $\boldsymbol{\beta}_t$ only depends on the choice of columns.

We want to control the term $r_t = \mathbb{E}\|(\boldsymbol{Id}_n - \boldsymbol{P}_i)(\boldsymbol{\beta}_t - \boldsymbol{\beta_{LN}})\|_2^2$ by bounding it by some α and B such that $r_t \leq \alpha^t B$. We calculate this here:

$$\begin{aligned}
\mathbb{E}\|(\boldsymbol{Id}_n - \boldsymbol{P}_i)(\boldsymbol{\beta}_t - \boldsymbol{\beta_{LN}})\|_2^2 &= \mathbb{E}[\mathbb{E}_{t-1}\|(\boldsymbol{Id}_n - \boldsymbol{P}_i)(\boldsymbol{\beta}_t - \boldsymbol{\beta_{LN}})\|_2^2] \\
&\leq \frac{\alpha\mathbb{E}\|\boldsymbol{X}(\boldsymbol{\beta}_{t-1} - \boldsymbol{\beta_{LN}})\|_2^2}{\|\boldsymbol{X}\|_F^2} \\
&\leq \alpha^t \frac{\|\boldsymbol{X}\boldsymbol{\beta}_0 - \boldsymbol{X}\boldsymbol{\beta_{LN}}\|_2^2}{\|\boldsymbol{X}\|_F^2}.
\end{aligned}$$

The first line follows by definition, the second is by Lemma 27, and the third by Lemma 26.

Finally, we take the expected value of $\|\boldsymbol{\beta}_t^{LN} - \boldsymbol{\beta_{LN}}\|_2^2$. From equation (7.23) and using Fact 1 we obtain:

$$\begin{aligned}
\mathbb{E}\|\boldsymbol{\beta}_t^{LN} - \boldsymbol{\beta_{LN}}\|_2^2 &= \mathbb{E}\|\boldsymbol{P}_i(\boldsymbol{\beta}_{t-1}^{LN} - \boldsymbol{\beta_{LN}})\|_2^2 + \mathbb{E}\|(\boldsymbol{Id}_n - \boldsymbol{P}_i)(\boldsymbol{\beta}_t - \boldsymbol{\beta_{LN}})\|_2^2 \\
&\leq \alpha\mathbb{E}\|(\boldsymbol{\beta}_{t-1}^{LN} - \boldsymbol{\beta_{LN}})\|_2^2 + \mathbb{E}\|(\boldsymbol{Id}_n - \boldsymbol{P}_i)(\boldsymbol{\beta}_t - \boldsymbol{\beta_{LN}})\|_2^2.
\end{aligned}$$

We complete the proof using the following lemma from [235]:

Lemma 28. *([235, Thm. 8]) Suppose that for some $\alpha, \bar{\alpha} < 1$, the following bounds hold for all $t^* \geq 0$:*

$$\mathbb{E}\|\boldsymbol{\beta}_{t^*}^{LN} - \boldsymbol{\beta_{LN}}\|_2^2 \leq \alpha\mathbb{E}\|\boldsymbol{\beta}_{t^*-1}^{LN} - \boldsymbol{\beta_{LN}}\|_2^2 + r_{t^*} \text{ and } r_{t^*} \leq \bar{\alpha}^{t^*} B.$$

Then for any $T > 0$,

$$\mathbb{E}\|\boldsymbol{\beta}_T^{LN} - \boldsymbol{\beta_{LN}}\|_2^2 \leq \alpha^T \|\boldsymbol{\beta}_0^{LN} - \boldsymbol{\beta_{LN}}\|_2^2 + (\alpha^{\lfloor T/2 \rfloor} + \bar{\alpha}^{\lfloor T/2 \rfloor})\frac{B}{1-\alpha}.$$

Letting $\alpha = \bar{\alpha} = \alpha$, $r_t^* = \mathbb{E}\|(\boldsymbol{Id}_n - \boldsymbol{P}_i)(\boldsymbol{\beta}_t - \boldsymbol{\beta_{LN}})\|_2^2$, $B = \frac{\|\boldsymbol{X}\boldsymbol{\beta}_0 - \boldsymbol{X}\boldsymbol{\beta_{LN}}\|_2^2}{\|\boldsymbol{X}\|_F^2}$, and noting that $\boldsymbol{\beta}_0^{LN} = \boldsymbol{\beta}_0 = 0$, we complete the proof of Theorem 23.

Comparison

Theorem 23 shows that, like the RK and REK methods, REGS converges linearly to the least-norm solution in the underdetermined case. We believe it serves to complement existing analysis and completes the theory of these iterative methods in all three cases of interest. For that reason, we compare the three approaches for that setting here. For ease of comparison, set α as in Theorem 23, and write $\kappa = \sigma_{\max}(\boldsymbol{X})/\sigma_{\min}(\boldsymbol{X})$ for the condition number of $\boldsymbol{X}$. From the convergence rate bounds for RK [206] and REK [235] given in Section 7.3, and after applying elementary bounds to (7.22) of Theorem 23, we have:

$$\text{(RK)} \qquad \mathbb{E}\|\boldsymbol{\beta}_t - \boldsymbol{\beta_{LN}}\|_2^2 \leq \alpha^t \|\boldsymbol{\beta_{LN}}\|_2^2 \tag{7.24}$$

$$\text{(REK)} \qquad \mathbb{E}\|\boldsymbol{\beta}_{2t} - \boldsymbol{\beta_{LN}}\|_2^2 \leq \alpha^t (1 + 2\kappa^2) \|\boldsymbol{\beta_{LN}}\|_2^2 \tag{7.25}$$

$$\text{(REGS)} \qquad \mathbb{E}\|\boldsymbol{\beta}_{2t} - \boldsymbol{\beta_{LN}}\|_2^2 \leq \alpha^t (1 + 2\kappa^2) \|\boldsymbol{\beta_{LN}}\|_2^2. \tag{7.26}$$

Thus, up to constant terms (which are likely artifacts of the proofs), the bounds provide the same convergence rate α, which is not surprising in light of the connections between the methods. We compare these approaches experimentally in the next section.

7.5 Empirical Results

In this section we present our experimental results. The code used to run these experiments can be found at [7]. For each experiment, we initialize a matrix $\boldsymbol{X}$ and vector $\boldsymbol{\beta}$ with independent standard normal entries and run 50 trials. The right hand side $\boldsymbol{y}$ is taken to be $\boldsymbol{X\beta}$. At each iteration t, we keep track of the ℓ_2-error $\|\boldsymbol{\beta}_t^{LN} - \boldsymbol{\beta_{LN}}\|_2^2$ and fix the stopping criterion to be $\|\boldsymbol{\beta}_t^{LN} - \boldsymbol{\beta_{LN}}\|_2^2 < 10^{-6}$ (of course in practice one chooses a more practical criterion). In each plot, the solid blue line represents the median ℓ_2-error at iteration k, the light blue shaded region captures the range of error across trials, and the red line represents the theoretical upper bound at each iteration. In Figure 7.1, we show the convergence of $\boldsymbol{\beta}_t^{LN}$ for varying sized underdetermined linear systems. In Figure 7.2, we show the convergence of a matrix $\boldsymbol{X}$ of size 700x1000 and its theoretical upper bound. As it turns out, the REGS algorithm often converges much faster than the theoretical worst-case bound.

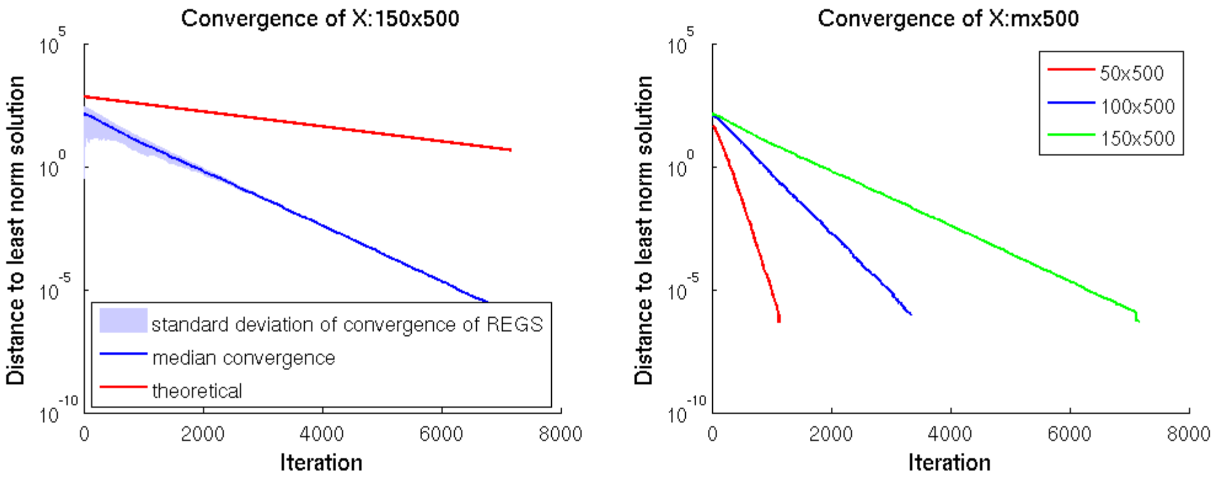


Figure 7.1: Left: ℓ_2-error of REGS on a 150×500 matrix and its the theoretical bound. Right: Comparison of ℓ_2-error of REGS for $m \times 500$ sized matricies with $m = 50, 100, 150$.

We also tested REGS on tomography problems using the Regularization toolbox by Hansen [93] (`http://www.imm.dtu.dk/~pcha/Regutools/`). For the 2D tomography problem $\boldsymbol{X\beta} = \boldsymbol{y}$ with $\boldsymbol{X}$ an $m \times n$ matrix where $n = dN^2$ and $m = N^2$, we use $N = 20$ and $d = 3$ for our experiments. Here, $\boldsymbol{X}$ consists of samples of absorption along a random line on an $N \times N$ grid and d is the oversampling factor. The results from this experiment are shown in Figure 7.2.

We also compare the performance of all four algorithms (RK, REK, RGS, REGS) under the different settings discussed in this chapter. Each line in each plot represents the median ℓ_2-error at that iteration or CPU time over 50 trials using a stopping criterion of 10^{-6}. For the underdetermined case, $\boldsymbol{X}$ is a 50×500 Gaussian matrix and a 500×50 Gaussian matrix for the overdetermined cases. In the overdetermined, inconsistent case, we set $\boldsymbol{y} = \boldsymbol{X\beta} + \mathbf{r}$ where $\mathbf{r} \in \text{null}(\mathbf{X}^*)$ (computed in Matlab using the `null()` function). Figure 7.3, Figure 7.4, and Figure 7.5 show the empirical results for the underdetermined, overdetermined inconsistent, and overdetermined consistent cases respectively. Note we only plot the methods which actually converge to the desired solution in each case. Looking at iterations to convergence, it seems that RK and RGS converge faster than their extended counterparts while REGS and REK converge to the desired solution at about the same rate.

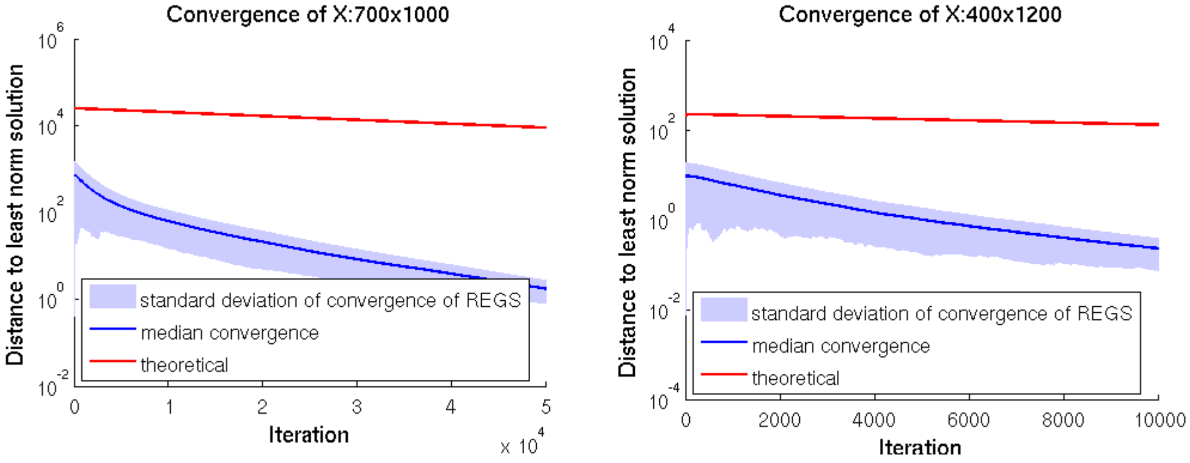


Figure 7.2: Left: ℓ_2-error of REGS on a 700×1000 matrix and its the theoretical bound. Right: ℓ_2-error of REGS on the tomography problem with a 400×1200 matrix.

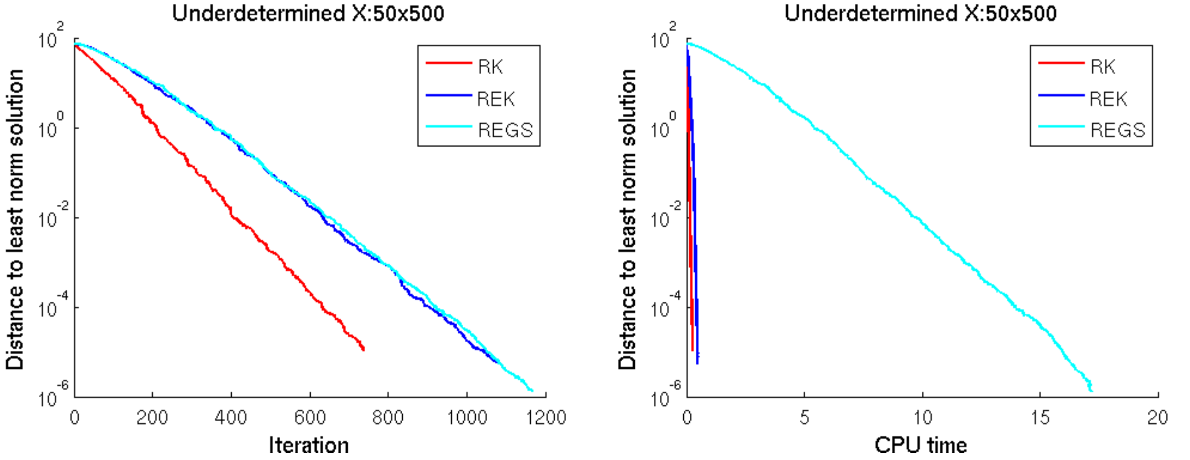


Figure 7.3: Comparison of median ℓ_2-error of RK, REK, and REGS for an underdetermined system.

Conclusion

The Kaczmarz and Gauss-Seidel methods operate in two different spaces (ie. row versus column space), but share many parallels. In this chapter we drew connections between these two methods, highlighting the similarities and differences in convergence analysis. The approaches possess conflicting convergence properties; RK converges to the desired solution in the underdetermined case but not the inconsistent overdetermined setting, while RGS does the exact opposite. The extended method REK in the Kaczmarz framework fixes this issue, converging to the solution in both scenarios. Here, we present the REGS method, a natural extension of RGS, which completes the overall picture. With these results, we present a unified analysis of all four methods which we hope will assist researchers working with these approaches.

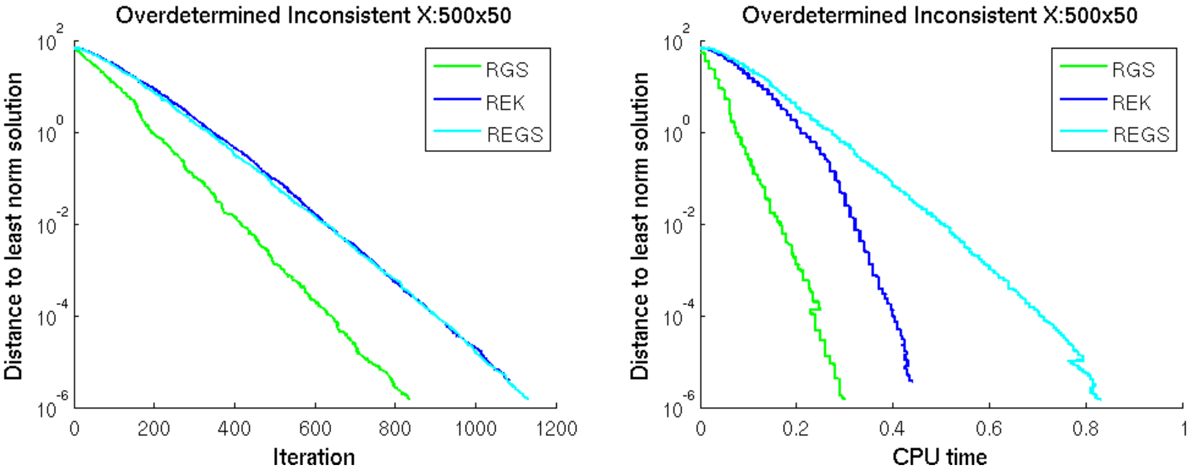


Figure 7.4: Comparison of median ℓ_2-error of RGS, REK, and REGS for an overdetermined, inconsistent system.

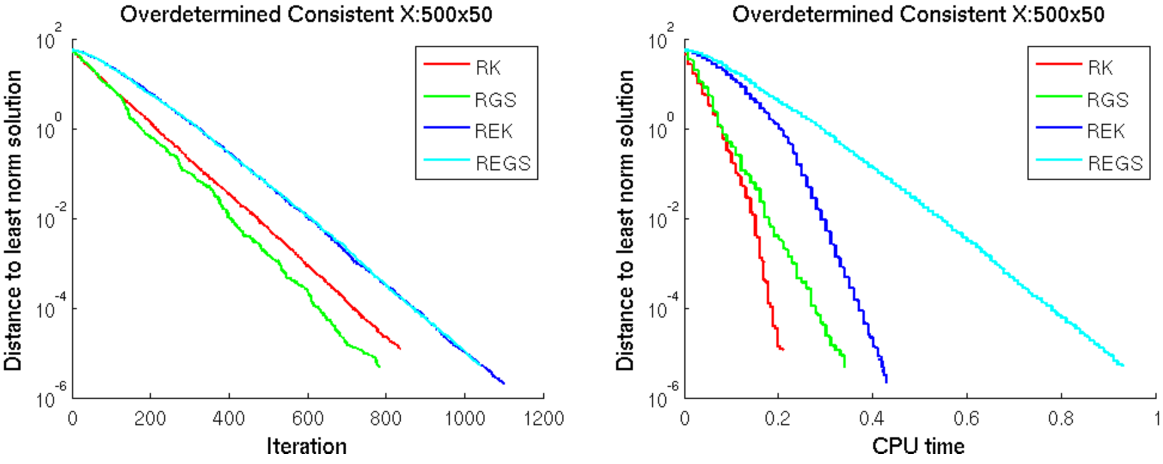


Figure 7.5: Comparison of median ℓ_2-error of RK, RGS, REK, and REGS for an overdetermined, consistent system

Chapter 8

Ridge Regression : Randomized algorithms for Tikhonov regularization

The Kaczmarz and Gauss-Seidel methods aim to solve a linear $m \times n$ system $\boldsymbol{X}\boldsymbol{\beta} = \boldsymbol{y}$ by iteratively refining the solution estimate; the former uses random rows of $\boldsymbol{X}$ and the latter uses random columns of $\boldsymbol{X}$. Interest in these methods was recently revitalized by a proof of Strohmer and Vershynin showing linear convergence in expectation for a *randomized* Kaczmarz method variant (RK), and a similar result for the randomized Gauss-Seidel algorithm (RGS) was later proved by Lewis and Leventhal. Recent work unified the analysis of these algorithms for the overcomplete and undercomplete systems, converging to the ordinary least squares (OLS) solution and the minimum Euclidean norm solution respectively. This paper considers the natural follow-up to the OLS problem, ridge regression, which solves $(\boldsymbol{X}^*\boldsymbol{X} + \lambda \boldsymbol{I})\boldsymbol{\beta} = \boldsymbol{X}^*\boldsymbol{y}$. We present particular variants of RK and RGS for solving this system and derive their convergence rates. We argue that a recent proposal by Ivanov and Zhdanov to solve this system, that can be interpreted as randomly sampling both rows and columns, is suboptimal. Instead, we claim that one should always use RGS (columns) when $m > n$ and RK (rows) when $m < n$. This difference in behavior is simply related to the minimum eigenvalue of two related positive semidefinite matrices, $\boldsymbol{X}^*\boldsymbol{X} + \lambda \boldsymbol{I}_n$ and $\boldsymbol{X}\boldsymbol{X}^* + \lambda \boldsymbol{I}_m$ when $m > n$ or $m < n$.

8.1 Introduction

We consider solving the linear system of equations given by Tikhonov-regularized regression or ridge regression,

$$(\boldsymbol{X}^*\boldsymbol{X} + \lambda \boldsymbol{I})\boldsymbol{\beta} = \boldsymbol{X}^*\boldsymbol{y}, \tag{8.1}$$

for a (real or complex) $m \times n$ matrix $\boldsymbol{X}$, in two settings – when $m < n$ and when $m > n$, using randomized iterative algorithms. Recently, the work of Strohmer and Vershynin [206] sparked a revival of interest in using the Kaczmarz method for solving linear systems of the form $\boldsymbol{X}\boldsymbol{\beta} = \boldsymbol{y}$. They proved a linear[1] convergence rate of the Randomized Kaczmarz (RK) algorithm that works on the rows of $\boldsymbol{X}$ (data points). Leventhal and Lewis [126] after proved linear convergence of Randomized Gauss-Seidel (RGS), (also known as Randomized Coordinate Descent, which we will use interchangeably), which instead operates on the columns of $\boldsymbol{X}$ (features). Recently, Ma* et al. [134] provided a unifying analysis of RK and RGS in a variety of settings.

[1] Mathematicians often refer to linear convergence as exponential convergence.

Solving linear systems of equations $\boldsymbol{X\beta} = \boldsymbol{y}$, also sometimes called ordinary least squares (OLS) regression, dates back to the times of Gauss, who introduced what we now know as Gaussian elimination. A dominant iterative approach to solving linear systems is the conjugate gradient method; it can also be seen as solving a convex optimization problem $\min_{\boldsymbol{\beta}} \|\boldsymbol{y} - \boldsymbol{X\beta}\|_2^2$.

For statistical as well as computational reasons, one often prefers not just to solve for the OLS solution, but instead what is called *ridge* regression or *Tikhonov-regularized* least squares regression. This corresponds to solving the convex optimization problem $\min_{\boldsymbol{\beta}} \|\boldsymbol{y} - \boldsymbol{X\beta}\|_2^2 + \lambda\|\boldsymbol{\beta}\|^2$ for a given regularization level λ.

There exist a large number of algorithms, iterative and not, randomized and not, for this problem. In this work, we will only be concerned with a particular subclass of algorithms. Specifically, we present an approach for this problem which is motivated by recent work on the randomized Kaczmarz and Gauss-Seidel methods. We analyze the convergence rates of our two proposed methods (variants of RK and RGS), showing again linear convergence in expectation, but the emphasis will be on the effective condition number that comes to play for our algorithms when $m > n$ and $m < n$. We contrast this with a previous approach, showing both analytically and empirically the drawbacks of the prior method. Our contribution thus extends the unifying framework of these iterative approaches (as done by Ma* et al. [134] for OLS) to the setting of ridge regression, while also providing methods with improved performance.

8.1.1 Paper Outline

We first introduce the two most relevant algorithms for our paper, Randomized Kaczmarz (RK) and Randomized Gauss-Seidel (RGS), for solving the ordinary least squares problem in Section 8.2. In Section 8.3, we describe what happens when RK or RGS is naively applied to the ridge regression problem, and discuss a recent proposal to tackle this issue (which can coincidentally be viewed as a combination of an RK-like and RGS-like update), and present its drawbacks. Then, in Section 8.4 we describe our proposed algorithms, that overcomes these drawbacks, and provide a simple unified analysis in various settings. We conclude with detailed experiments that agree with the theory in Section 8.5.

8.2 Randomized Algorithms for OLS

We begin by briefly describing the randomized Kaczmarz and Gauss-Siedel methods, which serve as the foundation to our approach for ridge regression. Throughout the paper we will consider an $m \times n$ (real or complex) matrix $\boldsymbol{X}$ and write $\boldsymbol{X}^i$ to represent the ith row of $\boldsymbol{X}$ (or ith entry of a vector) and $\boldsymbol{X}_{(j)}$ to denote the jth column. We will write solution estimations $\boldsymbol{\beta}$ as column vectors. We write vectors and matrices in boldface, and constants in standard font. The singular values of a matrix $\boldsymbol{X}$ are written as $\sigma(\boldsymbol{X})$ or just σ, with subscripts $\min$, $\max$ or integer values corresponding to the smallest, largest, and numerically ordered values. We denote the identity matrix by $\boldsymbol{I}$, with a subscript denoting the dimension when needed. We use the norm notation $\|\boldsymbol{z}\|^2_{\boldsymbol{A}^*\boldsymbol{A}}$ to mean $\langle \boldsymbol{z}, \boldsymbol{A}^*\boldsymbol{A}\boldsymbol{z}\rangle = \|\boldsymbol{A}\boldsymbol{z}\|^2$. Unless otherwise specified, the norm $\|\cdot\|$ denotes the standard Euclidean (or spectral) norm.

8.2.1 Randomized Kaczmarz (RK) for $\boldsymbol{X\beta} = \boldsymbol{y}$

The Kaczmarz method [111] is also known in the tomography setting as the *Algebraic Reconstruction Technique* (ART) [29, 79, 97, 140]. It has long been observed that selecting the rows i in a random fashion improves the algorithm's performance, reducing the possibility of slow convergence due to adversarial or unfortunate row ordering [90, 98]. Recently, Strohmer and Vershynin [206] showed that the RK method converges linearly to the solution $\boldsymbol{\beta}^\star$ of $\boldsymbol{X}\boldsymbol{\beta} = \boldsymbol{y}$ in expectation, with a rate that depends on natural geometric properties of the system, improving upon previous convergence analysis (e.g. [232]). In particular, they propose the variant of the Kaczmarz update with the following selection strategy:

$$\boldsymbol{\beta}_{t+1} := \boldsymbol{\beta}_t + \frac{(y^i - \boldsymbol{X}^i\boldsymbol{\beta}_t)}{\|\boldsymbol{X}^i\|_2^2}(\boldsymbol{X}^i)^*, \quad \text{where} \quad \Pr(\text{row} = i) = \frac{\|\boldsymbol{X}^i\|_2^2}{\|\boldsymbol{X}\|_F^2}, \tag{8.2}$$

where the first estimation $\boldsymbol{\beta}_0$ is chosen arbitrarily and $\|\boldsymbol{X}\|_F$ denotes the Frobenius norm of $\boldsymbol{X}$.

Strohmer and Vershynin [206] then prove that the iterates $\boldsymbol{\beta}_t$ of this method satisfy the following,

$$\|\boldsymbol{\beta}_t - \boldsymbol{\beta}^\star\|_2^2 \leq \left(1 - \frac{\sigma_{\min}^2(\boldsymbol{X})}{\|\boldsymbol{X}\|_F^2}\right)^t \|\boldsymbol{\beta}_0 - \boldsymbol{\beta}^\star\|_2^2. \tag{8.3}$$

This result was extended to the inconsistent case [141], derived probabilistically [37], accelerated in multiple ways [62, 63, 142, 143, 154], and generalized to other settings [126, 144, 172].

8.2.2 Randomized Gauss-Seidel (RGS) for $\boldsymbol{X}\boldsymbol{\beta} = \boldsymbol{y}$

The Randomized Gauss-Seidel (RGS) method (or the Randomized Coordinate Descent (RCD) method) selects columns rather than rows in each iteration. For a selected coordinate j, RGS attempts to minimize the objective function $L(\boldsymbol{\beta}) = \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2$ with respect to coordinate j in that iteration. It can thus similarly be defined by the following update rule:

$$\boldsymbol{\beta}_{t+1} := \boldsymbol{\beta}_t + \frac{\boldsymbol{X}_{(j)}^*(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}_t)}{\|\boldsymbol{X}_{(j)}\|_2^2}\boldsymbol{e}_{(j)} \tag{8.4}$$

where $\boldsymbol{e}_{(j)}$ is the jth coordinate basis column vector (all zeros with a 1 in the jth position).

Leventhal and Lewis [126] showed that the residuals of RGS converge again at a linear rate,

$$\|\boldsymbol{X}\boldsymbol{\beta}_t - \boldsymbol{X}\boldsymbol{\beta}^\star\|_2^2 \leq \left(1 - \frac{\sigma_{\min}^2(\boldsymbol{X})}{\|\boldsymbol{X}\|_F^2}\right)^t \|\boldsymbol{X}\boldsymbol{\beta}_0 - \boldsymbol{X}\boldsymbol{\beta}^\star\|_2^2. \tag{8.5}$$

Of course when $m > n$ and the system is full-rank, this convergence also implies convergence of the iterates $\boldsymbol{\beta}_t$ to the solution $\boldsymbol{\beta}^\star$. Connections between the analysis and performance of RK and RGS were recently studied in [134], which also analyzed extended variants to the Kacmarz [235] and Gauss-Siedel method [134] which always converge to the least-squares solution in both the under and overdetermined cases. Analysis of RGS usually applies more generally than our OLS problem, see e.g. Nesterov [147] or Richtárik and Takáč [172] for further details. Also, see [134] for a unified viewpoint and analysis of RK and RGS.

8.3 Suboptimal RK/RGS Algorithms for Ridge Regression

It is well known that the solution to

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^n} \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|^2 + \lambda\|\boldsymbol{\beta}\|^2 \tag{8.6}$$

can be given in two equivalent forms (using the covariance and gram matrices) as

$$\begin{aligned} \boldsymbol{\beta}_{RR} &= (\boldsymbol{X}^*\boldsymbol{X} + \lambda\boldsymbol{I}_n)^{-1}\boldsymbol{X}^*\boldsymbol{y} && (8.7)\\ \text{(and also)} &= \boldsymbol{X}^*\boldsymbol{\alpha}_{RR} && (8.8)\\ \text{where} \quad \boldsymbol{\alpha}_{RR} &= (\boldsymbol{X}\boldsymbol{X}^* + \lambda\boldsymbol{I}_m)^{-1}\boldsymbol{y}. && (8.9) \end{aligned}$$

The presented algorithms are of computational interest because they completely avoid inverting, storing or even forming $\boldsymbol{X}\boldsymbol{X}^*$ and $\boldsymbol{X}^*\boldsymbol{X}$. One can view $\boldsymbol{\beta}_{RR}$ and $\boldsymbol{\alpha}_{RR}$ simply as solutions to two the two linear systems

$$(\boldsymbol{X}^*\boldsymbol{X} + \lambda \boldsymbol{I}_n)\boldsymbol{\beta} = \boldsymbol{X}^*\boldsymbol{y}$$

and

$$(\boldsymbol{X}\boldsymbol{X}^* + \lambda \boldsymbol{I}_m)\boldsymbol{\alpha} = \boldsymbol{y}.$$

If we naively use RK or RGS on either of these systems (treating them as solving $\mathbf{Ax} = \mathbf{b}$ for some given $\boldsymbol{A}$ and $\mathbf{b}$), then we may apply the bounds (8.3) and (8.5) to the matrix $\boldsymbol{X}^*\boldsymbol{X} + \lambda\boldsymbol{I}_n$ or $\boldsymbol{X}\boldsymbol{X}^* + \lambda\boldsymbol{I}_m$. This, however, yields a bound on the convergence rate which depends on the *squared* scaled condition number of $\boldsymbol{X}^*\boldsymbol{X} + \lambda\boldsymbol{I}$, which is approximately the *fourth power* of the scaled condition number of $\boldsymbol{X}$. This dependence is suboptimal, so much so that it becomes highly impractical to solve large scale problems using these methods. This is of course not surprising since this naive solution does not utilize any structure of the ridge regression problem. One thus searches for more tailored approaches. Later, we will propose updates whose computation is still only $O(n)$ or $O(m)$ and yield linear convergence with desired properties; specifically they depend only on the scaled condition number of $\boldsymbol{X}^*\boldsymbol{X} + \lambda\boldsymbol{I}_n$ or $\boldsymbol{X}\boldsymbol{X}^* + \lambda\boldsymbol{I}_m$, and not their square.

The aforementioned updates and their convergence rates are motivated by a clear understanding of how RK and RGS methods relate to each other as in [134] and jointly to positive semi-definite systems of equations.

8.3.1 Ivanov and Zhdanov's Approach

We first consider the regularized normal equations of the system (8.1), as demonstrated in [108, 234]. Here, the authors recognize that the solution to the system (8.1) can be given by

$$\begin{pmatrix} \sqrt{\lambda}\boldsymbol{I}_m & \boldsymbol{X} \\ \boldsymbol{X}^* & -\sqrt{\lambda}\boldsymbol{I}_n \end{pmatrix} \begin{pmatrix} \boldsymbol{\alpha}' \\ \boldsymbol{\beta} \end{pmatrix} = \begin{pmatrix} \boldsymbol{y} \\ \mathbf{0_n} \end{pmatrix}.$$

Here we use $\boldsymbol{\alpha}'$ to differentiate this variable from $\boldsymbol{\alpha}$, which is traditionally defined as the variable involved in the "dual" system $(\boldsymbol{K} + \lambda\boldsymbol{I}_m)\boldsymbol{\alpha} = y$ (though $\boldsymbol{\alpha}'$ and $\boldsymbol{\alpha}$ are related by a constant factor $\sqrt{\lambda}$). The authors propose to solve the system (8.1) by applying the Kaczmarz algorithm (and in the experiments, RK) to the aforementioned system. As they mention, the advantage of rewriting it in this fashion is that the condition number of the $(m + n) \times (m + n)$ matrix

$$\mathbf{A} := \begin{pmatrix} \sqrt{\lambda}\boldsymbol{I}_m & \boldsymbol{X} \\ \boldsymbol{X}^* & -\sqrt{\lambda}\boldsymbol{I}_n \end{pmatrix}$$

is the square-root of the condition number of the $n \times n$ matrix $\boldsymbol{X}^*\boldsymbol{X} + \lambda\boldsymbol{I}_n$. Hence, the RK algorithm on the aforementioned system converges an order of magnitude faster than running RK on (8.1) using the matrix $\boldsymbol{X}^*\boldsymbol{X} + \lambda\boldsymbol{I}_n$.

Let us look at what the algorithm does in more detail. The two sets of equations are:

$$\sqrt{\lambda}\boldsymbol{\alpha}' + \boldsymbol{X}\boldsymbol{\beta} = \boldsymbol{y} \quad \text{and} \quad \boldsymbol{X}^*\boldsymbol{\alpha}' = \sqrt{\lambda}\boldsymbol{\beta}. \tag{8.10}$$

First note that the first m rows of $\boldsymbol{A}$ correspond to rows of $\boldsymbol{X}$ and have a norm $\|\boldsymbol{X}^i\|^2 + \lambda$ and the next n rows of $\boldsymbol{A}$ correspond to columns of $\boldsymbol{X}$ and have a norm $\|\boldsymbol{X}_{(j)}\|^2 + \lambda$. Hence, $\|\boldsymbol{A}\|_F^2 = 2\|\boldsymbol{X}\|_F^2 + (m + n)\lambda$. Hence, one can think of picking a random row of the $(m+n) \times (m+n)$ matrix $\boldsymbol{A}$ (with probability proportional to its row norm, as done by RK) as a two step process. We first choose between doing "row updates" or "column updates" using $\mathbf{X}$ (choosing to do a row update with probability $\frac{\|\boldsymbol{X}\|_F^2+m\lambda}{2\|\boldsymbol{X}\|_F^2+(m+n)\lambda}$ and a column update otherwise). If we had chosen to do row updates, we then choose a random row of $\mathbf{X}$ (with probability proportional to $\frac{\|\boldsymbol{X}^i\|^2+\lambda}{\|\boldsymbol{X}\|_F^2+m\lambda}$ as done by RK). If we had chosen to do column updates, we then choose a random column of $\mathbf{X}$ (with probability proportional to $\frac{\|\boldsymbol{X}_{(j)}\|^2+\lambda}{\|\boldsymbol{X}\|_F^2+n\lambda}$ as done by RGS).

If one selects a random row $i \leq m$ with probability proportional to $\|\boldsymbol{X}^i\|^2 + \lambda$, the equation we greedily satisfy is

$$\sqrt{\lambda}\boldsymbol{e}^*_{(i)}\boldsymbol{\alpha}' + \boldsymbol{X}^i\boldsymbol{\beta} = y^i$$

using the update

$$(\boldsymbol{\alpha}'_{t+1}, \boldsymbol{\beta}_{t+1}) = (\boldsymbol{\alpha}'_t, \boldsymbol{\beta}_t) + \frac{y^i - \sqrt{\lambda}\boldsymbol{e}^*_{(i)}\boldsymbol{\alpha}'_t - \boldsymbol{X}^i\boldsymbol{\beta}_t}{\|\boldsymbol{X}^i\|^2 + \lambda}(\sqrt{\lambda}\boldsymbol{e}_{(i)}, \boldsymbol{X}^i), \tag{8.11}$$

which can be computed in $O(m+n)$ time. Similarly, if a random column $j \leq n$ is selected with probability proportional to $\|\boldsymbol{X}_{(j)}\|^2 + \lambda$, the equation we greedily satisfy is

$$\boldsymbol{X}^*_{(j)}\boldsymbol{\alpha}' = \sqrt{\lambda}\boldsymbol{e}^*_{(j)}\boldsymbol{\beta}$$

with the update in $O(m+n)$ time of

$$(\boldsymbol{\alpha}'_{t+1}, \boldsymbol{\beta}_{t+1}) = (\boldsymbol{\alpha}'_t, \boldsymbol{\beta}_t) + \frac{\sqrt{\lambda}\boldsymbol{e}^*_{(j)}\boldsymbol{\beta}_t - \boldsymbol{X}^*_{(j)}\boldsymbol{\alpha}'_t}{\|\boldsymbol{X}_{(j)}\|^2 + \lambda}(\boldsymbol{X}^*_{(j)}, -\sqrt{\lambda}\boldsymbol{e}_{(j)}). \tag{8.12}$$

Next, we further study the behavior of this method, which the authors called the *augmented projection method.*

8.3.2 The behavior of the augmented projection Approach

Ivanov and Zhandov's approach attempts to find $\boldsymbol{\alpha}'$ and $\boldsymbol{\beta}$ that satisfy conditions (8.10). It is insightful to examine the behavior of that approach when one of these conditions is already satisfied.

Claim 1. *Assume $\boldsymbol{\alpha}'_0$ and $\boldsymbol{\beta}_0$ are initialized such that*

$$\boldsymbol{\beta}_0 = \frac{\boldsymbol{X}^*\boldsymbol{\alpha}'_0}{\sqrt{\lambda}}.$$

(for example, all zeros). Then:

1. *The update equation* (8.11) *is an RK-style update on $\boldsymbol{\alpha}$.*
2. *The condition $\boldsymbol{\beta}_t = \frac{\boldsymbol{X}^*\boldsymbol{\alpha}'_t}{\sqrt{\lambda}}$ is automatically maintained for all t.*
3. *Update equation* (8.12) *has absolutely no effect.*

Proof: Suppose at some iteration $\boldsymbol{\beta}_t = \frac{\boldsymbol{X}^*\boldsymbol{\alpha}'_t}{\sqrt{\lambda}}$ holds. Then assuming we do a row update, substituting this into (8.11) gives, for the ith variable being updated,

$$\alpha'^{(i)}_{t+1} = \alpha'^{(i)}_t + \frac{y^i\sqrt{\lambda} - \lambda\alpha^{(i)}_t - \boldsymbol{X}^i\boldsymbol{X}^*\boldsymbol{\alpha}'}{\|\boldsymbol{X}^i\|^2 + \lambda} = \frac{\|\boldsymbol{X}^i\|^2}{\|\boldsymbol{X}^i\|^2 + \lambda}\alpha'^{(i)}_t + \frac{y^i\sqrt{\lambda} - \boldsymbol{X}^i\boldsymbol{X}^*\boldsymbol{\alpha}'}{\|\boldsymbol{X}^i\|^2 + \lambda},$$

which (as we will later see in more detail) can be viewed as an RK-style update on $\boldsymbol{\alpha}$. The parallel update to $\boldsymbol{\beta}$ can then be rewritten as

$$\boldsymbol{\beta}_{t+1} = \boldsymbol{\beta}_t + \frac{\alpha^i_{t+1} - \alpha^i_t}{\sqrt{\lambda}} \boldsymbol{X}^i,$$

which automatically keeps condition $\boldsymbol{\beta} = \frac{\boldsymbol{X}^* \boldsymbol{\alpha}'}{\sqrt{\lambda}}$ satisfied! Since this condition is already satisfied, if we then run any column update from (8.12) we get

$$(\boldsymbol{\alpha}'_{t+1}, \boldsymbol{\beta}_{t+1}) = (\boldsymbol{\alpha}'_t, \boldsymbol{\beta}_t).$$

□

Claim 2. *Assume $\boldsymbol{\alpha}'_0$ and $\boldsymbol{\beta}_0$ are initialized such that*

$$\boldsymbol{\alpha}'_0 = \frac{\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}_0}{\sqrt{\lambda}}.$$

(for example, $\boldsymbol{\beta}_0$ is zero, $\boldsymbol{\alpha}'_0 = \boldsymbol{y}/\sqrt{\lambda}$). Then:

1. *The update equation* (8.12) *is an RGS-style update on $\boldsymbol{\beta}$.*
2. *The condition $\boldsymbol{\alpha}'_t = \frac{\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}_t}{\sqrt{\lambda}}$ is automatically maintained for all t.*
3. *Update equation* (8.11) *has absolutely no effect.*

Proof: Suppose at some iteration $\boldsymbol{\alpha}'_t = \frac{\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}_t}{\sqrt{\lambda}}$ holds. Then assuming we do a column update, substituting this in (8.12) gives, for the jth variable being updated

$$\beta^j_{t+1} = \beta^j_t + \frac{\boldsymbol{X}^*_{(j)}(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}_t) - \lambda \beta^j_t}{\|\boldsymbol{X}_{(j)}\|^2 + \lambda},$$

which (as we will later see in more detail) is an RGS-style update. The parallel update on $\boldsymbol{\alpha}'$ can then be rewritten as

$$\boldsymbol{\alpha}'_{t+1} = \boldsymbol{\alpha}'_t - \frac{\beta^j_{t+1} - \beta^j_t}{\sqrt{\lambda}} \boldsymbol{X}^*_{(j)},$$

which automatically keeps the condition $\boldsymbol{\alpha}' = \frac{\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}}{\sqrt{\lambda}}$ satisfied! Since this condition is already satisfied, if we then run any row update from (8.11) we get

$$(\boldsymbol{\alpha}'_{t+1}, \boldsymbol{\beta}_{t+1}) = (\boldsymbol{\alpha}'_t, \boldsymbol{\beta}_t).$$

□

In summary, Ivanov and Zhandov's approach effectively executes RK-style updates as well as RGS updates. We can think of update (8.11) (resp. (8.12)) as attempting to satisfy the first (resp. second) condition of (8.10) while maintaining the status of the other condition. If one of the two conditions is already satisfied at the start of the algorithm, then the corresponding update will have no effect. This implies that under typical initial conditions (e.g. $\boldsymbol{\alpha}' = 0, \boldsymbol{\beta} = 0$), this approach is prone to executing many iterations that make absolutely no progress towards convergence! We will see later in Section 8.5 how this behavior affects empirical convergence as well.

8.4 Our Proposed Approach

Both RK and RGS can be viewed in the following fashion. Suppose we have a positive definite matrix $\boldsymbol{A}$, and we want to solve $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$. Instead of casting it as $\min_{\boldsymbol{x}} \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|^2$ which involves $\boldsymbol{A}^T\boldsymbol{A}$ and squares its condition number, we can alternatively pose the different problem $\min_{\boldsymbol{x}} \frac{1}{2}\boldsymbol{x}^*\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}^*\boldsymbol{x}$. Then one could use the update

$$\boldsymbol{x}_{t+1} = \boldsymbol{x}_t + \frac{b_i - \boldsymbol{A}^i\boldsymbol{x}_t}{A_{ii}}\boldsymbol{e}_{(i)},$$

where $b_i - \boldsymbol{A}^i\boldsymbol{x}_t$ is basically the i-th coordinate of the gradient, and A_{ii} is the Lipschitz constant of the i-th coordinate of the gradient (see related works e.g. Leventhal and Lewis [126], Nesterov [149], Richtárik and Takáč [172], Lee and Sidford [123]).

In this light, the original RK update in (8.2) can be seen as the randomized coordinate descent rule for the positive semidefinite system $\boldsymbol{X}\boldsymbol{X}^*\boldsymbol{\alpha} = \boldsymbol{y}$ (substituting $\boldsymbol{\beta} = \boldsymbol{X}^*\boldsymbol{\alpha}$) and treating $\boldsymbol{A} = \boldsymbol{X}\boldsymbol{X}^*$ and $\boldsymbol{b} = \boldsymbol{y}$. Similarly, the RGS update in (8.4) can be seen as the randomized coordinate descent rule for the positive semidefinite system $\boldsymbol{X}^*\boldsymbol{X}\boldsymbol{\beta} = \boldsymbol{X}^*\boldsymbol{y}$ and treating $\boldsymbol{A} = \boldsymbol{X}^*\boldsymbol{X}$ and $\boldsymbol{b} = \boldsymbol{X}^*\boldsymbol{y}$.

Using this connection, we propose the following update rule:

$$\begin{aligned}
\delta_t &= \frac{y^i - \boldsymbol{\beta}_t^*\boldsymbol{X}^i - \lambda\alpha_t^i}{\|\boldsymbol{X}^i\|^2 + \lambda} && (8.13)\\
\alpha_{t+1}^i &= \alpha_t^i + \delta_t && (8.14)\\
\boldsymbol{\beta}_{t+1} &= \boldsymbol{\beta}_t + \delta_t\boldsymbol{X}^i, && (8.15)
\end{aligned}$$

where the ith row is selected with probability proportional to $\|\boldsymbol{X}^i\|^2 + \lambda$. If all rows are normalized, this is a uniform distribution. However, it is more typical to normalize the columns in statistics, and hence one pass over the data must be made to calculate row norms (see e.g. [144] for other alternatives in the general setting). The update for $\boldsymbol{\alpha}$ can be rewritten in the form

$$\begin{aligned}
\alpha_{t+1}^i &= \frac{K_{ii}}{K_{ii} + \lambda}\alpha_t^i + \frac{y_i - \sum_j K_{ij}\alpha_t^j}{K_{ii} + \lambda} && (8.16)\\
&= S_{\frac{\lambda}{K_{ii}}}\left(\alpha_t^i + \frac{r_i}{K_{ii}}\right) && (8.17)
\end{aligned}$$

where $\boldsymbol{K} = \boldsymbol{X}\boldsymbol{X}^*$, $S_a(z) = \frac{z}{1+a}$ and $r_i = y_i - \sum_j K_{ij}\alpha_t^j$ is the ith residual and row i is picked proportional to $K_{ii} + \lambda$.

Let us contrast this with the randomized coordinate descent update rule for the loss function $\min_x \frac{1}{2}\boldsymbol{\beta}^*(\boldsymbol{X}^*\boldsymbol{X} + \lambda\boldsymbol{I}_m)\boldsymbol{\beta} - \boldsymbol{y}^*\boldsymbol{X}\boldsymbol{\beta}$, i.e. the system $(\boldsymbol{X}^*\boldsymbol{X} + \lambda\boldsymbol{I}_n)\boldsymbol{\beta} = \boldsymbol{X}^*\boldsymbol{y}$. In this case we instead have, calling $\mathbf{r}_t = \boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}_t$

$$\begin{aligned}
\beta_{t+1}^j &= \beta_t^j + \frac{\boldsymbol{X}_{(j)}^*\boldsymbol{y} - \boldsymbol{X}_{(j)}^*\boldsymbol{X}\boldsymbol{\beta}_t - \lambda\beta_t^j}{\|\boldsymbol{X}_{(j)}\|^2 + \lambda} && (8.18)\\
&= \frac{\|\boldsymbol{X}_{(j)}\|^2}{\|\boldsymbol{X}_{(j)}\|^2 + \lambda}\beta_t^j + \frac{\boldsymbol{X}_{(j)}^* r_t}{\|\boldsymbol{X}_{(j)}\|^2 + \lambda} && (8.19)\\
&= S_{\frac{\lambda}{\|\boldsymbol{X}_{(j)}\|^2}}\left(\beta_t^j + \frac{\boldsymbol{X}_{(j)}^* r_t}{\|\boldsymbol{X}_{(j)}\|^2}\right). && (8.20)
\end{aligned}$$

Next, we analyze the behavior of these approaches, the first one being referred to as RK updates and the second being referred to as RGS updates.

8.4.1 Computation and Convergence

The RGS updates in (8.18)-(8.20) take $O(m)$ time, since each column (feature) is of size m. In contrast, the proposed RK updates in (8.13)-(8.15) take $O(n)$ time since that is the length of a data point.

While the RK and RGS algorithms are similar and related, one should not be tempted into thinking their convergence rates are the same. Indeed, with no normalization assumption, using a similar style proof as presented in [134], one can analyze the convergence rates in parallel as follows. Let us denote

$$\boldsymbol{\Sigma}' := \boldsymbol{X}^*\boldsymbol{X} + \lambda \boldsymbol{I}_n$$

and

$$\boldsymbol{K}' := \boldsymbol{X}\boldsymbol{X}^* + \lambda \boldsymbol{I}_m$$

for brevity, and let $\sigma_1, \sigma_2, ...$ be the singular values of $\boldsymbol{X}$ in increasing order.

RK: $\mathbb{E}\|\boldsymbol{\alpha}_{t+1} - \boldsymbol{\alpha}^*\|^2_{\boldsymbol{K}'}$	RGS: $\mathbb{E}\|\boldsymbol{\beta}_{t+1} - \boldsymbol{\beta}^*\|^2_{\boldsymbol{\Sigma}'}$
$= \mathbb{E}\left(\|\boldsymbol{\alpha}_t - \boldsymbol{\alpha}^*\|^2_{\boldsymbol{K}'} - \|\boldsymbol{\alpha}^{t+1} - \boldsymbol{\alpha}_t\|^2_{\boldsymbol{K}'}\right)$	$= \mathbb{E}\left(\|\boldsymbol{\beta}_t - \boldsymbol{\beta}^*\|^2_{\boldsymbol{\Sigma}'} - \|\boldsymbol{\beta}^{t+1} - \boldsymbol{\beta}_t\|^2_{\boldsymbol{\Sigma}'}\right)$
$= \|\boldsymbol{\alpha}_t - \boldsymbol{\alpha}^*\|^2_{\boldsymbol{K}'} - \sum_i \frac{\boldsymbol{K}_{ii}+\lambda}{Tr(\boldsymbol{K})+m\lambda} \frac{y_i - \sum_j \boldsymbol{K}_{ij}\alpha_t^j - \lambda\alpha_t^i}{\boldsymbol{K}_{ii}+\lambda}$	$= \|\boldsymbol{\beta}_t - \boldsymbol{\beta}^*\|^2_{\boldsymbol{\Sigma}'} - \sum_j \frac{\|\boldsymbol{X}_{(j)}\|^2+\lambda}{\|\boldsymbol{X}\|^2_F+n\lambda} \frac{(\boldsymbol{X}^*_{(j)}(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta}_t)-\lambda\beta_t^j)^2}{\|\boldsymbol{X}_{(j)}\|^2+\lambda}$
$= \|\boldsymbol{\alpha}_t - \boldsymbol{\alpha}^*\|^2_{\boldsymbol{K}'} - \frac{\|\boldsymbol{y}-(\boldsymbol{K}')\boldsymbol{\alpha}_t)\|^2}{Tr(\boldsymbol{K})+m\lambda}$	$= \|\boldsymbol{\beta}_t - \boldsymbol{\beta}^*\|^2_{\boldsymbol{\Sigma}'} - \frac{\|\boldsymbol{X}^*\boldsymbol{y}-(\boldsymbol{\Sigma}')\boldsymbol{\beta}_t\|^2}{\|\boldsymbol{X}\|^2_F+n\lambda}$
$= \|\boldsymbol{\alpha}_t - \boldsymbol{\alpha}^*\|^2_{\boldsymbol{K}'} - \frac{\|(\boldsymbol{K}')(\boldsymbol{\alpha}^*-\boldsymbol{\alpha}_t)\|^2}{Tr(\boldsymbol{K})+m\lambda}$	$= \|\boldsymbol{\beta}_t - \boldsymbol{\beta}^*\|^2_{\boldsymbol{\Sigma}'} - \frac{\|(\boldsymbol{\Sigma}')(\boldsymbol{\beta}^*-\boldsymbol{\beta}_t)\|^2}{\|\boldsymbol{X}\|^2_F+n\lambda}$
$\leq \|\boldsymbol{\alpha}_t - \boldsymbol{\alpha}^*\|^2_{\boldsymbol{K}'} - \frac{\sigma_{\min}(\boldsymbol{K}')\|\boldsymbol{\alpha}^*-\boldsymbol{\alpha}_t\|^2_{\boldsymbol{K}'}}{Tr(\boldsymbol{K}')}$	$\leq \|\boldsymbol{\beta}_t - \boldsymbol{\beta}^*\|^2_{\boldsymbol{\Sigma}'} - \frac{\sigma_{\min}(\boldsymbol{\Sigma}')\|\boldsymbol{\beta}^*-\boldsymbol{\beta}_t\|^2_{\boldsymbol{\Sigma}'}}{Tr(\boldsymbol{\Sigma}')}$
$= \begin{cases} \left(1 - \frac{\lambda}{\sum_i \sigma_i^2+m\lambda}\right)^t \|\boldsymbol{\alpha}_t - \boldsymbol{\alpha}^*\|^2_{\boldsymbol{K}'} \text{ if } m > n \\ \left(1 - \frac{\sigma_1^2+\lambda}{\sum_i \sigma_i^2+m\lambda}\right)^t \|\boldsymbol{\alpha}_t - \boldsymbol{\alpha}^*\|^2_{\boldsymbol{K}'} \text{ if } n > m \end{cases}$	$= \begin{cases} \left(1 - \frac{\sigma_1^2+\lambda}{\sum_i \sigma_i^2+n\lambda}\right)^t \|\boldsymbol{\beta}_t - \boldsymbol{\beta}^*\|^2_{\boldsymbol{\Sigma}'} \text{ if } m > n \\ \left(1 - \frac{\lambda}{\sum_i \sigma_i^2+n\lambda}\right)^t \|\boldsymbol{\beta}_t - \boldsymbol{\beta}^*\|^2_{\boldsymbol{\Sigma}'} \text{ if } n > m \end{cases}$

Applying these bounds recursively, we obtain the following convergence guarantee for RK,

$$\mathbb{E}\|\boldsymbol{\alpha}_t - \boldsymbol{\alpha}^*\|^2_{\boldsymbol{K}+\lambda\boldsymbol{I}_n} \quad \leq \quad \begin{cases} \left(1 - \frac{\lambda}{\sum_i \sigma_i^2+m\lambda}\right)^t \|\boldsymbol{\alpha}_t - \boldsymbol{\alpha}^*\|^2_{\boldsymbol{K}+\lambda\boldsymbol{I}_m} & \text{if } m > n \\ \left(1 - \frac{\sigma_1^2+\lambda}{\sum_i \sigma_i^2+m\lambda}\right)^t \|\boldsymbol{\alpha}_t - \boldsymbol{\alpha}^*\|^2_{\boldsymbol{K}+\lambda\boldsymbol{I}_m} & \text{if } n > m. \end{cases} \tag{8.21}$$

The rate of convergence for RGS for Ridge Regression is subtly different,

$$\mathbb{E}\|\boldsymbol{\beta}_t - \boldsymbol{\beta}^*\|^2_{\boldsymbol{X}^*\boldsymbol{X}+\lambda\boldsymbol{I}_n} \quad \leq \quad \begin{cases} \left(1 - \frac{\sigma_1^2+\lambda}{\sum_i \sigma_i^2+n\lambda}\right)^t \|\boldsymbol{\beta}_t - \boldsymbol{\beta}^*\|^2_{\boldsymbol{X}^*\boldsymbol{X}+\lambda\boldsymbol{I}_n} & \text{if } m > n \\ \left(1 - \frac{\lambda}{\sum_i \sigma_i^2+n\lambda}\right)^t \|\boldsymbol{\beta}_t - \boldsymbol{\beta}^*\|^2_{\boldsymbol{X}^*\boldsymbol{X}+\lambda\boldsymbol{I}_n} & \text{if } n > m. \end{cases} \tag{8.22}$$

We see immediately by these bounds that the RGS method is preferable in the overdetermined case while RK is preferable in the underdetermined case. Substituting appropriately into (8.3), we get very similar convergence rates to the above for the Ivanov-Zhdanov algorithm, except that it bounds the quantity $\|\boldsymbol{\alpha}'^T - \boldsymbol{\alpha}'^*\|^2 + \|\boldsymbol{\beta}^T - \boldsymbol{\beta}^*\|^2$. However, we have already argued that these updates are suboptimal, since a large proportion of updates to not perform any action, as we shall once more verify in the experimental section.

To summarize, our final proposal for solving such systems as as follows : when $m > n$ use RGS, and when $m < n$ use RK, and never use IZ.

8.5 Empirical Results

We next present simulation experiments to test the performance of RK, RGS and IZ (Ivanov and Zhdanov's) algorithms in different settings of ridge regression. For given dimensions m and n, We generate a design matrix $\boldsymbol{X} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^\top$, where $\boldsymbol{U} \in \mathbb{R}^{m\times k}$, $\boldsymbol{V} \in \mathbb{R}^{n\times k}$, and $k = \min(m,n)$. Elements of $\boldsymbol{U}$ and $\boldsymbol{V}$ are generated from a standard Gaussian distribution and then columns are orthonormalized. The matrix $\boldsymbol{S}$ is a diagonal $k \times k$ matrix of singular values of $\boldsymbol{X}$. The maximum singular value is 1.0 and the values decay exponentially to $\sigma_{\min}$. The true parameter vector $\boldsymbol{\beta}$ is generated from a multivariate Gaussian distribution with zero mean and identity covariance. The vector $\boldsymbol{y}$ is generated by adding independent standard Gaussian noise to the coordinates of $\boldsymbol{X}\boldsymbol{\beta}$. We used different values of m, n, λ and σ_{min} as listed in Table 8.1. For each configuration of the simulated parameters, we run RGS and RK and IZ for 10^4 iterations on a random instance of that configuration and report the Euclidean difference between estimated and optimal parameters after each 100 iterations. We used several different initializations for the IZ algorithm as shown in Table 8.2.

The results are reported in Figures 8.1, 8.2 and 8.3. Figure 8.1 shows that RGS and RK exhibit similar behavior when $m = n$. Poor conditioning of the design matrix results in slower convergence. However, the effect of conditioning is most apparent when the regularization parameter is small. Figures 8.2 and 8.3 show that RGS consistently outperforms other methods when $m > n$ while RK consistently outperforms other methods when $m < n$. The difference is again most apparent when the regularization parameter is small. We also notice that IZ0 (resp. IZ1) exhibit similar convergence behavior as that of RK (resp. RGS) although typically slower. This agrees with our analysis which reveals that, depending on the initialization, IZ can perform RGS or RK-style updates except that some iterations can be ineffective, which causes slower convergence. Interestingly, IZMIX, where α is initialized midway between IZ0 and IZ1 exhibits convergence behavior that is in between IZ0 and IZ1.

Parameter	Definition	Values
(m,n)	Dimensions of the design matrix $\boldsymbol{X}$	$(1000, 1000)$, $(10^4, 100)$, $(100, 10^4)$
λ	Regularization parameter	$10^{-3}, 1.0, 10.0$
σ_{min}	Minimum singular value of the design matrix	$1.0, 0.1, 10^{-3}, 10^{-5}$

Table 8.1: Different parameters used in simulation experiments

Algorithm	Description
RGS	Randomized coordinate descent updates using (8.18) with initialization $\beta_0 = 0$
RK	Randomized Kaczmarz updates using (8.17) with initialization $\alpha_0 = 0$
IZ0	Ivanov and Zhdanov's algorithm with $\boldsymbol{\alpha}_0 = 0, \boldsymbol{\beta}_0 = 0$
IZ1	Ivanov and Zhdanov's algorithm with $\boldsymbol{\alpha}_0 = \boldsymbol{y}/\sqrt{\lambda}, \boldsymbol{\beta}_0 = 0$
IZMIX	Ivanov and Zhdanov's algorithm with $\boldsymbol{\alpha}_0 = \boldsymbol{y}/2\sqrt{\lambda}, \boldsymbol{\beta}_0 = 0$
IZRND	Ivanov and Zhdanov's algorithm with elements of $\boldsymbol{\beta}_0$ and $\boldsymbol{\alpha}_0$ randomly drawn from a standard normal distribution

Table 8.2: List of algorithms compared in simulation experiment.

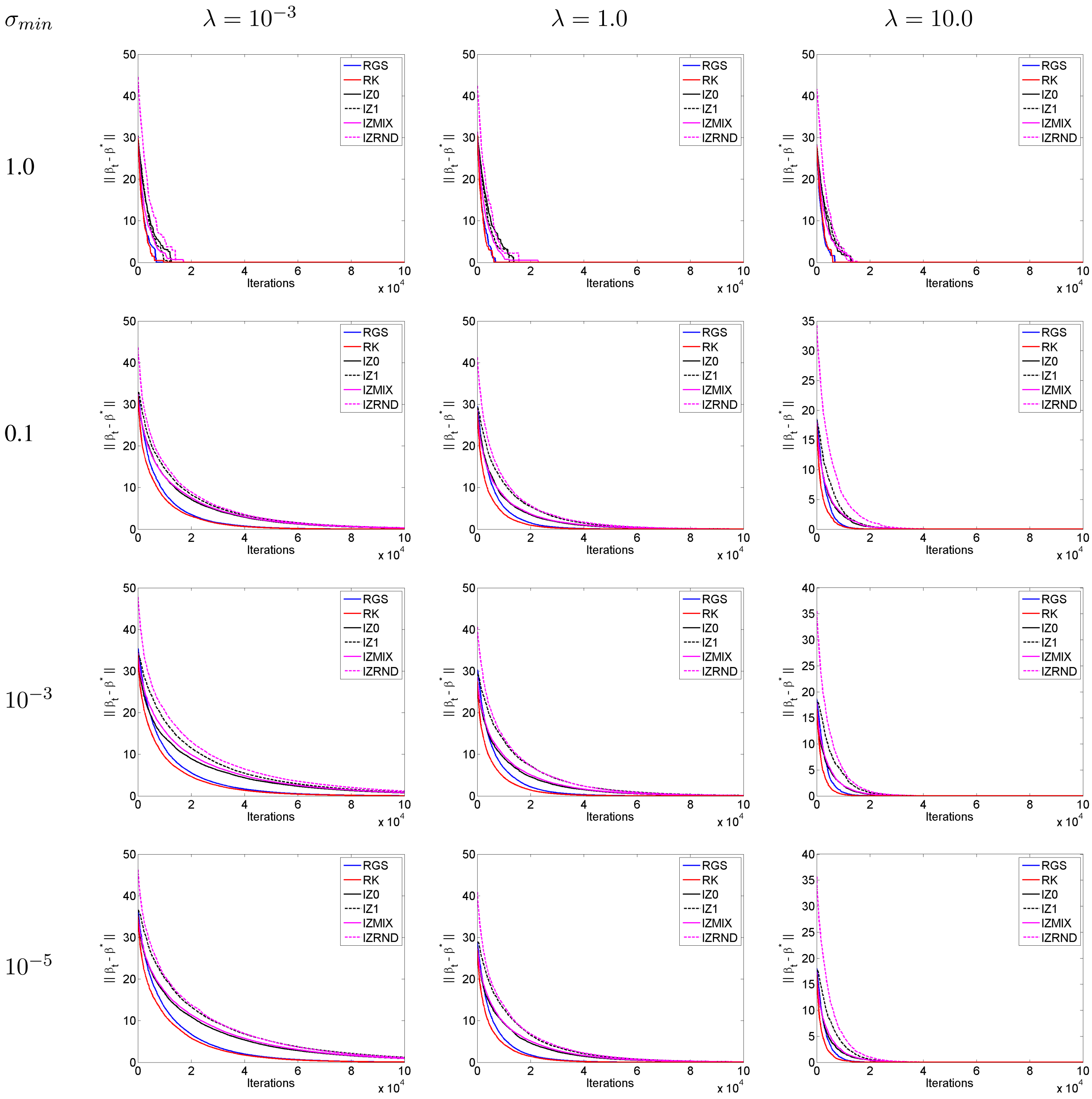


Figure 8.1: Simulation results for $m = n = 1000$: Euclidean error $\|\boldsymbol{\beta}_t - \boldsymbol{\beta}^{\star}\|$ versus iteration count.

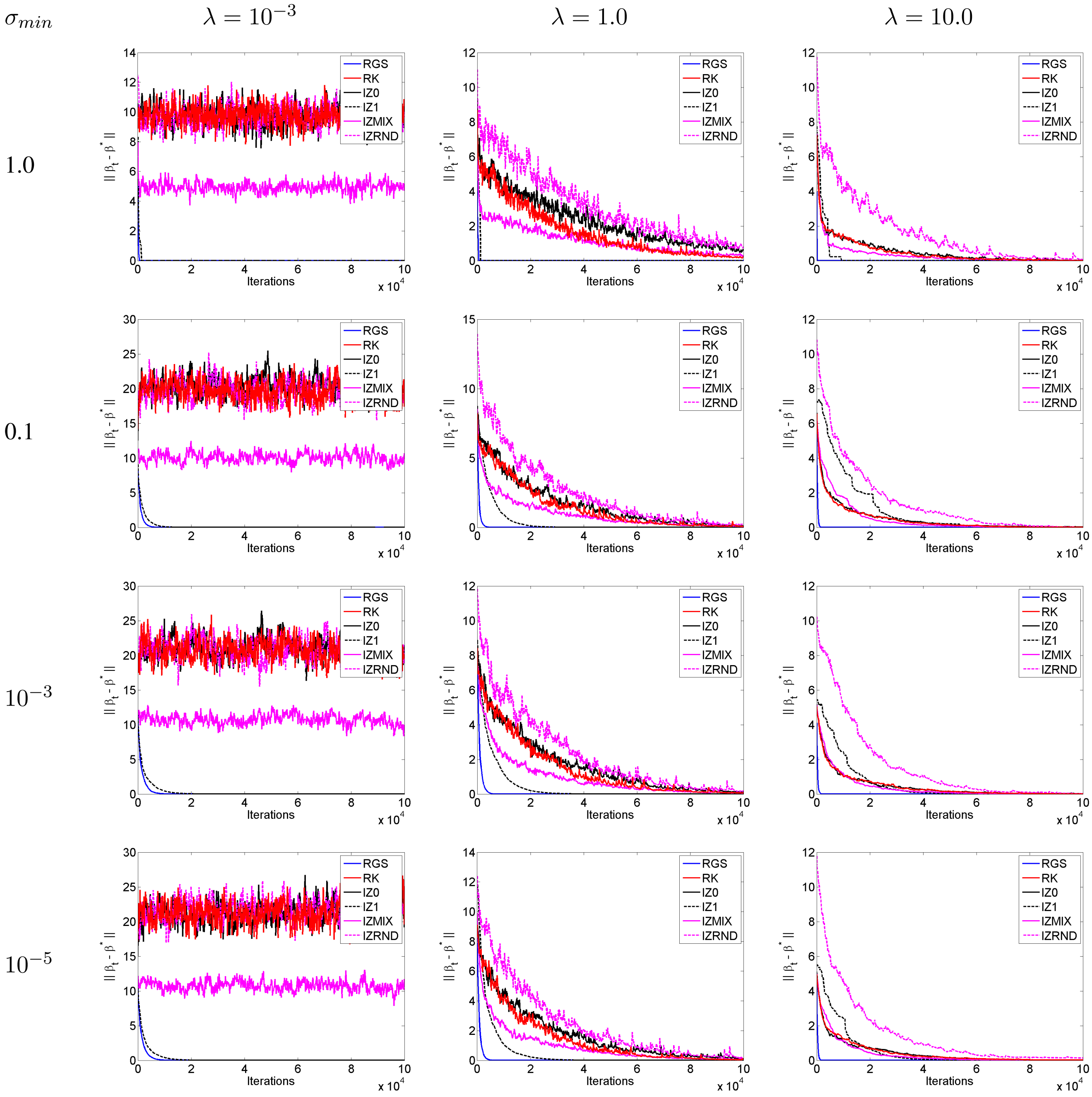


Figure 8.2: Simulation results for $m = 10^4, n = 100$: Euclidean error $\|\boldsymbol{\beta}_t - \boldsymbol{\beta}^\star\|$ versus iteration count.

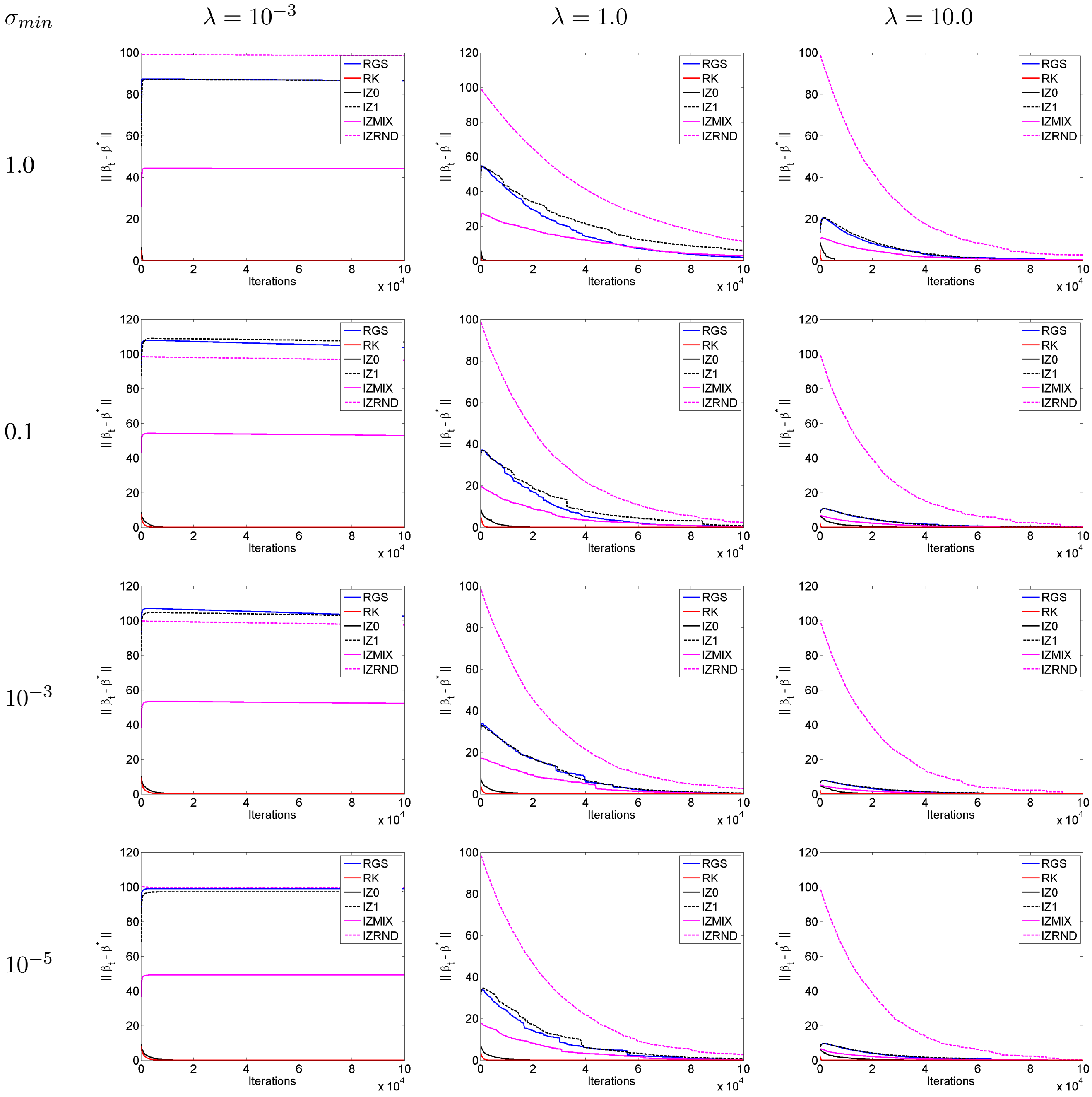


Figure 8.3: Simulation results for $m = 100, n = 10^4$: Euclidean error $\|\boldsymbol{\beta}_t - \boldsymbol{\beta}^{\star}\|$ versus iteration count.

Other Applications

Here we present a few simple settings where the above algorithms can be useful.

Kernel Ridge Regression

If $\mathcal{H}_k$ is a Reproducing Kernel Hilbert Space (RKHS, see e.g. Schölkopf and Smola [184] for an introduction) associated to positive definite kernel k and feature map $\boldsymbol{\phi}_x$, it is well known that the solution to the corresponding Kernel Ridge Regression (KRR) [182] problem is

$$\begin{aligned} f_{KRR} &= \arg\min_{\boldsymbol{f}\in\mathcal{H}_k} \sum_{i=1}^{m} (y^i - \boldsymbol{f}(x_i))^2 + \lambda \|\boldsymbol{f}\|_{\mathcal{H}_k}^2 & (8.23) \\ &= \boldsymbol{\Phi}^*(\boldsymbol{K} + \lambda \boldsymbol{I})^{-1}\boldsymbol{y}, & (8.24) \end{aligned}$$

where $\boldsymbol{\Phi} = (\boldsymbol{\phi}_{x_1}, ..., \boldsymbol{\phi}_{x_n})^*$ and $\boldsymbol{K}$ is the gram matrix with $K_{ij} = k(x_i, x_j)$.

One of the main problems with kernel methods is as data size grows, the gram matrix becomes too large to store. This has motivated the study of approximation techniques for such kernel matrices, but we have an alternate suggestion. The aim of a Kaczmarz style algorithm would be to solve the problem by never forming $\boldsymbol{K}$. Indeed, we will provide an update for KRR with cost $O(m)$ per iteration that exhibits linear convergence. Note that here RK for Kernel Ridge Regression costs $O(m)$ per iteration and RK for Ridge Regression cost $O(n)$ per iteration due to different parameterizations. In the latter, we can keep track of $\boldsymbol{\beta}_t$ as well as $\boldsymbol{\alpha}_t$ easily, but for KRR, calculations can only be performed via evaluations of the kernel only ($\boldsymbol{\beta}_t$ corresponds to a function and cannot be stored), and hence have a different cost.

Since one hopes to calculate $f_{KRR} = \boldsymbol{\Phi}^*(\boldsymbol{K} + \lambda \boldsymbol{I}_m)^{-1}\boldsymbol{y}$, the RK-style update is suitable to calculate the solution to the positive semidefinite system

$$(\boldsymbol{K} + \lambda \boldsymbol{I})\boldsymbol{\alpha} = \boldsymbol{y}$$

followed by setting $f_{KRR} = \boldsymbol{\Phi}^*\boldsymbol{\alpha}$.

The RK updates in (8.16) take $O(m)$ time (to update r) not counting time for kernel evaluations. The difference between the two RK updates for Ridge Regression and Kernel Ridge Regression is that for KRR, we cannot maintain $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ since $\boldsymbol{\beta}$ is a function in the RKHS. This different parameterization makes the updates to $\boldsymbol{\alpha}$ cost $O(m)$ instead of $O(n)$.

Conclusion

This work extends the unifying analysis of the randomized Kaczmarz (RK) and randomized Gauss-Seidel (RGS) methods to the setting of ridge regression. By presenting a parallel study of the behavior of these two methods in this setting, comparisons and connections can be made between the approaches as well as other existing approaches. In particular, we demonstrate that the augmented projection approach of Ivanov and Zhdanov exhibits a mix of RK and RGS style updates in such a way that many iterations yield no progress. Motivated by this unifying framework, we present a new approach which eliminates this drawback, and provide an analysis demonstrating that the RGS variant is preferred in the overdetermined case while RK is preferred in the underdetermined case. This extends previous analysis of these types of iterative methods in the classical ordinary least squares setting, which are highly suboptimal if directly applied to the setting of ridge regression.

Chapter 9

Univariate Regression : Fast & Flexible algorithms for trend filtering

This chapter[1] presents a fast and robust algorithm for trend filtering, a recently developed nonparametric regression tool. It has been shown that, for estimating functions whose derivatives are of bounded variation, trend filtering achieves the minimax optimal error rate, while other popular methods like smoothing splines and kernels do not. Standing in the way of a more widespread practical adoption, however, is a lack of scalable and numerically stable algorithms for fitting trend filtering estimates. This chapter presents a highly efficient, specialized ADMM routine for trend filtering. Our algorithm is competitive with the specialized interior point methods that are currently in use, and yet is far more numerically robust. Furthermore, the proposed ADMM implementation is very simple, and importantly, it is flexible enough to extend to many interesting related problems, such as sparse trend filtering and isotonic trend filtering. Software for our method is freely available, in both the C and R languages.

9.1 Introduction

Trend filtering is a relatively new method for nonparametric regression, proposed independently by Kim et al. [115], Steidl et al. [202]. Suppose that we are given output points $y = (y_1, \dots y_n) \in \mathbb{R}^n$, observed across evenly spaced input points $x = (x_1, \dots x_n) \in \mathbb{R}^n$, say, $x_1 = 1, \dots x_n = n$ for simplicity. The trend filtering estimate $\hat{\beta} = (\hat{\beta}_1, \dots \hat{\beta}_n) \in \mathbb{R}^n$ of a specified order $k \geq 0$ is defined as

$$\hat{\beta} = \underset{\beta \in \mathbb{R}^n}{\arg\min} \; \frac{1}{2}\|y - \beta\|_2^2 + \lambda \|D^{(k+1)}\beta\|_1. \tag{9.1}$$

Here $\lambda \geq 0$ is a tuning parameter, and $D^{(k+1)} \in \mathbb{R}^{(n-k)\times n}$ is the discrete difference (or derivative) operator of order $k+1$. We can define these operators recursively as

$$D^{(1)} = \begin{bmatrix} -1 & 1 & 0 & \dots & 0 & 0 \\ 0 & -1 & 1 & \dots & 0 & 0 \\ \vdots & & & & & \\ 0 & 0 & 0 & \dots & -1 & 1 \end{bmatrix}, \tag{9.2}$$

and

$$D^{(k+1)} = D^{(1)} D^{(k)} \quad \text{for } k = 1, 2, 3, \dots. \tag{9.3}$$

[1] See [162].

(Note that, above, we write $D^{(1)}$ to mean the $(n-k-1) \times (n-k)$ version of the 1st order difference matrix in (9.2).) When $k = 0$, we can see from the definition of $D^{(1)}$ in (9.2) that the trend filtering problem (9.1) is the same as the 1-dimensional fused lasso problem [211], also called 1-dimensional total variation denoising [177], and hence the 0th order trend filtering estimate $\hat{\beta}$ is piecewise constant across the input points $x_1, \ldots x_n$.

For a general k, the kth order trend filtering estimate has the structure of a kth order piecewise polynomial function, evaluated across the inputs $x_1, \ldots x_n$. The knots in this piecewise polynomial are selected adaptively among $x_1, \ldots x_n$, with a higher value of the tuning parameter λ (generally) corresponding to fewer knots. To see examples, the reader can jump ahead to the next subsection, or to future sections. For arbitrary input points $x_1, \ldots x_n$ (i.e., these need not be evenly spaced), the defined difference operators will have different nonzero entries, but their structure and the recursive relationship between them is basically the same; see Section 9.4.

Broadly speaking, nonparametric regression is a well-studied field with many celebrated tools, and so one may wonder about the merits of trend filtering in particular. For detailed motivation, we refer the reader to Tibshirani [212], where it is argued that trend filtering essentially balances the strengths of smoothing splines [50, 223] and locally adaptive regression splines [136], which are two of the most common tools for piecewise polynomial estimation. In short: smoothing splines are highly computationally efficient but are not minimax optimal (for estimating functions whose derivatives are of bounded variation); locally adaptive regression splines are minimax optimal but are relatively inefficient in terms of computation; trend filtering is both minimax optimal and computationally comparable to smoothing splines. Tibshirani [212] focuses mainly on the statistical properties trend filtering estimates, and relies on externally derived algorithms for comparisons of computational efficiency.

9.1.1 Overview of contributions

In this chapter, we propose a new algorithm for trend filtering. We do not explicitly address the problem of model selection, i.e., we do not discuss how to choose the tuning parameter λ in (9.1), which is a long-standing statistical issue with any regularized estimation method. Our concern is computational; if a practitioner wants to solve the trend filtering problem (9.1) at a given value of λ (or sequence of values), then we provide a scalable and efficient means of doing so. Of course, a fast algorithm such as the one we provide can still be helpful for model selection, in that it can provide speedups for common techniques like cross-validation.

For 0th order trend filtering, i.e., the 1d fused lasso problem, two direct, linear time algorithms already exist: the first uses a taut string principle [49], and the second uses an entirely different dynamic programming approach [110]. Both are extremely (and equally) fast in practice, and for this special 0th order problem, these two direct algorithms rise above all else in terms of computational efficiency and numerical accuracy. As far as we know (and despite our best attempts), these algorithms cannot be directly extended to the higher order cases $k = 1, 2, 3, \ldots$. However, our proposal *indirectly* extends these formidable algorithms to the higher order cases with a special implementation of the alternating direction method of multipliers (ADMM). In general, there can be multiple ways to reparametrize an unconstrained optimization problem so that ADMM can be applied; for the trend filtering problem (9.1), we choose a particular parametrization suggested by the recursive decomposition (9.3), leveraging the fast, exact algorithms that exist for the $k = 0$ case. We find that this choice makes a big difference in terms of the convergence of the resulting ADMM routine, compared to what may be considered the standard ADMM parametrization for (9.1).

Currently, the specialized primal-dual interior point (PDIP) method of Kim et al. [115] seems to be the preferred method for computing trend filtering estimates. The iterations of this algorithm are cheap because they reduce to solving banded linear systems (the discrete difference operators are themselves banded). Our specialized ADMM implementation and the PDIP method have distinct strengths. We summarize our main findings below.

- Our specialized ADMM implementation converges more reliably than the PDIP method, over a wide range of problems sizes n and tuning parameter values λ.
- In particular setups—namely, small problem sizes, and small values of λ for moderate and large problem sizes—the PDIP method converges to high accuracy solutions very rapidly. In such situations, our specialized ADMM algorithm does not match the convergence rate of this second-order method.
- However, when plotting the function estimates, our specialized ADMM implementation produces solutions of visually perfectly acceptable accuracy after a small number of iterations. This is true over a broad range of problem sizes n and parameter values λ, and covers the settings in which its achieved criterion value has not converged at the rate of the PDIP method.
- Furthermore, our specialized ADMM implementation displays a greatly improved convergence rate over what may be thought of as the "standard" ADMM implementation for problem (9.1). Loosely speaking, standard implementations of ADMM are generally considered to behave like first-order methods [27], whereas our specialized implementation exhibits performance somewhere in between that of a first- and second-order method.
- One iteration of our specialized ADMM implementation has linear complexity in the problem size n; this is also true for PDIP. Empirically, an iteration of our ADMM routine runs about 10 times faster than a PDIP iteration.
- Our specialized ADMM implementation is quite simple (considerably simpler than the specialized primal-dual interior point method), and is flexible enough that it can be extended to cover many variants and extensions of the basic trend filtering problem (9.1), such as sparse trend filtering, mixed trend filtering, and isotonic trend filtering.
- Finally, it is worth remarking that extensions beyond the univariate case are readily available as well, as univariate nonparametric regression tools can be used as building blocks for estimation in broader model classes, e.g., in generalized additive models [94].

Readers well-versed in optimization may wonder about alternative iterative (descent) methods for solving the trend filtering problem (9.1). Two natural candidates that have enjoyed much success in lasso regression problems are proximal gradient and coordinate descent algorithms. Next, we give a motivating case study that illustrates the inferior performance of both of these methods for trend filtering. In short, their performance is heavily affected by poor conditioning of the difference operator $D^{(k+1)}$, and their convergence is many orders of magnitude worse than the specialized primal-dual interior point and ADMM approaches.

9.1.2 A motivating example

Conditioning is a subtle but ever-present issue faced by iterative (indirect) optimization methods. This issue affects some algorithms more than others; e.g., in a classical optimization context, it is well-understood that the convergence bounds for gradient descent depend on the smallest and largest eigenvalues of the Hessian of the criterion function, while those for Newton's method do not (Newton's method being affine invariant). Unfortunately, conditioning is a very real issue when solving the trend filtering problem in (9.1)—the discrete derivative operators $D^{(k+1)}$, $k = 0, 1, 2, \ldots$ are extremely ill-conditioned, and this only worsens as k increases.

This worry can be easily realized in examples, as we now demonstrate in a simple simulation with a reasonable polynomial order, $k = 1$, and a modest problem size, $n = 1000$. For solving the trend filtering problem (9.1), with $\lambda = 1000$, we compare proximal gradient descent and accelerated proximal gradient method (performed on both the primal and the dual problems), coordinate descent, a standard ADMM approach, our specialized ADMM approach, and the specialized PDIP method of Kim et al. [115]. Details of the simulation setup and these various algorithms are given in Appendix 9.6.1, but the main message can be seen from Figure 9.1. Different variants of proximal gradient methods, as well as coordinate descent, and a standard ADMM approach, all perform quite poorly in computing trend filtering estimate, but the second-order PDIP method and our specialized ADMM implementation perform drastically better—these two reach in 20 iterations what the others could not reach in many thousands. Although the latter two techniques perform similarly in this example, we will see later that our specialized ADMM approach generally suffers from far less conditioning and convergence issues than PDIP, especially in regimes of regularization (i.e., ranges of λ values) that are most interesting statistically.

The rest of this chapter is organized as follows. In Section 9.2, we describe our specialized ADMM implementation for trend filtering. In Section 9.3, we make extensive comparisons to PDIP. Section 9.4 covers the case of general input points $x_1, \dots x_n$. Section 9.5 considers several extensions of the basic trend filtering model, and the accompanying adaptions of our specialized ADMM algorithm. Section 9.5 concludes with a short discussion.

9.2 A specialized ADMM algorithm

We describe a specialized ADMM algorithm for trend filtering. This algorithm may appear to only slightly differ in its construction from a more standard ADMM algorithm for trend filtering, and both approaches have virtually the same computational complexity, requiring $O(n)$ operations per iteration; however, as we have glimpsed in Figure 9.1, the difference in convergence between the two is drastic.

The standard ADMM approach (e.g., Boyd et al. [27]) is based on rewriting problem (9.1) as

$$\min_{\beta \in \mathbb{R}^n,\, \alpha \in \mathbb{R}^{n-k-1}} \frac{1}{2}\|y-\beta\|_2^2 + \lambda\|\alpha\|_1 \;\; \text{subject to} \;\; \alpha = D^{(k+1)}\beta. \tag{9.4}$$

The augmented Lagrangian can then be written as

$$L(\beta,\alpha,u) = \frac{1}{2}\|y-\beta\|_2^2 + \lambda\|\alpha\|_1 + \frac{\rho}{2}\|\alpha - D^{(k+1)}\beta + u\|_2^2 - \frac{\rho}{2}\|u\|_2^2,$$

from which we can derive the standard ADMM updates:

$$\beta \leftarrow \big(I + \rho (D^{(k+1)})^T D^{(k+1)}\big)^{-1}\big(y + \rho (D^{(k+1)})^T(\alpha + u)\big), \tag{9.5}$$

$$\alpha \leftarrow S_{\lambda/\rho}(D^{(k+1)}\beta - u), \tag{9.6}$$

$$u \leftarrow u + \alpha - D^{(k+1)}\beta. \tag{9.7}$$

The β-update is a banded linear system solve, with bandwidth $k+2$, and can be implemented in time $O(n(k+2)^2)$ (actually, $O(n(k+2)^2)$ for the first solve, with a banded Cholesky, and $O(n(k+2))$ for each subsequent solve). The α-update, where $S_{\lambda/\rho}$ denotes coordinate-wise soft-thresholding at the level λ/ρ, takes time $O(n-k-1)$. The dual update uses a banded matrix multiplication, taking time $O(n(k+2))$, and therefore one full iteration of standard ADMM updates can be done in linear time (considering k as a constant).

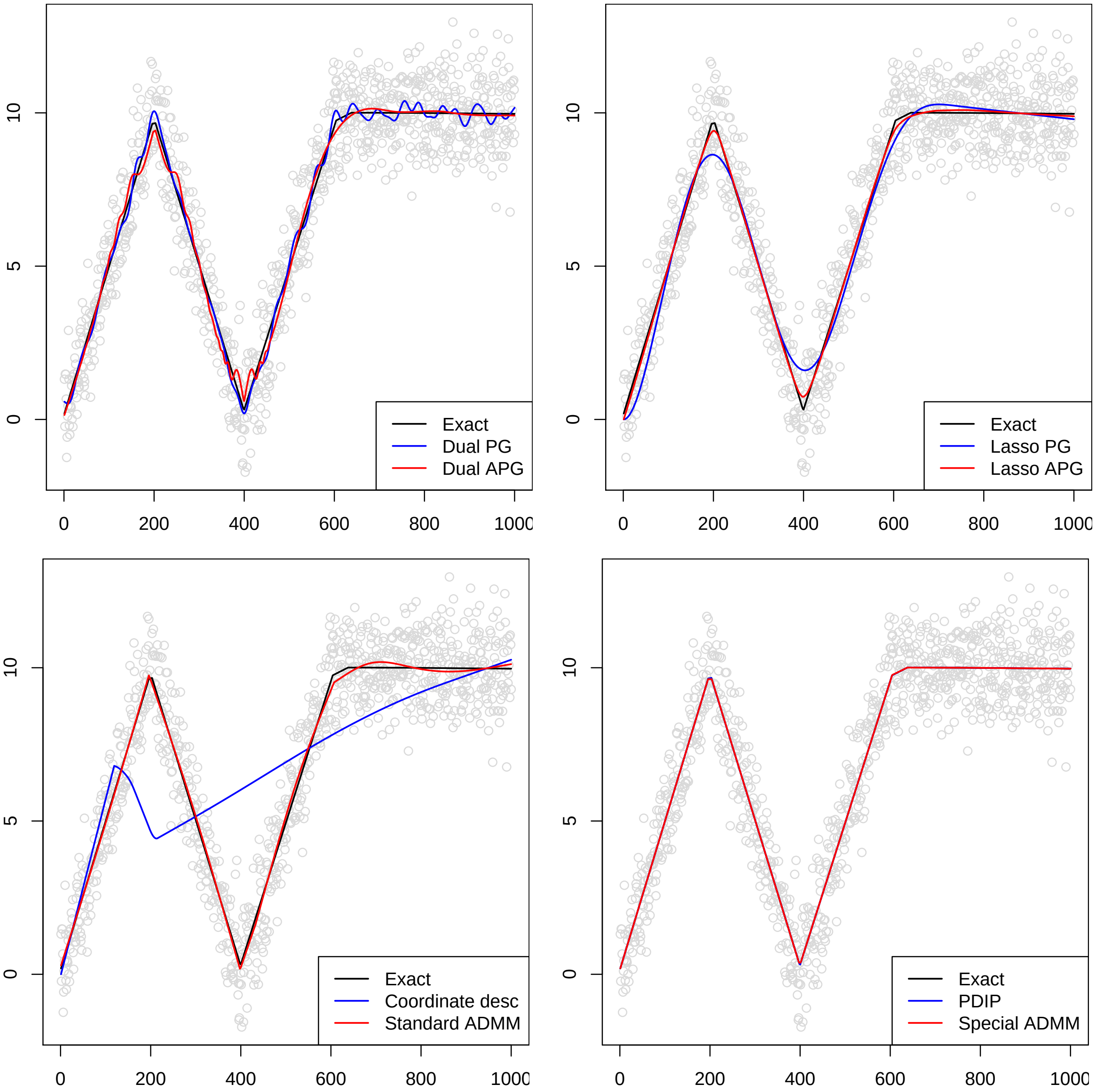


Figure 9.1: *All plots show $n = 1000$ simulated observations in gray and the exact trend filtering solution as a black line, computed using the dual path algorithm of Tibshirani and Taylor [213]. The top left panel shows proximal gradient descent and its accelerated version applied to the dual problem, after 10,000 iterations. The top right show proximal gradient and its accelerated version after rewriting trend filtering in lasso form, again after 10,000 iterations. The bottom left shows coordinate descent applied to the lasso form, and a standard ADMM approach applied to the original problem, each using 5000 iterations (where one iteration for coordinate descent is one full cycle of coordinate updates). The bottom right panel shows the specialized PDIP and ADMM algorithms, which only need 20 iterations, and match the exact solution to perfect visual accuracy. Due to the special form of the problem, all algorithms here have $O(n)$ complexity per iteration (except coordinate descent, which has a higher iteration cost).*

Our specialized ADMM approach instead begins by rewriting (9.1) as

$$\min_{\beta\in\mathbb{R}^n,\, \alpha\in\mathbb{R}^{n-k}} \frac{1}{2}\|y-\beta\|_2^2 + \lambda\|D^{(1)}\alpha\|_1 \;\; \text{subject to} \;\; \alpha = D^{(k)}\beta, \tag{9.8}$$

where we have used the recursive property $D^{(k+1)} = D^{(1)}D^{(k)}$. The augmented Lagrangian is now

$$L(\beta,\alpha,u) = \frac{1}{2}\|y-\beta\|_2^2 + \lambda\|D^{(1)}\alpha\|_1 + \frac{\rho}{2}\|\alpha - D^{(k)}\beta + u\|_2^2 - \frac{\rho}{2}\|u\|_2^2,$$

yielding the specialized ADMM updates:

$$\beta \leftarrow \big(I + \rho(D^{(k)})^T D^{(k)}\big)^{-1}\big(y + \rho(D^{(k)})^T(\alpha+u)\big), \tag{9.9}$$

$$\alpha \leftarrow \mathop{\mathrm{arg\,min}}_{\alpha\in\mathbb{R}^{n-k}} \frac{1}{2}\|D^{(k)}\beta - u - \alpha\|_2^2 + \lambda/\rho\|D^{(1)}\alpha\|_1, \tag{9.10}$$

$$u \leftarrow u + \alpha - D^{(k)}\beta. \tag{9.11}$$

The β- and u-updates are analogous to those from the standard ADMM, just of a smaller order k. But the α-update above is not; the α-update itself requires solving a constant order trend filtering problem, i.e., a 1d fused lasso problem. Therefore, the specialized approach would not be efficient if it were not for the extremely fast, direct solvers that exist for the 1d fused lasso. Two exact, linear time 1d fused lasso solvers were given by Davies and Kovac [49], Johnson [110]. The former is based on taut strings, and the latter on dynamic programming. Both algorithms are very creative and are a marvel in their own right; we are more familiar with the dynamic programming approach, and so in our specialized ADMM algorithm, we utilize (a custom-made, highly-optimized implementation of) this dynamic programming routine for the α-update, hence writing

$$\alpha \leftarrow \mathrm{DP}_{\lambda/\rho}(D^{(k)}\beta - u). \tag{9.12}$$

This uses $O(n-k)$ operations, and thus a full round of specialized ADMM updates runs in linear time, the same as the standard ADMM ones (the two approaches are also empirically very similar in terms of computational time; see Figure 9.4). As mentioned in the introduction, neither the taut string nor dynamic programming approach can be directly extended beyond the $k=0$ case, to the best of our knowledge, for solving higher order trend filtering problems; however, they can be wrapped up in the special ADMM algorithm described above, and in this manner, they lend their efficiency to the computation of higher order estimates.

9.2.1 Superiority of specialized over standard ADMM

We now provide further experimental evidence that our specialized ADMM implementation significantly outperforms the naive standard ADMM. We simulated noisy data from three different underlying signals: constant, sinusoidal, and Doppler wave signals (representing three broad classes of functions: trivial smoothness, homogeneous smoothness, and inhomogeneous smoothness). We examined 9 different problem sizes, spaced roughly logarithmically from $n = 500$ to $n = 500,000$, and considered computation of the trend filtering solution in (9.1) for the orders $k = 1, 2, 3$. We also considered 20 values of λ, spaced logarithmically between $\lambda_{\max}$ and $10^{-5}\lambda_{\max}$, where

$$\lambda_{\max} = \big\|\big((D^{(k+1)}(D^{(k+1)})^T\big)^{-1}(D^{(k+1)})^T y\big\|_\infty,$$

the smallest value of λ at which the penalty term $\|D^{(k+1)}\hat{\beta}\|_1$ is zero at the solution (and hence the solution is exactly a kth order polynomial). In each problem instance—indexed by the choice of underlying

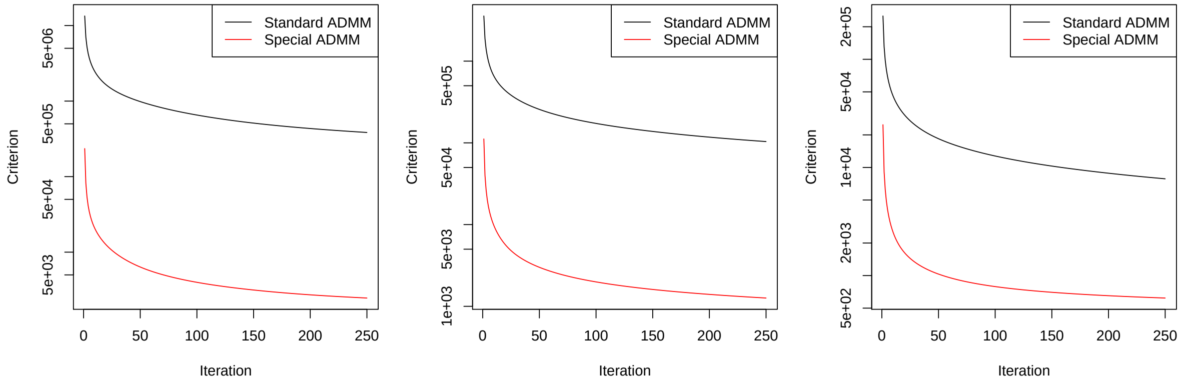


Figure 9.2: *All plots show values of the trend filtering criterion versus iteration number in the two ADMM implementations. The underlying signal here was the Doppler wave, with $n = 10,000$, and $k = 2$. The left plot shows a large value of λ (near $\lambda_{\max}$), the middle a medium value (halfway in between $\lambda_{\max}$ and $10^{-5}\lambda_{\max}$, on a log scale), and the right a small value (equal to $10^{-5}\lambda_{\max}$). The specialized ADMM approach easily outperforms the standard one in all cases.*

function, problem size, polynomial order k, and tuning parameter value λ—we ran a large number of iterations of the ADMM algorithms, and recorded the achieved criterion values across iterations.

The results from one particular instance, in which the underlying signal was the Doppler wave, $n = 10,000$, and $k = 2$, are shown in Figure 9.2; this instance was chosen arbitrarily, and we have found the same qualitative behavior to persist throughout the entire simulation suite. We can see clearly that in each regime of regularization, the specialized routine dominates the standard one in terms of convergence to optimum. Again, we reiterate that qualitatively the same conclusion holds across all simulation parameters, and the gap between the specialized and standard approaches generally widens as the polynomial order k increases.

9.2.2 Some intuition for specialized versus standard ADMM

One may wonder why the two algorithms, standard and specialized ADMM, differ so significantly in terms of their performance. Here we provide some intuition with regard to this question. A first, very rough interpretation: the specialized algorithm utilizes a dynamic programming subroutine (9.12) in place of soft-thresholding (9.6), therefore solving a more "difficult" subproblem in the same amount of time (linear in the input size), and likely making more progress towards minimizing the overall criterion. In other words, this reasoning follows the underlying intuitive principle that for a given optimization task, an ADMM parametrization with "harder" subproblems will enjoy faster convergence.

While the above explanation was fairly vague, a second, more concrete explanation comes from viewing the two ADMM routines in "basis" form, i.e., from essentially inverting $D^{(k+1)}$ to yield an equivalent lasso form of trend filtering, as explained in (9.21) of Appendix 9.6.1, where $H^{(k)}$ is a basis matrix. From this equivalent perspective, the standard ADMM algorithm reparametrizes (9.21) as in

$$\min_{\theta \in \mathbb{R}^n,\, w \in \mathbb{R}^n} \frac{1}{2}\|y - H^{(k)}w\|_2^2 + \lambda \cdot k! \sum_{j=k+2}^{n} |\theta_j| \;\; \text{subject to} \;\; w = \theta, \tag{9.13}$$

and the specialized ADMM algorithm reparametrizes (9.21) as in

$$\min_{\theta \in \mathbb{R}^n,\, w \in \mathbb{R}^n} \frac{1}{2}\|y - H^{(k-1)}w\|_2^2 + \lambda \cdot k! \sum_{j=k+2}^{n} |\theta_j| \;\text{ subject to }\; w = L\theta, \tag{9.14}$$

where we have used the recursion $H^{(k)} = H^{(k-1)}L$ [227], analogous (equivalent) to $D^{(k+1)} = D^{(1)}D^{(k)}$. The matrix $L \in \mathbb{R}^{n\times n}$ is block diagonal with the first $k \times k$ block being the identity, and the last $(n-k) \times (n-k)$ block being the lower triangular matrix of 1s. What is so different between applying ADMM to (9.14) instead of (9.13)? Loosely speaking, if we ignore the role of the dual variable, the ADMM steps can be thought of as performing alternating minimization over θ and w. The joint criterion being minimized, i.e., the augmented Lagrangian (again hiding the dual variable) is of the form

$$\frac{1}{2}\left\| z - \begin{bmatrix} H^{(k)} & 0 \\ \sqrt{\rho}I & -\sqrt{\rho}I \end{bmatrix} \begin{bmatrix} \theta \\ w \end{bmatrix} \right\|_2^2 + \lambda \cdot k! \sum_{j=k+2}^{n} |\theta_j| \tag{9.15}$$

for the standard parametrization (9.13), and

$$\frac{1}{2}\left\| z - \begin{bmatrix} H^{(k-1)} & 0 \\ \sqrt{\rho}I & -\sqrt{\rho}L \end{bmatrix} \begin{bmatrix} \theta \\ w \end{bmatrix} \right\|_2^2 + \lambda \cdot k! \sum_{j=k+2}^{n} |\theta_j| \tag{9.16}$$

for the special parametrization (9.14). The key difference between (9.15) and (9.16) is that the left and right blocks of the regression matrix in (9.15) are highly (negatively) correlated (the bottom left and right blocks are each scalar multiples of the identity), but the blocks of the regression matrix in (9.16) are not (the bottom blocks are the identity and the lower triangular matrix of 1s). Hence, in the context of an alternating minimization scheme, an update step in (9.16) should make more progress than an update step in (9.15), because the descent directions for θ and w are not as adversely aligned (think of coordinatewise minimization over a function whose contours are tilted ellipses, and over one whose contours are spherical). Using the equivalence between the basis form and the original (difference-penalized) form of trend filtering, therefore, we may view the special ADMM updates (9.9)–(9.11) as *decorrelated* versions of the original ADMM updates (9.5)–(9.7). This allows each update step to make greater progress in descending on the overall criterion.

9.2.3 Superiority of warm over cold starts

In the above numerical comparison between special and standard ADMM, we ran both methods with cold starts, meaning that the problems over the sequence of λ values were solved independently, without sharing information. Warm starting refers to a strategy in which we solve the problem for the largest value of λ first, use this solution to initialize the algorithm at the second largest value of λ, etc. With warm starts, the relative performance of the two ADMM approaches does not change. However, the performance of both algorithms does improve in an absolute sense, illustrated for the specialized ADMM algorithm in Figure 9.3.

This example is again representative of the experiments across the full simulation suite. Therefore, from this point forward, we use warm starts for all experiments.

9.2.4 Choice of the augmented Lagrangian parameter ρ

A point worth discussing is the choice of augmented Lagrangian parameter ρ used in the above experiments. Recall that ρ is not a statistical parameter associated with the trend filtering problem (9.1); it

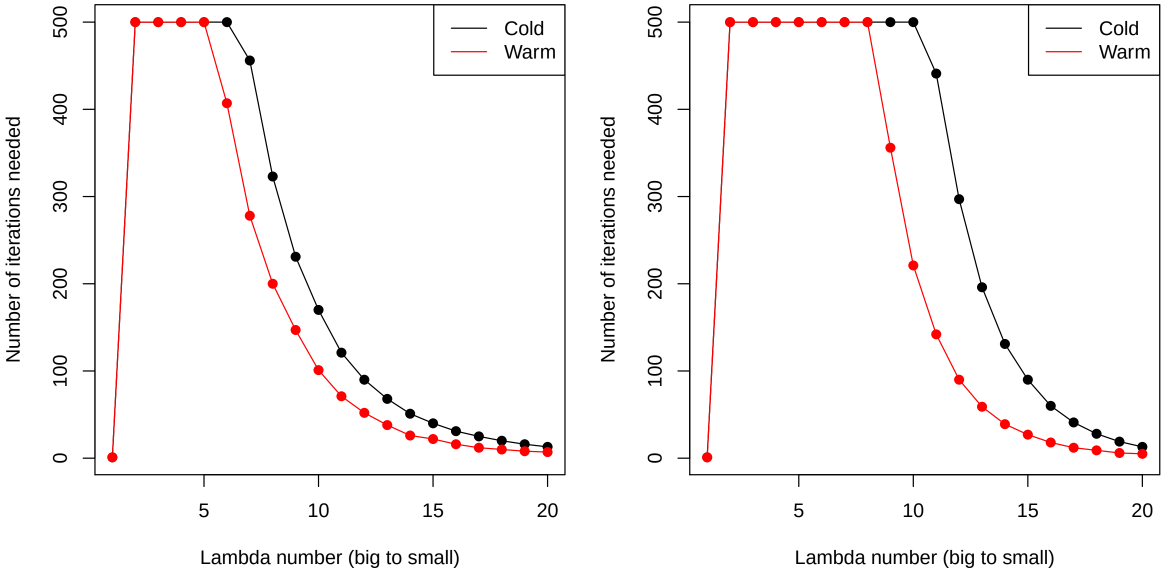


Figure 9.3: *The x-axis in both panels represents 20 values of λ, log-spaced between $\lambda_{\max}$ and $10^{-5}\lambda_{\max}$, and the y-axis the number of iterations needed by specialized ADMM to reach a prespecified level of accuracy, for $n = 10,000$ noisy points from the Doppler curve for $k = 2$ (left) and $k = 3$ (right). Warm starts (red) have an advantage over cold starts (black), especially in the statistically reasonable (middle) range for λ.*

is rather an optimization parameter introduced during the formation of the agumented Lagrangian in ADMM. It is known that under very general conditions, the ADMM algorithm converges to optimum for any fixed value of ρ [27]; however, in practice, the rate of convergence of the algorithm, as well as its numerical stability, can both depend strongly on the choice of ρ.

We found the choice of setting $\rho = \lambda$ to be numerically stable across all setups. Note that in the ADMM updates (9.5)–(9.7) or (9.9)–(9.11), the only appearance of λ is in the α-update, where we apply $S_{\lambda/\rho}$ or $\mathrm{DP}_{\lambda/\rho}$, soft-thresholding or dynamic programming (to solve the 1d fused lasso problem) at the level λ/ρ. Choosing ρ to be proportional to λ controls the amount of change enacted by these subroutines (intuitively making it neither too large nor too small at each step). We also tried adaptively varying ρ, a heuristic suggested by Boyd et al. [27], but found this strategy to be less stable overall; it did not yield consistent benefits for either algorithm.

Recall that this chapter is not concerned with the model selection problem of how to choose λ, but just with the optimization problem of how to solve (9.1) when given λ. All results in the rest of this chapter reflect the default choice $\rho = \lambda$, unless stated otherwise.

9.3 Comparison of specialized ADMM and PDIP

Here we compare our specialized ADMM algorithm and the PDIP algorithm of [115]. We used the C++/LAPACK implementation of the PDIP method (written for the case $k = 1$) that is provided freely by these authors, and generalized it to work for an arbitrary order $k \geq 1$. To put the methods on equal footing, we also wrote our own efficient C implementation of the specialized ADMM algorithm. This code has been interfaced to R via the `trendfilter` function in the R package `glmgen`, available at `https://github.com/statsmaths/glmgen`.

A note on the PDIP implementation: this algorithm is actually applied to the dual of (9.1), as given in (9.20) in Appendix 9.6.1, and its iterations solve linear systems in the banded matrix D in $O(n)$ time. The number of constraints, and hence the number of log barrier terms, is $2(n-k-1)$. We used 10 for the log barrier update factor (i.e., at each iteration, the weight of log barrier term is scaled by $1/10$). We used backtracking line search to choose the step size in each iteration, with parameters 0.01 and 0.5 (the former being the fraction of improvement over the gradient required to exit, and the latter the step size shrinkage factor). These specific parameter values are the defaults suggested by Boyd and Vandenberghe [26] for interior point methods, and are very close to the defaults in the original PDIP linear trend filtering code from Kim et al. [115]. In the settings in which PDIP struggled (to be seen in what follows), we tried varying these parameter values, but no single choice led to consistently improved performance.

9.3.1 Comparison of cost per iteration

Per iteration, both ADMM and PDIP take $O(n)$ time, as explained earlier. Figure 9.4 reveals that the constant hidden in the $O(\cdot)$ notation is about 10 times larger for PDIP than ADMM. Though the comparisons that follow are based on achieved criterion value versus iteration, it may be kept in mind that convergence plots for the criterion values versus time would be stretched by a factor of 10 for PDIP.

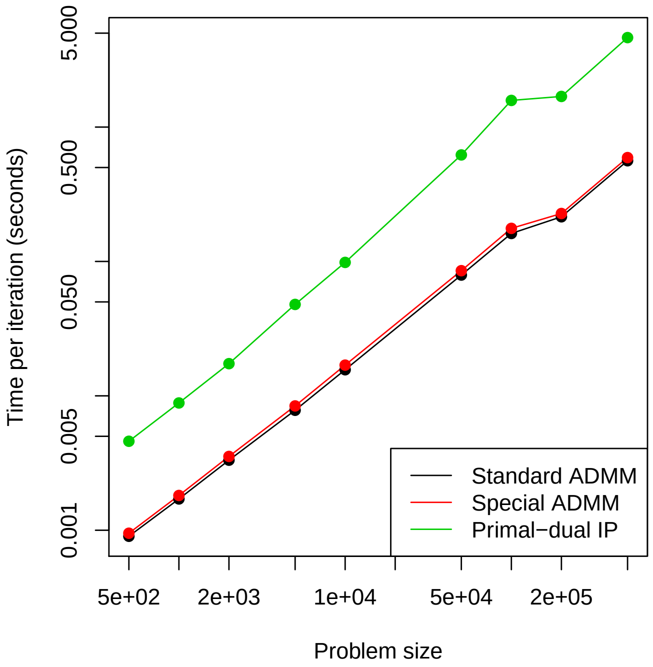


Figure 9.4: *A log-log plot of time per iteration of ADMM and PDIP routines against problem size n (20 values from 500 up to 500,000). The times per iteration of the algorithms were averaged over 3 choices of underlying function (constant, sinusoidal, and Doppler), 3 orders of trends ($k = 1, 2, 3$), 20 values of λ (log-spaced between $\lambda_{\max}$ and $10^{-5}\lambda_{\max}$), and 10 repetitions for each combination (except the two largest problem sizes, for which we performed 3 repetitions). This validates the theoretical $O(n)$ iteration complexities of the algorithms, and the larger offset (on the log-log scale) for PDIP versus ADMM implies a larger constant in the linear scaling: an ADMM iteration is about 10 times faster than a PDIP iteration.*

9.3.2 Comparison for $k = 1$ (piecewise linear fitting)

In general, for $k = 1$ (piecewise linear fitting), both the specialized ADMM and PDIP algorithms perform similarly, as displayed in Figure 9.7. The PDIP algorithm displays a relative advantage as λ becomes small, but the convergence of ADMM is still strong in absolute terms. Also, it is important to note that these small values of λ correspond to solutions that overfit the underlying trend in the problem context, and hence PDIP outperforms ADMM in a statistically uninteresting regime of regularization.

9.3.3 Comparison for $k = 2$ (piecewise quadratic fitting)

For $k = 2$ (piecewise quadratic fitting), the PDIP routine struggles for moderate to large values of λ, increasingly so as the problem size grows, as shown in Figure 9.10. These convergence issues remain as we vary its internal optimization parameters (i.e., its log barrier update parameter, and backtracking parameters). Meanwhile, our specialized ADMM approach is much more stable, exhibiting strong convergence behavior across all λ values, even for large problem sizes in the hundreds of thousands.

The convergence issues encountered by PDIP here, when $k = 2$, are only amplified when $k = 3$, as the issues begin to show at much smaller problem sizes; still, the specialized ADMM steadily converges, and is a clear winner in terms of robustness. Analysis of this case is deferred until Appendix 9.6.2 for brevity.

9.3.4 Some intuition on specialized ADMM versus PDIP

We now discuss some intuition for the observed differences between the specialized ADMM and PDIP. This experiments in this section showed that PDIP will often diverge for large problem sizes and moderate values of the trend order ($k = 2, 3$), regardless of the choices of the log barrier and backtracking line search parameters. That such behavior presents itself for large n and k suggests that PDIP is affected by poor conditioning of the difference operator $D^{(k+1)}$ in these cases. Since PDIP is affine invariant, in theory it should not be affected by issues of conditioning at all. But when $D^{(k+1)}$ is poorly conditioned, it is difficult to solve the linear systems in $D^{(k+1)}$ that lie at the core of a PDIP iteration, and this leads PDIP to take a noisy update step (like taking a majorization step using a perturbed version of the Hessian). If the computed update directions are noisy enough, then PDIP can surely diverge.

Why does specialized ADMM not suffer the same fate, since it too solves linear systems in each iteration (albeit in $D^{(k)}$ instead of $D^{(k+1)}$)? There is an important difference in the form of these linear systems. Disregarding the order of the difference operator and denoting it simply by D, a PDIP iteration solves linear systems (in x) of the form

$$(DD^T + J)x = b \tag{9.17}$$

where J is a diagonal matrix, and an ADMM iteration solves systems of the form

$$(D^T D + \rho I)x = b \tag{9.18}$$

Recall that by default we set the augmented Lagrangian parameter to be $\rho = \lambda$; this bounds ρ away from zero, and provides an important singular value "buffer" for the linear system (9.18): the eigenvalues values of $D^T D + \rho I$ are all at least ρ, which, if ρ is sizable, can make up for the possibly poor conditioning of D. Meanwhile, the diagonal elements of J in (9.17) can be driven to zero across iterations of the PDIP method; in fact, at optimality, complementary slackness implies that J_{ii} is zero whenever the ith dual variable lies strictly inside the interval $[-\lambda, \lambda]$. Hence, matrix J does not always provide the needed buffer for the linear system in (9.17), so $DD^T + J$ can remain poorly conditioned, causing numerical instability

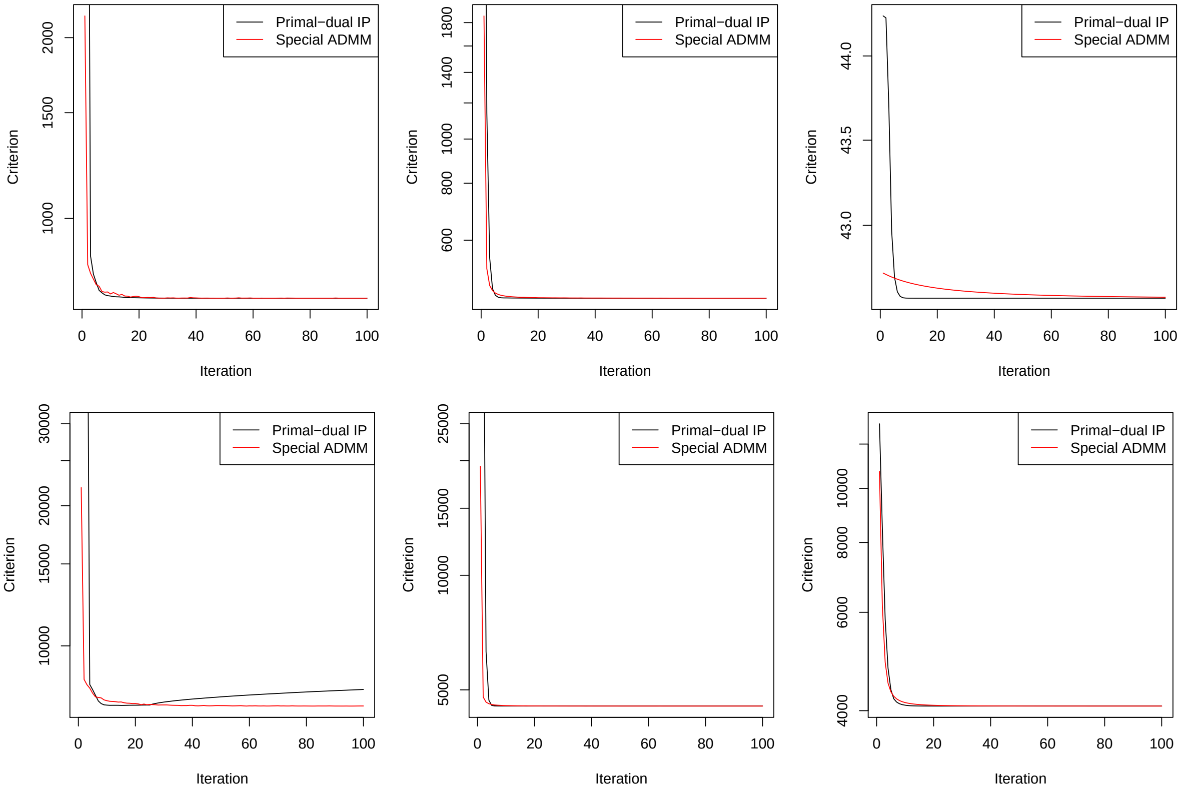


Figure 9.5: *Convergence plots for $k = 1$: achieved criterion values across iterations of ADMM and PDIP. The first row concerns $n = 10,000$ points, and the second row $n = 100,000$ points, both generated around a sinusoidal curve. The columns (from left to right) display high to low values of λ: near $\lambda_{\max}$, halfway in between (on a log scale) $\lambda_{\max}$ and $10^{-5}\lambda_{\max}$, and equal to $10^{-5}\lambda_{\max}$, respectively. Both algorithms exhibit good convergence.*

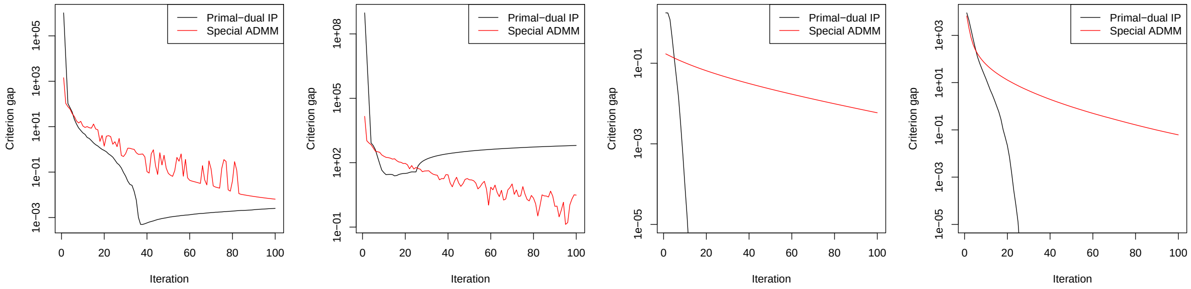


Figure 9.6: *Convergence gaps for $k = 1$: achieved criterion value minus the optimum value across iterations of ADMM and PDIP. Here the optimum value was defined as smallest achieved criterion value over 5000 iterations of either algorithm. The first two plots are for λ near $\lambda_{\max}$, with $n = 10,000$ and $n = 100,000$ points, respectively. In this high regularization regime, ADMM fares better for large n. The last two plots are for $\lambda = 10^{-5}\lambda_{\max}$, with $n = 10,000$ and $n = 100,000$, respectively. Now in this low regularization regime, PDIP converges at what appears to be a second-order rate, and ADMM does not. However, these small values of λ are not statistically interesting in the context of the example, as they yield grossly overfit trend estimates of the underlying sinusoidal curve.*

Figure 9.7: *Convergence plots and gaps ($k = 1$), for specialized ADMM and PDIP.*

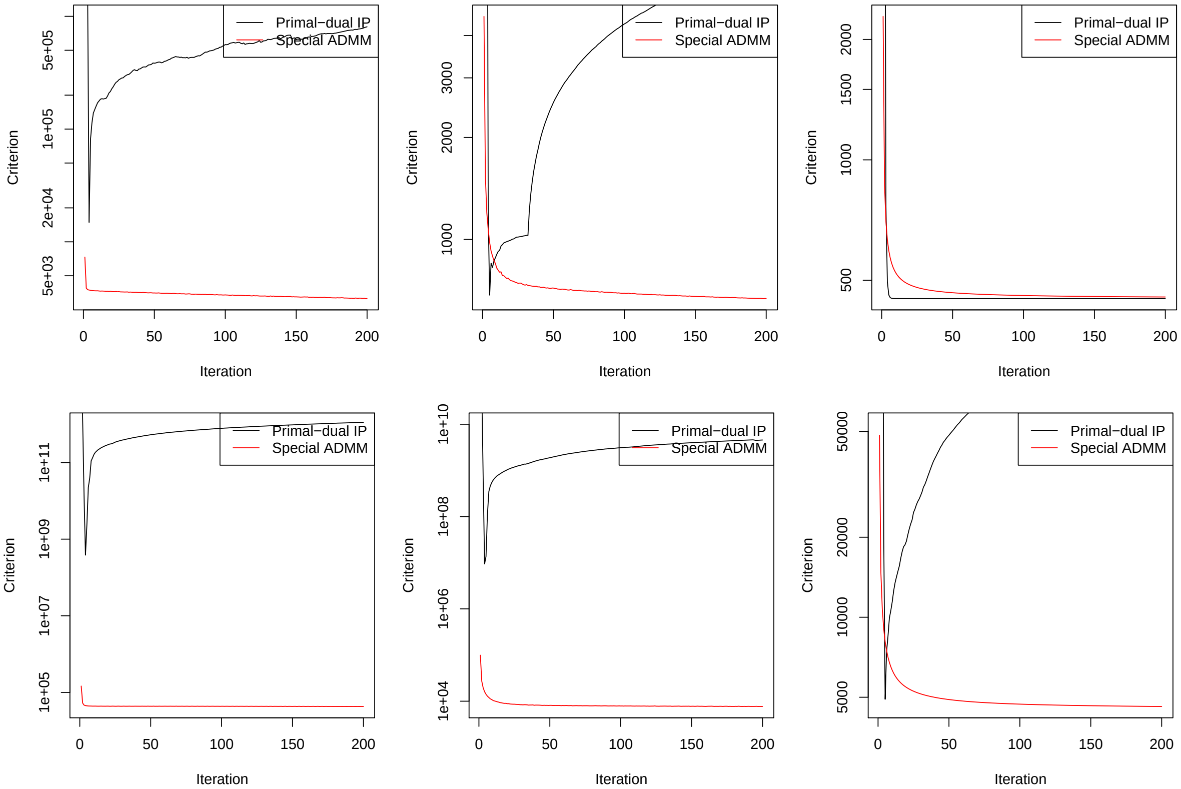


Figure 9.8: *Convergence plots for $k = 2$: achieved criterion values across iterations of ADMM and PDIP, with the same layout as in Figure 9.5. The specialized ADMM routine has fast convergence in all cases. For all but the smallest λ values, PDIP does not come close to convergence. These values of λ are so small that the corresponding trend filtering solutions are not statistically desirable in the first place; see below.*

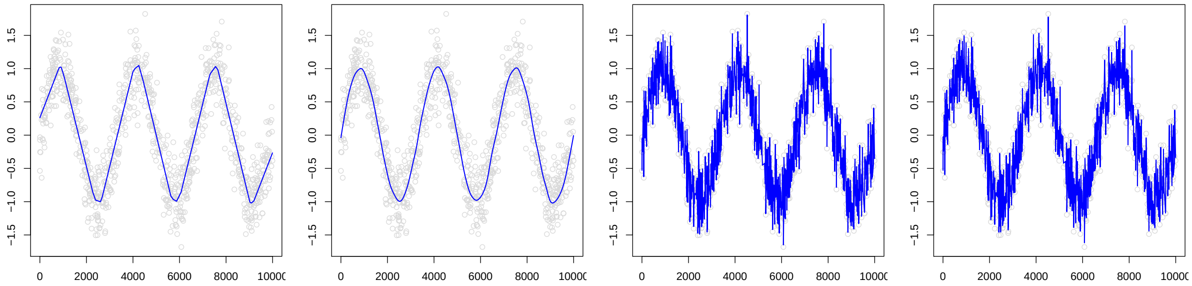


Figure 9.9: *Visualization of trend filtering estimates for the experiments in Figures 9.7, 9.10. The estimates were trained on $n = 10,000$ points from an underlying sinusoidal curve (but the above plots have been downsampled to 1000 points for visibility). The two left panels show the fits for $k = 1, 2$, respectively, in the high regularization regime, where λ is near $\lambda_{\max}$. The specialized ADMM approach outperforms PDIP (and shown are the ADMM fits). The two right panels show the fits for $k = 1, 2$, respectively, in the low regularization regime, with $\lambda = 10^{-5}\lambda_{\max}$. PDIP converges faster than ADMM (and shown are the PDIP fits), but this is not a statistically reasonable regime for trend estimation.*

Figure 9.10: *Convergence plots and estimated fits ($k = 2$) for special ADMM and PDIP.*

issues when solving (9.17) in PDIP iterations. In particular, when λ is large, many dual coordinates will lie strictly inside $[-\lambda, \lambda]$ at optimality, which means that many diagonal elements of J will be pushed towards zero over PDIP iterations. This explains why PDIP experiences particular difficulty in the large λ regime, as seen in our experiments.

9.4 Arbitrary input points

Up until now, we have assumed that the input locations are implicitly $x_1 = 1, \dots x_n = n$; in this section, we discuss the algorithmic extension of our specialized ADMM algorithm to the case of arbitrary input points $x_1, \dots x_n$. Such an extension is highly important, because, as a nonparametric regression tool, trend filtering is much more likely to be used in a setting with generic inputs than one in which these are evenly spaced. Fortuitously, there is little that needs to be changed with the trend filtering problem (9.1) when we move from unit spaced inputs $1, \dots n$ to arbitrary ones $x_1, \dots x_n$; the only difference is that the operator $D^{(k+1)}$ is replaced by $D^{(x,k+1)}$, which is adjusted for the uneven spacings present in $x_1, \dots x_n$. These adjusted difference operators are still banded with the same structure, and are still defined recursively. We begin with $D^{(x,1)} = D^{(1)}$, the usual first difference operator in (9.2), and then for $k \geq 1$, we define, assuming unique sorted points $x_1 < \dots < x_n$,

$$D^{(x,k+1)} = D^{(1)} \cdot \mathrm{diag}\left(\frac{k}{x_{k+1} - x_1}, \dots \frac{k}{x_n - x_{n-k}} \right) \cdot D^{(x,k)},$$

where $\mathrm{diag}(a_1, \dots a_m)$ denotes a diagonal matrix with elements $a_1, \dots a_m$; see Tibshirani [212], Wang et al. [227]. Abbreviating this as $D^{(x,k+1)} = D^{(1)} \widetilde{D}^{(x,k)}$, we see that we only need to replace $D^{(k)}$ by $\widetilde{D}^{(x,k)}$ in our special ADMM updates, replacing one $(k+1)$-banded matrix with another.

The more uneven the spacings among $x_1, \dots x_n$, the worse the conditioning of $\tilde{D}^{(x,k)}$, and hence the slower to converge our specialized ADMM algorithm (indeed, the slower to converge any of the alternative algorithms suggested in Section 9.1.2.) As shown in Figure 9.11, however, our special ADMM approach is still fairly robust even with considerably irregular design points $x_1, \dots x_n$.

9.4.1 Choice of the augmented Lagrangian parameter ρ

Aside from the change from $D^{(k)}$ to $\tilde{D}^{(x,k)}$, another key change in the extension of our special ADMM routine to general inputs $x_1, \dots x_n$ lies in the choice of the augmented Lagrangian parameter ρ. Recall that for unit spacings, we argued for the choice $\rho = \lambda$. For arbitrary inputs $x_1 < \dots < x_n$, we advocate the use of

$$\rho = \lambda \left(\frac{x_n - x_1}{n} \right)^k. \tag{9.19}$$

Note that this (essentially) reduces to $\rho = \lambda$ when $x_1 = 1, \dots x_n = n$. To motivate the above choice of ρ, consider running two parallel ADMM routines on the same outputs $y_1, \dots y_n$, but with different inputs: $1, \dots n$ in one case, and arbitrary but evenly spaced $x_1, \dots x_n$ in the other. Then, setting $\rho = \lambda$ in the first routine, we choose ρ in the second routine to try to match the first round of ADMM updates as best as possible, and this leads to ρ as in (9.19). In practice, this input-adjusted choice of ρ makes a important difference in terms of the progress of the algorithm.

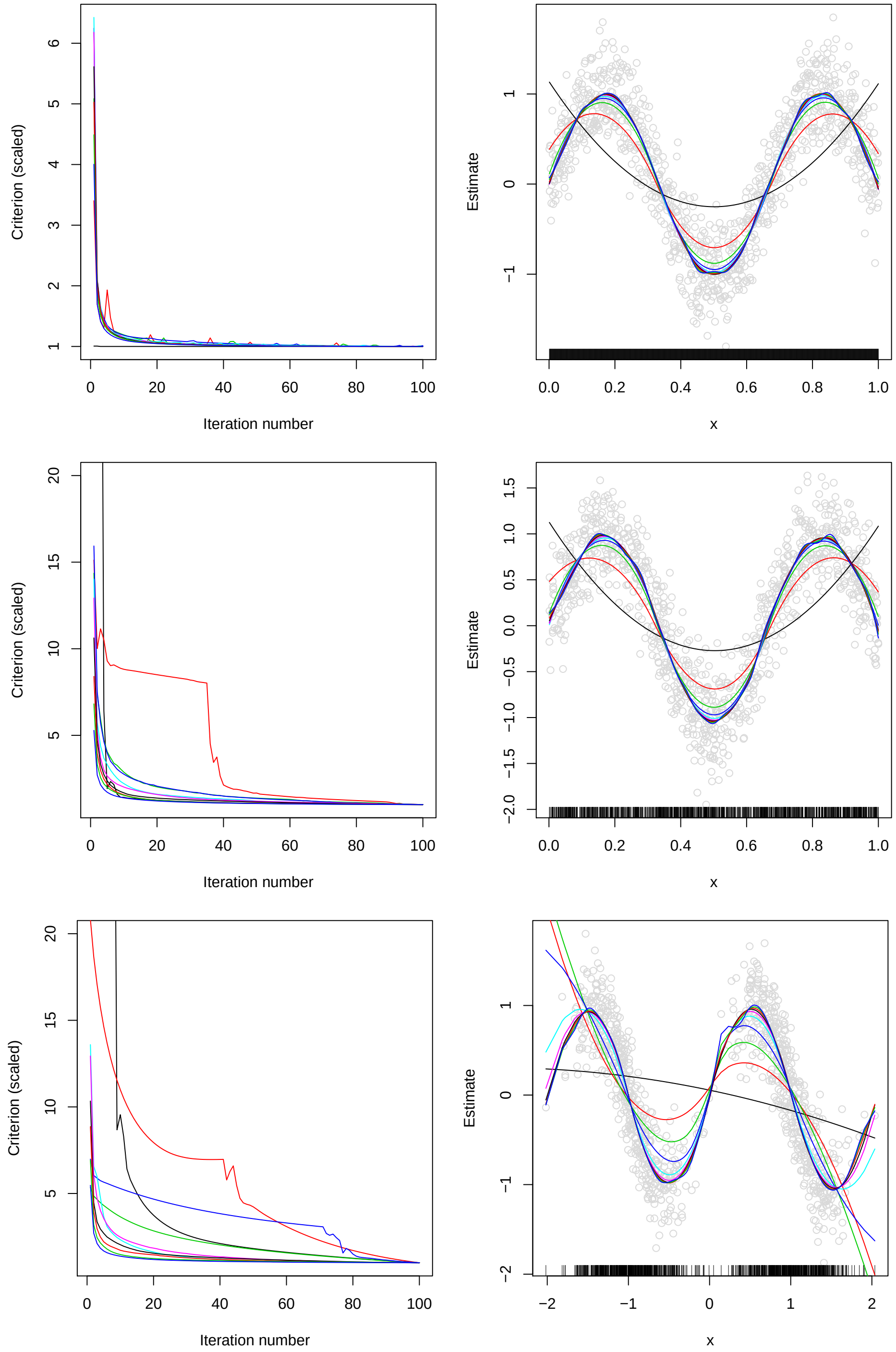


Figure 9.11: *Each row considers a different design for the inputs. Top row: evenly spaced over* $[0, 1]$*; middle row: uniformly at random over* $[0, 1]$*; bottom row: mixture of Gaussians. In each case, we drew* $n = 1000$ *points from a noisy sinusoidal curve at the prescribed inputs. The left panels show the achieved criterion values versus iterations of the specialized ADMM implementation, with* $k = 2$*, the different colored lines show convergence plots at different* λ *values (we used 20 values log-spaced between* $\lambda_{\max}$ *and* $10^{-5}\lambda_{\max}$*). The curves are all scaled to end at the same point for visibility. The ADMM algorithm experiences more difficulty as the input spacings become more irregular, due to poorer conditioning of the difference operator. The right panels plot the fitted estimates, with the ticks on the x-axis marking the input locations.*

9.5 ADMM algorithms for trend filtering extensions

One of the real strengths of the ADMM framework for solving (9.1) is that it can be readily adapted to fit modifications of the basic trend filtering model. Here we very briefly inspect some extensions of trend filtering—some of these extensions were suggested by Tibshirani [212], some by Kim et al. [115], and some are novel to this manuscript. Our intention is not to deliver an exhaustive list of such extensions (as many more can be conjured), or to study their statistical properties, but rather to show that the ADMM framework is a flexible stage for such creative modeling tasks.

Sparse trend filtering

In this sparse variant of trend filtering, we aim to estimate a trend that can be exactly zero in some regions of its domain, and can depart from zero in a smooth (piecewise polynomial) fashion. This may be a useful modeling tool when the observations $y_1, \ldots y_n$ represent a difference of signals across common input locations. We solve, as suggested by Tibshirani [212],

$$\hat{\beta} = \underset{\beta \in \mathbb{R}^n}{\arg\min} \ \frac{1}{2}\|y - \beta\|_2^2 + \lambda_1 \|D^{(k+1)}\beta\|_1 + \lambda_2 \|\beta\|_1,$$

where both λ_1, λ_2 are tuning parameters. A short calculation yields the specialized ADMM updates:

$$\begin{aligned}
&\beta \leftarrow \big((1+\rho_2)I + \rho_1 (D^{(k)})^T D^{(k)}\big)^{-1}\big(y + \rho_1 (D^{(k)})^T(\alpha + u) + \rho_2(\gamma + v)\big),\\
&\alpha \leftarrow \mathrm{DP}_{\lambda_1/\rho_1}(D^{(k)}\beta - u),\\
&\gamma \leftarrow S_{\lambda_2/\rho_2}(\beta - v),\\
&u \leftarrow u + \alpha - D^{(k)}\beta, \quad v \leftarrow v + \gamma - \beta.
\end{aligned}$$

This is still highly efficient, using $O(n)$ operations per iteration. An example is shown in Figure 9.12.

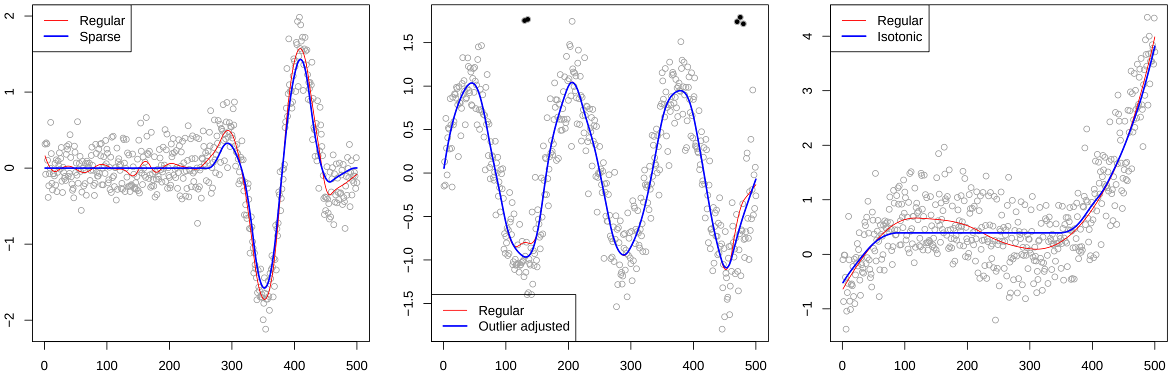


Figure 9.12: *Three examples, of sparse, outlier-corrected, and isotonic trend filtering, from left to right. These extensions of the basic trend filtering model were computed from $n = 500$ data points; their fits are drawn in blue, and the original (unmodified) trend filtering solutions are drawn in red, both using the same hand-chosen tuning parameter values. (In the middle panel, the points deemed outliers by the nonzero entries of $\hat{z}$ are colored in black.) These comparisons are not supposed to be statistically fair, but rather, illuminate the qualitative differences imposed by the extra penalties or constraints in the extensions.*

Mixed trend filtering

To estimate a trend with two mixed polynomial orders $k_1, k_2 \geq 0$, we solve

$$\hat{\beta} = \underset{\beta \in \mathbb{R}^n}{\arg\min} \ \frac{1}{2}\|y - \beta\|_2^2 + \lambda_1 \|D^{(k_1+1)}\beta\|_1 + \lambda_2 \|D^{(k_2+1)}\beta\|_1,$$

as discussed in Tibshirani [212]. The result is that either polynomial trend, of order k_1 or k_2, can act as the dominant trend at any location in the domain. More generally, for r mixed polynomial orders, $k_\ell \geq 0$, $\ell = 1, \ldots r$, we replace the penalty with $\sum_{\ell=1}^r \lambda_\ell \|D^{(k_\ell+1)}\beta\|_1$. The specialized ADMM routine naturally extends to this multi-penalty problem:

$$\begin{aligned}
&\beta \leftarrow \Big(I + \sum_{\ell=1}^r \rho_\ell (D^{(k_\ell)})^T D^{(k_\ell)}\Big)^{-1} \Big(y + \sum_{\ell=1}^r \rho_\ell (D^{(k_\ell)})^T (\alpha_\ell + u_\ell)\Big), \\
&\alpha_\ell \leftarrow \mathrm{DP}_{\lambda_\ell/\rho_\ell}(D^{(k_\ell)}\beta - u_\ell), \quad \ell = 1, \ldots r, \\
&u_\ell \leftarrow u_\ell + \alpha_\ell - D^{(k_\ell)}\beta, \quad \ell = 1, \ldots r.
\end{aligned}$$

Each iteration here uses $O(nr)$ operations (recall r is the number of mixed trends).

Trend filtering with outlier detection

To simultaneously estimate a trend and detect outliers, we solve

$$(\hat{\beta}, \hat{z}) = \underset{\beta, z \in \mathbb{R}^n}{\arg\min} \ \frac{1}{2}\|y - \beta - z\|_2^2 + \lambda_1 \|D^{(k+1)}\beta\|_1 + \lambda_2 \|z\|_1,$$

as in Kim et al. [115], She and Owen [189], where the nonzero components of $\hat{z}$ correspond to adaptively detected outliers. A short derivation leads to the updates:

$$\begin{aligned}
&\begin{pmatrix} \beta \\ z \end{pmatrix} \leftarrow \begin{pmatrix} I + \rho_1 (D^{(k)})^T D^{(k)} & I \\ I & (1+\rho_2) I \end{pmatrix}^{-1} \begin{pmatrix} y + \rho_1 (D^{(k)})^T (\alpha + u) \\ y + \rho_2 (\gamma + v) \end{pmatrix}, \\
&\alpha \leftarrow \mathrm{DP}_{\lambda_1/\rho_1}(D^{(k)}\beta - u), \\
&\gamma \leftarrow S_{\lambda_2/\rho_2}(z - v), \\
&u \leftarrow u + \alpha - D^{(k)}\beta, \quad v \leftarrow v + \gamma - z.
\end{aligned}$$

Again, this routine uses $O(n)$ operations per iteration. See Figure 9.12 for an example.

Isotonic trend filtering

A monotonicity constraint in the estimated trend is straightforward to encode:

$$\hat{\beta} = \underset{\beta \in \mathbb{R}^n}{\arg\min} \ \frac{1}{2}\|y - \beta\|_2^2 + \lambda \|D^{(k+1)}\beta\|_1 \ \text{ subject to } \ \beta_1 \leq \beta_2 \leq \ldots \leq \beta_n,$$

as suggested by Kim et al. [115]. The specialized ADMM updates are easy to derive:

$$\begin{aligned}
&\beta \leftarrow \big((1+\rho_2) I + \rho_1 (D^{(k)})^T D^{(k)}\big)^{-1} \big(y + \rho_1 (D^{(k)})^T (\alpha + u) + \rho_2 (\gamma + v)\big), \\
&\alpha \leftarrow \mathrm{DP}_{\lambda/\rho}(D^{(k)}\beta - u), \\
&\gamma \leftarrow \mathrm{IR}(\beta - v), \\
&u \leftarrow u + \alpha - D^{(k)}\beta, \quad v \leftarrow v + \gamma - \beta.
\end{aligned}$$

where $\mathrm{IR}(z)$ denotes an isotonic regression fit on z; since this takes $O(n)$ time (e.g., Stout [205]), a round of updates also takes $O(n)$ time. Figure 9.12 gives an example.

Nearly-isotonic trend filtering

Instead of enforcing strict monotonicity in the fitted values, we can penalize the pointwise nonmontonicities with a separate penalty, following Tibshirani et al. [214]:

$$\hat{\beta} = \operatorname*{arg\,min}_{\beta \in \mathbb{R}^n} \; \frac{1}{2}\|y - \beta\|_2^2 + \lambda_1 \|D^{(k+1)}\beta\|_1 + \lambda_2 \sum_{i=1}^{n-1} (\beta_i - \beta_{i+1})_+.$$

This results in a "nearly-isotonic" fit $\hat{\beta}$. Above, we use $x_+ = \max\{x, 0\}$ to denote the positive part of x. The specialized ADMM updates are:

$$\begin{aligned}
&\beta \leftarrow \big((1+\rho_2)I + \rho_1 (D^{(k)})^T D^{(k)}\big)^{-1}\big(y + \rho_1 (D^{(k)})^T(\alpha + u) + \rho_2(\gamma + v)\big), \\
&\alpha \leftarrow \mathrm{DP}_{\lambda_1/\rho_1}(D^{(k)}\beta - u), \\
&\gamma \leftarrow \mathrm{DP}^+_{\lambda_2/\rho_2}(\beta - v), \\
&u \leftarrow u + \alpha - D^{(k)}\beta, \quad v \leftarrow v + \gamma - \beta.
\end{aligned}$$

where $\mathrm{DP}^+_t(z)$ denotes a nearly-isotonic regression fit to z, with penalty parameter t. It can be computed in $O(n)$ time by modifying the dynamic programming algorithm of Johnson [110] for the 1d fused lasso, so one round of updates still takes $O(n)$ time.

Conclusion

We proposed a specialized but simple ADMM approach for trend filtering, leveraging the strength of extremely fast, exact solvers for the special case $k = 0$ (the 1d fused lasso problem) in order to solve higher order problems with $k \geq 1$. The algorithm is fast and robust over a wide range of problem sizes and regimes of regularization parameters (unlike primal-dual interior point methods, the current state-of-the-art). Our specialized ADMM algorithm converges at a far superior rate to (accelerated) first-order methods, coordinate descent, and (what may be considered as) the standard ADMM approach for trend filtering. Finally, a major strength of our proposed algorithm is that it can be modified to solve many extensions of the basic trend filtering problem. Software for our specialized ADMM algorithm is accessible through the `trendfilter` function in the R package `glmgen`, built around a lower level C package, both freely available at `https://github.com/statsmaths/glmgen`.

9.6 Appendix: further details and simulations

9.6.1 Algorithm details for the motivating example

First, we examine in Figure 9.13 the condition numbers of the discrete difference operators $D^{(k+1)} \in \mathbb{R}^{(n-k-1)\times n}$, for varying problem sizes n, and $k = 0, 1, 2$. Since the plot uses a log-log scale, the straight lines indicate that the condition numbers grow polynomially with n (with a larger exponent for larger k). The sheer size of the condition numbers (which can reach 10^{10} or larger, even for a moderate problem size of $n = 5000$) is worrisome from an optimization point of view; roughly speaking, we would expect the criterion in these cases to be very flat around its optimum.

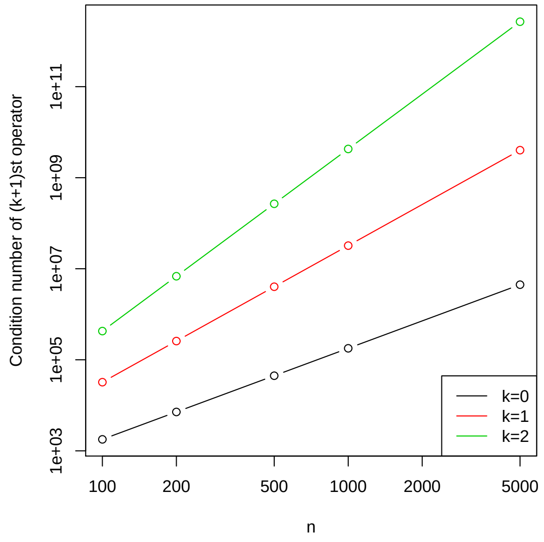


Figure 9.13: *A log-log plot of the condition number of $D^{(k+1)}$ versus the problem size n, for $k = 0, 1, 2$, where the condition numbers scale roughly like n^k.*

Figure 9.1 (in the introduction) provides evidence that such a worry can be realized in practice, even with only a reasonable polynomial order and moderate problem size. For this example, we drew $n = 1000$ points from an underlying piecewise linear function, and studied computation of the linear trend filtering estimate, i.e., with $k = 1$, when $\lambda = 1000$. We chose this tuning parameter value because it represents a statistically reasonable level of regularization in the example. The *exact solution* of the trend filtering problem at $\lambda = 1000$ was computed using the generalized lasso dual path algorithm [9, 213]. The problem size here is small enough that this algorithm, which tracks the solution in (9.1) as λ varies continuously from ∞ to 0, can be run effectively; however, for larger problem sizes, computation of the full solution path quickly becomes intractable. Each panel of Figure 9.1 plots the simulated data points, and the exact solution as a reference point. The results of using various algorithms to solve (9.1) at $\lambda = 1000$ are also shown. Below we give the details of these algorithms.

- Proximal gradient algorithms cannot be used directly to solve the primal problem (9.1) (note that evaluating the proximal operator is the same as solving the problem itself). However, proximal gradient descent can be applied to the dual of (9.1). Abbreviating $D = D^{(k+1)}$, the dual problem can be expressed as (e.g., see Tibshirani and Taylor [213])

$$\hat{u} = \underset{u \in \mathbb{R}^{n-k-1}}{\arg\min} \; \|y - D^T u\|_2^2 \;\; \text{subject to} \;\; \|u\|_\infty \le \lambda. \tag{9.20}$$

The primal and dual solutions are related by $\hat{\beta} = y - D^T \hat{u}$. We ran proximal gradient and accelerated proximal gradient descent on (9.20), and computed primal solutions accordingly. Each iteration here is very efficient and requires $O(n)$ operations, as computation of the gradient involves one multiplication by D and one by D^T, which takes linear time since these matrices are banded, and the proximal operator is simply coordinate-wise truncation (projection onto an ℓ_∞ ball). The step sizes for each algorithm were hand-selected to be the largest values for which the algorithms still converged; this was intended to give the algorithms the best possible performance. The top left panel of Figure 9.1 shows the results after 10,000 iterations of proximal gradient its accelerated version on the dual (9.20). The fitted curves are wiggly and not piecewise linear, even after such an unreasonably large number of iterations, and even with acceleration (though acceleration clearly provides an improvement).

- The trend filtering problem in (9.1) can alternatively be written in lasso form,

$$\hat{\theta} = \underset{\theta \in \mathbb{R}^n}{\arg\min} \; \frac{1}{2}\|y - H\theta\|_2^2 + \lambda \cdot k! \sum_{j=k+2}^{n} |\theta_j|, \tag{9.21}$$

  where $H = H^{(k)} \in \mathbb{R}^{n \times n}$ is kth order falling factorial basis matrix, defined over $x_1, \ldots x_n$, which, recall, we assume are $1, \ldots n$. The matrix H is effectively the inverse of D [212], and the solutions of (9.1) and (9.21) obey $\hat{\beta} = H\hat{\theta}$. The lasso problem (9.21) provides us with another avenue for proximal gradient descent. Indeed the iterations of proximal gradient descent on (9.21) are very efficient and can still be done in $O(n)$ time: the gradient computation requires one multiplication by H and H^T, which can be applied in linear time, despite the fact that these matrices are dense [227], and the proximal map is coordinate-wise soft-thresholding. After 10,000 iterations, as we can see from the top right panel of Figure 9.1, this method still gives an unsatisfactory fit, and the same is true for 10,000 iterations with acceleration (the output here is close, but it is not piecewise linear, having rounded corners).

- The bottom left panel in the figure explores two commonly used non-first-order methods, namely, coordinate descent applied to the lasso formulation (9.21), and a standard ADMM approach on the original formulation (9.1). The standard ADMM algorithm is described in Section 9.2, and has $O(n)$ per iteration complexity. As far as we can tell, coordinate descent requires $O(n^2)$ operations per iteration (one iteration being a full cycle of coordinate-wise minimizations), because the update rules involve multiplication by individual columns of H, and not H in its entirety. The plot shows the results of these two algorithms after 5000 iterations each. After such a large number of iterations, the standard ADMM result is fairly close to the exact solution in some parts of the domain, but overall fails to capture the piecewise linear structure. Coordinate descent, on the other hand, is quite far off (although we note that it does deliver a visually perfect piecewise linear fit after nearly 100,000 iterations).

- The bottom right panel in the figure justifies the perusal of this chapter, and should generate excitement in the curious reader. It illustrates that after just *20 iterations*, both the PDIP method of Kim et al. [115], and our special ADMM implementation deliver results that are visually indistinguishable from the exact solution. In fact, after only 5 iterations, the specialized ADMM fit (not shown) is visually passable. Both algorithms use $O(n)$ operations per iteration: the PDIP algorithm is actually applied to the dual problem (9.20), and its iterations reduce to solving linear systems in the banded matrix D; the special ADMM algorithm in described in Section 9.2.

9.6.2 ADMM vs. PDIP for $k = 3$ (piecewise cubic fitting)

For the case $k = 3$ (piecewise cubic fitting), the behavior of PDIP mirrors that in the $k = 2$ case, yet the convergence issues begin to show at problem sizes smaller by an order of magnitude. The specialized ADMM approach is slightly slower to converge, but overall still quite fast and robust. Figure 9.14 supports this point.

9.6.3 Prediction at arbitrary points

Continuing within the nonparametric regression context, an important task to consider is that of function prediction at arbitrary locations in the domain. We discuss how to make such predictions using trend

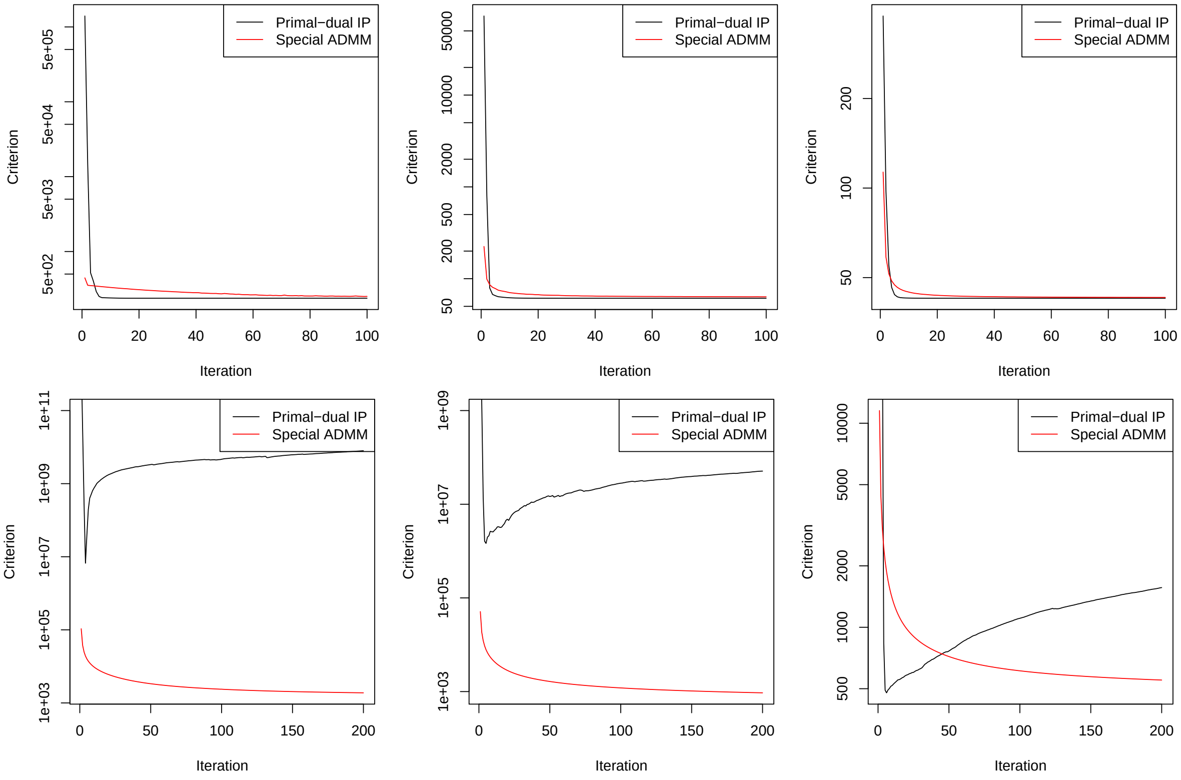


Figure 9.14: *Convergence plots for ($k = 3$): achieved criterion values across iterations of ADMM and PDIP, with the same layout as in Figures 9.5 and 9.8, except that the first row uses $n = 1000$ points, and the second row $n = 10,000$ points. Both algorithms comfortably converge when $n = 1000$. However, PDIP encounters serious difficulties when $n = 10,000$, reminiscent of its behavior for $k = 2$ but when $n = 100,000$ (see Figure 9.8). In all cases, the specialized ADMM algorithm demonstrates a strong and steady convergence behavior.*

filtering. This topic is not directly relevant to our particular algorithmic proposal, but our R software package that implements this algorithm also features the function prediction task, and hence we describe it here for completeness. The trend filtering estimate, as defined in (9.1), produces fitted values $\hat{\beta}_1, \ldots \hat{\beta}_n$ at the given input points $x_1, \ldots x_n$. We may think of these fitted values as the evaluations of an underlying fitted function $\hat{f}$, as in $\big(\hat{f}(x_1), \ldots \hat{f}(x_n)\big) = (\hat{\beta}_1, \ldots \hat{\beta}_n)$. Tibshirani [212], Wang et al. [227] argue that the appropriate extension of $\hat{f}$ to the continuous domain is given by

$$\hat{f}(x) = \sum_{j=1}^{k+1} \hat{\phi}_j \cdot h_j(x) + \sum_{j=1}^{n-k-1} \hat{\theta}_j \cdot h_{k+1+j}(x), \tag{9.22}$$

where $h_1, \ldots h_n$ are the falling factorial basis functions, defined as

$$h_j(x) = \prod_{\ell=1}^{j-1} (x - x_\ell), \quad j = 1, \ldots k+1,$$

$$h_{k+1+j}(x) = \prod_{\ell=1}^{k} (x - x_{j+\ell}) \cdot 1\{x \geq x_{j+k}\}, \quad j = 1, \ldots n-k-1,$$

and $\hat{\phi} \in \mathbb{R}^{k+1}$, $\hat{\theta} \in \mathbb{R}^{n-k-1}$ are inverse coefficients to $\hat{\beta}$. The first $k+1$ coefficients index the polynomial functions $h_1, \ldots h_{k+1}$, and defined by $\hat{\phi}_1 = \hat{\beta}_1$, and

$$\hat{\phi}_j = \frac{1}{(j-1)!} \cdot \left[\mathrm{diag}\left(\frac{1}{x_j - x_1}, \ldots \frac{1}{x_n - x_{n-j+1}} \right) \cdot D^{(x,j-1)} \right]_1 \cdot \hat{\beta}, \quad j = 2, \ldots k+1. \tag{9.23}$$

Above, we use A_1 to denote the first row of a matrix A. Note that $\hat{\phi}_1, \ldots \hat{\phi}_{k+1}$ are generally nonzero at the trend filtering solution $\hat{\beta}$. The last $n-k-1$ coefficients index the knot-producing functions $h_{k+2}, \ldots h_n$, and are defined by

$$\hat{\theta} = D^{(x,k+1)} \hat{\beta} / k!. \tag{9.24}$$

Unlike $\hat{\phi}$, it is apparent that many of $\hat{\theta}_1, \ldots \hat{\theta}_{n-k-1}$ will be zero at the trend filtering solution, more so for large λ. Given a trend filtering estimate $\hat{\beta}$, we can precompute the coefficients $\hat{\phi}, \hat{\theta}$ as in (9.23). Then, to produce evaluations of the underlying estimated function $\hat{f}$ at arbitrary points $x'_1, \ldots x'_m$, we calculate the linear combinations of falling factorial basis functions according to (9.22). From the precomputed coefficients $\hat{\phi}, \hat{\theta}$, this requires only $O(mr)$ operations, where $r = \|D^{(x,k+1)} \hat{\beta}\|_0$, the number of nonzero $(k+1)$st order differences at the solution (we are taking k to be a constant).

Part III

Nonparametric Hypothesis Testing

Chapter 10

Nonparametric testing : Adaptivity of kernel and distance based two sample tests

Nonparametric two sample testing is a decision theoretic problem that involves identifying differences between two random variables without making parametric assumptions about their underlying distributions. We refer to the most common settings as mean difference alternatives (MDA), for testing differences only in first moments, and general difference alternatives (GDA), which is about testing for any difference in distributions. A large number of test statistics have been proposed for both these settings. This paper connects three classes of statistics - high dimensional variants of Hotelling's t-test, statistics based on Reproducing Kernel Hilbert Spaces, and energy statistics based on pairwise distances. We ask the following question - *how much statistical power do popular kernel and distance based tests for GDA have, compared against specialized tests for MDA, when the unknown distributions do actually differ in their means?*

To answer this, we characterize the power of popular tests for GDA like the Maximum Mean Discrepancy with the Gaussian kernel (gMMD) and bandwidth-dependent variants of the Energy Distance with the Euclidean norm (eED) in the high-dimensional MDA regime. We prove several interesting properties relating these classes of tests under MDA, which include

(a) eED and gMMD have asymptotically equal power; furthermore they also enjoy a free lunch because, while they are additionally consistent for GDA, they have the same power as specialized high-dimensional t-tests for MDA. All these tests are asymptotically optimal (including matching constants) for MDA under spherical covariances, according to simple lower bounds.

(b) The power of gMMD is independent of the kernel bandwidth, as long as it is larger than the choice made by the median heuristic.

(c) There is a clear and smooth computation-statistics tradeoff for linear-time, subquadratic-time and quadratic-time versions of these tests, with more computation resulting in higher power.

All three observations are practically important, since point (a) implies that eED and gMMD while being consistent against all alternatives, are also automatically adaptive to simpler alternatives, point (b) suggests that the median "heuristic" has some theoretical justification for being a default bandwidth choice, and point (c) implies that expending more computation may yield direct statistical benefit by orders of magnitude.

10.1 Introduction

Nonparametric two sample testing (or homogeneity testing) deals with detecting differences between two distributions, given samples from both, without making any parametric distributional assumptions. More formally, given samples $X_1, ..., X_n \sim P$ and $Y_1, ..., Y_m \sim Q$, where P and Q are distributions in $\mathbb{R}^d$, the most common types of two sample tests involve testing for the following sets of null and alternate hypotheses

$$\begin{aligned} \text{General difference alternatives } \textbf{(GDA)}: &\quad H_0 : P = Q \qquad\qquad \text{vs} \quad H_1 : P \neq Q, \\ \text{Mean difference alternatives } \textbf{(MDA)}: &\quad H_0 : \mu_P = \mu_Q \qquad\qquad \text{vs} \quad H_1 : \mu_P \neq \mu_Q \end{aligned}$$

where $\mu_P := \mathbb{E}_P X, \mu_Q := \mathbb{E}_Q Y$. This problem has a sustained interest in both the statistics and machine learning literature, due to applications where the sample size might be limited compared to dimensionality, due to experimental or computational costs. For example, it can be used to answer questions in medicine (*is there a difference between pill and placebo?*) and neuroscience (*does a particular brain region respond differently to two different kinds of stimuli?*).

We will assume $m = n$ for simplicity, though our results may be extended to the case when $m/(n+m)$ converges to any constant $k \in (0, 1)$. A test η is a function from $X_1, ...X_n, Y_1, ..., Y_n$ to $\{0, 1\}$, where we reject H_0 when $\eta = 1$. We will only consider tests that have an asymptotic type-I error of at most α. Let us call the set of all such tests as

$$[\eta]_{n,d,\alpha} := \{\eta : \mathbb{R}^{n\times d} \times \mathbb{R}^{n\times d} \to \{0, 1\}, \mathbb{E}_{H_0}\eta \leq \alpha + o(1)\}. \tag{10.1}$$

In the Neyman-Pearson paradigm for the fixed d setting, a test is judged by its power $\phi = \phi(n, P, Q, \alpha) = \mathbb{E}_{H_1}\eta$, and we say that such a test $\eta \in [\eta]_{n,d,\alpha}$ is consistent in the fixed d setting when

$$\mathbb{E}_{H_1}\eta \to 1, \mathbb{E}_{H_0}\eta \leq \alpha \text{ as } n \to \infty \text{ for any fixed } \alpha > 0.$$

In contrast, we say that a test $\eta \in [\eta]_{n,d,\alpha}$ is consistent in the high-dimensional setting when its power $\phi = \phi(n, d_n, P_n, Q_n, \alpha) = \mathbb{E}_{H_1}\eta$ satisfies

$$\mathbb{E}_{H_1}\eta \to 1, \mathbb{E}_{H_0} \leq \alpha \text{ as } (n, d) \to \infty, \text{ for any fixed } \alpha > 0$$

where one also needs to specify the relative rate at which n, d can increase. The central question being considered in this paper is *"what is the power of tests designed for GDA, compared to those designed for MDA, when the distributions truly differ in their means?"*. We will explain this and other related questions in more detail in Section 10.3.

Remark 1. The tests considered in this paper have some common properties. All the test statistics T are centered under the null, i.e. $\mathbb{E}_{H_0}T = 0$, dividing the statistic by $\sqrt{var(T)}$ leads to an asymptotically standard normal statistic under the null, i.e. $T/\sqrt{var(T)} \rightsquigarrow N(0, 1)$ under H_0, where $\rightsquigarrow$ represents convergence in distribution as $n \to \infty$, and hence all tests are of the form:

$$\eta(X_1, ..., X_n, Y_1, ..., Y_n) = \mathbb{I}\left(\frac{T}{\sqrt{var(T)}} > z_\alpha\right)$$

where z_α is the $1 - \alpha$ quantile of the standard normal distribution.

Two-sample testing is a fundamental decision-theoretic problem, having a long history in statistics - for example, the past century has seen a wide adoption of the t-statistic by Hotelling [102] to decide if two samples have different population means (MDA). It was introduced in the parametric setting for univariate Gaussians, but it has been generalized to multivariate non-Gaussian settings as well. If $\bar{X}, \bar{Y}$ are the sample means, and S is a joint sample covariance matrix, then a statistician using the multivariate t-test calculates

$$T_H := (\bar{X} - \bar{Y})^T S^{-1} (\bar{X} - \bar{Y})$$

and the test is $\mathbb{I}(T_H / \sqrt{Var(T_H)} > t_\alpha)$ where t_α is chosen so that $\mathbb{E}_{H_0}\eta \leq \alpha + o(1))$. T_H is consistent for MDA whenever P, Q have different means, and further, it is known to be the "uniformly most powerful" test when P, Q are univariate Gaussians under fairly general assumptions [5, 112, 181, 191].

In a seminal paper by [13], the authors proved that T_H has asymptotic power tending to α in this high-dimensional setting (as discussed in the next section), motivating the study of alternative test statistics. Despite their increasing popularity and usage, many interesting questions remain unanswered, as will be discussed in Section 10.3 and partially answered in this paper. This paper deals with (moderately) high-dimensional and nonparametric two-sample testing, where d can grow polynomially with n, and there are no explicit parametric assumptions on P, Q. In Section 10.7, we experimentally validate our claims for a variety of distributions, even at quite small sample sizes and dimensions. This shows that the asymptotics accurately describe even finite sample behavior of these tests.

Paper Outline. The rest of this paper is organized as follows. In Section 10.2, we introduce three classes of tests in the literature - Hotelling-based tests for MDA, and kernel-based and distance-based tests for GDA, and we discuss related open questions in Section 10.3. In Section 10.4, we prove that three of the most popular tests (one from each class) have the same asymptotic power for MDA, showing the free adaptivity of GDA-based tests for the simpler MDA problem. In Section 10.5, we show that all these classes of tests are optimal for MDA under the diagonal covariance setting, by adapting a lower bound from the normal means problem. Section 10.6 discusses computation-statistics tradeoffs, where we compare the power of linear-time, sub-quadratic time and quadratic-time versions of these tests. In Section 10.7, we run experiments and discuss some practical implications of this work. We end with the proofs in Section 10.8.

Notation We use the standard o, o_P, O_P notation extensively. Also, for two non-random sequences A_n, B_n, $A_n = \Omega(B_n)$ is the negation of $A_n = o(B_n)$, $A_n = \omega(B_n)$ is the negation of $A_n = O(B_n)$, and $A_n \asymp B_n$ to mean $A_n = B_n(c + o(1))$ for some absolute constant c. $Tr()$ is the trace of a (square) matrix and $Tr^k()$ is the k-th power of the trace. $\circ$ is the elementwise or Hadamard product, $Ts()$ refers to the total sum of all the elements of a matrix, e_i is the i-th standard basis vector, 1 is the vector of ones. $\rightsquigarrow$ is convergence in distribution, and $\mathbb{I}(\cdot)$ is a 0-1 indicator function.

10.2 Hotelling-based MDA Tests and Kernel/Distance-based GDA tests

Tests for MDA. As mentioned in the introduction, [13] prove that Hotelling's T_H has power tending to α (this is called trivial power), when $(n, d) \to \infty$ with $d/n \to 1 - \epsilon$ for small ϵ, explained by the inherent difficulty of accurately estimating the $O(d^2)$ parameters of Σ^{-1} with very few samples (S^{-1} is not even defined if $d > n$ and is badly conditioned if d is of similar order as n). To avoid this problem, they proposed to use the test statistic

$$T_{BS} := \|\bar{X} - \bar{Y}\|^2 - \mathrm{tr}(S)/n$$

and showed that it has non-trivial power whenever $d/n \to c \in (0, \infty)$. An important precursor to this nonparametric work of [13] is that of [55] who proposed a high-dimensional t-test for Gaussians. [200] and [201] proposed to instead use $\mathrm{diag}(S)^{-1}$ instead of S^{-1}, in T_H, and showed its advantages in certain settings over T_{BS} (specifically its scale invariance, i.e. invariance when the data is rescaled by a diagonal matrix, gives it an advantage when the covariance matrices are diagonal but non-spherical).

In another extension of T_{BS} by [36], henceforth called CQ, the authors proposed a variant of T_{BS} of the form

$$T_{CQ} \;:=\; \frac{1}{n(n-1)}\sum_{i\neq j=1}^{n} X_i^T X_j + \frac{1}{n(n-1)}\sum_{i\neq j=1}^{n} Y_i^T Y_j - \frac{2}{n^2}\sum_{i,j=1}^{n} X_i^T Y_j,$$

analyzing its power for MDA when the covariances of X, Y are also unequal and without explicit restrictions on d, n, but rather in terms of conditions stated in terms of n, Σ and mean difference $\delta := \mu_P - \mu_Q$. We will return to these conditions later in this paper, since we will use assumptions of similar flavor.

Note that $\mathbb{E}[T_{CQ}] = \mu_P^T\mu_P + \mu_Q^T\mu_Q - 2\mu_P^T\mu_Q = \|\mu_P - \mu_Q\|^2$, and hence T_{CQ} is an unbiased estimator of $\|\mu_P - \mu_Q\|^2$. In this paper, instead of using T_{CQ} directly, we will analyze a minor variant, which is a U-statistic:

$$\begin{aligned} U_{CQ} \;&:=\; \frac{1}{n(n-1)}\sum_{i\neq j=1}^{n} h_{CQ}(X_i, X_j, Y_i, Y_j) \\ \text{where } h_{CQ}(X, X', Y, Y') \;&:=\; X^T X + Y^T Y - X^T Y' - X'^T Y. \end{aligned} \tag{10.2}$$

T_{CQ}'s difference from U_{CQ} is only in the third term, and this difference is asymptotically vanishing, making the asymptotic properties of U_{CQ} (especially its power) identical to T_{CQ}, and its usage is only for technical convenience.

There is also a large literature on the so-called parametric Behrens-Fisher problem, which is a parametric MDA problem where the distributions are Gaussian and heteroskedastic, and also the nonparametric Behrens-Fisher problem that deals with MDA when P, Q are nonparametric mean-scale families, in the univariate and multivariate settings. See [17] and [131] for recent such works, and references therein. Another related line of work analyzes the setting where p could be exponentially larger than n but assuming some kind of sparsity (say in the mean difference); see [30] for such an example.

Tests for GDA. It is well known that the Kolmogorov-Smirnov (KS) test by [116] and [194] involves differences in empirical CDFs. The KS test, the related Cramer von-Mises criterion by [43] and [222], and Anderson-Darling test by [6] are very popular in one dimension, but their usage has been more restricted in higher dimensions. This is mostly due to the curse of dimensionality involved with estimating multivariate empirical CDFs. While there has been work on generalizing these popular one-dimensional to higher dimensions, like [19], these are seemingly not the most common multivariate tests. Some other examples of univariate tests include rank based tests as covered by the book [124] and the runs test by [225], while some interesting multivariate tests include spanning tree methods by [74], nearest-neighbor based tests by [183] and [96], and the "cross-match" tests by [175]. Most of these have been proved to be consistent in the fixed d setting, but not much is known about their power in the high-dimensional setting.

One popular class of tests for the multivariate GDA problem that has emerged over the last decade, are kernel-based tests introduced in parallel by [70] and [85], and expanded on in [82]. The *Maximum Mean Discrepancy* between P, Q is defined as

$$\mathrm{MMD}(H_\kappa, P, Q) := \max_{\|f\|_{H_\kappa}\leq 1} \mathbb{E}_P f(x) - \mathbb{E}_Q f(y)$$

where H_κ is a Reproducing Kernel Hilbert Space associated with Mercer kernel $k(\cdot,\cdot)$, and $\{f : \|f\|_{H_\kappa} \leq 1\}$ is its unit norm ball. It is easy to see that $\mathrm{MMD} \geq 0$, and also that $P = Q$ implies $\mathrm{MMD} = 0$. For the converse, [85] show that under fairly general conditions involving H_κ or equivalently κ, the equality holds iff $P = Q$. The authors prove that

$$\mathrm{MMD}(H_\kappa, P, Q) = \|\mathbb{E}_P \kappa(x, .) - \mathbb{E}_Q \kappa(y, .)\|_{H_\kappa}.$$

This gives rise to a natural associated test, that involves thresholding the following U-statistic, an unbiased estimator of MMD^2:

$$\begin{aligned} \mathrm{MMD}_u^2(k(\cdot,\cdot)) &:= \frac{1}{n(n-1)} \sum_{i \neq j}^{n} h_\kappa(X_i, X_j, Y_i, Y_j) \\ \text{where } h_\kappa(X, X', Y, Y') &:= \kappa(X, X') + \kappa(Y, Y') - \kappa(X, Y') - \kappa(X', Y). \end{aligned} \tag{10.3}$$

Note once again that we can form a gMMD statistic having 3 summations like T_{CQ}, but for technical convenience we mimic the form of the U-statistic U_{CQ}, the asymptotic properties of both being the same. Note that U_{CQ} is just the MMD when we use the linear kernel $k(a, b) = a^T b$. The most popular kernel for GDA is the Gaussian kernel with bandwidth parameter γ, leading to the test statistic that we henceforth call gMMD:

$$\begin{aligned} \mathrm{gMMD}_\gamma^2 &:= \mathrm{MMD}_u^2(g_\gamma(\cdot,\cdot)) \\ \text{where } \; g_\gamma(a, b) &:= \exp\left(-\frac{\|a-b\|_2^2}{\gamma^2}\right). \end{aligned}$$

Apart from the fact that the population $\mathrm{gMMD}^2(P, Q) = 0$ iff $P = Q$ the other fact that makes this a useful test statistic is that its estimation error, i.e. the error of MMD_u^2 in estimating MMD^2, scales like $1/\sqrt{n}$, independent of d; see [82] for a detailed proof of this fact. This is unlike the KL divergence, for example, which is 0 iff $P = Q$ but is hard to estimate in high-dimensions. However, it was recently argued in [167] that the study of estimation error covers only one side of the story, and that test power still degrades with d even if estimation error does not.

A related but different class of tests are distance-based "energy statistics" as introduced in parallel by [16] and [207], and generalized to some kinds of metrics, denoted ρ, for a related independence testing problem, by [133]. The test statistic is called the *Cramer statistic* by the former paper but we use the term *Energy Distance* as done by the latter, and once more, we study the U-statistic form:

$$\begin{aligned} \mathrm{ED}_u(\rho(\cdot,\cdot)) &:= \frac{1}{n(n-1)} \sum_{i \neq j}^{n} h_\rho(X_i, X_j, Y_i, Y_j) \\ \text{where } h_\rho(X, X', Y, Y') &:= \rho(X, Y') + \rho(X', Y) - \rho(X, X') - \rho(Y, Y'). \end{aligned} \tag{10.4}$$

The most popular or "default" choice within this class (the only one studied by both sets of authors who introduced it) is the Energy Distance with the Euclidean distance, henceforth called eED, defined as

$$\begin{aligned} \mathrm{eED}_u &:= \mathrm{ED}_u(e(\cdot,\cdot)) \\ \text{where } \; e(a, b) &:= \|a - b\|_2. \end{aligned}$$

Appropriately thresholding gMMD_u^2 and eED_u leads to tests that are consistent for GDA in the fixed d setting against all fixed alternatives where $P \neq Q$ (and some local alternatives, i.e. alternatives that change with n) under fairly general conditions and such results can be found in the associated references. However not much is known about them in the high dimensional regime.

Remark 2. This paper will deal largely with gMMD and eED, because these are the most popular choices for kernel and distance used in practice, but similar inferences can possibly be made about other kernels and distances, using the same proof technique. Similarly, we will focus on U_{CQ}, though one may draw similar inferences about T_{BS} and T_{SD} and their corresponding GDA variants.

10.3 Open Questions and Summary of Results

The test statistics for MDA, like $U_{CQ}, T_{BS}, T_{SD}, T_H$ have all been analysed in the high-dimensional setting. However, there is presently poor understanding of gMMD and eED in high dimensions. Below we list some of these open questions (along with explanations) that we are going to answer in this paper, followed by our partial answers to these questions.

Q1. How can one characterize the power of nonparametric tests like gMMD and/or eED in high dimensions, either for GDA or MDA?

Explanation [Q1]. In the fixed d setting, gMMD and eED are well understood, and their null and alternate distributions are given in [82] and [207] respectively. However, their behavior in high dimensions seems to be essentially unanswered in the current literature. A general characterization of power is impossible since P, Q could be different yet arbitrarily similar to each other (see Section 3.2 of [82] for a formal statement and proof of this claim). Due to this reason, one is somewhat restricted to trying to characterize the power in limited settings. For example, one can hope to characterize the power by parameterizing the problem in terms of the smallest moment in which P, Q differ.

Result [Q1]. One way that we propose to analyze them is to consider two nonparametric distributions P, Q that *only* differ in one specific moment and see how much power gMMD or eED have to identify this difference and reject the null. As a first step, this paper will characterize their power for MDA, when P, Q differ only in their first moment.

Q2. How does the choice of bandwidth parameter γ affect power of gMMD_u^2, for GDA or MDA?

Explanation [Q2]. The most popular choice of bandwidth is the "median heuristic" where it is chosen as the median Euclidean distance between all pairs of points (see [184]). However, the effect of this choice on test power is unclear. [83] also make suggestions for choosing the bandwidth parameter, but only for the linear-time gMMD_l^2 (see Section 10.6), and also with guarantees only in the fixed d setting. Hence the study of how the kernel bandwidth affects power is a work in progress in the current literature. For any fixed γ, consistency for GDA was proved in [85]; further, the power of gMMD_u^2 against any fixed GDA alternative was also explicitly derived in the fixed d setting to be $\Phi(\sqrt{n})$, ignoring constants, where Φ is the Gaussian CDF. Notice that consistency of the gMMD test for any *fixed* γ is in stark contrast to using Gaussian kernels for density estimation, where we must let the bandwidth go to zero with increasing n, and hence the gMMD statistic does not behave in the same way as the L2-distance between kernel density estimates, as done in [4].

Result [Q2]. In Section 10.4, we prove that the power of gMMD_u^2 does not depend on the bandwidth parameter γ, as long as γ is chosen to be asymptotically larger than the choice made by the aforementioned median heuristic.

Q3. Can one directly compare the power of eED and gMMD for GDA or MDA? Is one of them more powerful than the other?

Explanation [Q3]. [187] describes connections between kernel and distance based tests for independence testing. Informally speaking, there is a near one-to-one correspondence between the class of kernels and distances for which such tests make sense. However, while there is *some* metric/semimetric that corresponding to Gaussian kernel g, that metric/semimetric is not the Euclidean distance e (and vice versa). eED seems to be more popular in the statistics literature, and gMMD in machine learning - it is of practical importance to both fields to know how one should choose between eED and gMMD.

Result [Q3]. In Section 10.4, we show that (under fairly general conditions) gMMD and eED have asymptotically equal power for MDA, both in theory and practice.

Q4. How do the powers of tests for GDA compare to tests for MDA, when (unknown to us) P, Q actually differ in only their means?

Explanation [Q4]. Given a nonparametric two-sample testing problem, one generally does not know if the distributions differed in their means or not. If they did differ in their means, presumably the former statistics may perform worse than the latter, since the latter are designed specifically for that purpose, and can concentrate all their power in detecting first moment differences. But how much worse? What is the price one must pay for the extra generality of gMMD and eED? One of the main questions considered in this paper is actually one of comparing the powers of $\mathrm{eED}, \mathrm{gMMD}$ and U_{CQ}.

Result [Q4]. In Section 10.4, we prove that one does not pay any price for the generality of $\mathrm{gMMD}_u^2, \mathrm{eED}_u$ (they enjoy a "free lunch") - gMMD_u^2 and eED_u have the same power as U_{CQ} against MDA in high dimensions, both in theory and practice, even though gMMD_u^2 and eED_u are also consistent against GDA whereas U_{CQ} is not. We would like to note that this result has actually been observed in practice, but seemingly not been explicitly acknowledged or conjectured. Figures 1 and 4 of [16] are quite convincing for eED, and the authors explicitly point this out in their experiments and conclusion sections, while Figures 3 and 4 of [131] also show same phenomenon for gMMD, though the latter authors do not comment on their experimental observation. As far as we know, this paper has the first rigorous justification of such a phenomenon.

Q5. How does computation affect power in high dimensions?

Explanation [Q5]. A final question we consider is the relationship between computation and power. Noting that gMMD_u^2 takes quadratic time i.e. $O(n^2)$ to compute, [82] and [233] introduce linear-time and block-based subquadratic-time statistics gMMD_l^2 and gMMD_b^2. The main related work in this regard is [168], which analyses a linear-time version of gMMD_l^2 in the high-dimensional setting. We will discuss this last question in detail in Section 10.6.

Result [Q5]. In Section 10.6, we show that expending more computation yields a direct statistical benefit of higher power; there is clear and smooth statistics-computation tradeoff for a family of earlier proposed sub-quadratic and linear time (kernel) two sample tests.

Q6. What are the lower bounds for two sample testing in high dimensions?

Explanation [Q6]. We have not seen any lower bounds for the two sample testing problem in the literature, and definitely none for the high dimensional setting, even under MDA.

Result [Q6]. In Section 10.5, we prove tight lower bounds for two-sample testing under MDA, for the case of diagonal covariance, which show that all three tests are optimal in this setting, even including constants.

10.4 Adaptivity of gMMD and eED to MDA

This section will aim to provide some answers to questions Q1-4. Our main assumptions are inspired by those in [13] and [36], and related followup papers.

[A1] Model. $X_i = \Gamma Z_{1i} + \mu_P$ and $Y_i = \Gamma Z_{2i} + \mu_Q$ for $i = 1, ..., n$ where Z_{1i}, Z_{2i} are k-dimensional independent zero mean, identity covariance random variables and Γ is a $d \times D$ unknown full-rank deterministic transformation matrix for some $D \geq d$ satisfying $\Gamma\Gamma' = \Sigma$ (hence the $d \times d$ population covariance Σ is full-rank). Denote the mean difference as $\delta := \mu_P - \mu_Q$.

Remark 3. Assumption [A1] implies that X, Y have means μ_1, μ_2 and covariances Σ, like in [13]. We do not assume that X, Y have different covariances Σ_1, Σ_2 like in [36]. The reason for this choice is as follows. gMMD and eED can detect differences in distributions P, Q that occur in any finite moment. For example, by Bochner's theorem (see [178]), the population quantity gMMD^2 is precisely (up to constants)

$$\int_{\mathbb{R}^d} |\varphi_X(t) - \varphi_Y(t)|^2 e^{-\gamma^2 \|t\|^2} \mathrm{d}t$$

where $\varphi_X(t) = \mathbb{E}_{x \sim P}[e^{-it^T x}]$ is the characteristic function of X at frequency t (similarly $\varphi_Y(t)$), and the population eED is precisely (up to constants)

$$\int\limits_{(a,t) \in S^{d-1} \times \mathbb{R}} [F_X(a,t) - F_Y(a,t)]^2 \mathrm{d}a \, \mathrm{d}t$$

where $F_X(a,t) = P(a^T X \leq t)$ (similarly $F_Y(a,t)$) is the population CDF of X when projected along direction a and S^{d-1} is the surface of the d dimensional unit sphere; see [207] for a proof. Because of this, gMMD and eED are sensitive to differences in second (and higher) moments of distributions. To analyze their power against MDA, it makes sense to nullify all other sources of signal like $\|\Sigma_1 - \Sigma_2\|_F^2$ that might alter the power of gMMD or eED.

[A2] Moment assumption. Each of the D coordinates of Z_{1i} and Z_{2i} have $m \geq 8$ moments, each moment being a finite constant. For all $i = 1, ..., n$ and $s = 1, 2$, we have $\mathbb{E}(Z_{si1}^{\alpha_1} Z_{si2}^{\alpha_2}, ..., Z_{siD}^{\alpha_D}) = \mathbb{E}(Z_{si1}^{\alpha_1})\mathbb{E}(Z_{si2}^{\alpha_2}) ... \mathbb{E}(Z_{siD}^{\alpha_D})$ for all $\sum_{j=1}^D \alpha_j \leq 8$.

Remark 4. Assumption [A2] was made in essentially the same form in [13] and [36]. Some of our calculations explicitly involve how much these moments deviate from those of a standard Gaussian. We show in Section 10.7 that many of our results hold experimentally for a variety of non-Gaussian distributions.

[A3] Fairly good conditioning of Σ. (a) We assume that $Tr(\Sigma^{2k}) = o(Tr^2(\Sigma^k))$ for $k = 1, 2$. (b) We also assume that $Tr(\Sigma) \asymp d$ and for $S_i \in \{X_i, Y_i\}$, the average $\|S_i - S_j\|^2/d$ exponentially concentrates around its expectation, i.e.

$$P\left(\left|\frac{\|S_i - S_j\|^2}{d} - \frac{\mathbb{E}\|S_i - S_j\|^2}{d}\right| > d^{-\nu}\right) \to 0 \text{ exponentially fast in (some polynomial of) } d.$$

for some $\nu = \nu(\Sigma, m) \in (1/3, 1/2]$.

Remark 5. Assumption [A3] essentially means that Σ is fairly well conditioned, and was also made in the aforementioned earlier works. To see this, note that if $\Sigma = \sigma^2 I$ then the conditions reduce to requiring $d = o(d^2)$. If all the eigenvalues of Σ are bounded, this assumption is still met. When Σ's eigenvalues are not bounded, this condition will be satisfied as long as Σ is not terribly conditioned. This assumption is discussed in detail with several nontrivial examples in [36]. Similarly, $\nu(\Sigma, m)$ reflects the conditioning of Σ, and the number m of moments of S. In the best case, with d independent coordinates i.e. identity covariance $\Sigma = I$ and infinite moments, $\nu(\Sigma, m) = 1/2$. As we assume fewer moments or as we deviate away from diagonal covariance to more ill-conditioned matrices, $\nu(\Sigma, m)$ strays away from half, but we assume it is fairly well-conditioned, being at least $1/3$. We think that some such good conditioning is necessary for our theorems to hold, but that the scalar $1/3$ can be lowered.

[A4] Low signal strength. $\|\delta\|^2 = o\left(\min\left\{\frac{Tr^2(\Sigma)}{Tr(\Sigma^2)}\lambda_{\min}(\Sigma), \frac{Tr(\Sigma)}{d^\nu}\right\}\right)$ and $\delta^T\Sigma^k\delta = o(Tr(\Sigma^{k+1}))$ for $k = 0, 1, 2, 3$.

Remark 6. First recall that we assumed Σ is full rank in Assumption [A1], so $\lambda_{\min}(\Sigma) > 0$. Assumption [A4] essentially means that the signal strength is not very *large* relative to the noise. For example, when $\Sigma = \sigma^2 I$, the assumption requires that $\|\delta\|^2/\sigma^2 = o(\sqrt{d})$. Indeed, it more generally implies that $\|\delta\|^2 = o(Tr(\Sigma))$[1]. We need this assumption for technical reasons, and we conjecture that our results hold under a weaker assumption. Even in its present form, this is not such a strong assumption since (as we shall see in the theorem statements) if the signal strength is large then the decision problem becomes too easy and such a regime is rather uninteresting. Further note that $\delta^T\delta = o(Tr(\Sigma))$ implies, by Cauchy-Schwarz,

$$\begin{aligned}
\delta^T\Sigma\delta &\leq \lambda_{\max}(\Sigma)\|\delta\|^2 = o(\lambda_{\max}(\Sigma)Tr(\Sigma)),\\
\delta^T\Sigma^2\delta &\leq Tr(\Sigma^2)\|\delta\|^2 = o(Tr(\Sigma^2)Tr(\Sigma)),\\
\delta^T\Sigma^3\delta &= o(Tr(\Sigma^3)Tr(\Sigma)) \leq o(Tr(\Sigma^2)Tr^2(\Sigma)).
\end{aligned}$$

[A5] High-dimensional setting. $n = o(d^{3\nu-1}Tr(\Sigma^2)) = o(\sqrt{d}Tr^2(\Sigma)) = o(d^{2.5})$.

Remark 7. Currently, Assumption [A5] is needed only for a technicality in proving our main theorem, and we conjecture that it can be relaxed.

As in [36], we do not assume that $(n, d) \to \infty$ at any particular rate. Instead, we will analyze their behavior in two regimes that have implicit control on n, d. For notational convenience, denote

$$\sigma_{n1}^2 := 8\frac{Tr(\Sigma^2)}{n^2}, \tag{10.5}$$

$$\sigma_{n2}^2 := 8\frac{\delta^T\Sigma\delta}{n}. \tag{10.6}$$

Recalling that $\delta := \mu_P - \mu_Q$, the first theorem summarizes the power of U_{CQ}.

Theorem 24. *Under [A1], [A2] and [A3a], U_{CQ} has asymptotic power which equals*

$$\phi_{CQ} = \Phi\left(-\frac{\sqrt{\frac{Tr(\Sigma^2)}{n^2}}}{\sqrt{\frac{Tr(\Sigma^2)}{n^2} + \frac{\delta^T\Sigma\delta}{n}}}\cdot z_\alpha + \frac{\|\delta\|^2}{\sqrt{8\frac{Tr(\Sigma^2)}{n^2} + 8\frac{\delta^T\Sigma\delta}{n}}}\right) + o(1) \tag{10.7}$$

where Φ is the Gaussian CDF and z_α is the threshold representing the α-quantile of the standard Gaussian distribution.

This theorem follows from the main result of [36] for U_{CQ}, and hence we do not reproduce it here. There, the authors prove that U_{CQ} is asymptotically normally distributed with variance $\sigma_{n1}^2 + \sigma_{n2}^2$ under

[1] This holds because $Tr(\Sigma) = Tr(\Sigma^2\Sigma^{-1}) \leq Tr(\Sigma^2)\lambda_{\min}^{-1}(\Sigma)$ by Cauchy-Schwarz inequality that $Tr(A^TB) \leq \|A\|_*\|B\|_{op}$ where $*$, op refer to the nuclear and operator norms respectively.

the alternative, and variance σ_{n1}^2 under the null (with $\Sigma_1 = \Sigma_2 = \Sigma$ and $n_1 = n_2 = n$ being used by us). This then gives rise to the above expression for the power ϕ fairly easily, except that the authors made a small mistake by interchanging σ_{n1} and σ_{n2} in one crucial expression (confirmed by email correspondence with the authors, summarized in the Appendix Sec. 10.8.2). Another minor difference is that we write down the power as a single expression, while [36] prefer to write them down in the two aforementioned special cases of low and high SNR.

Remark 8. The null distribution of U_{CQ} is asymptotically Gaussian under MDA in this high-dimensional setting. This is in stark contrast to the fixed-d, increasing-n setting, where the null distribution is an infinite sum of weighted chi-squared distributions, due to the properties of degenerate U-statistics (see [188]). This seems to have first been proved by [13] for T_{BS} using a martingale central limit theorem (see [89]).

The next theorem summarizes the power of gMMD, which is also one of the main results of the paper.

Theorem 25. *Assume [A1], [A2], [A3], [A4] and [A5], and let the bandwidth be chosen as* $\gamma^2 = \omega(2Tr(\Sigma))$. *Then* gMMD_γ *has asymptotic power which is independent of* γ, *and equals the power of* U_{CQ}. *In other words, the power is*

$$\phi_{\mathrm{gMMD}} = \Phi\left(- \frac{\sqrt{\frac{Tr(\Sigma^2)}{n^2}}}{\sqrt{\frac{Tr(\Sigma^2)}{n^2} + \frac{\delta^T \Sigma \delta}{n}}} \cdot z_\alpha + \frac{\|\delta\|^2}{\sqrt{8\frac{Tr(\Sigma^2)}{n^2} + 8\frac{\delta^T\Sigma\delta}{n}}} \right) + o(1)$$

for all $\gamma^2 = \omega(2Tr(\Sigma))$.

The proof of this theorem is covered in Section 10.8. While one may conjecture a result like the above due to the claims of [61] that the Gaussian kernel often behaves like the linear kernel in high dimensions, their results only hold true when $n \asymp d$ (apart from other differences in assumptions). Further, they also interpret the results rather pessimistically, by saying that these kernels do not provide an advantage in the high-dimensional setting, but we will demonstrate in experiments that when the linear kernel does not suffice (the distributions have the same mean but differ in their variances), then U_{CQ} has trivial power but gMMD's power tends to one in reasonable scenarios. Of course, more samples are probably needed to detect differences in second moments compared to differences in first moments.Hence, we choose to interpret the above result optimistically — *not only is* gMMD *capable of detecting any difference in distributions, but it also detects differences in means as well as* U_{CQ} *which is designed to test only mean differences.*

For the purpose of mathematical analysis, we now introduce a family of statistics, for which eED_u is a special case. These are defined (recalling Eq.(10.4)) as

$$\begin{aligned} \mathrm{eED}_\gamma &:= \mathrm{ED}_u(e_\gamma(\cdot,\cdot)) \\ \text{where } e_\gamma(a,b) &:= \sqrt{\gamma^2 - 2Tr(\Sigma) + \|a-b\|_2^2} \end{aligned}$$

where $\gamma^2 \geq 2Tr(\Sigma)$ is a constant user-chosen bandwidth parameter. Note that

$$\lim_{\gamma^2 \to 2Tr(\Sigma)^+} \mathrm{eED}_\gamma = \mathrm{eED}_u$$

The next theorem summarizes the power of eED_γ, in all cases when $\gamma^2 = \omega(2Tr(\Sigma))$.

Theorem 26. *Assume [A1], [A2], [A3], [A4] and [A5], and let the bandwidth be chosen as* $\gamma^2 = \omega(2Tr(\Sigma))$. *Then* eED_γ *has asymptotic power which is independent of* γ, *and equals the power of* U_{CQ}.

In other words, the power is

$$\phi_{\text{eED}} = \Phi\left(-\frac{\sqrt{\frac{Tr(\Sigma^2)}{n^2}}}{\sqrt{\frac{Tr(\Sigma^2)}{n^2} + \frac{\delta^T\Sigma\delta}{n}}} \cdot z_\alpha + \frac{\|\delta\|^2}{\sqrt{8\frac{Tr(\Sigma^2)}{n^2} + 8\frac{\delta^T\Sigma\delta}{n}}} \right) + o(1)$$

for all $\gamma^2 = \omega(2Tr(\Sigma))$.

The proof of this theorem is similar to the proof of Theorem 25, and hence is briefly covered at the end of Section 10.8, after the proof of Theorem 25.

Remark 9. We remark on our inability to prove the above theorems for the limiting case of $\gamma^2 \asymp 2Tr(\Sigma)$. The proofs of Theorems 25 and 26 are based on a Taylor expansion of the h_κ and h_ρ respectively (recall Eqs.(10.3),(10.4) for their definition). This leads to a "dominant" Taylor term U_2/γ^2 which is a U-statistic in h_2 and a "remainder" term U_4/γ^4 which is a U-statistic in h_4, where

$$h_2(X, X', Y, Y') = \|X - X'\|^2 + \|Y - Y'\|^2 - \|X - Y'\|^2 - \|X' - Y\|^2, \quad (10.8)$$
$$h_4(X, X', Y, Y') = \|X - X'\|^4 + \|Y - Y'\|^4 - \|X - Y'\|^4 - \|X' - Y\|^4. \quad (10.9)$$

One can easily observe that $h_2 = -2h_{CQ}$ (see Eq.(10.2)) and hence the behavior of U_2 is immediately captured by the behavior of U_{CQ}, the most important fact being that U_2 is always Gaussian under the null and the alternative (as mentioned after Theorem 24 and its following remarks). When $\gamma^2 = \omega(Tr(\Sigma))$, we prove that $U_4/\gamma^4 = o_P(U_2/\gamma^2)$. However, when $\gamma^2 \asymp 2Tr(\Sigma)$, our results suggest that $U_4/\gamma^4 = O_P(U_2/\gamma^2)$. However, while we know that U_2/γ^2 is asymptotically Gaussian, we do not know the limiting distribution of U_4/γ^4, even though we undertake tedious calculations to find the mean and variance of U_4. Hence, while this allows us to make arguments about the mean and variance of gMMD and eED, we cannot make power claims since for that purpose we require knowing the limiting distribution of U_4 under the null. While we conjecture that it is indeed Gaussian and simulations support this, the proof is vastly more complicated than for U_2 because the number of terms to be controlled in the martingale central limit theorem is larger (by an order of magnitude, as the number of terms grows exponentially). Proving the above theorem statements for the limiting case is an important direction for future work, and may require development of the theory of U-statistics for high dimensional variables. However, for the moment we show a variety of experiments that support our conjecture, implying that the borderline case is probably a technical limitation.

10.4.1 The Special Case of $\Sigma = \sigma^2 I$

Though no explicit assumptions are placed on n, d for the above expression (and hence for consistency to hold), for further understanding of the power of these tests, let us consider the situation when $\Sigma = \sigma^2 I$ and define the signal-to-noise ratio (SNR) as

$$\text{SNR} \quad \Psi := \frac{\|\delta\|}{\sigma}.$$

One can think of Ψ^2 as the problem-dependent constant, which determines how hard the testing problem is - of course, the larger the SNR, the easier the distributions are to distinguish. Indeed, in the special case of P, Q being spherical Gaussians, Ψ^2 is just the KL-divergence between these distributions. Then, the expression for power from Eq.(10.7) simplifies to

$$\Phi\left(-\frac{\sqrt{d}}{\sqrt{d + n\Psi^2}} z_\alpha + \frac{\Psi^2}{\sqrt{8d/n^2 + 8\Psi^2/n}} \right) + o(1). \quad (10.10)$$

We are most interested in the regimes where Ψ is small. Let us define the three regimes as follows:

$$\begin{aligned} \text{Low SNR:}\ \ \Psi &= o(\sqrt{d/n}), & (10.11)\\ \text{Medium SNR:}\ \ \Psi &\asymp \sqrt{d/n}, & (10.12)\\ \text{High SNR:}\ \ \Psi &= \omega(\sqrt{d/n}). & (10.13) \end{aligned}$$

Remark 10. We find it worthy to note that the behavior is *different*[2] in the low and high SNR regime. Specifically, in the Low SNR regime, the asymptotic power is

$$\phi_L = \Phi\left(-z_\alpha + \frac{n\Psi^2}{\sqrt{8d}}\right) \ \text{ when } \ \Psi = o(\sqrt{d/n}) \tag{10.14}$$

while in the high SNR regime, the asymptotic power is

$$\phi_H = \Phi(\sqrt{n}\Psi/\sqrt{8}) \ \text{ when } \ \Psi = \omega(\sqrt{d/n}). \tag{10.15}$$

The above two rates match in the Medium SNR regime, yielding a power $\asymp \Phi(\sqrt{d})$.

10.5 Lower Bounds when $\Sigma = \sigma^2 I$

Here we show that the form of the power achieved in Theorem 24 is not improvable under certain assumptions. For example, in the case when $\Sigma = \sigma^2 I$, we can provide matching lower bounds to Eq. 10.10 using techniques from [106] designed for Gaussian normal means problem. The proof relies on the Gaussian approximations of the central and noncentral chi-squared distributions.

Proposition 27. *Let $G_d(x,0)$ be the cdf of a central chi-squared distribution with d degrees of freedom and $G_d(x,r)$ be the cdf of a noncentral chi-squared distribution with d degrees of freedom and noncentrality parameter r. Then as $d \to \infty$, we have uniformly over x, r*

$$\begin{aligned} G_d(x,0) &= \Phi\left(\frac{x-d}{\sqrt{2d}}\right) + o(1), & (10.16)\\ G_d(x,r^2) &= \Phi\left(\frac{x-d-r^2}{\sqrt{2d+4r^2}}\right) + o(1), & (10.17)\\ G_d(T_{d\alpha},r^2) &= \Phi\left(\frac{\sqrt{2d}}{\sqrt{2d+4r^2}}z_\alpha - \frac{r^2}{\sqrt{2d+4r^2}}\right) + o(1) & (10.18) \end{aligned}$$

where $T_{d\alpha}$ is $1-\alpha$ quantile cutoff of the χ^2_d and z_α is the corresponding quantile of the standard normal.

Remark 11. Our Eq.(10.18) differs from [106][Ch 1.3, Pg 13, Eq. 1.14] where the authors applied the additional approximation that $d \to \infty$ with r fixed (or just $d >> r$) to get

$$G(T_{d\alpha}, r^2) = \Phi(z_\alpha - \rho^2/\sqrt{2d}) + o(1). \tag{10.19}$$

We do not make this approximation.

[2] There is a mistake/typo in the paper by [36], which causes them to miss this surprising observation. We have confirmed this important typo with the authors, and describe the context of its occurrence in more detail in the Appendix Sec. 10.8.2.

Proof: [Proof of Proposition 27] The first two expressions appear verbatim in [106][Ch 1.3, Pg 12]. Substituting $x = T_{d\alpha}$ into the second expression yields

$$G_d(T_{d\alpha}, r^2) = \Phi\left(\frac{T_{d\alpha} - d}{\sqrt{2d + 4r^2}} - \frac{r^2}{\sqrt{2d + 4r^2}}\right) + o(1)$$

The last expression then follows due to the following fact:

$$\frac{T_{d\alpha} - d}{\sqrt{2d}} = z_\alpha + o(1), \tag{10.20}$$

Eq.(10.20) holds by the following argument. First note that

$$(\chi_d^2 - d)/\sqrt{2d} \rightsquigarrow N(0, 1).$$

Then by definition of $T_{d\alpha}$,

$$P(\chi_d^2 > T_{d\alpha}) \leq \alpha$$

which then implies

$$P\left(Z > \frac{T_{d\alpha} - d}{\sqrt{2d}} + o(1)\right) \leq \alpha$$

for standard normal Z. Since we know that $P(Z > z_\alpha) \leq \alpha$, Eq.(10.20) follows.

Next, define $S_d(\rho) = \{\delta \in \mathbb{R}^d \mid \|\delta\| = \rho\}$ to be the surface of the d-dimensional sphere of radius ρ. For the normal means problem, we are given $Z \sim N(\delta, I_d)$ and we test $H_0 : \delta = 0$ against $H_1 : \delta \in S_d(\rho)$. Recalling the definition of $[\eta]_{n,d,\alpha}$ from Eq.(10.1), we analogously define $[\eta]_{d,\alpha}$ for the normal means problem as the set of all tests from $\mathbb{R}^d \to [0, 1]$ with expected type-1 error at most α. Define the minimax power at level α as

$$\beta(\rho, \alpha) := \inf_{\eta \in [\eta]_{d,\alpha}} \sup_{\delta \in S_d(\rho)} \mathbb{E}_\delta \eta.$$

Proposition 28. *Given $Z \sim N(\delta, I_d)$ where $\|\delta\| = \rho$, the minimax power for the normal means problem is*

$$\beta(\rho, \alpha) = 1 - G_d(T_{d\alpha}, \rho^2) = \Phi\left(-\frac{\sqrt{2d}}{\sqrt{2d + 4\rho^2}} T_\alpha + \frac{\rho^2}{\sqrt{2d + 4\rho^2}}\right) + o(1).$$

Proof: This proposition is *almost* verbatim from Proposition 2.15 of Pg 69 of [106]. Its proof is given in Example 2.2 on pg 51 of [106], the end of the example yielding the expression for power as $G_d(T_{d\alpha}, \rho^2)$. The only difference in our proposition statement is that we directly use the expression $G_d(T_{d\alpha}, \rho^2)$ in Eq.(10.18) instead of the approximation in Eq.(10.19).

The above proposition now directly yields a lower bound for two sample testing when $\Sigma = \sigma^2 I$. Let $\mathcal{F}_d(\rho, \sigma) := \{(P, Q) : \mathbb{E}_P[X] - \mathbb{E}_Q[Y] \in S_d(\rho), \mathbb{E}[XX^T] - \mathbb{E}[X]\mathbb{E}[X]^T = \mathbb{E}[YY^T] - \mathbb{E}[Y]\mathbb{E}[Y]^T = \sigma^2 I\}$ represent the set of all pairs of d-dimensional distributions P, Q whose means differ by $\delta \in S_d(\rho)$ and whose covariances are both $\sigma^2 I$. Define the minimax power at level α as

$$\beta(\rho, \alpha, \sigma) := \inf_{\eta \in [\eta]_{n,d,\alpha}} \sup_{(P,Q) \in \mathcal{F}_d(\rho, \sigma)} \mathbb{E}_{P,Q} \eta.$$

Theorem 29. *Given $X_1, ..., X_n \sim N(0, \sigma^2 I_d)$ and $Y_1, ...Y_n \sim N(\delta, \sigma^2 I_d)$, suppose we want to test $\delta = 0$ against $\delta \in S_d(\rho)$. Then putting $\Psi := \rho/\sigma$, the minimax power is*

$$\beta(\rho, \alpha, \sigma) = \Phi\left(-\frac{\sqrt{d}}{\sqrt{d + n\Psi^2}} T_\alpha + \frac{\Psi^2}{\sqrt{8d/n^2 + 8\Psi^2/n}}\right) + o(1)$$

Proof: Denote

$$Z = \sum_i \frac{X_i - Y_i}{\sqrt{2}\sigma\sqrt{n}} = \sqrt{n/2}\frac{(\bar{X} - \bar{Y})}{\sigma}.$$

Under the null,

$$Z \sim N(0, I_d)$$

and under the alternate

$$Z \sim N(\delta, I_d)$$

for $\delta \in S_d(\rho')$, where $\rho' = \sqrt{n/2}\rho/\sigma$, i.e. $\rho'^2 = n\Psi^2/2$. Our claim follows by direct substitution into proposition 28.

Remark 12. This lower bound expression *exactly* matches the upper bound expression in Eq.(10.10), including matching constants, showing that *all* of the discussed tests are minimax optimal in this setting of $\Sigma = \sigma^2 I$. Even though the current lower bounds can possibly be strengthened to include nondiagonal Σ, we remark that we have not been able to find even these diagonal-covariance lower bounds in the two sample testing literature, especially which are accurate even to constants.

10.6 Computation-Statistics Tradeoffs

In this section we will consider computationally cheaper alternatives to computing the quadratic time gMMD^2 that were suggested in [82] and [233], namely a block-based gMMD^2_B and a linear-time gMMD^2_L. While it is clear that gMMD^2 is the minimum variance unbiased estimator (it is a Rao-Blackwellized U-statistic), it is not clear *how much* worse the other options are - if they are only slightly worse, the computational benefits could be worth it if there is a large amount of data. Due to the lack of a high-dimensional analysis in [82], it was inferred that one suffers for cheaper computation with power that is worse, by a constant factor compared to the power of gMMD^2. We will show that, for MDA, the power is worse not by constants but by exponents of n (presumably this would only get worse for GDA). At all points, the Assumptions in Section 3 are assumed to hold wherever needed, so that we can proceed directly to comparisons.

Assume that we divide the data into $B = B(n)$ blocks of size n/B with $n/B \to \infty$. Let $\mathrm{gMMD}^2(b)$ be the gMMD^2 statistic evaluated only on the samples in block $b \in \{1, ..., B\}$, and let the block-based MMD be defined as

$$\mathrm{gMMD}^2_B = \frac{1}{B}\sum_{b=1}^{B} \mathrm{gMMD}^2(b).$$

We note that this statistic takes $(n/B)^2 B = n^2/B$ time to compute.

Also, when using $B = n/2$, i.e. using blocks of size just 2, since $n/B \to \infty$ does not hold, we look at this case separately. This statistic just takes linear-time to compute, since each block b is just of size 2, and we define the linear time MMD as

$$\mathrm{gMMD}^2_L = \frac{1}{n/2}\sum_{b=1}^{n/2} \mathrm{gMMD}^2(b). \tag{10.21}$$

Theorem 30. *Under assumptions [A1], [A2], [A3], [A4], [A5] (appropriately holding for n/B points), and the bandwidth is chosen as $\gamma^2 = \omega(Tr(\Sigma))$, the power of* gMMD^2_B *is*

$$\phi^B_{\mathrm{gMMD}} = \Phi\left(\frac{\sqrt{B}\|\delta\|^2}{\sqrt{8\frac{B^2 Tr(\Sigma^2)}{n^2} + 8\frac{B\delta^T\Sigma\delta}{n}}} - z_\alpha \frac{\sigma_{B1}}{\sigma_B}\right) + o(1).$$

Proof: Let σ_{B1}^2 and σ_{B2}^2 be as defined in Eqs.(10.5),(10.6), but each calculated on n/B points instead of n points, and scaled by γ^4, i.e.

$$\begin{aligned}\sigma_{B1}^2 &:= 8\frac{B^2Tr(\Sigma^2)}{\gamma^4 n^2}\\ \sigma_{B2}^2 &:= 8\frac{B\delta^T\Sigma\delta}{\gamma^4 n}.\end{aligned}$$

Define $\sigma_B^2 = \sigma_{B1}^2 + \sigma_{B2}^2$. Then from our earlier arguments we have that

$$\text{Under } H_0, \ \ \text{gMMD}^2(b) \ \rightsquigarrow \ N(0, \sigma_{B1}^2), \tag{10.22}$$

$$\text{Under } H_1, \ \ \text{gMMD}^2(b) \ \rightsquigarrow \ N(0, \sigma_{B1}^2 + \sigma_{B2}^2). \tag{10.23}$$

Hence, the distribution of gMMD_B^2 is $N(0, \sigma_{B1}^2/B)$ under null and $N(\text{gMMD}^2, \sigma_B^2/B)$ under alternative. Hence, from our earlier results it is straightforward to note that under H_0,

$$\sqrt{B}\frac{\text{gMMD}_B^2}{\sigma_{B1}} \rightsquigarrow N(0,1)$$

and under H_1,

$$\sqrt{B}\frac{\text{gMMD}_B^2 - \text{gMMD}^2}{\sigma_B} \rightsquigarrow N(0,1).$$

Hence our test statistic will be

$$T_B := \sqrt{B}\frac{\text{gMMD}_B^2}{\sigma_1}$$

with our test being given by $\mathbb{I}(T_B > z_\alpha)$ where z_α is the α quantile cutoff of the standard normal distribution. Note that in practice, we would simply use a studentized statistic by plugging in the estimated σ_1. Then, the power of this test is

$$\begin{aligned} P_{H1}\left(\sqrt{B}\frac{\text{gMMD}_B^2}{\sigma_{B1}} > z_\alpha\right) &= P_{H1}\left(\sqrt{B}\frac{\text{gMMD}_B^2 - \text{gMMD}^2}{\sigma_B} > z_\alpha\frac{\sigma_{B1}}{\sigma_B} - \frac{\sqrt{B}\text{gMMD}^2}{\sigma_B}\right) & (10.24)\\ &= 1 - \Phi\left(z_\alpha\frac{\sigma_{B1}}{\sigma_B} - \frac{\sqrt{B}\text{gMMD}^2}{\sigma_B}\right) & (10.25)\\ &= \Phi\left(\frac{\sqrt{B}\|\delta\|^2}{\sqrt{8\frac{B^2Tr(\Sigma^2)}{n^2} + 8\frac{B\delta^T\Sigma\delta}{n}}} - z_\alpha\frac{\sigma_{B1}}{\sigma_B}\right). & (10.26)\end{aligned}$$

It is again useful to consider the case of $\Sigma = \sigma^2 I$ for some insight, and recall $\Psi = \|\delta\|/\sigma$. Specifically, the power is

$$\phi_L^B = \Phi\left(\frac{n\Psi^2}{\sqrt{8Bd}} - z_\alpha\right) \ \text{ when } \ \Psi = o(\sqrt{Bd/n}) \tag{10.27}$$

while in the *very* high SNR regime, the power behaves like

$$\phi_H^B = \Phi(\sqrt{n}\Psi/\sqrt{8}) \ \text{ when } \ \Psi = \omega(\sqrt{Bd/n}). \tag{10.28}$$

Of course, the above two rates match in the Medium SNR regime. Here we use the italicized *very* because it is a $\sqrt{B}$ times larger SNR requirement than the high SNR regime given in Eq.(10.13) of $\Psi =$

$\omega(\sqrt{d/n})$. Comparing to Eqs.(10.14),(10.15) to the ones above, in the very high SNR regime i.e. $\Psi = \omega(\sqrt{Bd/n})$, we have

$$\phi_H^B = \phi_H.$$

However, the low SNR regime is statistically more interesting. In this case, the power of the block test is $\sqrt{B}$ times worse (inside the Φ transformation). Noting that the block based test takes time n^2/B to compute, we see the factor $n/\sqrt{B}$ in Eq.(10.27) quite illuminating (it is the square-root of the time taken).

It was proved in [168] that the power of the linear-time statistic is given by

$$\Phi\left(\frac{\sqrt{n}\Psi^2}{\sqrt{8d+8\Psi^2}} - z_\alpha\right)$$

and hence its power in the low SNR regime is given by $\Phi\left(\frac{\sqrt{n}}{\sqrt{8d}}\Psi^2\right)$ in the (very very) high SNR regime of $\Psi = \omega(\sqrt{d})$, its power does not suffer, and is exactly $\Phi(\sqrt{n}\Psi/\sqrt{8})$ like all the above statistics, but in the low SNR regime its dependence on n suffers (and again it is the square-root of the computation time taken).

Remark 13. We can summarize this section informally as follows. If the test statistic takes time n^t to compute for $1 \leq t \leq 2$ then the power behaves like $\Phi\left(\frac{n^{t/2}\Psi^2}{\sqrt{8d}}\right)$ in the low SNR regime.

10.7 Experiments

In our experience, our claimed theorems hold true much more generally in practice. For example:

1. While we need n, d to be polynomially related in theory, we find that our experiments show that $\phi_{CQ} = \phi_{\text{eED}} = \phi_{\text{gMMD}}$ even when n is fixed and d increases, or when d is fixed and n increases.
2. While our theory seems to suggest that $\gamma^2 = \omega(Tr(\Sigma))$ is needed, the experiments suggest that $\gamma^2 = \Omega(Tr(\Sigma))$ suffices.

Before we describe our experimental suite, let us first detour to mention the "median heuristic".

10.7.1 The Median Heuristic

The median heuristic chooses the bandwidth for the Gaussian kernel as the median pairwise distance between all pairs of points (see [184]). In other words, it chooses

$$\gamma^2 = \text{Empirical Median}\left\{\|S - S'\|^2\right\}$$

where $S \neq S' \in \{X_1, ..., X_n, Y_1, ..., Y_n\}$. To have some idea of the order of magnitude of the choice that median heuristic makes, let us make the reasonable supposition that this choice is similar to the *mean*-heuristic, which chooses it to be the average distance between all pairs of points, i.e. let us assume for argument's sake that

$$\text{Empirical Median}\left\{\|S - S'\|^2\right\} \asymp \text{Population Mean}\left\{\|S - S'\|^2\right\}.$$

Then the following proposition captures the order of magnitude of the bandwidth choice made by the common median heuristic.

Proposition 31. *Under [A1], the average distance between all pairs of points is $\asymp 2Tr(\Sigma)$. Hence, under [A1], the median-heuristic chooses $\gamma^2 \asymp 2Tr(\Sigma)$.*

Proof: There are $\binom{n}{2}$ pairs of xs and $\binom{n}{2}$ pairs of ys and n^2 xy pairs, the total number of pairs being $\binom{2n}{2}$. This implies that the population mean pairwise distance is $\frac{\binom{n}{2}}{\binom{2n}{2}}\mathbb{E}\|X - X'\|^2 + \frac{\binom{n}{2}}{\binom{2n}{2}}\mathbb{E}\|Y - Y'\|^2 + \frac{n^2}{\binom{2n}{2}}\mathbb{E}\|X - Y\|^2$.

$$\begin{aligned}\mathbb{E}\|X - X'\|^2 &= \mathbb{E}\|(X - \mu_1) - (X' - \mu_1)\|^2 = 2\mathbb{E}(X - \mu_1)^T(X - \mu_1) \\ &= 2\mathbb{E}Tr((X - \mu_1)(X - \mu_1)^T) = 2Tr(\Sigma).\end{aligned}$$

$$\begin{aligned}\mathbb{E}\|X - Y\|^2 &= \mathbb{E}\|X\|^2 + \mathbb{E}\|Y\|^2 - 2\mathbb{E}X^TY \\ &= \mathbb{E}\|X - \mu_1\|^2 + \|\mu_1\|^2 + \mathbb{E}\|Y - \mu_2\|^2 + \|\mu_2\|^2 - 2\mu_1^T\mu_2 \\ &= 2Tr(\Sigma) + \|\delta\|^2.\end{aligned}$$

Together, these imply our claim.

Remark 14. The above proposition implies that the choice made by the median heuristic is at the borderline of satisfying the condition under which our main theorem holds, which is $\gamma^2 = \omega(Tr(\Sigma))$. Practically, in our experiments that follow, it seems like all the claims still seem to hold even when $\gamma^2 \asymp Tr(\Sigma)$. This implies that the conditions currently needed for our theory are possibly stronger than needed. Hence, this "heuristic" actually provides a reasonable default bandwidth choice since Σ is usually unknown.

10.7.2 Practical accuracy of our theory

Here, we consider a wide variety of experiments and demonstrate that our claims hold true with great accuracy in practice, and actually in greater generality than we can currently prove.

The different test statistics considered in this simulation suite (as given in the legends) are:

1. **uMMD0.5** - gMMD with $\gamma \asymp d^{0.5}$ i.e. $\gamma^2 \asymp Tr(\Sigma)$.
2. **uMMD Median** - gMMD with γ chosen by the aforementioned median heuristic.
3. **uMMD0.75** - gMMD with $\gamma \asymp d^{0.75}$ i.e. $\gamma^2 = \omega(Tr(\Sigma))$.
4. **ED** - (Euclidean) energy distance eED, i.e eED$_\gamma$ with $\gamma^2 = 2Tr(\Sigma)$.
5. **uCQ** - The U-statistic U_{CQ} from [36].
6. **lMMD#** - The linear-time gMMD^2_L statistic from Eq.(10.21) with $\# \in \{0.5, 0.75, \text{ Median}\}$ specifying the bandwidth as in the case of gMMD above.
7. **lCQ** - The linear-time version of U_{CQ}.

We plot the power of all these tests statistics when $\alpha = 0.05$, for various P, Q by running 100 repetitions of the two sample test for each parameter setting. As a one sentence summary of all the experiments that follow, we find that all the U-statistics have exactly the same power under mean-differences, as claimed by our theorems, i.e. $\phi_{CQ} = \phi_{\text{gMMD}} = \phi_{ED}$ for all the above choices of bandwidth, while the linear-time statistics perform significantly worse, also as predicted by the theory (demonstrating the computation-statistics tradeoff).

Experiment 1. For this experiment we use the following distributions. We vary d from 40 to 200 and always draw $n = d$ samples from the corresponding P, Q.

- Normal distribution with diagonal covariance: $P = N(\mu_0, I_{d\times d})$ and $Q = N(\mu_1, I_{d\times d})$ where $\mu_0 = (0 \dots 0)^\top$ and $\mu_1 = \frac{1}{\sqrt{d}}(1 \dots 1)^\top$.

- Product of Laplace distributions: P and Q are shifted Laplace distributions with shifts $\mu_0 = (0\ldots0)^\top$ and $\mu_1 = \frac{1}{\sqrt{d}}(1\ldots1)^\top$ respectively and identity covariance matrix.
- Product of Beta distributions: P and Q are shifted Beta distributions $\text{BETA}(1,1)$ with shifts $\mu_0 = (0\ldots0)^\top$, $\mu_1 = \frac{1}{\sqrt{12d}}(1\ldots1)^\top$ respectively and identity covariance matrix.
- Mixture of Gaussian distributions: P and Q are shifted mixture of Gaussians $\frac{1}{3}N(0, I_{d\times d}) + \frac{1}{3}N(0, 2I_{d\times d}) + \frac{1}{3}N(0, 3I_{d\times d})$ with shifts $\mu_0 = (0\ldots0)^\top$ and $\mu_1 = \sqrt{\frac{2}{d}}$ respectively.

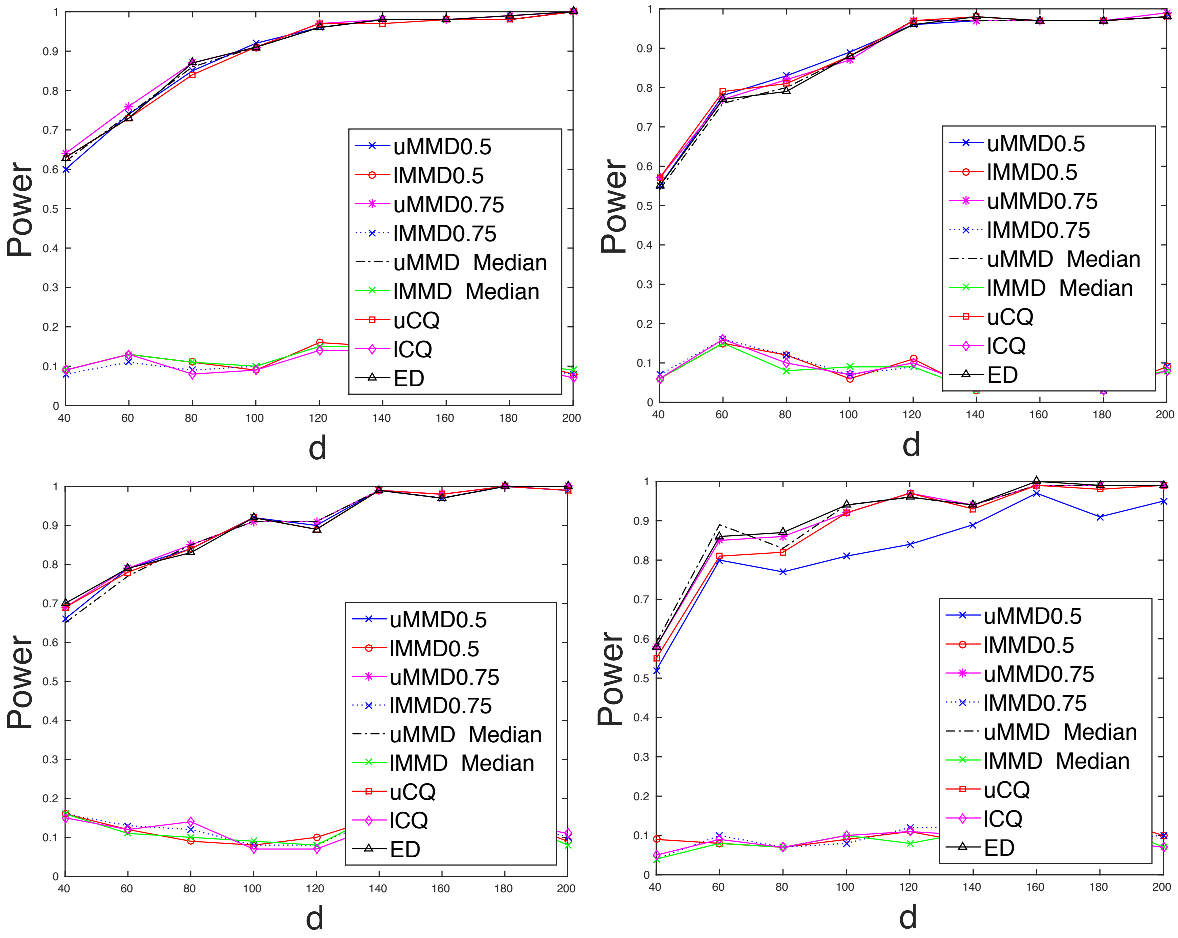


Figure 10.1: Power vs Dimension when P, Q are mean-shifted Normal (top left), Laplace (top right), Betas (bottom left) or Mixture (bottom right plot) distributions.

The values of shifts and covariance matrix are chosen to keep the asymptotic power same for all the distribution (see Theorem 25). Figure 10.1 shows the performance of various estimators for the aforementioned two sample test settings. It is clear that the power of eED, T_{CQ}, gMMD all coincide for any (sufficiently large) bandwidth, increasing as $\Phi(\sqrt{n})$ for the quadratic time statistic, and staying constant for the linear time statistics, both as predicted by the theory. Also note the fact that the plots look almost identical is consistent with our theory (see Theorem 25).

Experiment 2: In the previous experiment, we have seen the performance of the estimators for diagonal covariance matrix. Here, we empirically verify that similar effects can be observed in distributions with non-diagonal covariance matrix. To this end, we consider distributions $P = N(\mu_0, \Sigma')$ and $Q = N(\mu_1, \Sigma')$ where $\mu_0 = (0 \dots 0)^\top$, $\mu_1 = \frac{1}{\sqrt{d}}(1 \dots 1)^\top$ and $\Sigma' = U\Lambda' U^\top$. The matrix U is a random unitary matrix U obtained from the eigenvectors of a random Gaussian matrix. Λ' is set as follows. Let Λ be a diagonal matrix, the entries of which are equally spaced between 0.01 and 1, raised to the power 6. This experimental setup is similar to one used in [131]. The matrix Λ' is $d\frac{\Lambda}{tr(\Lambda)}$. Figure 10.2 shows that the qualitative performance of all statistics is similar to one observed in the previous experiment (see Figure 10.1).

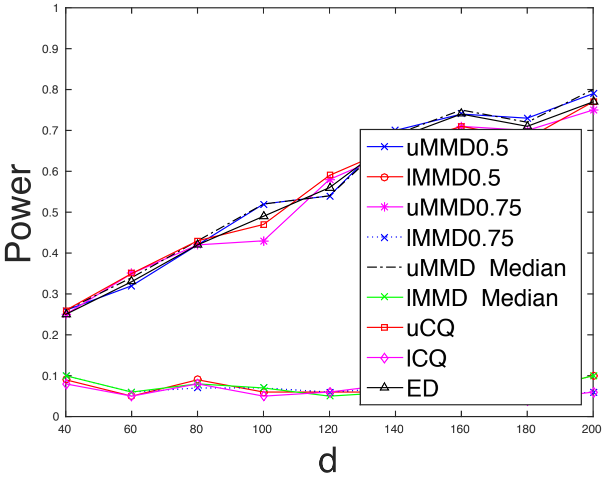


Figure 10.2: Power vs d when P, Q are mean-shifted Normal (top left) with non-diagonal covariance matrix.

Experiment 4. The aim of this experiment is to study the performance of the statistics when distributions differ in covariances rather than means. In this experiment, we set $P = N(0, \Sigma_1)$ and $Q = N(0, \Sigma_2)$ where $\Sigma_1 = \frac{50I}{\|\Sigma\|_F}$ and $\Sigma_2 = \frac{50(\Sigma+I)}{\|\Sigma\|_F}$. Here, Σ is a positive definite matrix $U\Lambda U^\top$ where U and Λ are generated as described in Experiment 2. Again, the experimental setup is similar to the one used in [131]. Not surprisingly, as seen in Figure 10.3, gMMD and eED perform better than CQ.

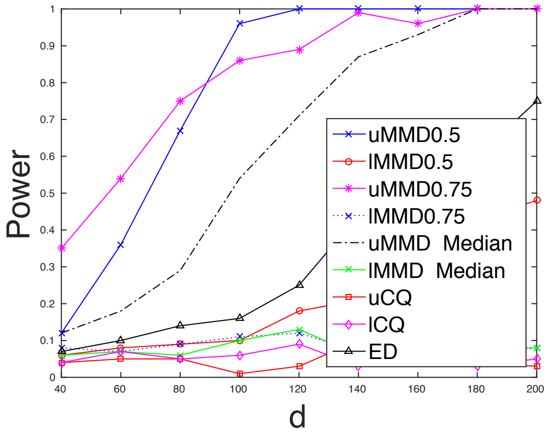


Figure 10.3: Power vs d when P, Q are distributions differing in Covariances.

This experiment demonstrates that gMMD and eED *dominate* U_{CQ} in some sense. This is due to the fact that CQ is designed for mean-shift alternatives while rest of them work for more general alternatives. Hence, they achieve the same power when the distributions differ in their means, and strictly higher power when the distributions do not differ in their means, but only in some higher moment. We can also see that the powers of the different statistics are no longer equal, and that the bandwidth does matter in this situation.

Experiment 5. Finally, we verify the nature of the asymptotic power for fixed dimension. For the purpose of this experiment, we hold d fixed to value 40 and vary n. Here, we consider two sample tests for normal distributions with diagonal and non-diagonal covariance matrices (used in Experiment 1 and Experiment 2 respectively). Figure 10.4 illustrates the power of the tests under this scenario. It can be seen that power increases with n in a manner similar to the ones observed in the previous experiments.

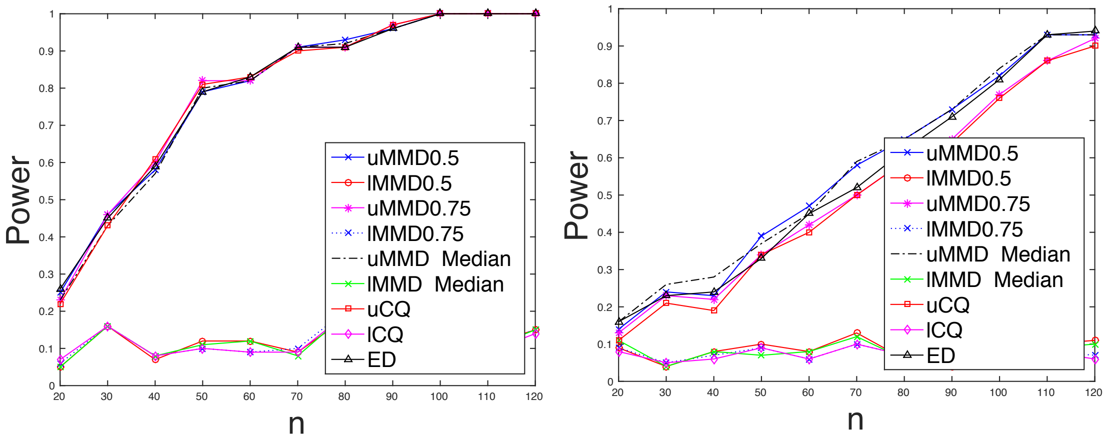


Figure 10.4: Power vs Sample size for fixed dimension when P, Q are normal distributions with diagonal (left plot) and non-diagonal (right plot) covariance matrices respectively.

This experiment suggests that assumption [A5] can probably be relaxed or dropped from the theory. We need it only to bound a certain Taylor remainder term R_3 in the proof of the theorems that follows, and it is perhaps possible to find a better way to bound this term.

10.8 Proofs of Theorems 25 and 26

Let us first note that the gMMD statistic can be written as

$$\begin{aligned} \text{gMMD} &= \begin{bmatrix} \mathbf{1}_n/\sqrt{n(n-1)} \\ -\mathbf{1}_n/\sqrt{n(n-1)} \end{bmatrix}^T \begin{bmatrix} K_{XX} & K_{XY} \\ K_{XY}^T & K_{YY} \end{bmatrix} \begin{bmatrix} \mathbf{1}_n/\sqrt{n(n-1)} \\ -\mathbf{1}_n/\sqrt{n(n-1)} \end{bmatrix} \\ &= \frac{2}{(n-1)} \cdot u^T K u \end{aligned} \tag{10.29}$$

where $u = \begin{bmatrix} \mathbf{1}_n/\sqrt{2n} \\ -\mathbf{1}_n/\sqrt{2n} \end{bmatrix}$ is a unit vector and $K = \begin{bmatrix} K_{XX} & K_{XY} \\ K_{YX} & K_{YY} \end{bmatrix}$ with its submatrices defined as

$$K_{XX} := \left\{ \exp\left(-\frac{\|X_i - X_j\|^2}{\gamma^2} \right) \mathbb{I}(i \neq j) \right\}$$

$$:= \begin{bmatrix} 0 & \exp\left(-\frac{\|X_1-X_2\|^2}{\gamma^2}\right) & \cdots & \exp\left(-\frac{\|X_1-X_n\|^2}{\gamma^2}\right) \\ \exp\left(-\frac{\|X_2-X_1\|^2}{\gamma^2}\right) & 0 & \cdots & \exp\left(-\frac{\|X_2-X_n\|^2}{\gamma^2}\right) \\ \vdots & \vdots & \ddots & \vdots \\ \exp\left(-\frac{\|X_n-X_1\|^2}{\gamma^2}\right) & \exp\left(-\frac{\|X_n-X_2\|^2}{\gamma^2}\right) & \cdots & 0 \end{bmatrix}$$

and we use the first expression to summarize the above matrix and similarly,

$$K_{YY} = \left\{\exp\left(-\frac{\|Y_i - Y_j\|^2}{\gamma^2}\right)\mathbb{I}(i \neq j)\right\}$$

$$K_{XY} = K_{YX}^T = \left\{\exp\left(-\frac{\|X_i - Y_j\|^2}{\gamma^2}\right)\mathbb{I}(i \neq j)\right\}$$

Note that there are 0s on the diagonal of K, but also on the diagonals of the other two submatrices. Note that $2Tr(\Sigma) + \|\delta\|^2 = \mathbb{E}\|X_i - Y_j\|^2 \asymp \mathbb{E}\|X_i - X_j\|^2 = \mathbb{E}\|Y_i - Y_j\|^2 = 2Tr(\Sigma)$ since $\|\delta\|^2 = o(Tr(\Sigma))$ by Assumption [A4]. For $i \neq j$, let

$$\tau := 2Tr(\Sigma)/\gamma^2 \asymp \mathbb{E}\|S_i - S_j\|^2/\gamma^2 = o(1) \tag{10.30}$$

for $S_i \in \{X_i, Y_i\}$. Let $a = \frac{\|S_i-S_j\|^2}{\gamma^2}$ Let us write the exact third order Taylor expansion of the terms $\exp(-a)$ around $\exp(-\tau)$ as

$$e^{-a} \;=\; e^{-\tau} - e^{-\tau}(a-\tau) + \frac{e^{-\tau}}{2}(a-\tau)^2 - \frac{e^{-\zeta_{ij}}}{3!}(a-\tau)^3 \tag{10.31}$$

for some ζ_{ij} between a and τ, and since $a, \tau > 0$, we have $\exp(-\zeta_{ij}) \leq 1$. For clarity in the following expressions, we drop the $\mathbb{I}(i \neq j)$ and assume it is understood. In this notation, the term-wise Taylor expansion of K is given by

$$\begin{aligned} K \;&=\; \begin{bmatrix} \left\{e^{-\frac{\|X_i-X_j\|^2}{\gamma^2}}\right\} & \left\{e^{-\frac{\|X_i-Y_j\|^2}{\gamma^2}}\right\} \\ \left\{e^{-\frac{\|Y_i-X_j\|^2}{\gamma^2}}\right\} & \left\{e^{-\frac{\|Y_i-Y_j\|^2}{\gamma^2}}\right\} \end{bmatrix} \\ &=\; e^{-\tau}\begin{bmatrix} \{1\} & \{1\} \\ \{1\} & \{1\} \end{bmatrix} - e^{-\tau}\begin{bmatrix} \left\{\frac{\|X_i-X_j\|^2}{\gamma^2} - \tau\right\} & \left\{\frac{\|X_i-Y_j\|^2}{\gamma^2} - \tau\right\} \\ \left\{\frac{\|Y_i-X_j\|^2}{\gamma^2} - \tau\right\} & \left\{\frac{\|Y_i-Y_j\|^2}{\gamma^2} - \tau\right\} \end{bmatrix} \\ &\quad + \frac{e^{-\tau}}{2!}\begin{bmatrix} \left\{\left(\frac{\|X_i-X_j\|^2}{\gamma^2} - \tau\right)^2\right\} & \left\{\left(\frac{\|X_i-Y_j\|^2}{\gamma^2} - \tau\right)^2\right\} \\ \left\{\left(\frac{\|Y_i-X_j\|^2}{\gamma^2} - \tau\right)^2\right\} & \left\{\left(\frac{\|Y_i-Y_j\|^2}{\gamma^2} - \tau\right)^2\right\} \end{bmatrix} \\ &\quad - \frac{1}{3!}\begin{bmatrix} \left\{e^{-\zeta_{ij}^{XX}}\left(\frac{\|X_i-X_j\|^2}{\gamma^2} - \tau\right)^3\right\} & \left\{e^{-\zeta_{ij}^{XY}}\left(\frac{\|X_i-Y_j\|^2}{\gamma^2} - \tau\right)^3\right\} \\ \left\{e^{-\zeta_{ij}^{YX}}\left(\frac{\|Y_i-X_j\|^2}{\gamma^2} - \tau\right)^3\right\} & \left\{e^{-\zeta_{ij}^{YY}}\left(\frac{\|Y_i-Y_j\|^2}{\gamma^2} - \tau\right)^3\right\} \end{bmatrix} \end{aligned}$$

Recalling Eq.(10.29) and expanding using the above Taylor expansion of K, we get

$$\text{gMMD} = 2e^{-\tau}\frac{U_{CQ}}{\gamma^2} + \frac{e^{-\tau}}{(n-1)}u^T T_2 u - \frac{2}{3!(n-1)}u^T(E \circ T_3)u \tag{10.32}$$

where, recalling that $\circ$ is the Hadamard product,

$$
\begin{aligned}
T_2 &:= \begin{bmatrix} \left\{\left(\frac{\|X_i-X_j\|^2}{\gamma^2} - \tau\right)^2\right\} & \left\{\left(\frac{\|X_i-Y_j\|^2}{\gamma^2} - \tau\right)^2\right\} \\ \left\{\left(\frac{\|Y_i-X_j\|^2}{\gamma^2} - \tau\right)^2\right\} & \left\{\left(\frac{\|Y_i-Y_j\|^2}{\gamma^2} - \tau\right)^2\right\} \end{bmatrix} \\
E &:= \begin{bmatrix} \{e^{-\zeta_{ij}^{XX}}\} & \{e^{-\zeta_{ij}^{XY}}\} \\ \{e^{-\zeta_{ij}^{YX}}\} & \{e^{-\zeta_{ij}^{YY}}\} \end{bmatrix} \\
T_3 &:= \begin{bmatrix} \left\{\left(\frac{\|X_i-X_j\|^2}{\gamma^2} - \tau\right)^3\right\} & \left\{\left(\frac{\|X_i-Y_j\|^2}{\gamma^2} - \tau\right)^3\right\} \\ \left\{\left(\frac{\|Y_i-X_j\|^2}{\gamma^2} - \tau\right)^3\right\} & \left\{\left(\frac{\|Y_i-Y_j\|^2}{\gamma^2} - \tau\right)^3\right\} \end{bmatrix}.
\end{aligned}
$$

Note that we have used the fact that for $u = \begin{bmatrix} \mathbf{1}_n/\sqrt{2n} \\ -\mathbf{1}_n/\sqrt{2n} \end{bmatrix}$ we have

$$u^T \begin{bmatrix} \{1\} & \{1\} \\ \{1\} & \{1\} \end{bmatrix} u = 0$$

and also that

$$U_{CQ} = \frac{1}{\binom{n}{2}} \sum_{i \neq j} \left\{ -\|X_i - X_j\|^2 - \|Y_i - Y_j\|^2 + \|X_i - Y_j\|^2 + \|X_j - Y_i\|^2 \right\}.$$

Further, recall from Eq.(10.30) that $\tau = o(1)$.

The proof of the theorem will proceed from Eq.(10.32) in three steps. Define

$$
\begin{aligned}
U_4 &:= \frac{1}{\binom{n}{2}} \sum_{i \neq j} h_4(X_i, X_j, Y_i, Y_j) \\
h_4(X_i, X_j, Y_i, Y_j) &:= \|X_i - X_j\|^4 + \|Y_i - Y_j\|^4 - \|X_i - Y_j\|^4 - \|X_j - Y_i\|^4
\end{aligned}
$$

to note that

$$\frac{1}{(n-1)} u^T T_2 u = \left(\frac{U_4}{2\gamma^4} + \frac{\tau U_{CQ}}{\gamma^2} \right)$$

(i) First we will show that the third order Taylor remainder term $R_3 := \frac{2}{3!(n-1)} u^T (E \circ T_3) u$ is a smaller order term than U_{CQ}/γ^2.

(ii) Denote $\theta_2 = \frac{1}{n-1} u^T \mathbb{E}[T_2] u$. We will show that $\theta_2 = o(\|\delta\|^2/\gamma^2)$.

(iii) Denote $s_4 = Var(U_4)$. We will show that $Var(U_4/\gamma^4) = o(Var(U_{CQ}/\gamma^2))$.

Both θ_4 and s_4 are tedious to calculate, especially under the alternative, and we will have to develop a series of lemmas on the way to calculate these quantities. Assuming for the moment that these above claims are true, we then have from Eq.(10.32) that

$$\text{gMMD} = \frac{U_{CQ}}{\gamma^2} (2e^{-\tau} + o_P(1))$$

Since we have assumed $m \geq 8$ moments, this immediately implies convergence of means and variances, i.e.

$$\mathbb{E}\text{gMMD} = \frac{\|\delta\|^2}{\gamma^2} (2e^{-\tau} + o(1)) \tag{10.33}$$

and

$$Var(\text{gMMD}) = \frac{Var(U_{CQ})}{\gamma^4}(2e^{-\tau} + o(1))^2 \tag{10.34}$$

which then implies that, ignoring smaller order terms,

$$\frac{\text{gMMD} - \mathbb{E}\text{gMMD}}{\sqrt{Var(\text{gMMD})}} = \frac{U_{CQ} - \|\delta\|^2}{\sqrt{8\frac{Tr(\Sigma^2)}{n^2} + 8\frac{\delta^T \Sigma \delta}{n}}}$$

and hence the distribution of gMMD matches the distribution of U_{CQ} under null and alternative (and the above expression has a standard normal distribution), and the two statistics hence also have the same power. The same argument also holds for the studentized statistics calculated in practice. The rest of the proof is devoted to proving the three steps (i), (ii) and (iii).

Step (i): Bounding $R_3 := \frac{2}{3!(n-1)} u^T (E \circ T_3) u$

Noting that every element of E is smaller than 1, and hence $u^T(E \circ T_3)u \leq \|E \circ T_3\|_2 \leq \max_{ij} E_{ij} \|T_3\|_2 \leq \|T_3\|_2$, implying that (ignoring constants)

$$R_3 \leq \frac{\|T_3\|_2}{n} \leq \frac{\|T_3\|_\infty}{\sqrt{n}}$$

Let us now bound every term of T_3. Taking a union bound on the statement of Assumption [A3], we see that the same exponential concentration bound holds uniformly for all $O(n^2) = o(d^4)$ pairs i, j, and hence w.p. tending to 1,

$$\max_{ij} \left| \frac{\|S_i - S_j\|^2}{\gamma^2} - \tau \right| \leq d^{-\nu(\Sigma,m)} \frac{d}{\gamma^2}$$

(we also multiplied both sides by d/γ^2). Hence we have w.p. tending to 1,

$$R_3 \leq \frac{1}{d^{3\nu}\sqrt{n}} \frac{d^3}{\gamma^6}$$

Since any random variable satisfies $X = O_P(\sqrt{Var(X)})$, we have that $U_{CQ}/\gamma^2 = O_P\left(\frac{\sqrt{Tr(\Sigma^2)}}{n\gamma^2}\right)$ under the null (its variance is even larger under the alternate), and hence $R_3 = o_P\left(U_{CQ}/\gamma^2\right)$ whenever

$$\frac{1}{d^{3\nu}\sqrt{n}} \frac{d^3}{\gamma^6} = o\left(\frac{\sqrt{Tr(\Sigma^2)}}{n\gamma^2}\right) \text{ i.e. } \sqrt{n} = o\left(\frac{\gamma^4 \sqrt{Tr(\Sigma^2)}}{d^{3-3\nu}}\right)$$

This is reasonably satisfied whenever $\gamma^2 > Tr(\Sigma) \asymp d$ and $n = o(d^{3\nu-1} Tr(\Sigma^2))$ as assumed. Hence, under our assumptions $R_3 = o_P(U_{CQ}/\gamma^2)$.

Remark. We conjecture that this holds true under much weaker conditions on γ, n, Σ, m.

Step (ii): The Behavior of $\theta_4 = \mathbb{E}[U_4]$ and $\theta_2 = \frac{1}{n-1} u^T \mathbb{E}[T_2] u$

Note the fact that for any random variable V, $\mathbb{E}(V-b)^2 = Var(V) + (\mathbb{E}V - b)^2$. Using $V = \|X-Y\|^2/\gamma^2$, $b = \tau$ and $\mathbb{E}V = \tau + \|\delta\|^2/\gamma^2$, we can write the off-diagonal terms as

$$\mathbb{E}\begin{bmatrix} \left\{\left(\frac{\|X_i - X_j\|^2}{\gamma^2} - \tau\right)^2\right\} & \left\{\left(\frac{\|X_i - Y_j\|^2}{\gamma^2} - \tau\right)^2\right\} \\ \left\{\left(\frac{\|Y_i - X_j\|^2}{\gamma^2} - \tau\right)^2\right\} & \left\{\left(\frac{\|Y_i - Y_j\|^2}{\gamma^2} - \tau\right)^2\right\} \end{bmatrix} = \begin{bmatrix} \left\{\frac{Var(\|X - X'\|^2)}{\gamma^4}\right\} & \left\{\frac{Var(\|X - Y\|^2)}{\gamma^4} + \frac{\|\delta\|^4}{\gamma^4}\right\} \\ \left\{\frac{Var(\|X - Y\|^2)}{\gamma^4} + \frac{\|\delta\|^4}{\gamma^4}\right\} & \left\{\frac{Var(\|Y - Y'\|^2)}{\gamma^4}\right\} \end{bmatrix}$$

Since $Var(\|X-X'\|^2) = Var(\|Y-Y'\|^2)$, we have

$$\theta_2 = Var(\|X-X'\|^2) - Var(\|X-Y\|^2) - \|\delta\|^4/\gamma^4.$$

The next two propositions imply that $\theta_2 = -8\delta^T\Sigma\delta/\gamma^4 - \|\delta\|^4/\gamma^4 = o(\|\delta\|^2/\gamma^2)$, as required for step (ii). They also imply that

$$\theta_4 = -16\delta^T\Sigma\delta - 8\|\delta\|^2 Tr(\Sigma) - 2\|\delta\|^4 \asymp -\|\delta\|^2 Tr(\Sigma).$$

Proposition 32. *Define $Z' = Z_1 - Z_2$ where Z_1, Z_2 are as in assumption [A1], [A2]. Then*

$$\begin{aligned} \mathbb{E}(Z'^T\Sigma Z') &= 2Tr(\Sigma) \\ Var(Z'^T\Sigma Z') &\asymp Tr(\Sigma^2) \\ \mathbb{E}[(Z'^T\Sigma Z')^2] &\asymp Tr^2(\Sigma) \end{aligned}$$

Proof: Since Z_1, Z_2 are independent, zero mean and identity covariance, we have Z' is mean zero and covariance $2I$ and fourth moment $\mathbb{E}Z_k'^4 = \mathbb{E}(Z_{1k} - Z_{2k})^4 = 3 + \Delta_4 + 6 + 3 + \Delta_4 = 12 + 2\Delta_4$. Firstly

$$\begin{aligned} \mathbb{E}[Z'^T\Sigma Z'] &= \mathbb{E}Tr(Z'^T\Sigma Z') = Tr\mathbb{E}(Z'^T\Sigma Z') = Tr(\mathbb{E}(\Sigma Z' Z'^T)) \\ &= 2Tr(\Sigma) \end{aligned}$$

where the last step follows since $\mathbb{E}[Z'Z'^T] = 2I$.

$$\begin{aligned} Var(Z'^T\Sigma Z') &= \mathbb{E}[Z'^T\Sigma Z']^2 - [2Tr(\Sigma)]^2 = \mathbb{E}\sum_{i,j,k,l} \Sigma_{ij}\Sigma_{kl} Z_i' Z_j' Z_k' Z_l' - 4(\sum_i \Sigma_{ii})^2 \\ &= 4\sum_i\sum_{j\neq i} \Sigma_{ii}\Sigma_{jj} + 8\sum_i\sum_{j\neq i}\Sigma_{ij}^2 + (12+4\Delta_4)\sum_i \Sigma_{ii}^2 - 4(\sum_i \Sigma_{ii}^2 + \sum_i\sum_{j\neq i}\Sigma_{ii}\Sigma_{jj}) \\ &= 8Tr(\Sigma^2) + 4\Delta_4 Tr(\Sigma\circ\Sigma) \end{aligned}$$

where the third step follows because the only nonzero terms in $\sum_{i,j,k,l}$ are because (a) $i = j$ and $k = l \neq i$ or (b) $i = k$ and $j = l \neq i$ or (c) $i = l$ and $j = k \neq i$ or (d) $i = j = k = l$ and the last step follows because $Tr(\Sigma^2) = \|\Sigma\|_F^2 = \sum_{i,j}\Sigma_{ij}^2$. The lemma is proved because $\sum_i \Sigma_{ii}^2 \leq \sum_{i,j}\Sigma_{ij}^2$.

$$\begin{aligned} \text{Hence } \mathbb{E}[(Z'^T\Sigma Z')^2] &= Var(Z'^T\Sigma Z') + (\mathbb{E}Z'^T\Sigma Z')^2 = 8Tr(\Sigma^2) + 2\Delta_4\sum_i \Sigma_{ii}^2 + 4Tr^2(\Sigma) \\ &\asymp Tr^2(\Sigma). \end{aligned}$$

Proposition 33. *Let X, Y be as in assumption [A1], [A2], [A3]. Then*

$$\begin{aligned} \mathbb{E}\|X-Y\|^2 &= 2Tr(\Sigma) + \|\delta\|^2, \\ Var(\|X-Y\|^2) &\asymp 8Tr(\Sigma^2) + 8\delta^T\Sigma\delta, \\ \mathbb{E}\|X-Y\|^4 &\asymp 4Tr^2(\Sigma) + 4\|\delta\|^2 Tr(\Sigma), \end{aligned}$$

Proof: Remember that $X - Y = \Gamma(Z_1 - Z_2) + \delta =: \Gamma Z' + \delta$. Note that Z' has zero mean, variance $2I$ and every component is independent with third moment zero. Hence

$$\mathbb{E}\|X-Y\|^2 = \mathbb{E}\|\Gamma Z' + \delta\|^2 = \mathbb{E}[Z'^T\Pi Z'] + \|\delta\|^2 + 2\mathbb{E}[\delta^T\Gamma Z']$$

$$\begin{array}{rcl} & = & 2Tr(\Sigma) + \|\delta\|^2. \\ \text{Hence } Var\|X - Y\|^2 & = & \mathbb{E}[\|\Gamma Z' + \delta\|^2 - (2Tr(\Sigma) + \|\delta\|^2)]^2 \\ & = & \mathbb{E}[Z'^T \Pi Z' + 2\delta^T \Gamma Z' - 2Tr(\Sigma)]^2 \\ & = & Var(Z'^T \Pi Z') + 4\mathbb{E}[\delta^T \Gamma Z' Z'^T \Gamma^T \delta] + 4\mathbb{E}[(Z'^T \Pi Z' - 2Tr(\Sigma))\delta^T \Gamma Z'] \\ & = & 8Tr(\Sigma^2) + 4\Delta_4 Tr(\Sigma \circ \Sigma) + 8\delta^T \Sigma \delta + 4\mathbb{E}\left[\sum_{i,j} \Pi_{ij} Z'_i Z'_j Z'^T\right] \Gamma^T \delta \\ & = & 8Tr(\Sigma^2) + 4\Delta_4 Tr(\Sigma \circ \Sigma) + 8\delta^T \Sigma \delta \end{array}$$

The second last step follows since $\mathbb{E}\sum_{i,j} \Pi_{ij} Z'_i Z'_j Z'_k = 0$ since Z' has first and third moments 0.

$$\begin{array}{rcl} \text{Hence } \mathbb{E}\|X - Y\|^4 & = & Var(\|X - Y\|^2) + (\mathbb{E}\|X - Y\|^2)^2 \\ & = & Var(Z'\Sigma Z') + 4Tr^2(\Sigma) \\ & = & 8Tr(\Sigma^2) + 4\Delta_4 Tr(\Sigma \circ \Sigma) + 8\delta^T \Sigma \delta + 4Tr^2(\Sigma) + 4\|\delta\|^2 Tr(\Sigma) + \|\delta\|^4 \end{array}$$

Step (iii): The Behavior of $s_4 = Var(U_4)$

We use the variance formula using the Hoeffding decomposition of the U-statistic U_4. We ignoring constants since we only aim to show that $Var(U_4/\gamma^4)$ is dominated by (is an order of magnitude smaller than) $Var(U_{CQ}/\gamma^2)$. Hence, we have by Lemma A of Section 5.2.1 of [188],

$$Var(U_4) \asymp \frac{Var(h_4)}{n^2} + \frac{Var(\mathbb{E}[h_4|X, Y])}{n}. \tag{10.35}$$

Some tedious algebra is required to estimate the second term. Recall that

$$\begin{array}{rcl} U_4 & := & \frac{1}{\binom{n}{2}} \sum_{i \neq j} h_4(X_i, X_j, Y_i, Y_j), \\ h_4(X_i, X_j, Y_i, Y_j) & := & \|X_i - X_j\|^4 + \|Y_i - Y_j\|^4 - \|X_i - Y_j\|^4 - \|X_j - Y_i\|^4, \\ \theta & := & \mathbb{E}\|X_i - X_j\|^4 + \mathbb{E}\|Y_i - Y_j\|^4 - \mathbb{E}\|X_i - Y_j\|^4 - \mathbb{E}\|X_j - Y_i\|^4. \end{array}$$

where $X, X' \sim P$ and $Y, Y' \sim Q$ from the model in [A1,A2] given by $X = \Gamma Z_1$ and $Y = \Gamma Z_2 + \delta$. (since h_4 depends only on differences, we have assumed $\delta_1 = 0$ and $\delta_2 = \delta$ without loss of generality). Firstly, it is easy to verify that h_4 is a degenerate U-statistic under the null, since $\mathbb{E}[h_4|(X, Y)] = 0$ when $P = Q$. We will now derive the variance of $\mathbb{E}[h_4|(X, Y)]$ when $P \neq Q$ under our assumptions. Let us first derive $\mathbb{E}[h_4|(X, Y)]$ below. For convenience of notation, denote

$$Y = \Gamma Z_Y$$

where $Z_Y = Z_2 + \eta$ and $\Gamma\eta = \delta$. Then

$$\begin{array}{rcl} \|X - Y'\|^4 & = & (X^T X + Y'^T Y' - 2X^T Y')^2 = (X^T X)^2 + (Y'^T Y')^2 + 4(X^T Y')^2 \\ & & +2X^T X Y'^T Y' - 4Y'^T Y' X^T Y' - 4X^T X X^T Y', \\ \mathbb{E}[\|X - Y'\|^4|(X, Y)] & = & (X^T X)^2 + \mathbb{E}[(Z'^T_Y \Pi Z'_Y)^2] + 4X^T(\Sigma + \delta\delta^T)X + 2X^T X(Tr(\Sigma) + \|\delta\|^2) \\ & & -4\mathbb{E}[Z'^T_Y \Pi Z'_Y Z'^T_Y \Gamma^T]\Gamma Z_1 - 4X^T X X^T \delta, \end{array}$$

$$\begin{aligned}
\|X'-Y\|^4 &= (X'^TX'+Y^TY-2X'^TY)^2 = (X'^TX')^2+(Y^TY)^2+4(X'^TY)^2 \\
&\quad +2X'^TX'Y^TY-4Y^TYX'^TY-4X'^TX'X'^TY, \\
\mathbb{E}[\|X'-Y\|^4|(X,Y)] &= \mathbb{E}[(Z_1'^T\Pi Z_1')^2]+(Y^TY)^2+4Y^T\Sigma Y+2Y^TYTr(\Sigma) \\
&\quad -4\mathbb{E}[Z_1'^T\Pi Z_1'Z_1'^T\Gamma^T](\Gamma Z_2+\delta).
\end{aligned}$$

Denoting $a_Y^T := \mathbb{E}[Z_Y^T\Pi Z_YZ_Y^T]$, we have

$$\begin{aligned}
a_{Yk} &= \mathbb{E}[(\sum_{i\neq j}\Pi_{ij}Z_{Yi}Z_{Yj}+\sum_i\Pi_{ii}Z_{Yi}^2)Z_{Yk}] \\
&= \mathbb{E}\left[\sum_{i\neq j}\Pi_{ij}(Z_{2i}Z_{2j}+\eta_jZ_{2i}+\eta_iZ_{2j}+\eta_i\eta_j)(Z_{2k}+\eta_k)\right] \\
&\quad +\mathbb{E}\left[\sum_i\Pi_{ii}(Z_{2i}^2+2Z_{2i}\eta_i+\eta_i^2)(Z_{2k}+\eta_k)\right] \\
&= \left[0+0+\sum_{j\neq k}\Pi_{kj}\eta_j+0+\sum_{i\neq k}\Pi_{ik}\eta_i+0+0+\eta_k\sum_{i\neq j}\eta_i\Pi_{ij}\eta_j\right] \\
&\quad +\left[\Delta_3\Pi_{kk}+\eta_k\sum_i\Pi_{ii}+2\Pi_{kk}\eta_k+0+0+\eta_k\sum_i\eta_i\Pi_{ii}\eta_i\right] \\
&= \left[2\sum_{j\neq k}\Pi_{jk}\eta_j\right]+\left[\Delta_3\Pi_{kk}+\eta_kTr(\Pi)+2\Pi_{kk}\eta_k\right]+\eta_k(\eta^T\Pi\eta) \\
&= \Delta_3\Pi_{kk}+\eta_kTr(\Pi)+2\Pi_k\eta+\eta_k\|\delta\|^2.
\end{aligned}$$

Since $\Pi\eta = \Gamma^T\Gamma\eta = \Gamma^T\delta$, we have $a_Y^T = \Delta_3 diag(\Pi)+\eta Tr(\Pi)+2\Gamma^T\delta+\|\delta\|^2\eta$. Using this and calling $a_X^T = \mathbb{E}[Z_1^T\Pi Z_1Z_1^T] = \Delta_3 diag(\Pi)$,

$$\begin{aligned}
-\mathbb{E}[\|X-Y'\|^4|(X,Y)] &= -(X^TX)^2-\mathbb{E}[(Z_Y'^T\Pi Z_Y')^2]-4X^T\Sigma X-4X^T\delta\delta^TX-2X^TXTr(\Sigma) \\
&\quad -2X^TX\|\delta\|^2+4a_X^T\Gamma^TX+4Tr(\Sigma)\delta^TX+8\delta^T\Sigma X+4\|\delta\|^2\delta^TX+4X^TXX^T\delta, \\
-\mathbb{E}[\|X'-Y\|^4|(X,Y)] &= -\mathbb{E}[(Z_1'^T\Pi Z_1')^2]-(Y^TY)^2-4Y^T\Sigma Y-2Y^TYTr(\Sigma)+4a_X^T\Gamma^TY, \\
\mathbb{E}[\|Y-Y'\|^4|(X,Y)] &= (Y^TY)^2+\mathbb{E}[(Z_Y'^T\Pi Z_Y')^2]+4Y^T\Sigma Y+4Y^T\delta\delta^TY+2Y^TYTr(\Sigma)+2Y^TY\|\delta\|^2 \\
&\quad -4a_X^T\Gamma^TY-4Tr(\Sigma)\delta^TY-8\delta^T\Sigma Y-4\|\delta\|^2\delta^TY-4Y^TYY^T\delta, \\
\mathbb{E}[\|X-X'\|^4|(X,Y)] &= \mathbb{E}[(Z_1'^T\Pi Z_1')^2]+(X^TX)^2+4X^T\Sigma X+2X^TXTr(\Sigma)-4a_X^T\Gamma^TX.
\end{aligned}$$

Adding the above 4 equations, we get

$$\begin{aligned}
\mathbb{E}[h_4|(X,Y)] &= 4\delta^T(YY^T-XX^T)\delta+2(Y^TY-X^TX)\|\delta\|^2-4Tr(\Pi)\delta^T(Y-X) \\
&\quad -8\delta^T\Sigma(Y-X)-4\|\delta\|^2\delta^T(Y-X)-4(Y^TYY^T-X^TXX^T)\delta. \qquad (10.36)
\end{aligned}$$

We will now take a detour to calculate the expectations and variances of products of quadratic forms, to aid us in bounding $Var(\mathbb{E}[h_4|(X,Y)])$ by bounding the variances of each term in Eq.(10.36) above.

Proposition 34. *Let $Q := \epsilon^T\Pi\epsilon$ be a quadratic form, where ϵ is standard normal. Then*

$$\mathbb{E}[Q] = Tr(\Pi)$$

$$\begin{aligned}
\mathbb{E}[Q^2] &= Tr^2(\Pi) + 2Tr(\Pi^2) \\
Var(Q) &= 2Tr(\Pi^2) \\
\mathbb{E}[Q^3] &= Tr^3(\Pi) + 6Tr(\Pi^2)Tr(\Pi) + 8Tr(\Pi^3) \\
\mathbb{E}[Q^4] &= Tr^4(\Pi) + 12Tr(\Pi^2)Tr^2(\Sigma) + 12Tr^2(\Pi^2) + 32Tr(\Pi)Tr(\Pi^3) + 48Tr(\Pi^4) \\
Var(Q^2) &= Tr^4(\Pi) + 12Tr(\Pi^2)Tr^2(\Pi) + 12Tr^2(\Pi^2) + 32Tr(\Pi)Tr(\Pi^3) + 48Tr(\Pi^4) \\
&\quad - \big(Tr^4(\Pi) + 4Tr^2(\Pi^2) + 4Tr(\Pi^2)Tr^2(\Pi)\big) \\
&\leq 96Tr(\Pi^2)Tr^2(\Pi)
\end{aligned}$$

Proof:

The expectations follow directly from the results of [135] and [113]. The last equation follows since $Tr(AB) \leq Tr(A)Tr(B)$ for any two psd matrices we have $Tr(\Pi^2) \leq Tr^2(\Pi)$ and $Tr(\Pi^3) \leq Tr(\Pi^2)Tr(\Pi)$ and $Tr(\Pi^4) \leq Tr(\Pi^2)Tr^2(\Pi)$. by Cauchy-Schwarz.

Proposition 35. *Let $Ts(A) = \sum_{ij} A_{ij}$ denote the Total sum of all entries of A and let $\circ$ denote Hadamard product. Let $Q = \epsilon^T \Pi \epsilon$, where the moments of the coordinates of ϵ are given by*

$$\begin{aligned}
m_1 &= 0, \\
m_2 &= 1, \\
m_3 &= \Delta_3, \\
m_4 &= 3 + \Delta_4, \\
m_5 &= \Delta_5 + 10\Delta_3, \\
m_6 &= \Delta_6 + 15\Delta_4 + 10\Delta_2^2 + 15, \\
m_7 &= \Delta_7 + 21\Delta_5 + 35\Delta_4\delta_3 + 105\Delta_3, \\
m_8 &= \Delta_8 + 28\Delta_6 + 56\Delta_5\Delta_3 + 35\Delta_4^2 + 210\Delta_4 + 280\Delta_3^2 + 105.
\end{aligned}$$

Here the Δs should be thought of as deviations from normality. Δ_3 is skewness and Δ_4 is kurtosis, and $\Delta_i = 0$ for all i if ϵ was standard Gaussian. Then, we have

$$\begin{aligned}
\mathbb{E}[Q] &= Tr(\Pi), \\
Var[Q] &= 2Tr(\Pi^2) + \Delta_4 Tr(\Pi \circ \Pi), \\
\mathbb{E}[Q^2] &= 2Tr(\Pi^2) + \Delta_4 Tr(\Pi \circ \Pi) + Tr^2(\Pi), \\
\mathbb{E}[Q^4] &= Tr^4(\Pi) + 12Tr(\Pi^2)Tr^2(\Pi) + 12Tr^2(\Pi^2) + 32Tr(\Pi)Tr(\Pi^3) + 48Tr(\Pi^4), \\
&\quad + \Delta_4 f_2 + \Delta_6 f_4 + \Delta_8 f_6 + \Delta_3^2 f_3 + \Delta_4^2 f_{42} + \Delta_3\Delta_5 f_{35} \\
\textit{where } f_4 &= 6Tr^2(\Pi)Tr(\Pi \circ \Pi) + 12Tr(\Pi^2)Tr(\Pi \circ \Pi) + 48Tr(\Pi)Tr(\Pi \circ \Pi^2) \\
&\quad + 96Tr(diag(\Pi)\Pi^3) + 48Tr(diag^2(\Pi^2)), \\
f_6 &= 4Tr(\Pi)Tr(\Pi \circ \Pi \circ \Pi) + 24Tr(\Pi \circ \Pi \circ \Pi^2), \\
f_8 &= Tr(\Pi \circ \Pi \circ \Pi \circ \Pi), \\
f_3 &= 24Ts(diag(\Pi)\Pi diag(\Pi))Tr(\Pi) + 48Ts(diag(\Pi)\Pi^2 diag(\Pi)) + 16Ts(\Pi \circ \Pi \circ \Pi)Tr(\Pi) \\
&\quad + 96Ts((\Pi \circ \Pi)\Pi diag(\Pi)) + 96Tr(\Pi(\Pi \circ \Pi)\Pi), \\
f_{42} &= 3Tr^2(\Pi \circ \Pi) + 24Ts(diag(\Pi)(\Pi \circ \Pi)diag(\Pi)) + 8Ts(\Pi \circ \Pi \circ \Pi \circ \Pi), \\
f_{35} &= 24Ts(diag(\Pi)\Pi diag^2(\Pi)) + 32Ts(diag(\Pi)(\Pi \circ \Pi \circ \Pi)), \\
Var(Q^2) &\asymp Tr(\Pi^2)Tr^2(\Pi).
\end{aligned}$$

Proof: The first four claims follow directly from the detailed work of [15]. Let us see how the last claim then follows. First note that $Tr(\Pi \circ \Pi) \leq Tr(\Pi^2) \leq Tr^2(\Pi)$. The first inequality follows because $\sum_i \Pi_{ii}^2 \leq \sum_{i,j} \Pi_{i,j}^2 = \|\Pi\|_F^2 = Tr(\Pi^2)$. The second follows because $0 \leq Tr(\Pi^2) = \langle \Pi, \Pi \rangle \leq \|\Pi\|_{op}\|\Pi\|_* \leq Tr^2(\Pi)$ by Cauchy-Schwarz. We also use the Hadamard product identity $diag(\Pi)(\Pi \circ \Pi)diag(\Pi) = (diag(\Pi)\Pi) \circ (\Pi diag(\Pi)) = (\Pi diag(\Pi)) \circ (diag(\Pi)\Pi) = \Pi \circ (diag(\Pi)\Pi diag(\Pi))$, see [101]. Since $Tr(AB) \leq Tr(A)Tr(B)$ for any two psd matrices, we similarly have

$$\begin{aligned}
Ts(\Pi \circ \Pi \circ \Pi) &= \sum_{ij} \Pi_{ij}^3 \leq \sum_{ij} |\Pi_{ij}|^3 \leq (\sum_{ij} \Pi_{ij}^2)^{3/2} = Tr^{3/2}(\Pi^2) \leq Tr(\Pi^2)Tr(\Pi) \\
Tr(\Pi \circ \Pi \circ \Pi) &= \sum_i \Pi_{ii}^3 \leq (\sum_i \Pi_{ii}^2)^{3/2} \leq (\sum_{ij} \Pi_{ij}^2)^{3/2} < Tr(\Pi^2)Tr(\Pi) \\
Ts(\Pi \circ \Pi \circ \Pi \circ \Pi) &= \sum_{ij} \Pi_{ij}^4 = \langle \Pi \circ \Pi, \Pi \circ \Pi \rangle \leq Tr^2(\Pi \circ \Pi) < Tr(\Pi^2)Tr^2(\Pi) \\
Tr(\Pi \circ \Pi \circ \Pi \circ \Pi) &< Tr(\Pi^2)Tr^2(\Pi) \\
Tr(diag(\Pi)\Pi^3) &\leq Tr(diag(\Pi))Tr(\Pi^3) \leq Tr(\Pi^2)Tr^2(\Pi) \\
Tr(\Pi(\Pi \circ \Pi)\Pi) &\leq Tr(\Pi)Tr(\Pi \circ \Pi)Tr(\Pi) \leq Tr(\Pi^2)Tr^2(\Pi) \\
Ts(diag(\Pi)(\Pi \circ \Pi)diag(\Pi)) &\leq Tr^2(\Pi)Tr(\Pi^2).
\end{aligned}$$

In this fashion, we can verify that the dominant term of $Var(Q^2)$ scales as $Tr(\Pi^2)Tr^2(\Pi)$.

We can now extend these results to the case where the quadratic form is uncentered.

Proposition 36. *$Q = \epsilon^T \Pi \epsilon$ and $Q' = Q + a^T\epsilon + b$, where ϵ satisfies the conditions of the previous proposition, $a^T a = 4\delta^T \Sigma \delta$ and $b = \delta^T \delta$. Then*

$$\begin{aligned}
\mathbb{E}[Q'] &= Tr(\Pi) + b \\
Q'^2 &= Q^2 + (a^T\epsilon)^2 + b^2 + 2Qa^T\epsilon + 2ba^T\epsilon + 2bQ \\
\mathbb{E}Q'^2 &\asymp Tr^2(\Pi) + 2Tr(\Pi^2) + a^T a + b^2 + 2\Delta_3 diag(\Pi)a + 2bTr(\Pi) \\
Var(Q') &\asymp 2Tr(\Pi^2) + a^T a + 2\Delta_3 diag(\Pi)a \\
Var(Q'^2) &\leq 2Var(Q^2) + 4(a^T a)^2 + 2\Delta_4 Tr(aa^T \circ aa^T) + 4Var(Qa^T\epsilon) \\
&\quad +4b^2 a^T a + 8b^2 Tr(\Pi^2) + 4b^2\Delta_4 Tr(\Pi \circ \Pi) \\
&\asymp Tr^2(\Pi)Tr(\Pi^2) \\
&\asymp Var(Q^2).
\end{aligned}$$

Proof: All statements hold simply by expansion and substitution from the previous proposition. Remembering that $Var(Q^2) \asymp Tr(\Sigma^2)Tr^2(\Sigma)$, we can see that the last claim holds. Indeed, Assumption [A4] implies that $a^T a = o(\lambda_{\max}(\Sigma)Tr(\Sigma))$ and hence $(a^T a)^2 = o(Tr(\Sigma^2)Tr^2(\Sigma))$ since $\lambda_{\max}^2(\Sigma) \leq \|\Sigma\|_F^2 = Tr(\Sigma^2)$. Similarly, $b^2 a^T a = o(Tr^2(\Sigma)Tr(\Sigma^2))$. In this fashion we deduce that the dominant term in $Var(Q'^2)$ is $Var(Q^2)$.

Since $Var(A + B) \leq 2Var(A) + 2Var(B)$ and $(a + b + c)^2 \leq 3a^2 + 3b^2 + 3c^2$, we can alternately derive the following bound for variances of quadratic forms involving $Y = \Gamma Z_2 + \delta$:

$$\begin{aligned}
Y^T Y &= Z_2^T \Pi Z_2 + \delta^T\delta + 2\delta^T\Gamma Z_2 \\
Y^T \Sigma Y &= Z_2^T \Pi^2 Z_2 + \delta^T \Sigma\delta + 2\delta^T\Sigma\Gamma Z_2 \\
(Y^T Y)^2 &\leq 3(Z_2^T\Pi Z_2)^2 + 3(\delta^T\delta)^2 + 3(\delta^T\Gamma Z_2)^2 \\
\mathbb{E}[Y^T Y] &= Tr(\Sigma) + \delta^T\delta
\end{aligned}$$

$$\begin{aligned}
\mathbb{E}[Y^T\Sigma Y] &= Tr(\Sigma^2) + \delta^T\Sigma\delta \\
Var(Y^TY) &\leq 4Tr(\Sigma^2) + 8\delta^T\Sigma\delta \\
\mathbb{E}[(Y^TY)^2] &= Var(Y^TY) + \mathbb{E}^2(Y^TY) \\
&\leq 4Tr(\Sigma^2) + 8\delta^T\Sigma\delta + (Tr(\Sigma) + \delta^T\delta)^2 \asymp Tr^2(\Sigma) \\
Var(Y^T\Sigma Y) &\leq 4Tr(\Sigma^4) + 8\delta^T\Sigma^3\delta \\
Var((Y^TY)^2) &\leq 18Var((Z^T\Pi Z_2)^2) + 18Var((\delta^T\Gamma Z_2)^2) \\
&\asymp Tr(\Sigma^2)Tr^2(\Sigma) + (\delta^T\Sigma\delta)^2
\end{aligned}$$

where we used $var((v^TZ)^2) = var(Z^Tvv^TZ) = 2Tr((vv^T)^2) = 2(v^Tv)^2$. Since $\delta^T\Sigma\delta = o(Tr(\Sigma^2))$ by our assumptions, the last expression is dominated by its first term.

Proposition 37.

$$\begin{aligned}
Var(X^TXX^T\delta) &\asymp Tr^2(\Sigma)\delta^T\Sigma\delta \\
Var(Y^TYY^T\delta) &\asymp Tr^2(\Sigma)\delta^T\Sigma\delta \\
&\asymp Var(X^TXX^T\delta).
\end{aligned}$$

Proof: Let us first calculate $Var(X^TXX^T\delta)$, for which we need to know $\mathbb{E}[XX^TXX^TXX^T]$. Let us first calculate $\mathbb{E}[XX^TXX^T]$. For this purpose, see that $\mathbb{E}(Z_1Z_1^T\Pi Z_1Z_1^T) = \mathbb{E}((Z_1^T\Pi Z_1)Z_1Z_1^T) = 2\Pi + Tr(\Pi)I$. This is true because its off-diagonal element is $\mathbb{E}(\sum_{ij}\Pi_{ij}z_iz_jz_az_b) = 2\Pi_{ab}$, and its diagonal is $\mathbb{E}(\sum_{ij}\Pi_{ij}z_iz_jz_a^2) = 3\Pi_{aa} + \sum_{k\neq a}\Pi_{kk} = Tr(\Pi) + 2\Pi_{aa}$. Hence $\mathbb{E}(XX^TXX^T) = \Gamma\mathbb{E}(Z_1Z_1^T\Pi Z_1Z_1^T)\Gamma^T = 2\Sigma^2 + Tr(\Sigma)\Sigma$. Now, we are ready to calculate $\mathbb{E}[XX^TXX^TXX^T]$.

$$\begin{aligned}
\text{Define } C &:= \mathbb{E}((Z_1^T\Pi Z_1)^2Z_1Z_1^T) \\
\text{Hence } C_{aa} &= \mathbb{E}(\sum_{ijkl}\Pi_{ij}\Pi_{kl}z_iz_jz_kz_lz_a^2) \\
&= 15\Pi_{aa}^2 + 6\Pi_{aa}(\sum_{t\neq a}\Pi_{tt}) + 12\sum_{t\neq a}\Pi_{ta}^2 + 3\sum_{t\neq a}\Pi_{tt}^2 + 2\sum_{s\neq t\neq a}\Pi_{ss}\Pi_{tt} + 4\sum_{s\neq t\neq a}\Pi_{st}^2
\end{aligned}$$

Let us simplify this expression. Notice the following identities:

$$\begin{aligned}
2Tr(\Pi^2) &= 2\Pi_{aa}^2 + 4\sum_{t\neq a}\Pi_{ta}^2 + 2\sum_{t\neq a}\Pi_{tt}^2 + 4\sum_{s\neq t\neq a}\Pi_{st}^2 \\
Tr^2(\Pi) &= \Pi_{aa}^2 + \sum_{t\neq a}\Pi_{tt}^2 + 2\sum_{t\neq a}\Pi_{tt}\Pi_{aa} + 2\sum_{s\neq t\neq a}\Pi_{ss}\Pi_{tt} \\
8\Pi_{.a}^T\Pi_{.a} &= 8\Pi_{aa}^2 + 8\sum_{t\neq a}\Pi_{ta}^2 \\
4Tr(\Pi)\Pi_{aa} &= 4\Pi_{aa}^2 + 4\sum_{t\neq a}\Pi_{tt}\Pi_{aa} \\
\text{Hence, we see that } C_{aa} &= 6\Pi_{aa}^2 + 4Tr(\Pi)\Pi_{aa} + 2(\Pi^2)_{aa} + 2Tr(\Pi^2) + Tr^2(\Pi) \\
\text{Similarly } C_{ab} &= \mathbb{E}(\sum_{ijkl}\Pi_{ij}\Pi_{kl}z_iz_jz_kz_lz_az_b) \\
&= 8\sum_{t\neq a\neq b}\Pi_{at}\Pi_{bt} + 4\sum_{t\neq a\neq b}\Pi_{ab}\Pi_{tt} + 12\Pi_{aa}\Pi_{ab} + 12\Pi_{bb}\Pi_{ab}
\end{aligned}$$

$$\begin{aligned} &= 4\Pi_{ab}Tr(\Pi) + 8(\Pi^2)_{ab} \\ \text{Hence } C &= 8\Pi^2 + 4Tr(\Pi)\Pi + (2Tr(\Pi^2) + Tr^2(\Pi))I \end{aligned}$$

Hence

$$\begin{aligned} \mathbb{E}[XX^TXX^TXX^T] &= 8\Sigma^3 + 4Tr(\Sigma)\Sigma^2 + 2Tr(\Sigma^2)\Sigma + Tr^2(\Sigma)\Sigma \qquad (10.37) \\ \text{and} \quad Var(X^TXX^T\delta) &\asymp \delta^T\Sigma^3\delta + Tr(\Sigma)\delta^T\Sigma^2\delta + Tr^2(\Sigma)\delta^T\Sigma\delta \\ &\asymp Tr^2(\Sigma)\delta^T\Sigma\delta. \end{aligned}$$

Next, let us calculate $Var(Y^TYY^T\delta)$. We keep only the higher order terms in the following expansions, to avoid the tediousness of Proposition 35 for clarity.

$$\begin{aligned} \mathbb{E}[YY^T] &= \Sigma + \delta\delta^T \\ \mathbb{E}(Y^TYY^T\delta) &= \mathbb{E}[(\Gamma Z_2 + \delta)^T(\Gamma Z_2 + \delta)(Z_2^T\Gamma^T\delta + \delta^T\delta)] \\ &= \|\delta\|^2(Tr(\Sigma) + \delta^T\delta) + 2\delta^T\Sigma\delta \\ &\asymp \|\delta\|^2Tr(\Sigma) \\ \mathbb{E}[YY^TYY^T] &= \mathbb{E}[(\Gamma Z_2 + \delta)(\Gamma Z_2 + \delta)^T(\Gamma Z_2 + \delta)(\Gamma Z_2 + \delta)^T] \\ &\asymp \Gamma B\Gamma^T + \delta\delta^T(\Sigma + \delta\delta^T) + \delta(Tr(\Sigma) + \delta^T\delta)\delta^T + (\Sigma + \delta\delta^T)\delta\delta^T + \|\delta\|^2(\Sigma + \delta\delta^T) \\ &\quad + \mathbb{E}[\delta Z_2^T\Gamma^T\delta Z_2^T\Gamma^T] + \mathbb{E}[\Gamma Z_2\delta^T\Gamma Z_2\delta^T] + \|\delta\|^2\delta\delta^T \\ \mathbb{E}[\delta^TYY^TYY^T\delta] &= 2\delta^T\Sigma^2\delta + Tr(\Sigma)\delta^T\Sigma\delta + 5\|\delta\|^2\delta^T\Sigma\delta + 5\|\delta\|^6 + \|\delta\|^4Tr(\Sigma) \\ &\asymp \delta^T\Sigma\delta Tr(\Sigma) + \|\delta\|^4Tr(\Sigma) \\ \mathbb{E}[\delta^TYY^TYY^TYY^T\delta] &= \delta^T\mathbb{E}[(\Gamma Z_2 + \delta)(\Gamma Z_2 + \delta)^T(\Gamma Z_2 + \delta)(\Gamma Z_2 + \delta)^T(\Gamma Z_2 + \delta)(\Gamma Z_2 + \delta)^T]\delta \\ &\asymp \|\delta\|^2(\mathbb{E}[\delta^TYY^TYY^T\delta]) + \delta^T\mathbb{E}[\Gamma Z_2Z_2^T\Gamma^TYY^TYY^T]\delta \\ &\quad + \mathbb{E}[\delta^T\Gamma Z_2\delta^TYY^TYY^T]\delta + \|\delta\|^2\mathbb{E}[Z_2^T\Gamma^TYY^TYY^T]\delta \\ &:= G_1 + G_2 + G_3 + G_4 \\ \text{Define } \Phi &:= \Gamma^T\delta\delta^T\Gamma, \text{ and let us expand the 4 terms above.} \\ G_2 = \delta^T\mathbb{E}[\Gamma Z_2Z_2^T\Gamma^TYY^TYY^T]\delta &= \delta^T\mathbb{E}[XX^TXX^TXX^T]\delta + \|\delta\|^2\delta^T\mathbb{E}[XX^TXX^T]\delta + 3\|\delta\|^2\mathbb{E}[Z_2^T\Phi Z_2Z_2^T\Pi Z_2] \\ &\quad + 2\mathbb{E}[(Z_2^T\Phi Z_2)^2] + \mathbb{E}[Z_2^T\Phi Z_2]\|\delta\|^4 \\ &\asymp \delta^T\Sigma^3\delta + Tr(\Sigma)\delta^T\Sigma^2\delta + Tr^2(\Sigma)\delta^T\Sigma\delta + \|\delta\|^2\delta^T\Sigma^2\delta + \|\delta\|^2\delta^T\Sigma\delta Tr(\Sigma) \\ &\quad + (\delta^T\Sigma\delta)^2 + \delta^T\Sigma\delta\|\delta\|^4 \\ &\asymp Tr^2(\Sigma)\delta^T\Sigma\delta \\ G_1 = \|\delta\|^2(\mathbb{E}[\delta^TYY^TYY^T\delta]) &= \|\delta\|^6Tr(\Sigma) + \|\delta\|^2\delta^T\Sigma\delta Tr(\Sigma) \\ &\preceq G_2 \\ G_3 = \mathbb{E}[\delta^T\Gamma Z_2\delta^TYY^TYY^T]\delta &= 2\mathbb{E}[Z_2^T\Phi Z_2Z_2^T\Pi Z_2]\|\delta\|^2 + 2\mathbb{E}[(Z_2^T\Phi Z_2)^2] + 4\mathbb{E}[Z_2^T\Phi Z_2]\|\delta\|^4 \\ &\asymp \|\delta\|^2\delta^T\Sigma\delta Tr(\Sigma) + \|\delta\|^2\delta^T\Sigma^2\delta + (\delta^T\Sigma\delta)^2 + \delta^T\Sigma\delta\|\delta\|^4 \\ &\preceq G_2 \\ G_4 = \|\delta\|^2\mathbb{E}[Z_2^T\Gamma^TYY^TYY^T]\delta &= \|\delta\|^4\mathbb{E}[(Z_2^T\Pi Z_2)^2] + 3\|\delta\|^2\mathbb{E}[Z_2^T\Pi Z_2Z_2^T\Phi Z_2] \\ &\quad + \|\delta\|^6\mathbb{E}[Z_2^T\Pi Z_2] + 3\|\delta\|^4\mathbb{E}[Z_2^T\Phi Z_2] \end{aligned}$$

$$\begin{aligned}
&\asymp \|\delta\|^4 Tr^2(\Sigma) + \|\delta\|^2 \delta^T \Sigma \delta Tr(\Sigma) + \|\delta\|^2 \delta^T \Sigma^2 \delta + \|\delta\|^6 Tr(\Sigma) + \|\delta\|^4 \delta^T \Sigma \delta \\
&\asymp \|\delta\|^4 Tr^2(\Sigma) \\
\text{Hence } Var(Y^T Y Y^T \delta) &= \mathbb{E}[\delta^T Y Y^T Y Y^T Y Y^T \delta] - \mathbb{E}^2[Y^T Y Y^T \delta] \\
&\asymp G_1 + G_2 + G_3 + G_4 - \|\delta\|^4 Tr^2(\Sigma) \\
&\asymp Tr^2(\Sigma)\delta^T \Sigma \delta \\
&\asymp Var(X^T X X^T \delta)
\end{aligned}$$

Lemma 29.

$$Var(\mathbb{E}[h_4|(X,Y)]) \asymp Tr^2(\Sigma)\delta^T \Sigma \delta$$

Proof:

Returning back to Eq.(10.36), the 4 different variance terms involved in $Var(\mathbb{E}[h_4|(X,Y)])$ are

$$\begin{aligned}
Var(Y^T \delta \delta^T Y) &= Var((\Gamma Z_2 + \delta)^T \delta \delta^T (\Gamma Z_2 + \delta)) \asymp (\delta^T \Sigma \delta)^2 + \|\delta\|^4 \delta^T \Sigma \delta \\
Var(Y^T Y \|\delta\|^2) &\asymp \|\delta\|^4 Tr(\Sigma^2) \\
Var(Tr(\Pi)\delta^T \Gamma (Z_2 - Z_1)) &\asymp Tr^2(\Sigma)\delta^T \Sigma \delta \\
Var(Y^T Y Y^T \delta) &\asymp Tr^2(\Sigma)\delta^T \Sigma \delta
\end{aligned}$$

Under our assumptions, one can verify that the dominant term of $Var(\mathbb{E}[h_4|X,Y])$ is $\asymp Tr^2(\Sigma)\delta^T \Sigma \delta$.

Lemma 30.

$$Var(h_4) \asymp Tr^2(\Sigma) Tr(\Sigma^2)$$

Proof:

$$\begin{aligned}
h_4 &= 4[(X^T X')^2 + (Y^T Y')^2 - (X^T Y')^2 - (X'^T Y)^2] \\
&+ 2[X^T X (X'^T X' - Y'^T Y') + Y^T Y (Y'^T Y' - X'^T X')] \\
&+ 4[Y'^T Y' Y'^T (X - Y) + X'^T X' X'^T (Y - X) + X^T X X^T (Y' - X') + Y^T Y Y^T (X' - Y')]
\end{aligned} \tag{10.38}$$

For example, let us calculate $Var((X^T X')^2)$. Defining $S' = X' X'^T$, we have

$$\begin{aligned}
\mathbb{E}[(X^T X')^4] &= \mathbb{E}_{X'} \mathbb{E}_X [(X^T S' X)^2] = \mathbb{E}_{X'} \mathbb{E}_{Z_1} [(Z_1^T \Gamma^T S' \Gamma Z_1)^2] \\
&= \mathbb{E}_{X'} [Tr(\Gamma^T X' X'^T \Gamma \Gamma^T X' X'^T \Gamma) + Tr^2(\Gamma^T X' X'^T \Gamma)] \\
&= \mathbb{E}_{X'} [(X'^T \Sigma X')^2 + (X'^T \Sigma X')^2] \\
&= \mathbb{E}_{X'} [(Z_1'^T \Pi^2 Z_1')^2 + (Z_1'^T \Pi^2 Z_1')^2] \\
&= 2Tr(\Pi^4) + Tr^2(\Pi^2) \\
\mathbb{E}[(X^T X')^2] &= \mathbb{E}_{X'} \mathbb{E}_X [Z_1^T \Gamma^T S' \Gamma Z_1] = \mathbb{E}_{X'} Tr(\Gamma^T X' X'^T \Gamma) = \mathbb{E}_{Z_1'} Z_1'^T \Pi^2 Z_1' \\
&= Tr(\Pi^2) \\
Var((X^T X')^2) &= \mathbb{E}[(X^T X')^4] - \mathbb{E}[(X^T X')^2]^2 = Tr(\Pi^4) = Tr(\Sigma^4) = o(Tr^2(\Sigma) Tr(\Sigma^2))
\end{aligned}$$

Similarly, let us calculate $Var(X'^T X' X^T X)$ and $Var(Y'^T Y' Y^T Y)$ as follows.

$$\begin{aligned}
Var(X'^T X' X^T X) &= \mathbb{E}[(X^T X)^2 (X'^T X')^2] - \mathbb{E}^2[X^T X X'^T X'] \\
&= \mathbb{E}^2[(X^T X)^2] - \mathbb{E}^4[X^T X] \asymp (8Tr(\Sigma^2) + 4Tr^2(\Sigma))^2 - (2Tr(\Sigma))^4
\end{aligned}$$

$$
\begin{aligned}
& \asymp && Tr(\Sigma^2)Tr^2(\Sigma) \\
\text{and } Var(Y'^TY'Y^TY) & = && \mathbb{E}^2[(Y^TY)^2] - \mathbb{E}^4(Y^TY) \\
& = && (Tr^2(\Sigma) + 2Tr(\Sigma^2) + 4\delta^T\Sigma\delta + \delta^T\delta \\
& && \quad + 8\Delta_3 diag(\Pi)\delta^T\Sigma\delta + 2\delta^T\delta Tr(\Sigma))^2 - (Tr(\Sigma) + \delta^T\delta)^4 \\
& \asymp && Tr^2(\Sigma)Tr(\Sigma^2)
\end{aligned}
$$

where we use Proposition 36 and the last step follows by larger terms canceling after direct expansion.

Next, let us bound $Var(X^TXX^TX')$ and $Var(Y^TYY^TY')$ as follows (other terms are similar). Multiplying Eq.(10.37) by Σ, we see that

$$\mathbb{E}[XX^TXX^TXX^T\Sigma] = 8\Sigma^4 + 4Tr(\Sigma)\Sigma^3 + 2Tr(\Sigma^2)\Sigma^2 + Tr^2(\Sigma)\Sigma^2.$$

Now taking traces on both sides, and applying trace rotation to the left, we see that the dominant term is

$$Tr(\mathbb{E}[XX^TXX^TXX^T\Sigma]) = \mathbb{E}[Tr(X^TXX^TXX^T\Sigma X)] = \mathbb{E}[(X^TX)^2X^T\Sigma X] \asymp Tr(\Sigma^2)Tr^2(\Sigma).$$

Since $Var(P) \leq \mathbb{E}[P^2]$, we conclude that

$$Var(X^TXX^TX') \leq \mathbb{E}[X^TXX^T(X'X'^T)XX^TX] = \mathbb{E}[X^T\Sigma X(X^TX)^2] \asymp Tr^2(\Sigma)Tr(\Sigma^2).$$

Then, taking expectations with respect to Y' first, we get

$$
\begin{aligned}
Var(Y^TYY^TY') &= \mathbb{E}[Y^T(\Sigma + \delta\delta^T)YY^TYY^TY] - \mathbb{E}^2[Y^TYY^T\delta] \\
&= \mathbb{E}[Y^T\Sigma Y(Y^TY)^2] + Var(Y^TYY^T\delta) \\
&\asymp \mathbb{E}[Z_Y^T\Sigma^2 Z_Y(Z_Y^T\Sigma Z_Y)^2] + Tr^2(\Sigma)\delta^T\Sigma\delta \\
&\asymp (\delta^T\Sigma\delta)^2\delta^T\Sigma^2\delta + 4(\delta^T\Sigma^2\delta)^2 + 8(\delta^T\Sigma\delta)(\delta^T\Sigma^3\delta) + 8\delta^T\Sigma^3\delta \\
&\quad + 4Tr(\Sigma^2)[\delta^T\Sigma^2\delta + (\delta^T\Sigma\delta)^2] + 8Tr(\Sigma)[\delta^T\Sigma^3\delta + (\delta^T\Sigma^2\delta)(\delta^T\Sigma\delta)] \\
&\quad + 3Tr(\Sigma^2)\delta^T\Sigma^2\delta + 6Tr(\Sigma)\delta^T\Sigma^3\delta + Tr^2(\Sigma)Tr(\Sigma^2) \\
&\quad + 4Tr(\Sigma^3)Tr(\Sigma) + 2Tr^2(\Sigma^2) + 8Tr(\Sigma^4) \\
&\asymp Tr^2(\Sigma)Tr(\Sigma^2).
\end{aligned}
$$

The above results are obtained in a fashion similar to Proposition 36 for variance of uncentered quadratic forms, or Proposition 37 for $Var(Y^TYY^T\delta)$, or from the results of [15] about momnents of products of non-normal quadratic forms (Pg. 255 of [219] for the Gaussian case). Hence, bounding the $Var(h_4)$ by (a constant times) the sum of variances of the terms in the expansion Eq.(10.38), we see that

$$Var(h_4) \asymp Tr^2(\Sigma)Tr(\Sigma^2)$$

as required, concluding the proof of the lemma.

In summary, using Eq.(10.35), we have the variance of U_4 as

$$Var(U_4) \leq C_1\frac{Tr(\Sigma^2)Tr^2(\Sigma)}{n^2} + C_2\frac{Tr^2(\Sigma)\delta^T\Sigma\delta}{n} \leq CTr^2(\Sigma)Var(U_{CQ})$$

for some absolute constants $C_1, C_2, C = \max\{C_1, C_2\}$.

Since $\gamma^2 = \omega(Tr(\Sigma))$, we see that

$$Var(U_4/\gamma^4) = o(Var(U_{CQ}/\gamma^2))$$

as required for step (iii).

Remark 15. Recall that it is typically stated in textbooks like [188], that for degenerate U-statistics, the variance under the null is $O(1/n^2)$, and variance under the alternative is $O(1/n)$. While this is true asymptotically when $n \to \infty$ in the fixed d setting, the variance under the alternative can still be $O(1/n^2)$ in the high-dimensional setting, depending on the signal to noise ratio and dimension when $d, n \to \infty$.

The conclusion of step (iii) also concludes the proof of Theorem 25.

10.8.1 Proof of Theorem 26

The only difference from the above proof, is that instead of taking the Taylor expansion of the Gaussian kernel, we take the expansion of the (modified) Euclidean distance. This gives rise to the exact same set of terms to bound, with different constants. Indeed, when $\gamma^2 = \omega(Tr(\Sigma))$, by the exact form of Taylor's theorem for $f(\cdot) = (1+\cdot)^{1/2}$ at $a = \frac{\|S_i - S_j\|^2}{\gamma^2 - 2Tr(\Sigma)}$ around $\tau = \frac{2Tr(\Sigma)}{\gamma^2 - 2Tr(\Sigma)} = o(1)$,

$$f(a) = f(\tau) + \frac{(a-\tau)}{2(1+\tau)^{1/2}} - \frac{(a-\tau)^2}{8(1+\tau)^{3/2}} + \frac{3(a-\tau)^3}{48}(1+\zeta)^{-5/2} \tag{10.39}$$

for some ζ between a and τ. Comparing Eq.(10.39) with Eq.(10.31), we see that all the terms are exactly the same, except for constants. Hence, exactly the same proof of Theorem 25 goes through for Theorem 26 as well.

Acknowledgments

This project was supported by the grant NSF IIS-1247658.

10.8.2 An error in [36] : the power for high SNR

We briefly describe an error in [36], that has a few important repercussions. All notations, equation numbers and theorems in this paragraph refer to those in [36]. Using the test statistic $T_n/\hat{\sigma}_{n1}$ defined below Theorem 2 in [36], we can derive the power under their assumption (3.5) as

$$\begin{aligned}
& P_1\left(\frac{T_n}{\hat{\sigma}_{n1}} > \xi_\alpha\right) = \\
= \quad & P_1\left(\frac{T_n - \|\mu_1 - \mu_2\|^2}{\hat{\sigma}_{n2}} > \frac{\hat{\sigma}_{n1}}{\hat{\sigma}_{n2}}\xi_\alpha - \frac{\|\mu_1 - \mu_2\|^2}{\hat{\sigma}_{n2}}\right) \\
\to \quad & \Phi\left(\frac{\|\mu_1 - \mu_2\|^2}{\hat{\sigma}_{n2}}\right) \text{ (the denominator is } \textit{not } \hat{\sigma}_{n1}\text{)} \\
= \quad & \Phi\left(\frac{\sqrt{n}\|\mu_1 - \mu_2\|^2}{\sqrt{(\mu_1 - \mu_2)^T \Sigma (\mu_1 - \mu_2)}}\right)
\end{aligned}$$

which should be the expression for power that they derive in Eq.(3.12), the most important difference being the presence of $\sqrt{n}$ instead of n in the numerator.

Chapter 11

Nonparametric testing : Sequential two sample testing using the Martingale LIL

Consider the problem of nonparametric two-sample mean testing, where we have access to i.i.d. samples from two multivariate distributions and wish to test whether they have the same mean. We propose a *sequential* test for this problem suitable for data-rich, memory-constrained situations. It is novel in several ways: it takes linear time and constant space to compute on the fly, and has robust high-dimensional statistical performance, including basically the same power guarantee (for a given false positive rate) as a batch/offline version of the test with the same computational constraints. Most notably, it has a distinct computational advantage over the batch test, because it accesses only as many samples as are required – its stopping time is adaptive to the unknown difficulty of the problem!

We analyze the test and prove these properties in a rigorously finite-sample fashion, using a novel uniform empirical Bernstein version of the law of the iterated logarithm (LIL). which may be of independent interest and allows analysis of sequential tests in a general framework. We demonstrate how to extend our ideas to nonparametric homogeneity and independence testing, and make a case for their even broader applicability.

11.1 Introduction

Nonparametric decision theory poses the problem of making a decision between a null and alternate hypothesis over a dataset with the aim of controlling both false positives and false negatives (or in statistical lingo, maximize power while controlling type-1 error), all without making distributional assumptions about the data being analyzed. There is increasing interest in solving such problems in a "big data" regime, in which both the sample size N and its dimensionality d can be large.

We present a sequential testing framework that is particularly suitable for two related scenarios:

1) The dataset is extremely large, and even a single pass through the data is prohibitive.

2) The data is arriving as a stream, and decisions must be made with minimal storage.

Hypothesis testing can be thought of as a "stochastic proof by contradiction" – the null hypothesis is assumed by default to be true, and is rejected only if the observed data are statistically very unlikely under the null. A sequential test accesses the data in an online/streaming fashion, assessing after every new datapoint whether it *then* has enough evidence to reject the null hypothesis.

Even outside the streaming setting, this problem setup is well-motivated. Suppose we have a gigantic amount of data (say a few million points of thousand-dimensional X and Y), enough to detect even the most minute differences in mean if they exist. Further suppose that, unknown to us, the decision problem at hand is actually statistically easy, meaning that one can conclude $\mu_1 \neq \mu_2$ with high confidence by just looking at a tiny fraction of the dataset. Can we take advantage of this structure despite our ignorance of its existence?

While one option would be to just discard most of the data and run an expensive test on a small subset, the main problem with this is the subsampling dilemma of not knowing how hard the problem is, and hence how large a subset will suffice — sampling too little data might miss the signal, and sampling too much could unnecessarily waste computational resources. In addressing this issue for the rest of this paper, we only compare algorithms with the *same computational budget*. Our sequential test avoids the subsampling dilemma entirely by automatically stopping after seeing an essentially optimal number of samples (where "optimal" is defined as the unknown number of samples that would suffice for a linear-time batch test to have a prespecified target type-2 error).

More specifically, we devise and formally analyze a sequential algorithm for nonparametric two-sample mean testing, where we have $X_1, ..., X_n, ... \sim P$ and $Y_1, ..., Y_n, ... \sim Q$ with P, Q are distributions on $\mathbb{R}^d$ with means $\mu_1 = \mathbb{E}_{X\sim P}[X], \mu_2 = \mathbb{E}_{Y\sim Q}[Y]$, and we need to decide between

$$H_0 : \mu_1 = \mu_2 \qquad \text{and} \qquad H_1 : \mu_1 \neq \mu_2 \tag{11.1}$$

We think of the data as arriving in two parallel infinite streams, with $X_1, ..., X_t$ and $Y_1, ..., Y_t$ having been seen by time point t, and hence it is not feasible to store the data and process it afterwards. So we resort to a sequential test with a constant memory requirement to make this decision.

Our proposed procedure only keeps track of a single scalar statistic. We construct this "test statistic" to be a zero-mean random walk under the null hypothesis. It is used to test for H_0 vs. H_1 each time a new data point is processed. The main statistical issue is dealing with the apparent multiple hypothesis testing problem – if our algorithm observes its first rejection of the null at time t, it might raise suspicions of being a false rejection, because $t-1$ hypothesis tests were already conducted and the t-th may have been rejected purely by chance. Applying some kind of multiple testing correction, like the Bonferroni or Benjamini-Hochberg procedure, is very conservative and produces inferior results over so many tests.

But the tests are far from independent, because the random walk moves a relatively small amount every iteration. Formalizing this intuition requires a classical probability theory result, the law of the iterated logarithm (LIL), with which we control for type-1 error (when H_0 is true).

Alternatively when H_1 is true, instead of needing the whole dataset as a batch algorithm would, we prove that the sequential algorithm automatically stops after processing *just enough data points to detect* H_1, depending on the unknown difficulty of the instance being solved. The near-optimal nature of this adaptive type-2 error control (when H_1 is true) is again due to the remarkable LIL.

The LIL can be described as follows: imagine tossing a fair coin, assigning $+1$ to heads and -1 to tails, and keeping track of the sum S_t of t coin flips. The LIL basically states that asymptotically, S_t remains always bounded between $-\sqrt{2t \ln \ln t}$ and $+\sqrt{2t \ln \ln t}$ (and this "LIL envelope" is tight).

As mentioned earlier, our test statistic will be a random walk, which behaves like S_t under H_0 (each quadruple of samples $X_{2t}, Y_{2t}, X_{2t+1}, Y_{2t+1}$ plays the role of a new fair coin flip). The LIL then characterizes how this random walk behaves under H_0 – our algorithm will keep observing new data since the random walk values will simply bounce around within the LIL envelope. Under H_1, this random walk is designed to have nonzero mean, and hence will eventually stray outside the LIL envelope, at which point the process stops and rejects the null hypothesis.

For practically applying this argument to finite samples and examining power tradeoffs, we cannot use the classical asymptotic form of the LIL typically stated in textbooks [69]. We instead use recent results from [14] to derive a novel finite-sample empirical Bernstein version of the LIL that depends on an easily calculated running estimate of the statistic's empirical variance. This tool may be of independent interest to those interested in non-asymptotic guarantees. This technical contribution is required to control both the type I and type II errors non-asymptotically *and* uniformly over all t.

In summary, our sequential nonparametric mean test has the following properties:

(A) Each update takes linear time in d and constant memory.

(B) Under H_0, it controls type I error, using a uniform empirical Bernstein LIL.

(C) Under *any* H_1, and with desired type II error controlled at β, it automatically stops after an optimal number of samples (or earlier if we had too few samples).

We discuss related work and our proposed test for problem (11.1) after a simple but instructive example.

11.2 A Detailed Illustrative Example

We first build intuition by studying in detail an introductory example which shows how a simple sequential test can perform statistically as well as the best batch test in hindsight, while automatically stopping essentially as soon as possible. We will show that such early stopping can be viewed as quite a general consequence of concentration of measure. Just for this section, let C, C_1, C_2 represent constants that may take different values on each appearance, but are always absolute.

Consider observing i.i.d. binary flips $A_1, A_2, \dots \in \{-1, +1\}$ of a coin, which may be fair or biased towards $+1$. We want to test for fairness, detecting unfairness as soon as possible. Concretely, we therefore wish to test, for $\delta \in (0, \frac{1}{2}]$:

$$H_0: \quad P(A_1 = +1) = \frac{1}{2} \qquad\qquad H_1(\delta): \quad P(A_1 = +1) = \frac{1}{2} + \delta$$

For any sample size n, the natural test statistic for this problem is $S_n = \sum_{i=1}^n A_i$. S_n is a (scaled) simple mean-zero random walk under H_0. A standard way to approach our problem is the *batch test* involving S_N, which tests for deviations from the null for a fixed sample size N. A basic Hoeffding bound shows that $S_N \leq \sqrt{\frac{N}{2} \ln \frac{1}{\alpha}} =: p_N$ with probability $\geq 1 - \alpha$ under the null, hence controls type I error at level α : $P_{H_0}(\text{reject } H_0) = P_{H_0}(S_N > p_N) \leq e^{-2p_N^2/N} = \alpha$.

```
Fix $N$;
if $S_N > p_N$ then
    Reject $H_0$;
else
    Fail to reject $H_0$;
```

```
Fix $N$;
for $n = 1, \dots, N$: do
    if $S_n > q_n$ then
        Reject $H_0$;
        break;
Fail to reject $H_0$;
```

Figure 11.1: General one-sided batch (left) and sequential (right) tests.

A Sequential Test

The main test we propose will be a sequential test in the framework of [224], see Fig. 11.1. It sees examples as they arrive one at a time, up to a large time N, the maximum sample size we can afford. The sequential test is defined with a sequence of positive thresholds $\{q_n\}_{n \in [N]}$. We show how to set q_n, making rough but instructive arguments for statements (B) and (C) in the introduction. These sketch the formal proofs of the corresponding results for the sequential two-sample test given later.

Type I Error. Just as the batch threshold p_N is determined by controlling the type I error with a concentration inequality, the sequential test also chooses $q_1, \dots, q_N$ to control the type I error at α:

$$P_{H_0}(\text{reject } H_0) = P_{H_0}\left(\exists n \leq N : S_n > q_n\right) \leq \alpha \tag{11.2}$$

This inequality concerns the uniform concentration over infinite tails of S_n, but what $\{q_n\}_{n \in [N]}$ satisfies it? Asymptotically, the answer is governed by a foundational result, the LIL:

Theorem 38 (Law of the iterated logarithm ([114])). *With probability* 1,

$$\limsup_{t \to \infty} \frac{S_t}{\sqrt{t \ln \ln t}} = \sqrt{2}$$

Theorem 2 in [14] proves a non-asymptotic LIL that is key to our sequential testing insights: w.p. at least $1 - \alpha$, we have $|S_n| \leq \sqrt{Cn \ln\left(\frac{\ln n}{\alpha}\right)} =: q_n$ simultaneously for *all* $n \geq C_1 \ln(\frac{4}{\alpha}) := n_0$. This choice of q_n satisfies (11.2) for $n_0 \leq n \leq N$.

Type II Error. For practical purposes, $\sqrt{\ln \ln n} \leq \sqrt{\ln \ln N}$ can be treated as a small constant (even when $N = 10^{20}$, $\sqrt{\ln \ln N} < 2$). Hence, $q_N \approx p_N$. With this approximation, the power is

$$P_{H_1(\delta)}\left(\exists n \leq N : S_n > q_n\right) \geq P_{H_1(\delta)}\left(S_N > q_N\right) \approx P_{H_1(\delta)}\left(S_N > p_N\right). \tag{11.3}$$

So the sequential test is essentially as powerful as a batch test with N samples (and similarly the n^{th} round of the sequential test is like an n-sample batch test).

Early Stopping. What is to be gained from using the sequential test? Following our earlier discussion, the standard motivation for using sequential tests is that they often require few samples to reject statistically distant alternatives. To investigate this with our working example, suppose N is large and the coin is actually biased, with a fixed unknown $\delta > 0$. Then, if we somehow had full knowledge of δ when using the batch test, we would use just enough samples $n^* = n^*(\delta)$ to ensure a desired type II error $\beta < 1$:

$$n^*(\delta) = \min\left\{n : P_{H_1(\delta)}\left(S_n \leq p_n\right) \leq \beta\right\} \tag{11.4}$$

so that for all $n \geq n^*(\delta)$, since $p_n = o(n)$,

$$\begin{aligned} \beta \;&\geq\; P_{H_1(\delta)}\left(S_n \leq p_n\right) \;=\; P_{H_1(\delta)}\left(S_n - n\delta \leq p_n - n\delta\right) \\ &\geq\; P_{H_1(\delta)}\left(S_n - n\delta \leq -Cn\delta\right) \end{aligned} \tag{11.5}$$

Examining (11.5), note that $S_n - n\delta$ is a mean-zero random walk; therefore, standard lower bounds for the binomial tail tell us that $n^* \geq \frac{C \ln(1/\beta)}{\delta^2}$ suffices.

How many samples does the sequential test use? The quantity of interest is the test's stopping time τ, which is $< N$ when it rejects H_0 and N otherwise. For any sufficiently high n, our definitions for q_n and p_n tell us that

$$\begin{aligned} P_{H_1}\left(\tau \geq n\right) \;&=\; P_{H_1}\left(\forall t \leq n : S_n \leq q_n\right) \;\leq\; P_{H_1}\left(S_n \leq q_n\right) && (11.6) \\ &=\; P_{H_1}\left(S_n - n\delta \leq q_n - n\delta\right) \;\leq\; P_{H_1}\left(S_n - n\delta \leq -Cn\delta\right) && (11.7) \\ &\leq\; \beta && (11.8) \end{aligned}$$

for $n \geq n^*$, from (11.5) and the definition of n^*. Also, using a Hoeffding bound on (11.7), we see that $P_{H_1}\left(\tau \geq n\right) \leq e^{-Cn\delta^2}$, exponentially decreasing in n. In particular, this implies that the expected stopping time of our algorithm $\mathbb{E}_{H_1}[\tau]$ is of the same order as n^*, because:

$$\mathbb{E}_{H_1}[\tau] \;=\; \sum_{n=1}^{\infty} P_{H_1}\left(\tau \geq n\right) \;\leq\; n^* + \sum_{n=n^*}^{\infty} P_{H_1}\left(\tau \geq n\right) \;\leq\; n^* + \sum_{n=n^*}^{\infty} e^{-Cn\delta^2} \tag{11.9}$$

$$\leq n^* + \frac{\beta^C}{1 - e^{-C\delta^2}} \leq n^* + \beta^C \left(\frac{1}{C\delta^2} + 1 \right) \tag{11.10}$$

$$\leq \left(1 + \frac{C_1 \beta^{C_2}}{\ln \frac{1}{\beta}} \right) n^* \tag{11.11}$$

Here (11.10) first sums the infinite geometric series with first term $(e^{-n^*\delta^2})^C \leq \beta^C$, and then uses the fact that $\frac{1}{1-e^{-x}} \leq \frac{1}{x} + 1$; and (11.11) uses $n^* \geq \frac{C \ln \frac{1}{\beta}}{\delta^2}$.

This analysis, culminating in Eq. (11.11), proves that the sequential test stops as soon as we could hope for, under any alternative δ, despite our ignorance of δ! In fact, with an increasingly stringent $\beta \to 0$, we see that $\frac{\mathbb{E}_{H_1}[\tau]}{n^*} \to 1$; so the sequential test in fact stops closer to n^*, and hence τ is almost *deterministically* best possible. We formalize this precise line of non-asymptotic reasoning in the analysis of the more nontrivial sequential test presented in Section 11.5.

11.3 Related Work

In a seminal line of work, Robbins and colleagues delved into sequential hypothesis testing in an asymptotic sense [174]. Apart from being asymptotic, his tests were most often for simple hypotheses (point nulls and alternatives), were univariate, or parametric (assuming Gaussianity or known density). That said, two of his most relevant papers are [173] and [46], that discuss statistical methods related to the LIL. They give an asymptotic version of the argument of Section 11.2, using it to design sequential Kolmogorov-Smirnov tests with power one, but once more is univariate and asymptotic; most of the other problems in these papers are parametric. Other old works that mention using the LIL for testing various simple or univariate or parametric problems include [48], [119], [125] and [47].

For testing a simple null against a simple alternative, Wald's sequential probability ratio test (SPRT) was proved to be optimal by the seminal work [226], but this applies when both the null and alternative have a known parametric form (and hence their probability ratio can be explicitly calculated). The same authors also suggested a univariate nonparametric two-sample test in [225], but presumably did not find it clear how to combine these two lines of work.

We emphasize that there are several advantages to our proposed framework and analysis which, taken together, are unique in the literature to our knowledge. Firstly we tackle the nonparametric setting, with composite hypotheses. Secondly, we work in the multivariate setting, and even in the high dimensional setting with $d, n \to \infty$. Thirdly, we do not only prove that the power is asymptotically one, but also derive finite sample rates that illuminate dependence of other parameters on β. Fourthly, we take computational considerations into account – we provide a fair comparison by proving that when compared to a single batch test with the same computational resources, our sequential test has (essentially) the same power. Lastly, our sequential test has an optimal stopping property, not provable via asymptotic arguments.

Empirical Bernstein inequalities have been used for stopping algorithms in Hoeffding races [130] and other even more general contexts [138]. This line of work uses the empirical bounds very similarly to us, albeit in the nominally different context of direct estimation of a mean. As such, they too require uniform concentration over time, but achieve it with a crude union bound (failure probability $\xi_n \propto \frac{\xi}{n^2}$), resulting in a deviation bound of $\sqrt{\widehat{V}_n \log \frac{n}{\xi}}$. In fact, this is arbitrarily inferior to our bound of $\sqrt{\widehat{V}_n \log \log \frac{\widehat{V}_n}{\xi}}$, precisely in the case $\widehat{V}_n \ll n$ in which we expect the empirical Bernstein bounds to be most useful over Hoeffding bounds. Further exploration of the consequences of the non-asymptotic LIL bound of [14] in this context is an interesting topic outside our scope here.

To our knowledge, implementing sequential testing in practice has previously invariably relied upon CLT-type results arbitrarily patched together with heuristic adjustments of the CLT threshold (e.g. the well-known Haybittle-Peto for clinical trials [151] has an arbitrary conservative choice of $q_n = 0.001$ through the sequential process and $q_N = 0.05 = \alpha$ at the last datapoint). These perform as loose versions of our uniform finite-sample LIL upper bound, though further discussion is outside the scope of this current work. In general, it is unsound to use an asymptotically normal distribution under the null at stopping time τ – the central limit theorem (CLT) applies to any *fixed* time t, and it may not apply to a *random* stopping time τ. The additional conditions required are given by Anscombe's random-sum CLT [8, 88], This has caused myriad practical complications in implementing such tests (see [120], Section 4). One of our contributions is to rigorously derive a directly usable finite-sample sequential test, in a way we believe can be generically extended.

11.4 A Linear-Time Batch Two-Sample Mean Test

We now propose and study a simple test (proofs in appendices) for problem (11.1), following the conventional hypothesis testing template referred to as the "batch test" in Fig. 11.1. The test is linear-time and its power analysis is of independent interest, but more importantly for us, as in Section 11.2, it is a reference point for the properties of the main sequential test introduced in the next section.

Denote the covariances of P, Q by $\Sigma_1, \Sigma_2, \Sigma := \frac{1}{2}(\Sigma_1 + \Sigma_2)$. Define $\delta := \mu_1 - \mu_2$ so that under H_0, $\delta = 0$. Assume for simplicity that the data is bounded: $\|x\|, \|y\| \leq B$ a.s. and let $\Phi(\cdot)$ denote the standard normal CDF. Define $[\ln\ln]_+(x) = \ln\ln[\max(x, e^e)]$.

Consider the linear-time statistic after seeing $2N$ data points:

$$U_N = \sum_{i=1}^{N} h_i$$

where $h_i = (x_{2i-1} - y_{2i-1})^\top (x_{2i} - y_{2i})$. Note that the h_is are also i.i.d.

Proposition 39. $\mathbb{E}\left[U_N\right] = \mathbb{E}\left[h\right] = N\|\delta\|^2$ *and* $\mathsf{var}\,(U_N) = N\,\mathsf{var}\,(h) = N(4\mathrm{tr}(\Sigma^2) + 4\delta^\top \Sigma \delta)$

Let V_{N0}, V_{N1} be $\mathsf{var}\,(U_N)$ under H_0, H_1 respectively: $V_{N0} := NV_0 := 4N\mathrm{tr}(\Sigma^2), V_{N1} := NV_1 := N(4\mathrm{tr}(\Sigma^2) + 4\delta^\top\Sigma\delta)$. Then since U_N is an i.i.d. sum, the central limit theorem (CLT) implies that (where $\overset{d}{\to}$ is convergence in distribution)

$$\frac{U_N}{\sqrt{V_{N0}}} \overset{d}{\to}_{H_0} \mathcal{N}(0,1) \quad , \quad \frac{U_N - N\|\delta\|^2}{\sqrt{V_{N1}}} \overset{d}{\to}_{H_1} \mathcal{N}(0,1) \tag{11.12}$$

Based on this information, our test rejects the null hypothesis whenever $U_N > \sqrt{V_{N0}}\, z_\alpha$, where z_α is the $1-\alpha$ quantile of the standard normal distribution. So Eq. (11.12) ensures that $P_{H_0}\left(\frac{U_N}{\sqrt{V_{N0}}} > z_\alpha\right) \leq \alpha$ giving us type-1 error control under H_0. In practice, we may not know V_{N0}, so we standardize the statistic using the empirical variance – since we assume N is large, these scalar variance estimates do not change the effective power analysis (unlike for the sequential test which needs variance estimates at each $n << N$). [1] The (asymptotic) power of the batch test is

$$P_{H_1}\left(\frac{U_N}{\sqrt{V_{N0}}} > z_\alpha\right) \quad = \quad P_{H_1}\left(\frac{U_N - N\|\delta\|^2}{\sqrt{V_{N1}}} > z_\alpha\sqrt{\frac{V_{N0}}{V_{N1}}} - \frac{N\|\delta\|^2}{\sqrt{V_{N1}}}\right)$$

[1] For non-asymptotic type-I error control, we can use an empirical Bernstein inequality [137, Thm 11], based on an unbiased estimator of V_N, specifically $\widehat{V}_N$, the empirical variance of h_is, to reject the null whenever $U_N > \sqrt{2\widehat{V}_N \ln(2/\alpha)} + \frac{7N\ln(2/\alpha)}{3(N-1)}$.

$$= \Phi\left(\frac{\sqrt{N}\|\delta\|^2}{\sqrt{8\text{tr}(\Sigma^2)+8\delta^\top\Sigma\delta}} - z_\alpha\sqrt{\frac{\text{tr}(\Sigma^2)}{\text{tr}(\Sigma^2)+\delta^\top\Sigma\delta}}\right) \tag{11.13}$$

Note that the second term is a constant less than z_α. As a concrete example, when $\Sigma = \sigma^2 I$, and we denote $\Psi := \frac{\|\delta\|}{\sigma}$, then the power of the linear-time batch test is at least $\Phi\left(\frac{\sqrt{N}\Psi^2}{\sqrt{8d+8\Psi^2}} - z_\alpha\right)$. This expression implies that the batch test is consistent, at constant SNR, whenever $d = o(N)$, a property which is inherited by the sequential test.

11.5 A Linear-Time Sequential Two-Sample Mean Test

In this section, we present our main sequential two-sample test. It follows the scheme in Fig. 11.1, so we only need specify a sequence of rejection thresholds q_n.

To do this, we interpret the unnormalized statistic $T_n = \sum_{i=1}^n h_i$ as a *stochastic process* evolving with n. Under the null, h_i has zero mean, and T_n is a zero-mean random walk computable from the data. We assume that our data is bounded i.e. $\|X\|, \|Y\| \leq B$. Though we assume bounded random variables for convenience, Bernstein moment conditions [24], bounded or subgaussian random variables being a special case, suffice to prove the non-asymptotic Bernstein LIL in [14], exactly as is the case for the usual Bernstein concentration inequalities for averages. Note that by the Cauchy-Schwarz inequality, w.p. 1,

$$|T_n - T_{n-1}| = |h_n| = |(x_{2n-1} - y_{2n-1})^\top (x_{2n} - y_{2n})| \leq (B+B)^2 \tag{11.14}$$

For convenience, we assume $B = \frac{1}{2}$, so that Eq. (11.14) ≤ 1. Since T_n has bounded differences, it exhibits Gaussian-like concentration under the null. However, analogously to the batch test, tighter concentration is desirable, in pursuit of which we examine the cumulative variance process of T_n under H_0, defined in the previous section:

$$\sum_{i=1}^n \mathbb{E}\left[(T_i - T_{i-1})^2 \mid h_{1:(i-1)}\right] = \sum_{i=1}^n \mathsf{var}\,(h_i) = n\mathbb{E}\left[h^2\right] = nV_0$$

This is the stochastic process version of the variance that we considered for the batch test. Using, this we have the following theorem that controls the behavior of T_n under H_0.

Theorem 40 (Uniform Bernstein Inequality for Random Walks). *Take any $\xi > 0$. Then with probability $\geq 1 - \xi$, for all n simultaneously,*

$$|T_n| < C_0(\xi) + \sqrt{2C_1 nV_0[\ln\ln]_+(nV_0) + C_1 nV_0 \ln\left(\frac{4}{\xi}\right)}$$

where $C_0(\xi) = 3(e-2)e^2 + 2\left(1+\sqrt{\frac{1}{3}}\right)\ln\left(\frac{8}{\xi}\right)$, and $C_1 = 6(e-2)$.

Its proof is in the Appendix. Unfortunately, we cannot use it directly to get computable deviation bounds for type I error control, because the covariance matrix Σ (and therefore V_0 under the null) is completely unknown a priori. nV_0 must instead be estimated on the fly as part of the sequential test, and *its estimate must be concentrated tightly and uniformly over time*, so as not to present a statistical bottleneck if the test runs for very many samples. We prove such a novel result, necessary for sequential testing, about the empirical variance process $\widehat{V}_n = \sum_i h_i^2$.

Lemma 31. *With probability $\geq 1 - \xi$, for all n simultaneously, there is an absolute constant C_3 such that*

$$nV_0 \leq C_3(\widehat{V}_n + C_0(\xi))$$

Its proof uses a self-bounding argument and is in the Appendix. Now, we can combine these to prove a novel uniform *empirical* Bernstein inequality to (practically) establish concentration of T_n under H_0.

Theorem 41 (Uniform Empirical Bernstein Inequality for Random Walks)**.** *Take any $\xi > 0$. Then with probability $\geq 1 - \xi$, for all n simultaneously, there exists an absolute constant C_3 such that*

$$|T_n| < C_0(\xi) + \sqrt{2C_3(\widehat{V}_n + C_0(\xi)) \left([\ln \ln]_+ (C_3(\widehat{V}_n + C_0(\xi))) + \ln\left(\frac{4}{\xi}\right) \right)}$$

where $C_0(\xi) = 3(e-2)e^2 + 2\left(1 + \sqrt{\frac{1}{3}}\right) \ln\left(\frac{8}{\xi}\right)$.

Its proof follows immediately from a union bound on Theorem 40 and Lemma 31. Theorem 41 depends on $\widehat{V}_n$, which is easily calculated by the algorithm on the fly in constant time per iteration [71]. Armed with this inequality, we can now compare this sequential test's statistical performance to the batch test, exactly following the generic example of Section 11.2.

Type I Error. By Thm. 41, the test controls type I error at α by setting, for a constant C,

$$q_n = \sqrt{C\widehat{V}_n \left(\ln \ln \left(\widehat{V}_n\right) + \ln\left(\frac{1}{\alpha}\right) \right)}$$

just as argued in Sec. 11.2. This choice of q_n is basically unimprovable up to constants (a finite-time optimality result is in [14]).

Type II Error. Again arguing as in Section 11.2, the threshold p_n that the batch test uses is within a $\sqrt{\ln \ln N}$ factor of q_n, so our sequential test has basically the same power as the batch test, in particular inheriting its favorable high-d statistical performance.

Early Stopping. The argument is again identical to that Section 11.2, proving that $\mathbb{E}_{H_1}[\tau]$ is nearly optimal, and arbitrarily close to optimal as β tends to zero.

This section hints at the generality of our arguments. The critical piece is just designing h so that $\sum_{i=1}^n h_i$ is a mean zero random walk under the null. Then we can use the LIL to control type-I error, and the rest of the arguments are identical, holding for any such random walk.

Finally, note that the LIL itself, as well as the non-asymptotic LIL bounds of [14], apply to martingales – much more general versions of random walks capable of modeling dependence. Our ideas could conceivably be extended to this setting, outside the scope of this paper.

11.6 Extensions

A General Two-Sample Test. Given two independent multivariate streams of i.i.d. data, instead of testing for differences in mean, we could also test for differences in *any* moment, i.e. differences in distribution, a subtler problem which may require much more data to ascertain differences in higher moments. In other words, we would be testing

$$H_0 : P = Q \text{ versus } H_1 : P \neq Q \,. \tag{11.15}$$

One simple way to do this is by using a *kernel* two-sample test, like the Maximum Mean Discrepancy (MMD) test proposed by [82]. The population MMD is defined as

$$MMD(P, Q) = \sup_{f \in H_k} \left(\mathbb{E}_{X \sim P} f(X) - \mathbb{E}_{Y \sim Q} f(Y) \right)$$

where H_k is the unit ball of functions in the Reproducing Kernel Hilbert Space corresponding to some positive semidefinite Mercer kernel k. One common choice is the Gaussian kernel $k(a, b) = \exp(-\|a -$

$b\|^2/\gamma^2)$. With this choice, the population MMD has an interesting interpretation, given by Bochner's theorem [179] as $MMD = \int_{\mathbb{R}^d} |\varphi_X(t) - \varphi_Y(t)|^2 e^{-\gamma^2\|t\|^2} \mathrm{d}t$ where $\varphi_X(t), \varphi_Y(t)$ are the characteristic functions of P, Q. This means that the population MMD is nonzero iff the distributions differ (i.e. the alternative holds).

The authors propose the following (linear-time) batch test statistic after seeing $2N$ samples: $MMD_N = \frac{1}{N}\sum_{i=1}^N h_i$. where $h_i = k(x_{2i}, x_{2i+1}) + k(y_{2i}, y_{2i+1}) - k(x_{2i}, y_{2i+1}) - k(x_{2i+1}, y_{2i})$. The associated test is consistent against all fixed (and some local) alternatives where $P \neq Q$; see [82] for a proof, and [168] for a high-dimensional analysis of this test (in the limited setting of mean-testing). Both properties are inhertied by the following sequential test.

Note that $\mathbb{E}\left[MMD_N\right] = \mathbb{E}\left[h_i\right] = 0$ under H_0. Hence, the sequential statistic we construct after seeing n batches ($2n$ samples) is the mean zero random walk $T_n = \sum_{i=1}^n h_i$. The similarity with our mean-testing statistic is not coincidental; when $k(a, b) = a^\top b$, they coincide. As before, we use the LIL to get type-1 error control, the same power up to a $\sqrt{\ln\ln N}$ factor, and also early stopping much before seeing N points if the problem at hand is simple.

A General Independence Test. Given a single multivariate stream of i.i.d data, where each datapoint is a pair $(X_i, Y_i) \in \mathbb{R}^{p+q}$, the independence testing problem involves testing whether X is independent of Y or not. More formally, we want to test

$$H_0 : X \perp Y \ \text{ versus } \ H_1 : X \not\perp Y \ . \tag{11.16}$$

A test of linear correlation/covariance only detects linear dependence. As an alternative to this, [208] proposed a population quantity called *distance covariance*, given by the formula

$$\mathrm{dCov}(X, Y) = \mathbb{E}\|X - X'\|\|Y - Y'\| + \mathbb{E}\|X - X'\|\mathbb{E}\|Y - Y'\| - 2\mathbb{E}\|X - X'\|\|Y - Y''\|$$

where $(X, Y), (X', Y'), (X'', Y'')$ are i.i.d. pairs from the joint distribution on (X, Y). Remarkably, an alternative representation is $\mathrm{dCov}(X, Y) = \int_{\mathbb{R}^{p+q}} |\phi_{X,Y}(t, s) - \phi_X(t)\phi_Y(s)|^2 w(t, s) \ dt \ ds$ where $\phi_X, \phi_Y, \phi_{X,Y}$ are the characteristic functions of the marginals and joint distribution of X, Y and $w(t, s) \propto \|t\|_p^{1+p}\|s\|_q^{1+q}$. Using this, the authors conclude that $\mathrm{dCov}(X, Y) = 0$ iff $X \perp Y$. One way to form a linear-time statistic to estimate dCov is to process the data in batches of size four — let us denote $B_i := \{(X_{4i}, Y_{4i}), (X_{4i+1}, Y_{4i+1}), (X_{4i+2}, Y_{4i+2}), (X_{4i+3}, Y_{4i+3})\}$, and calculate the scalar

$$h_i = \frac{1}{6}\sum_{\binom{4}{2}} \|X - X'\|\|Y - Y'\| + \frac{1}{6}\sum_{\binom{4}{2}} \|X - X'\|\|Y'' - Y'''\| - \frac{1}{24}\sum_{4\times 3} \|X - X'\|\|Y - Y''\|$$

where the summations are over all possible ways of assigning $(X, Y) \neq (X', Y') \neq (X'', Y'') \neq (X''', Y''')$, each pair being one from B_i. The expectation of this quantity is exactly dCov, and the batch test statistic given $2N$ datapoints, is simply $\mathrm{dCov}_N = \frac{1}{N}\sum_{i=1}^N h_i$. As before, the associated test is consistent for any fixed alternatives where $X \not\perp Y$. Noting that $\mathbb{E}\left[\mathrm{dCov}_N\right] = \mathbb{E}\left[h_i\right] = 0$ under the null, our random walk after seeing n batches (i.e. $4n$ points) will just be $T_n = \sum_{i=1}^n h_i$. As in previous sections, we use the LIL to get type-1 control; the same power up to a $\sqrt{\ln\ln N}$ factor; and also optimal early stopping much before seeing N points, if the problem at hand is simple.

Conclusion

In this paper we present a sequential scheme for multivariate nonparametric hypothesis testing against composite alternatives, which comes with a full finite-sample analysis in terms of on-the-fly estimable quantities. Its desirable properties include type-1 error control thanks to a new uniform-over-time empirical Bernstein LIL (of independent interest); near-optimal type-2 error compared to linear-time batch tests, due to the $\sqrt{\ln \ln n}$ term in the LIL; and most importantly, essentially optimal early stopping, uniformly over a large class of alternatives. We presented some simple applications in learning and statistics, but our design and analysis techniques are general, and their extensions to other settings are of continuing future interest.

11.7 Proof of Proposition 39

Proof: [Proof of Proposition 39] Since x, x', y, y' are all independent, $\mathbb{E}[h] = (\mathbb{E}[x] - \mathbb{E}[y])^\top (\mathbb{E}[x'] - \mathbb{E}[y']) = \delta^\top \delta$. Next,

$$\begin{aligned}
\mathbb{E}\left[h^2\right] &= \mathbb{E}\left[((x-y)^\top (x'-y'))^2\right] = \mathbb{E}\left[(x-y)^\top (x'-y')(x'-y')^\top (x-y)\right] \\
&= \mathbb{E}\left[\operatorname{tr}((x-y)(x-y)^\top (x'-y')(x'-y')^\top))\right] \\
&= \operatorname{tr}\left(\mathbb{E}\left[(x-y)(x-y)^\top\right] \mathbb{E}\left[(x'-y')(x'-y')^\top\right]\right)
\end{aligned}$$

Since $\mathbb{E}\left[(x-y)(x-y)^\top\right] = \Sigma_1 + \Sigma_2 + \delta\delta^\top = 2\Sigma + \delta\delta^\top$, we have

$$\begin{aligned}
\mathsf{var}\,(h) &= \mathbb{E}\left[h^2\right] - (\mathbb{E}h)^2 = \operatorname{tr}[(2\Sigma + \delta\delta^\top)^2] - \|\delta\|^4 \\
&= 4\operatorname{tr}(\Sigma^2) + 4\delta^\top \Sigma \delta
\end{aligned}$$

from which the result is immediate.

11.8 Proof of Theorem 40

We rely upon a variance-dependent form of the LIL. Upon noting that $\mathbb{E}[T_n - T_{n-1}] = 0$ and $\mathbb{E}\left[(T_n - T_{n-1})^2\right] = V_0$, it is an instance of a general martingale concentration inequality from [14].

Theorem 42 (Uniform Bernstein Bound (Instantiation of [14], Theorem 4)). *Suppose* $|T_n - T_{n-1}| \leq 1$ *w.p.* 1 *for all* $n \geq 1$. *Fix any* $\xi < 1$ *and define* $\tau_0(\xi) = \min\left\{ s : sV_0 \geq \frac{\left(1+\sqrt{1/3}\right)^2}{e-2} \ln\left(\frac{4}{\xi}\right) \right\}$. *Then with probability* $\geq 1 - \xi$, *for all* $n \geq \tau_0$ *simultaneously,* $|T_n| \leq \frac{2(e-2)}{\left(1+\sqrt{1/3}\right)} tV_0$ *and*

$$|T_n| \leq \sqrt{6(e-2)tV_0 \left(2 \ln \ln \left(\frac{3(e-2)e^2 tV_0}{|T_n|}\right) + \ln\left(\frac{2}{\xi}\right)\right)}$$

In principle this tight control by the second moment is enough to achieve our goals, just as the second-moment Bernstein inequality for random variables suffices for proving empirical Bernstein inequalities.

However, the version we use for our empirical Bernstein bound is a more convenient though looser restatement of Theorem 42. To derive it, we refer to the appendices of [14] for the following result:

Lemma 32 ([14], Theorem 16). *Take any* $\xi > 0$, *and define* T_n *and* $\tau_0(\xi)$ *as in Theorem 42. With probability* $\geq 1 - \frac{\xi}{2}$, *for all* $n < \tau_0(\xi)$ *simultaneously,*

$$|T_n| \leq 2\left(1 + \sqrt{1/3}\right) \ln\left(\frac{4}{\xi}\right)$$

Theorem 40 follows by loosely combining the above two uniform bounds.

Proof: [Proof of Theorem 40] Recall $V_n := nV_0$. Theorem 42 gives that w.p. $1 - \frac{\xi}{2}$, for all $n \geq \tau_0(\xi/2)$, $|T_n| \leq \frac{2(e-2)}{\left(1+\sqrt{1/3}\right)} V_n$ and

$$|T_n| \leq \max\left(3(e-2)e^2, \sqrt{2C_1 V_n \ln\ln V_n + C_1 V_n \ln\left(\frac{4}{\xi}\right)}\right) \tag{11.17}$$

Taking a union bound of (11.17) with Lemma 32 gives that w.p. $\geq 1 - \xi$, the following is true for all n simultaneously:

$$|T_n| \leq \begin{cases} 2\left(1 + \sqrt{\frac{1}{3}}\right)\ln\left(\frac{8}{\xi}\right) & \text{if } t < \tau_0(\xi/2) \\ \frac{2(e-2)}{\left(1+\sqrt{1/3}\right)} V_n \quad \text{and} \quad \max\left(3(e-2)e^2, \sqrt{2C_1 V_n \ln\ln V_n + C_1 V_n \ln\left(\frac{4}{\xi}\right)}\right) & \text{if } n \geq \tau_0(\xi/2) \end{cases}$$

For all n we have $|T_n|$ bounded by the maximum of the two cases above. The result can be seen to follow, by relaxing the explicit bound $|T_n| \leq \frac{2(e-2)}{\left(1+\sqrt{1/3}\right)} V_n$ to instead transform $\ln\ln$ into $[\ln\ln]_+$.

11.9 Proof of Lemma 31

Proof: Here, $\nu_i := h_i^2 - \mathbb{E}\left[h_i^2\right]$ has mean zero by definition. It has a cumulative variance process that is self-bounding:

$$\begin{aligned} B_n := \sum_{i=1}^n \mathbb{E}\left[\nu_i^2\right] &= \sum_{i=1}^n \mathbb{E}\left[\left(h_i^2 - \mathbb{E}\left[h_i^2\right]\right)^2\right] = \sum_{i=1}^n \left(\mathbb{E}\left[h_i^4\right] - \left(\mathbb{E}\left[h_i^2\right]\right)^2\right) \leq \sum_{i=1}^n \mathbb{E}\left[h_i^4\right] \\ &\overset{(a)}{\leq} \sum_{i=1}^n \mathbb{E}\left[h_i^2\right] = nV_0 := A_n \end{aligned} \tag{11.18}$$

where the last inequality (a) uses that $|h_i| \leq 1$, and we define the process A_n for convenience.

Applying Theorem 40 to the mean-zero random walk $\sum_{i=1}^n \nu_i$ gives $(1-\xi)$-a.s. for all t that:

$$\begin{aligned} \left|\widehat{V}_n - A_n\right| = \left|\sum_{i=1}^n \left(h_i^2 - \mathbb{E}\left[h_i^2\right]\right)\right| &< C_0(\xi) + \sqrt{2C_1 B_n [\ln\ln]_+(B_n) + C_1 B_n \ln\left(\frac{4}{\xi}\right)} \\ &\leq C_0(\xi) + \sqrt{2C_1 A_n [\ln\ln]_+(A_n) + C_1 A_n \ln\left(\frac{4}{\xi}\right)} \end{aligned}$$

This can be relaxed to

$$A_n - \sqrt{2C_1 A_n [\ln\ln]_+(A_n) + C_1 A_n \ln\left(\frac{4}{\xi}\right)} - C_0(\xi) - \widehat{V}_n \leq 0 \tag{11.19}$$

Suppose $A_n \geq 108 \ln\left(\frac{4}{\xi}\right)$. Then a straightforward case analysis confirms that

$$A_n \geq 8 \max\left(2C_1 [\ln\ln]_+(A_n), C_1 \ln\left(\frac{4}{\xi}\right)\right)$$

This is precisely the condition needed to invert (11.19) using Lemma 33. Doing this yields that

$$\sqrt{A_n} \leq \sqrt{2C_1[\ln\ln]_+\left(2C_0(\xi)+2\widehat{V}_n\right)+C_1\ln\left(\frac{4}{\xi}\right)}+\sqrt{C_0(\xi)+\widehat{V}_n} \tag{11.20}$$

For sufficiently high $\widehat{V}_n$ ($\Omega\left(\ln\left(\frac{4}{\xi}\right)\right)$ suffices), the first term on the right-hand side of (11.20) is bounded as $\sqrt{2C_1[\ln\ln]_+\left(2C_0(\xi)+2\widehat{V}_n\right)+C_1\ln\left(\frac{4}{\xi}\right)} \leq \sqrt{4C_1[\ln\ln]_+\left(2C_0(\xi)+2\widehat{V}_n\right)} \leq \sqrt{8C_1\left(C_0(\xi)+\widehat{V}_n\right)}$. Resubstituting into (11.20) and squaring both sides yields the result. It remains to check the case $A_n \leq 108\ln\left(\frac{4}{\xi}\right)$. But this bound clearly holds in the statement of the result, so the proof is finished.

The following lemma is useful to invert inequalities involving the iterated logarithm.

Lemma 33. *Suppose b_1, b_2, c are positive constants, $x \geq 8\max(b_1[\ln\ln]_+(x), b_2)$, and*

$$x - \sqrt{b_1 x[\ln\ln]_+(x)+b_2 x} - c \leq 0 \tag{11.21}$$

Then

$$\sqrt{x} \leq \sqrt{b_1[\ln\ln]_+(2c)+b_2}+\sqrt{c}$$

Proof: Suppose $x \geq 8\max(b_1[\ln\ln]_+(x), b_2)$. Since $x \geq 8b_2$, we have

$$0 \leq \frac{x}{8}-b_2 = \frac{x}{4}-b_1\left(\frac{x}{8b_1}\right)-b_2 \quad\implies\quad 0 \leq \frac{x^2}{4}-b_1 x\left(\frac{x}{8b_1}\right)-b_2 x$$

Substituting the assumption $\frac{x}{8b_1} \geq [\ln\ln]_+(x)$ gives

$$0 \leq \frac{x^2}{4}-b_1 x[\ln\ln]_+(x)-b_2 x \quad\implies\quad \sqrt{b_1 x[\ln\ln]_+(x)+b_2 x} \leq \frac{1}{2}x$$

Substituting this into (11.21) gives $x \leq 2c$. Therefore, again using (11.21),

$$0 \geq x-\sqrt{b_1 x[\ln\ln]_+(x)+b_2 x}-c \geq x-\sqrt{b_1 x[\ln\ln]_+(2c)+b_2 x}-c$$

This is now a quadratic in $\sqrt{x}$. Solving it (using $\sqrt{x} \geq 0$) gives

$$\sqrt{x} \leq \frac{1}{2}\left(\sqrt{b_1[\ln\ln]_+(2c)+b_2}+\sqrt{b_1[\ln\ln]_+(2c)+b_2+4c}\right) \leq \sqrt{b_1[\ln\ln]_+(2c)+b_2}+\sqrt{c}$$

using the subadditivity of $\sqrt{\cdot}$.

Chapter 12

Nonparametric testing : Smoothed Wasserstein two sample testing

Nonparametric two sample or homogeneity testing is a decision theoretic problem that involves identifying differences between two random variables without making parametric assumptions about their underlying distributions. The literature is old and rich, with a wide variety of statistics having being intelligently designed and analyzed, both for the unidimensional and the multivariate setting. Our contribution is to tie together many of these tests, drawing connections between seemingly very different statistics. Specifically, we form a chain of connections from univariate methods like the Kolmogorov-Smirnov test, QQ plots and ROC curves, to multivariate tests involving the Wasserstein distance, energy statistics and kernel based maximum mean discrepancy, that proceeds through the construction of a *smoothed* Wasserstein distance. Some observations in this chain are implicit in the literature, while others seem to have not been noticed thus far. We hope this will be a useful resource for theorists and practitioners familiar with one subset of methods but not with others.

12.1 Introduction

Nonparametric two sample testing (or homogeneity testing) deals with detecting differences between two d-dimensional distributions, given samples from both, without making any parametric distributional assumptions. The popular tests for $d = 1$ are rather different from those for $d \geq 1$, and our interest is in tying together different tests used in both settings.

There is a massive literature on the two-sample problem, having been formally studied for nearly a century, and there is no way we can cover the breadth of this huge and historic body of work. Our aim is much more restricted — we wish to form connections between several seemingly distinct families of such tests, both intuitively and formally, in the hope of informing both practitioners and theorists who may have familiarity with some sets of tests, but not others. We will also only introduce related work that has a direct relationship with this chapter.

There are also a large number of tests for *parametric* two-sample testing (assuming a form for underlying distributions, like Gaussianity), and yet others for testing only differences in *means* of distributions (like Hotelling's t-test, Wilcoxon's signed rank test, Mood's median test). Our focus will be much more restricted — in this chapter, we will restrict our attention only to *nonparametric* tests for testing differences in (any moment of the underlying) *distribution*.

This chapter started as an attempt to understand testing with the Wasserstein distance (also called earth-mover's distance or transportation distance). The main prior work in this area is the study of univariate *goodness-of-fit testing* (or one-sample testing) by del Barrio and his colleagues in [52, 53, 54], and summarized extremely well in [51]. There are other (more parametric) works specific to goodness-of-fit testing for location-scale families that we do not mention here. The only papers related to Wasserstein two-sample testing seem to involve studying the "trimmed comparison of distributions" by [2, 3].

In this chapter, we uncover an interesting relationship between the multivariate Wasserstein test and the (Euclidean) Energy distance test, also called the Cramer test, proposed independently by [207] and [16]. This proceeds through the construction of a *smoothed Wasserstein distance*, by adding an entropic penalty/regularization — varying the weight of the regularization interpolates between the Wasserstein distance at one extreme and the Energy distance at the other extreme.

This also gives rise to a new connection between the univariate Wasserstein test and popular univariate data analysis tools like quantile-quantile (QQ) plots and the Cramer von-Mises (CvM) test. Due to the relationship between distances and kernels, we will also establish connections to the kernel-based multivariate test by [82, 85] called the Maximum Mean Discrepancy, or MMD. Finally, the desire to design a univariate *distribution-free* Wasserstein test will lead us to the formal study of Receiver Operating Characteristic (ROC) curves, relating to work by [104].

Intuitively, the underlying reasons for the similarities and differences between these above tests can be seen through two lenses. First is the *population* viewpoint of how different tests work with different *representations* of distributions; most of these tests are based on differences between quantities that completely specify a distribution — (a) cumulative distribution functions (CDFs), (b) quantile functions (QFs), and (c) characteristic functions (CFs). Second is from the *sample* viewpoint of the behavior these statistics under the null hypothesis that the distributions are identical; most of these tests have null distributions based on norms of Brownian bridges, alternately viewed as infinite sums of weighted chi-squared distributions (due to the Karhunen-Loeve expansion). We will return to these points later in this chapter.

While we connect a wide variety of popular and seemingly disparate families of tests, there are still further classes of tests that we do not discuss. Some examples of tests quite different from the ones studied here include rank based tests as covered by the excellent book [124], the runs test by [225], spanning tree methods by [74], nearest-neighbor based tests by [183] and [96], and the "cross-match" tests by [175]. We also found the book by [210] to be a very useful reference for a broader perspective on comparing distributions.

Chapter Outline and Contributions. The rest of this chapter proceeds as follows. In Section 12.2, we formally present the notation and setup of nonparametric two sample testing, as well as briefly introduce three different ways of comparing distributions —using CDFs, QFs and CFs. In Section 12.3 we will introduce the multivariate Wasserstein distance, and connect it to the multivariate Energy Distance, and to the kernel MMD, through an entropy-smoothed Wasserstein distance. In Section 12.4 we will discuss a univariate Wasserstein two-sample test, and connect it to QQ plots and the KS test. Lastly, in Section 12.5, we will design a different univariate Wasserstein test that is also distribution-free, connecting it to ROC curves, but providing a careful and rigorous analysis of its limiting distribution.

12.2 Nonparametric Two Sample Testing

More formally, given samples $X_1, ..., X_n \sim P$ and $Y_1, ..., Y_m \sim Q$, where P and Q are distributions on $\mathbb{R}^d$. A test η is a function from the data $D_{m,n} := \{X_1, ...X_n, Y_1, ..., Y_m\} \in \mathbb{R}^{d(m+n)}$ to $\{0, 1\}$ (or to $[0, 1]$ if it is a randomized test).

Most tests proceed by calculating a scalar test statistic $T_{m,n} := T(D_{m,n}) \in \mathbb{R}$ and deciding H_0 or H_1 depending on whether $T_{m,n}$, after suitable normalization, is smaller or larger than a threshold t_α. t_α is calculated based on a prespecified false positive rate α, chosen so that, $\mathbb{E}_{H_0}\eta \leq \alpha$, at least asymptotically. Indeed, all tests considered in this chapter are of the form

$$\eta(X_1, ..., X_n, Y_1, ..., Y_m) = \mathbb{I}\left(T_{m,n} > t_\alpha\right)$$

We follow the Neyman-Pearson paradigm, we a test is judged by its power $\phi = \phi(m, n, d, P, Q, \alpha) = \mathbb{E}_{H_1}\eta$. We say that a test η is consistent, in the classical sense, when

$$\phi \to 1 \text{ as } m, n \to \infty, \alpha \to 0.$$

All the tests we consider in this chapter will be consistent in the classical sense mentioned above. Establishing general conditions under which these tests are consistent in the high-dimensional setting is largely open.

12.2.1 Three Ways to Compare Distributions

The literature broadly has three dominant ways of comparing distributions, both in one and in multiple dimensions. These are based on three different ways of characterizing distributions — cumulative distribution functions (CDFs), characteristic functions (CFs) and quantile functions (QFs). Many of the tests we will consider involve calculating differences between (empirical estimates of) these quantities.

For example, it is well known that the Kolmogorov-Smirnov (KS) test by [116] and [194] involves differences in empirical CDFs. We shall later see that in one dimension, the Wasserstein distance calculates differences in QFs.

While the KS test, and the related Cramer von-Mises and Anderson-Darling tests are very popular in one dimension, their usage has been slightly more restricted in higher dimension. This is mostly due to the curse of dimensionality involved with estimating multivariate empirical CDFs. While there has been work on generalizing these popular one-dimensional to higher dimensions, like [19], but these are seemingly not the most common multivariate tests.

Two classes of tests that are actually quite popular are kernel and distance based tests. As we will recap in more detail in later sections, it is also known that the Gaussian kernel MMD implicitly calculates a difference in CFs and the Euclidean energy distance implicitly works with a difference in (projected) CDFs.

12.2.2 PP and QQ plots

Let us consider two distributions P and Q on $\mathbb{R}$ and let $X_1, \ldots, X_n$ and $Y_1, \ldots, Y_m$ be two independent samples of P and Q respectively. We denote by P_n and Q_m the corresponding empirical measures.

We present some results on the asymptotic distribution of the difference between P_n and Q_m when using the distance between the cdfs F_n and G_m or the distance between the quantile functions F_n^{-1} and G_m^{-1}. For simplicity we assume that both distributions P and Q are supported on the interval $[0, 1]$; we remark that under mild assumptions on P and Q, the results we present in this section still hold without such a boundedness assumption. Moreover we assume for simplicity that F and G have positive densities on $[0, 1]$.

Note that F_n may be interpreted as a random element taking values in the space $\mathcal{D}([0, 1])$ of right continuous functions with left limits. It is well known that

$$\sqrt{n}\left(F_n - F\right) \to_w \mathbb{B} \circ F \tag{12.1}$$

where $\mathbb{B}$ is a standard Brownian bridge in $[0, 1]$ and where the weak convergence is understood as convergence of probability measures in the space $\mathcal{D}([0, 1])$; see Chapter 3 in [20] for details.

From this fact and the independence of the samples, it follows that under the null hypothesis H_0 : $P = Q$, as $n, m \to \infty$

$$\sqrt{\frac{nm}{n+m}}\,(F_n - G_m) = \sqrt{\frac{mn}{m+n}}\,(F_n - F) + \sqrt{\frac{mn}{m+n}}\,(G - G_m) \to_w \mathbb{B} \circ F. \tag{12.2}$$

The previous fact, and continuity of the function $h \in \mathcal{D}([0,1]) \mapsto \int_0^1 (h(t))^2 dt$, imply that as $n, m \to \infty$,

$$\frac{nm}{n+m} \int_0^1 (F_n(t) - G_m(t))^2 \, dt \to_w \int_0^1 (\mathbb{B}(F(t)))^2 dt.$$

We observe from the previous expression that the asymptotic distribution of

$$\frac{nm}{n+m} \int_0^1 (F_n(t) - G_m(t))^2 \, dt$$

depends on F which is unknown in practice. This observation creates an obstacle when considering a two sample test problem based on the L^2-distance (or any L^p-distance with $1 \leq p < \infty$) between the empirical cdfs F_n and G_m.

In the context of goodness-of-fit testing, that is when we want to test wether the sample $X_1, \dots, X_n$ was drawn from a *known* cdf F or not, there is a way to go around the dependence on F of the asymptotic distribution of the L^2 difference between F_n and F. This is the original purpose of the L^2-statistics of the von Mises type. In fact, (12.1) and the fact that the function $f \in \mathcal{D}([0,1]) \mapsto \int_0^1 (f(t))^2 dF(t)$ is continuous imply that

$$\int_0^1 (F_n(t) - F(t))^2 dF(t) \to_w \int_0^1 \mathbb{B}(F(t))^2 dF(t).$$

After changing variables we deduce that

$$\int_0^1 \mathbb{B}(F(t))^2 dF(t) = \int_0^1 (\mathbb{B}(s))^2 ds,$$

which we observe does not depend on F.

For the two sample problem an analogous procedure to the one presented above is not possible because in practice the distribution F is unknown. Nevertheless, a different situation occurs when one considers the L^∞-distance between F_n and G_m as opposed to their L^p-distance for $1 \leq p < \infty$. In fact, using again (12.1) we deduce that

$$\sqrt{\frac{mn}{m+n}} \|F_n - G_m\|_\infty \to_w \|\mathbb{B} \circ F\|_\infty = \|\mathbb{B}\|_\infty, \tag{12.3}$$

where the equality in the previous expression follows from the fact that the continuity of F implies that the interval $[0,1]$ is mapped onto the interval $[0,1]$. In other words, we conclude that the asymptotic distribution of $\sqrt{\frac{mn}{m+n}} \|F_n - G_m\|_\infty$ is distribution free. This makes the Kolmogorov-Smirnov test appropriate for two sample problems.

We now turn our attention to the QQ plots and specifically the L^2-distance between F_n^{-1} and G_m^{-1}. In fact, it can be shown that if F has a differentiable density f which (for the sake of simplicity) we assume is bounded away from zero, then

$$\sqrt{n}(F_n^{-1} - F^{-1}) \to_w \frac{\mathbb{B}}{f \circ F^{-1}}.$$

For a proof of the above statement see Chapter 18 in [190] or [51] for a proof where the weak convergence is considered in the space of probability measures on $L^2((0,1))$.

We note that from this result, independence, and assuming the null hypothesis $H_0 : P = Q$, it follows that

$$\sqrt{\frac{mn}{m+n}}(F_n^{-1} - G_m^{-1}) \to_w \frac{\mathbb{B}}{f \circ F^{-1}}.$$

In particular by continuity of the function $h \in L^2((0,1)) \mapsto \int_0^1 (h(t))^2 dt$, we deduce that

$$\frac{mn}{m+n} \int_0^1 (F_n^{-1} - G_m^{-1})^2 dt \to_w \int_0^1 \frac{(\mathbb{B}(t))^2}{(f \circ F^{-1}(t))^2} dt.$$

As when we considered the difference of the cdfs F_n and G_m, we remark that the asymptotic distribution of the L^2-difference of the empirical quantile functions is also distribution dependent.

Note however that there is an important difference between QQ and PP plots when using the L^∞ norm. In fact, we saw that the asymptotic distribution of the L^∞ norm of the difference of F_n and G_m is distribution free. Unfortunately, in the quantile case, we obtain

$$\sqrt{\frac{mn}{m+n}} \|F_n^{-1} - G_n^{-1}\|_\infty \to_w \frac{\mathbb{B}}{f \circ F^{-1}},$$

which of course is also distribution dependent.

12.3 Entropy Smoothed Wasserstein Distances ($d > 1$)

The theory of optimal transport [221] provides a set of powerful tools to compare probability measures and distributions on $\mathbb{R}^d$ through the knowledge of a metric D on $\mathbb{R}^d$, which we assume to be the usual Euclidean metric between vectors in what follows. Among that set of tools, the family of p-Wasserstein distances between probability measures is the best known and the subject of the next section.

12.3.1 Wasserstein Distance

Given an exponent $p \geq 1$, the definition of the p-Wasserstein distance reads:

Definition 43 (Wasserstein Distances)**.** *For $p \in [1, \infty)$ and probability measures μ, ν in $P(\Omega)$, their p-Wasserstein distance [221, Sect. 6] is*

$$W_p(\mu, \nu) = \left(\inf_{\pi \in \Gamma(\mu,\nu)} \int_{\Omega^2} \|x - y\|^p d\pi(x, y) \right)^{1/p}, \tag{12.4}$$

where $\Gamma(\mu, \nu)$ is the set of all joint probability measures on $\mathbb{R}^d \times \mathbb{R}^d$ whose marginals are μ, ν, i.e. such that for all subsets $A \subset \mathbb{R}^d$ we have $\pi(A \times \mathbb{R}^d) = \mu(A)$ and $\pi(\mathbb{R}^d \times A) = \nu(A)$.

A remarkable feature of Wasserstein distances is that Definition 43 applies to all measures regardless of their absolute continuity with respect to the Lebesgue measure: the same definition works for both empirical measures and for their densities if they exist.

When comparing two empirical measures μ, ν supported respectively on $X = (X_1, \ldots, X_n) \in \mathbb{R}^{d \times n}, Y = (Y_1, \ldots, Y_m) \in \mathbb{R}^{d \times m}$, with uniform[1] weight vectors $\mathbf{1}_n/n$ and $\mathbf{1}_m/m$ $\mu = \frac{1}{n} \sum_{i=1}^n \delta_{X_i}$, $\nu =$

[1] The Wasserstein machinery works also for non-uniform weights. We do not mention this in this chapter because all of the measures we consider in the context of two-sample testing are uniform.

$\frac{1}{m}\sum_{j=1}^{m}\delta_{Y_j}$, the Wasserstein distance $W_p(\mu,\nu)$ between μ and ν exponentiated to the power p is the optimum of a network flow problem known as the transportation problem [18, Section 7.2]. This problem has a linear objective and a polyhedral feasible set, defined respectively through the matrix M_{XY} of pairwise distances between elements of X and Y raised to the power p,

$$M_{XY} := [\|X_i - Y_j\|^p]_{ij} \in \mathbb{R}^{n\times m}, \tag{12.5}$$

and the polytope U_{nm} defined as the set of $n \times m$ nonnegative matrices such that their row and column marginals are equal to $\mathbf{1}_n/n$ and $\mathbf{1}_m/m$ respectively. Writing $\mathbf{1}_n$ for the n-dimensional vector of ones,

$$U_{nm} := \{T \in \mathbb{R}_+^{n\times m} \;:\; T\mathbf{1}_m = \mathbf{1}_n/n,\, T^T\mathbf{1}_n = \mathbf{1}_m/m\}. \tag{12.6}$$

Let $\langle A, B\,\rangle := \operatorname{tr}(A^T B)$ be the Frobenius dot-product of matrices. Combining Eq. (12.5) & (12.6), we have that $W_p^p(\mu,\nu)$ is the optimum of a linear program S of $n \times m$ variables,

$$W_p^p(\mu,\nu) = \min_{T\in U_{nm}} \langle T, M_{XY}\,\rangle, \tag{12.7}$$

of feasible set U_{nm} and cost matrix M_{XY}.

12.3.2 Smoothed Wasserstein Distance

Aside from the slow convergence rate of the Wasserstein distance between samples from two different measures to their distance in population, computing the optimum of Equation (12.7) is expensive. This can be easily seen by noticing that the transportation problem boils down to an optimal assignment problem when $n = m$. Since the resolution of the latter has a cubic cost in n, all known algorithms that can solve the optimal transport problem scale at least super-cubicly in n.

Using an idea can be traced back as far as Schrodinger [186], Cuturi [44] has recently proposed to use an entropic regularization of the optimal transport problem, to define the Sinkhorn divergence between two measures μ, ν

$$S_\lambda^p(\mu,\nu) := \langle T_\lambda, M_{XY}\,\rangle \tag{12.8}$$

where $\lambda > 0$,

$$T_\lambda := \operatorname*{arg\,min}_{T\in U_{nm}} \langle T, M_{XY}\,\rangle - \frac{1}{\lambda}E(T), \tag{12.9}$$

and $E(T)$ is the entropy of T seen as a discrete joint probability distribution, namely

$$E(T) := -\sum_{ij} T_{ij}\log(T_{ij}).$$

This regularization has two benefits: *(i)* because the entropic penalization term in Equation (12.9) is 1-strongly convex with respect to the ℓ_1 norm, the regularized problem is itself strongly convex and admits a unique optimal solution T_λ (as opposed to the initial OT problem, for which the minimizer may not be unique); *(ii)* the optimal solution T_λ in Equation (12.9) is a diagonal scaling of $e^{-M_{XY}}$, the element-wise exponential matrix of $-M_{XY}$. Indeed, one can easily show using the Lagrange method of multipliers that there must exist two non-negative vectors $u \in \mathbb{R}^n, v \in \mathbb{R}^m$ such that $T_\lambda := D_u e^{-M_{XY}} D_v$, where D_u D_v are diagonal matrices with u and v on their diagonal. The solution to this diagonal scaling problem can be found efficiently through Sinkhorn's algorithm [193], which has a linear convergence rate [72]. Sinkhorn's algorithm can be implemented in a few lines of code that only require matrix vector products and elementary operations, which can all be easily parallelized on modern hardware.

12.3.3 Smoothing the Wasserstein Distance to Energy Distance

An interesting class of modern tests are distance-based "energy statistics" as introduced in parallel by [16] and [207] (and generalized to other metrics, for a related independence testing problem, by [133]). The test statistic is called the *Cramer statistic* by the former paper but we use the term *Energy Distance* as done by the latter, and corresponds to the population quantity

$$\mathrm{ED} := 2\mathbb{E}\|X - Y\| - \mathbb{E}\|X - X'\| - \mathbb{E}\|Y - Y'\|$$

(where our convention is always $X, X' \sim P$ and $Y, Y' \sim Q$). An unbiased and a biased test statistic can be calculated as

$$\begin{aligned}
\mathrm{ED}_u &:= \frac{2}{mn}\sum_{i=1}^{n}\sum_{j=1}^{m}\|X_i - Y_j\| - \frac{1}{n(n-1)}\sum_{i\neq j=1}^{n}\|X_i - X_j\| - \frac{1}{m(m-1)}\sum_{i\neq j=1}^{m}\|Y_i - Y_j\| \\
\mathrm{ED}_b &:= \frac{2}{mn}\sum_{i=1}^{n}\sum_{j=1}^{m}\|X_i - Y_j\| - \frac{1}{n^2}\sum_{i,j=1}^{n}\|X_i - X_j\| - \frac{1}{m^2}\sum_{i,j=1}^{m}\|Y_i - Y_j\| \qquad (12.10)
\end{aligned}$$

Appropriately thresholding ED_u or ED_b leads to a test which is consistent (in the classical sense) against all fixed (and some local) alternatives where $P \neq Q$ under very general conditions (natural restrictions do exist, like finiteness of $\mathbb{E}[X], \mathbb{E}[Y]$ and so on) and such results can be found in the associated references.

Writing, as we did in Section 12.3.1, $\mu = \frac{1}{n}\sum_{i=1}^{n}\delta_{X_i}$ and $\nu = \frac{1}{n}\sum_{j=1}^{m}\delta_{Y_j}$ for the empirical measures corresponding to the samples of P and Q, the Sinkhorn divergence defined in Equation (12.8) can be linked to the the energy distance through the following formula

$$\mathrm{ED}_u = \frac{n-1}{n}\lim_{\lambda\to 0}\left(2S_\lambda^1(\mu,\nu) - S_\lambda^1(\mu,\mu) - S_\lambda^1(\nu,\nu)\right)$$

This can proved by noticing that $T_0 := \lim_{\lambda\to 0} T_\lambda = ab^T$, namely the maximal entropy table in $U(a, b)$ is the tensor product of the marginals a and b. Following this, we have that

$$S_0^1(\mu,\nu) = \frac{1}{nm}\sum_{ij}\|X_i - Y_j\|$$

and we recover the four terms described in Equation (12.10).

12.3.4 From Energy Distance to Kernel Maximum Mean Discrepancy

Another popular class of tests that has emerged over the last decade, are kernel-based tests introduced independently by [85] and [70], and expanded on in [82]. Without getting into technicalities that are irrelevant for this chapter, the *Maximum Mean Discrepancy* between P, Q is defined as

$$\mathrm{MMD}(H_k, P, Q) := \max_{\|f\|_{\mathcal{H}_k}\leq 1} \mathbb{E}_P f(X) - \mathbb{E}_Q f(Y)$$

where $\mathcal{H}_k$ is a Reproducing Kernel Hilbert Space associated with Mercer kernel $k(\cdot,\cdot)$, and $\|f\|_{\mathcal{H}_k} \leq 1$ is its unit norm ball. While it is easy to see that $\mathrm{MMD} \geq 0$ always, and also that $P = Q$ implies $\mathrm{MMD} = 0$, [85] show that if k is "characteristic", the equality holds iff $P = Q$ (see their paper for more details; the Gaussian kernel $k(a, b) = \exp(-\|a - b\|^2/\gamma^2)$ is a popular example).

One can easily argue that

$$\mathrm{MMD}(\mathcal{H}_k, P, Q) = \|\mathbb{E}_P k(X, .) - \mathbb{E}_Q k(Y, .)\|_{\mathcal{H}_k}$$

and hence by the reproducing property,

$$\mathrm{MMD}^2 = \mathbb{E}k(X, X') + \mathbb{E}k(Y, Y') - 2\mathbb{E}k(X, Y)$$

This gives rise to a natural associated test, that involves thresholding the following unbiased estimator of MMD^2:

$$\mathrm{MMD}_u^2(k(\cdot,\cdot)) \;\; := \;\; \frac{1}{n(n-1)} \sum_{i \neq j = 1}^{n} k(X_i, X_j) + \frac{1}{m(m-1)} \sum_{i \neq j = 1}^{m} k(Y_i, Y_j) - \frac{2}{mn} \sum_{i=1}^{n} \sum_{j=1}^{m} k(X_i, Y_j)$$

Apart from the fact that $\mathrm{MMD}(P, Q) = 0$ iff $P = Q$ (which is also satisfied by the KL-divergence, for example) the other fact that makes this a useful test statistic is that its estimation error, i.e. the error of MMD_u^2 in estimating MMD^2, scales like $1/\sqrt{n}$, independent of d (unlike the KL-divergence which is in general hard to estimate in high dimensions). See [85] for a detailed proof of this fact.

At first sight, the Energy Distance and the MMD look like fairly different tests. However, there is a natural connection that proceeds in two steps. Firstly, there is no reason to stick to only the Euclidean norm $\|\cdot\|_2$ to measure distances for ED — the test can be extended to other norms, and in fact also other metric spaces (where the corresponding metric replaces the Euclidean distance in the calculation of the test statistic); [133] explains the details for the closely related independence testing problem. Following that, [187] discuss the relationship between distances and kernels (again for independence testing, but the same arguments hold in the two sample testing setting also), and show that there is a (nearly) one-to-one mapping between these two concepts and corresponding test statistics. Loosely speaking, for every kernel, there exists a metric such that MMD with that kernel equals ED with that metric, and also vice versa. This is a very strong connection between these two families of tests.

12.4 Wasserstein Distance and QQ plots ($d = 1$)

We recall that in general, for $p \in [1, \infty)$ the p-Wasserstein distance between two probability measures P, Q on $\mathbb{R}$ with finite p-moments is given by

$$W_p(P, Q) := \inf_{\pi \in \Gamma(P,Q)} \left(\int_{\mathbb{R}\times\mathbb{R}} \|x - y\|^p d\pi(x, y) \right)^{1/p}. \tag{12.11}$$

Because the Wasserstein distance measures the cost of transporting mass from the original distribution P into the target distribution Q, one can say that it measures "horizontal" discrepancies between P and Q. Intuitively, two probability distributions P and Q that are different over "long" (horizontal) regions will be far away from each other in the Wasserstein distance sense, because in that case mass has to travel long distances to go from the original distribution to the target distribution.

In the one dimensional case (in contrast with what happens in dimension $d \geq 2$), the p-Wasserstein distance has a simple interpretation in terms of the quantile functions F^{-1} and G^{-1} of P and Q respectively. The reason for this is that the optimal way to transport mass from P to Q has to satisfy certain monotonicity property which we describe in the proof of the following Lemma.

Proposition 44. *The p-Wasserstein distance between two probability measures P and Q on $\mathbb{R}$ with p-finite moments can be written as*

$$W_p^p(P, Q) = \int_0^1 |F^{-1}(t) - G^{-1}(t)|^p dt.$$

Proof: This is a well known fact that can be found in [210], nevertheless here we present its proof for the sake of completeness. We first observe that the *infimum* in the definition of $W_p(P,Q)$ can be replaced by *minimum*, namely, there exists a transportation plan $\pi \in \Gamma(P,Q)$ that achieves the infimum in (12.11). This can be deduced in a straightforward way by noting that the expression $\int_{\mathbb{R}\times\mathbb{R}}|x-y|^p d\pi(x,y)$ is linear in π and that the set $\Gamma(P,Q)$ is compact in the sense of weak convergence of probability measures on $\mathbb{R}\times\mathbb{R}$. Let us denote by π^* an element in $\Gamma(P,Q)$ realizing the minimum in (12.11). Let $(x_1,y_1) \in \mathrm{supp}(\pi^*)$ and $(x_2,y_2) \in \mathrm{supp}(\pi^*)$ and suppose that $x_1 < x_2$. We claim that the optimality of π^* implies that $y_1 \leq y_2$. To see this, suppose for the sake of contradiction that this is not the case, that is, suppose that $y_2 < y_1$. We claim that in that case

$$|x_1 - y_2|^p + |x_2 - y_1|^p < |x_1 - y_1|^p + |x_2 - y_2|^p. \tag{12.12}$$

Note that for $p = 1$ this follows in a straightforward way. For the case $p > 1$, first note that $x_1 < x_2$ and $y_2 < y_1$ imply that there exists $t \in (0,1)$ such that $tx_1 + (1-t)y_1 = tx_2 + (1-t)y_2$. Now, note that

$$|x_1 - y_2| = |x_1 - (tx_1 + (1-t)y_1)| + |(tx_1 + (1-t)y_1) - y_2|$$

because the points x_1, y_2 and $tx_1 + (1-t)y_1$ all lie on the same line segment. But then, using the fact that $tx_1 + (1-t)y_1 = tx_2 + (1-t)y_2$, we can rewrite the previous expression as

$$|x_1 - y_2| = (1-t)|x_1 - y_1| + t|y_2 - x_2|.$$

Using the strict convexity of the function $t \mapsto t^p$ (when $p > 1$), we deduce that

$$|x_1 - y_2|^p < (1-t)|x_1 - y_1|^p + t|x_2 - y_2|^p.$$

In a similar fashion, we obtain

$$|x_2 - y_1|^p < t|x_1 - y_1|^p + (1-t)|x_2 - y_2|^p.$$

Adding the previous two inequalities we obtain (12.12). However, we notice that this inequality contradicts the optimality of π^*, because it shows that π^* is not *cyclically monotone*, which essentially means that it is possible to rearrange the way mass is transported from P to Q by π^* in order to reduce the transportation cost. Therefore, we conclude that if $(x_1,y_1) \in \mathrm{supp}(\pi^*)$ and $(x_2,y_2) \in \pi^*$ and $x_1 < x_2$, then $y_1 \leq y_2$. The previous monotonicity property of π^*, together with the fact that $\pi^* \in \Gamma(P,Q)$ imply that if $x \in \mathrm{supp}(P)$ and $y \in \mathrm{supp}(Q)$ then $(x,y) \in \mathrm{supp}(\pi^*)$ if and only if $F(x) = G(y)$. From this fact we conclude that

$$\int_{\mathbb{R}\times\mathbb{R}} |x-y|^p d\pi^*(x,y) = \int_{\mathrm{supp}(\pi^*)} |x-y|^p d\pi^*(x,y) = \int_0^1 |F^{-1}(t) - G^{-1}(t)|^p dt,$$

as we wanted to show.

Having consider the p-Wasserstein distance $W_p(P,Q)$ for $p \in [1,\infty)$, we turn to the case $p = \infty$. Let P, Q be two probability measures on $\mathbb{R}$ with bounded support. That is, assume that there exist a number $N > 0$ such that $\mathrm{supp}(P) \subseteq [-N,N]$ and $\mathrm{supp}(Q) \subseteq [-N,N]$. We define the ∞-Wasserstein distance between P and Q by

$$W_\infty(P,Q) := \inf_{\pi\in\Gamma(P,Q)} esssup_\pi |x-y|.$$

Proceeding as in the case $p \in [1,\infty)$, it is possible to show that the ∞-Wasserstein distance between P and Q with bounded supports can be written in terms of the difference of the corresponding quantile functions as

$$W_\infty(P,Q) = \|F^{-1} - G^{-1}\|_\infty.$$

12.5 A Distribution-Free Wasserstein Test

As we saw in Section 12.2.2, under the null hypothesis $H_0 : P = Q$, the statistic

$$\frac{mn}{m+n}\int_0^1 \left(F_n^{-1}(t) - G_m^{-1}(t)\right)^2 dt$$

has an asymptotic distribution which is not distribution free, i.e., it depends on F. We also saw that as opposed to what happens with the asymptotic distribution of the L^∞ distance between F_n and G_m, the asymptotic distribution of $\|F_n^{-1} - G_m^{-1}\|_\infty$ does depend on the cdf F.

In this section we introduce the ROC and ODC curves associated to two distributions. The ultimate goal is to relate those curves to a distribution-free Wasserstein test.

12.5.1 ROC and ODC curves

Let P and Q be two distributions on $\mathbb{R}$ with cdfs F and G and quantile functions F^{-1} and G^{-1} respectively. We define the *ROC* curve between F and G as the function.

$$ROC(t) := 1 - F(G^{-1}(1-t)), \quad t \in [0,1].$$

In addition, we define their *ODC* curve by,

$$ODC(t) := G(F^{-1}(t)), \quad t \in [0,1].$$

We observe that the ROC curve can be obtained from the ODC curve after reversing the axes. In addition, the following are straightforward properties of the ROC curve (see [103]).

1. The ROC curve is increasing and $ROC(0) = 0$, $ROC(1) = 1$.
2. If $G(t) \geq F(t)$ for all t then $ROC(t) \geq t$ for all t.
3. If F and G have densities with monotone likelihood ratio, then the ROC curve is concave.
4. The area under the ROC curve is equal to $\mathbb{P}(Y < X)$, where $Y \sim Q$ and $X \sim P$.

Intuitively speaking, the faster the ROC curve increases towards the value 1, the easier it is to distinguish the distributions P and Q. Given that the ROC curve can be obtained from the ODC curve by reversing the axes, we focus from this point on the ODC curve.

The first observation about the ODC curve is that it can be regarded as the quantile function of the distribution $G_\sharp P$ (the push forward of P by G) on $[0,1]$ which is defined by

$$G_\sharp P([0,\alpha)) := P\left(G^{-1}\left([0,\alpha)\right)\right), \quad \alpha \in [0,1].$$

Similarly, for $X_1, \dots, X_n$ and $Y_1, \dots, Y_m$ two independent samples drawn from P and Q respectively, we can consider the measure $G_{m\sharp}P_n$, that is, the push forward of P_n by G_m. We note that the empirical ODC curve $G_m \circ F_n^{-1}$ is the quantile function of $G_{m\sharp}P_n$.

From the results of the previous section, we deduce that

$$W_p^p(G_{m\sharp}P_n, G_\sharp P) = \int_0^1 |G_m \circ F_n^{-1}(t) - G \circ F^{-1}(t)|^p dt$$

for every $p \in [1,\infty)$ and also

$$W_\infty(G_{m\sharp}P_n, G_\sharp P) = ||G_m \circ F_n^{-1} - G \circ F^{-1}||_\infty.$$

That is, the p-Wasserstein distance between the measures $G_{m\sharp}P_n$ and $G_{\sharp}P$ can be computed by considering the L^p distance of the ODC curve and its empirical version.

First we observe that under the null hypothesis $H_0 : P = Q$, the distribution of empirical ODC curve is actually independent of P. In particular, $W_p^p(G_{m\sharp}P_n, G_{\sharp}P)$ and $W_\infty(G_{m\sharp}P_n, G_{\sharp}P)$ are distribution free. This is the content of the next lemma.

Lemma 34 (Reduction to uniform distribution)**.** *Let F be a continuous and strictly increasing cdf and let $X_1, \dots, X_n$ and $Y_1, \dots, Y_m$ be two independent samples of F. Consider*

$$\widetilde{U}_k := F(X_k),$$

and

$$\widehat{U}_k := F(Y_k).$$

Let $\widehat{G}_m$ be the c.d.f. associated to $\widehat{U}_1, \dots, \widehat{U}_m$. Then,

$$G_m(X_k) = \widehat{G}_m(\widetilde{U}_k), \quad \forall k \in \{1, \dots, n\}.$$

In particular,

$$G_m \circ F_n^{-1}(t) = \widehat{G}_m \circ \widetilde{F}_n^{-1}(t), \quad \forall t \in [0,1].$$

Proof: We denote by $Y_{(1)} \leq \cdots \leq Y_{(m)}$ the order statistic associated to the Ys. For $k = 1, \dots, m-1$ and $t \in (0,1)$, we have $G_m(t) = \frac{k}{m}$ if and only if $t \in [Y_{(k)}, Y_{(k+1)})$ which holds if and only if $t \in [F^{-1}(\widehat{U}_{(k)}), F^{-1}(\widehat{U}_{(k+1)}))$, which in turn is equivalent to $F(t) \in [\widehat{U}_{(k)}, \widehat{U}_{(k+1)})$. Thus, $G_m(t) = \frac{k}{m}$ if and only if $\widehat{G}_m(F(t)) = \frac{k}{m}$. From the previous observations we conclude that $G_m = \widehat{G}_m \circ F$. Finally, since $X_k = F^{-1}(\widetilde{U}_k)$ we conclude that

$$G_m(X_k) = \widehat{G}_m \circ F \circ F^{-1}(\widetilde{U}_k) = \widehat{G}_m(\widetilde{U}_k).$$

Now we establish a result related to the asymptotic distribution of $W_p^p(G_{m\sharp}P_n, G_{\sharp}P)$ and $W_\infty(G_{m\sharp}P_n, G_{\sharp}P)$. We do this by first considering the asymptotic distribution of the difference between the empirical ODC curve and the population ODC curve regarding both of them as elements in the space $\mathcal{D}([0,1])$. This is the content of the following Theorem which follows directly from the work of [117] (see [103]).

Theorem 45. *Suppose that F and G are two cdfs with densities f, g satisfying*

$$\frac{g(F^{-1}(t))}{f(F^{-1}(t))} \leq C,$$

for all $t \in [0,1]$. Also, assume that

$$\frac{n}{m} \to \lambda \in [0, \infty)$$

as $n, m \to \infty$. Then,

$$\sqrt{\frac{mn}{m+n}} \left(G_m(F_n^{-1}(\cdot)) - G(F^{-1}(\cdot)) \right) \to_w \sqrt{\frac{\lambda}{\lambda+1}} B_1(G \circ F^{-1}(\cdot)) + \sqrt{\frac{1}{\lambda+1}} \frac{g(F^{-1}(\cdot))}{f(F^{-1}(\cdot))} B_2(\cdot),$$

where B_1 and B_2 are two independent Brownian bridges and where the weak convergence must be interpreted as weak convergence in the space of probability measures on the space $\mathcal{D}([0,1])$.

As a corollary, under the null hypothesis $H_0 : P = Q$ we obtain the following. Suppose that F is a continuous, strictly increasing cdf. Then,

$$\frac{mn}{n+m} W_2^2(G_{m\sharp}P_n, G_\sharp P) = \frac{mn}{n+m} \int_0^1 (G_m(F_n^{-1}(t)) - t)^2 dt \to_w \int_0^1 (\mathbb{B}(t))^2 dt$$

and

$$\sqrt{\frac{mn}{n+m}} W_\infty(G_{m\sharp}P_n, G_\sharp P) = \sqrt{\frac{mn}{n+m}} \sup_{t\in[0,1]} |G_m(F_n^{-1}(t)) - t| \to_w \sup_{t\in[0,1]} |\mathbb{B}(t)|.$$

To see this, note that by Lemma 34 it suffices to consider $F(t) = t$ in $[0, 1]$. In that case, the assumptions of Theorem 45 are satisfied and the result follows directly.

Conclusion

In this chapter, we connect a wide variety of univariate and multivariate test statistics, with the central piece being the Wasserstein two-sample test statistic. The Wasserstein statistic is closely related to univariate tests like Kolmogorov-Smirnov, Cramer-von-Mises, and Anderson Darling, QQ plots and a distribution-free variant of the test is connected to ROC curves. Through entropic smoothing, the Wasserstein test is also related to the multivariate tests of Energy Distance and Kernel Maximum Mean Discrepancy. We hope that this is a useful resource to connect the seemingly vastly different families of two sample tests.

Chapter 13

Nonparametric testing : Stein shrinkage for kernel independence testing

This chapter[1] deals with the problem of nonparametric independence testing, a fundamental decision-theoretic problem that asks if two arbitrary (possibly multivariate) random variables X, Y are independent or not, a question that comes up in many fields like causality and neuroscience. While quantities like correlation of X, Y only test for (univariate) linear independence, natural alternatives like mutual information of X, Y are hard to estimate due to a serious curse of dimensionality. A recent approach, avoiding both issues, estimates norms of an *operator* in Reproducing Kernel Hilbert Spaces (RKHSs). Our main contribution is strong empirical evidence that by employing *shrunk* operators when the sample size is small, one can attain an improvement in power at low false positive rates. We analyze the effects of Stein shrinkage on a popular test statistic called HSIC (Hilbert-Schmidt Independence Criterion). Our observations provide insights into two recently proposed shrinkage estimators, SCOSE and FCOSE - we prove that SCOSE is (essentially) the optimal linear shrinkage method for *estimating* the true operator; however, the non-linearly shrunk FCOSE usually achieves greater improvements in *test power*. This work is important for more powerful nonparametric detection of subtle nonlinear dependencies for small samples.

13.1 Introduction

The problem of *nonparametric* independence testing deals with ascertaining if two random variables are independent or not, making no parametric assumptions about their underlying distributions. Formally, given n samples (x_i, y_i) for $i \in \{1, ..., n\}$ where $x_i \in \mathbb{R}^p, y_i \in \mathbb{R}^q$, that are drawn from a joint distribution P_{XY} supported on $\mathcal{X} \times \mathcal{Y} \subseteq \mathbb{R}^{p+q}$, we want to decide between the *null* and *alternate* hypotheses

$$\mathcal{H}_0 : P_{XY} = P_X \times P_Y \text{ vs. } \mathcal{H}_1 : P_{XY} \neq P_X \times P_Y$$

where P_X, P_Y are the marginals of P_{XY} w.r.t. X, Y. A test is a function from the data to $\{0, 1\}$. Tests aim to have high power (probability of detecting dependence, when it exists) at a prespecified allowable type-1 error rate α (probability of detecting dependence when there isn't any).

Independence testing is often a precursor to further analysis. Consider for instance conditional independence testing for inferring causality, say by the PC algorithm [198], whose first step is (unconditional) independence testing. It is also useful for scientific discovery like in neuroscience, to see if a stimulus X (say an image) is independent of the brain activity Y (say fMRI) in a relevant part of the brain. Since

[1] See Ramdas* and Wehbe* [163].

detecting nonlinear correlations is much easier than *estimating* a nonparametric regression function (of Y onto X), it can be done at smaller sample sizes, with further samples collected for estimation only if an effect is detected by the hypothesis test. For such situations, correlation only tests for univariate linear independence, while other statistics like mutual information that do characterize multivariate independence are hard to estimate from data, suffering from a serious curse of dimensionality. A recent popular approach for this problem (and a related two-sample testing problem) involve the use of quantities defined in reproducing kernel Hilbert spaces (RKHSs) - see [66, 81, 82, 84].

This chapter will concern itself with increasing the statistical power at small samples of a popular kernel statistic called HSIC, by using *shrunk* empirical estimators of the unknown population quantity (introduced below).

13.1.1 Hilbert Schmidt Independence Criterion

Due to limited space, familiarity with RKHS terminology is assumed - see [184] for an introduction. Let $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ and $l : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ be two positive-definite reproducing kernels that correspond to RKHSs $\mathcal{H}_k$ and $\mathcal{H}_l$ respectively with inner-products $\langle \cdot, \cdot \rangle_k$ and $\langle \cdot, \cdot \rangle_l$. Let k, l arise from (implicit) feature maps $\phi : \mathcal{X} \to \mathcal{H}_k$ and $\psi : \mathcal{Y} \to \mathcal{H}_l$. In other words, ϕ, ψ are not functions, but mappings to the Hilbert space. i.e. $\phi(x) \in \mathcal{H}_k, \psi(y) \in \mathcal{H}_l$ respectively. These functions, when evaluated at points in the original spaces, must satisfy $\phi(x)(x') = \langle \phi(x), \phi(x') \rangle_k = k(x, x')$ and $\psi(y)(y') = \langle \psi(y), \psi(y') \rangle_l = l(y, y')$.

The mean embedding of P_X and P_Y are defined as $\mu_X := \mathbb{E}_{x \sim P_X} \phi(x) \in \mathcal{H}_k$ and $\mu_Y := \mathbb{E}_{y \sim P_Y} \psi(y) \in \mathcal{H}_l$ whose empirical estimates are $\widehat{\mu}_X := \frac{1}{n} \sum_{i=1}^n \phi(x_i)$ and $\widehat{\mu}_Y := \frac{1}{n} \sum_{i=1}^n \psi(y_i)$. Finally, the cross-covariance operator of X, Y is defined as

$$\Sigma_{XY} := \mathbb{E}_{(x,y) \sim P_{XY}} (\phi(x) - \mu_X) \otimes (\psi(y) - \mu_Y)$$

where $\otimes$ is an outer-product. For unfamiliar readers, if we used the linear kernel $k(x, x') = x^T x'$ and $l(y, y') = y^T y'$, then the cross-covariance operator is just the cross-covariance matrix. The plug-in empirical estimator of Σ_{XY} is

$$S_{XY} := \frac{1}{n} \sum_{i=1}^n (\phi(x_i) - \widehat{\mu}_X) \otimes (\psi(y_i) - \widehat{\mu}_Y)$$

For conciseness, define $\widetilde{\phi}(x_i) = \phi(x_i) - \widehat{\mu}_X$, $\widetilde{\psi}(y_i) = \psi(y_i) - \widehat{\mu}_Y$, $\widetilde{k}(x, x') = \langle \widetilde{\phi}(x), \widetilde{\phi}(x') \rangle_k$ and $\widetilde{l}(y, y') = \langle \widetilde{\psi}(y), \widetilde{\psi}(y') \rangle_l$. The test statistic Hilbert-Schmidt Independence Criterion (HSIC) defined in [81] is the squared Hilbert-Schmidt norm of S_{XY}, and can be calculated using centered kernel matrices $\widetilde{K}, \widetilde{L}$, where $\widetilde{K}_{ij} = \widetilde{k}(x_i, x_j), \widetilde{L}_{ij} = \widetilde{l}(y_i, y_j)$, as

$$\text{HSIC} := \|S_{XY}\|_{HS}^2 = \frac{1}{n^2} \text{tr}(\widetilde{K}\widetilde{L}) \tag{13.1}$$

For unfamiliar readers, if we used the linear kernel, this just corresponds to the Frobenius norm of the cross-covariance matrix. The most important property is: *when the kernels k, l are "characteristic", then the corresponding population statistic $\|\Sigma_{XY}\|_{HS}^2$ is zero iff X, Y are independent* [81]. This gives rise to a natural test - calculate $\|S_{XY}\|_{HS}^2$ and reject the null if it is large.

Examples of characteristic kernels include Gaussian $k(x, x') = \exp\left(-\frac{\|x-x'\|_2^2}{\gamma^2}\right)$ and Laplace $k(x, x') = \exp\left(-\frac{\|x-x'\|_1}{\gamma}\right)$, for any bandwidth γ, while the aforementioned linear kernel is not characteristic — the corresponding HSIC tests only linear relationships, and a zero cross-covariance matrix characterizes independence only for multivariate Gaussian distributions. Working with the infinite dimensional operator with characteristic kernels, allows us to identify any general nonlinear dependence (in the limit) between any pair of distributions, not just Gaussians.

13.1.2 Independence Testing using HSIC

A permutation-based test is described in [81], and proceeds in the following manner. From the given data, calculate the test statistic $T := \|S_{XY}\|_{HS}^2$. Keeping the order of $x_1, ..., x_n$ fixed, randomly permute $y_1, ..., y_n$ a large number of times, and recompute the *permuted* HSIC each time. This destroyed any dependence between x, y simulating a draw from the product of marginals, making the empirical distribution of the permuted HSICs behave like the null distribution of the test statistic (distribution of HSIC when $\mathcal{H}_0$ is true). For a pre-specified type-1 error α, calculate threshold t_α in the right tail of the null distribution. Reject $\mathcal{H}_0$ if $T > t_\alpha$. This test was proved to be *consistent* against any fixed alternative, meaning for any fixed type-1 error α, the power goes to 1 as $n \to \infty$. Empirically, the power can be calculated using simulations by repeating the above permutation test many times for a fixed P_{XY} (for which dependence holds), and reporting the empirical probability of rejecting the null (detecting the dependence). Note that the power depends on P_{XY} (unknown to the user of the test).

Shrunk Estimators of S_{XY}

Even though S_{XY} is an unbiased estimator of Σ_{XY}, it typically has high variance at low sample sizes. The idea of Stein shrinkage [203] is to trade-off bias and variance, first introduced in the context of Gaussian mean estimation. This strategy of introducing some bias and decreasing the variance to get different estimators of Σ_{XY} was followed by [139] who define a linear shrinkage estimator of S_{XY} called SCOSE (Simple Covariance Shrinkage Estimator) and a nonlinear shrinkage estimator called FCOSE (Flexible Covariance Shrinkage Estimator). When we refer to shrunk estimators, we implicitly mean SCOSE and FCOSE. We will describe these briefly in the next section.

Contributions

Our first contribution is the following :

1. We provide evidence that employing shrunk estimators of Σ_{XY}, instead of S_{XY}, to calculate the aforementioned test statistic, can increase the power of the associated independence test at low false positive rates, when the sample size is small (there is higher variance in estimating infinite-dimensional operators).

Our second contribution is to analyze the effect of shrinkage on the test statistic, to provide some practical insight.

2. The effect of shrinkage on the test-statistic is very similar to soft-thresholding, shrinking very small statistics to zero, and shrinking other values nearly (but not) linearly, and nearly (but not) monotonically.

Our last contribution is an insight on the two estimators considered in this chapter, SCOSE and FCOSE.

3. We prove that SCOSE is (essentially, up to lower order terms) the optimal/oracle linear shrinkage estimator with respect to quadratic risk. However, we observe that FCOSE typically achieves higher power than SCOSE. This indicates that it may be useful to search for the optimal estimator in a larger class than linearly shrunk estimators, and also that quadratic loss may not be the right loss function for the purposes of test power.

The rest of this chapter is organized as follows. Section 13.2 introduces SCOSE, FCOSE and their corresponding shrunk test statistics. Section 13.4 presents illuminating experiments that bring out the statistically significant improvement in power over HSIC. Section 13.5 conducts a deeper investigation into the effect of shrinkage and proves the oracle optimality of SCOSE under quadratic risk.

13.2 Shrunk Estimators and Test Statistics

Let $\mathcal{HS}(\mathcal{H}_k, \mathcal{H}_l)$ represent the set of Hilbert-Schmidt operators from $\mathcal{H}_k$ to $\mathcal{H}_l$. We first note that S_{XY} can be written as the solution to the following optimization problem.

$$S_{XY} := \min_{Z \in \mathcal{HS}(\mathcal{H}_k,\mathcal{H}_l)} \frac{1}{n}\sum_{i=1}^{n} \left\| \widetilde{\phi}(x_i) \otimes \widetilde{\psi}(y_i) - Z \right\|_{HS}^2$$

Using this idea [139] suggest the following two shrunk/regularized estimators.

From SCOSE to HSIC^S

This is derived in [139] by solving

$$\min_{Z \in \mathcal{HS}(\mathcal{H}_k,\mathcal{H}_l)} \frac{1}{n}\sum_{i=1}^{n} \left\| \widetilde{\phi}(x_i) \otimes \widetilde{\psi}(y_i) - Z \right\|_{HS}^2 + \lambda \|Z\|_{HS}^2$$

and the optimal solution (called SCOSE) is

$$S_{XY}^S := \left(1 - \frac{\lambda}{1+\lambda}\right) S_{XY}$$

where λ (and hence the shrinkage intensity) is estimated by leave-one-out cross-validation (LOOCV), in closed form as

$$\begin{aligned}\rho^S &:= \left(\frac{\lambda^{CV}}{1+\lambda^{CV}}\right) \\ &= \frac{\left[\frac{1}{n}\sum_{i=1}^n \widetilde{K}_{ii}\widetilde{L}_{ii} - \frac{1}{n^2}\sum_{i,j=1}^n \widetilde{K}_{ij}\widetilde{L}_{ij}\right]}{(n-2)\frac{1}{n^2}\sum_{i,j=1}^n \widetilde{K}_{ij}\widetilde{L}_{ij} + \frac{1}{n^2}\sum_{i=1}^n \widetilde{K}_{ii}\widetilde{L}_{ii}}\end{aligned}$$

Observing the expression for λ^{CV} in [139], the denominator can be negative (for example, with the Gaussian kernel for small bandwidths, resulting in a kernel matrix close to the identity). This can cause λ^{CV} to be negative, and ρ^S to be (unintentionally) outside the range $[0, 1]$. Though not discussed in [139], we shall follow the convention that when $\rho^S < 0$, we shall use $\rho^S = 0$ and if $\rho^S > 1$, we use $\rho_S = 1$. Indeed, one can show that $\left(1 - \frac{\lambda}{1+\lambda}\right)_+ S_{XY}$ dominates $\left(1 - \frac{\lambda}{1+\lambda}\right) S_{XY}$ where $(x)_+ = \max\{x, 0\}$. We later prove that S_{XY}^S is (essentially) the optimal/oracle linear shrinkage estimator with respect to quadratic risk.

We can now calculate the corresponding shrunk statistic HSIC^S $= \|S_{XY}^S\|_{HS}^2 =$

$$\left(1 - \frac{\frac{1}{n}\sum_{i=1}^n \widetilde{K}_{ii}\widetilde{L}_{ii} - \text{HSIC}}{(n-2)\text{HSIC} + \frac{\frac{1}{n}\sum_{i=1}^n \widetilde{K}_{ii}\widetilde{L}_{ii}}{n}}\right)_+^2 \text{HSIC} \qquad (13.2)$$

While the above expression looks daunting, one thing to note is that the amount that HSIC is shrunk (i.e. the multiplicative factor) depends on the value of HSIC. As we shall later, small HSIC values get shrunk to zero, but as can be seen above, the shrinkage of HSIC is non-monotonic.

From FCOSE to HSIC^F

The Flexible Covariance Shrinkage Estimator is derived by relying on the Representer theorem, see [184], to instead minimize

$$\frac{1}{n}\sum_{i=1}^{n}\left\|\widetilde{\phi}(x_i)\otimes\widetilde{\psi}(y_i)-\sum_{i=1}^{n}\frac{\beta_i}{n}\widetilde{\phi}(x_i)\otimes\widetilde{\psi}(y_i)\right\|_{HS}^{2}+\lambda\|\beta\|_2^2$$

over all $\beta\in\mathbb{R}^n$, and the optimal solution (called FCOSE) is

$$\begin{aligned} S_{XY}^{F} &:= \sum_{i=1}^{n}\frac{\beta_i^{\lambda}}{n}\widetilde{\phi}(x_i)\otimes\widetilde{\psi}(y_i)\\ \text{where } \beta^{\lambda} &= (\widetilde{K}\circ\widetilde{L}+\lambda I)^{-1}\widetilde{K}\circ\widetilde{L}\mathbf{1}\end{aligned}$$

where $\circ$ denotes elementwise (Hadamard) product, $\mathbf{1}$ is the vector $[1,1,...,1]^T$, and as before the best λ is determined by LOOCV. The procedure to evaluate the optimal λ efficiently is described by [139] - a single eigenvalue decomposition of $\widetilde{K}\circ\widetilde{L}$ costing $O(n^3)$ can be done, following which evaluating LOOCV is only $O(n^2)$ per λ, see [139], section 3.1 for more details. As before, after picking the λ by LOOCV, we can derive the corresponding shrunk test statistic as

$$\begin{aligned}\text{HSIC}^F &= \|S_{XY}^{S}\|_{HS}^2\\ &= \frac{1}{n^2}\text{tr}(M(M+\lambda I)^{-1}M(M+\lambda I)^{-1}M)\end{aligned}$$

where $M=\widetilde{K}\circ\widetilde{L}$. Note here that the shrinkage is not linear, and the effect on HSIC cannot be seen immediately. Similar to SCOSE, as we shall later see, small HSIC values get shrunk to zero (LOOCV chooses a large λ).

13.3 Linear Shrinkage and Quadratic Risk

In this section, we prove that SCOSE is (essentially) optimal within a particular class of estimators. Such "oracle" arguments also exist elsewhere in the literature, like [121], so we provide only a brief proof outline.

Proposition 46. *The oracle (with respect to quadratic risk) linear shrinkage estimator and intensity is defined as*

$$S^*,\rho^* := \operatorname*{arg\,min}_{Z\in\mathcal{HS},Z=(1-\rho)S_{XY},0\le\rho\le1}\|Z-\Sigma_{XY}\|_{HS}^2$$

and is given by $S^*:=(1-\rho^*)S_{XY}$ *where*

$$\rho^* := \frac{\mathbb{E}\|S_{XY}-\Sigma_{XY}\|_{HS}^2}{\mathbb{E}\|S_{XY}\|^2}$$

Proof: Define $\alpha^2=\|\Sigma_{XY}\|_{HS}^2$, $\beta^2=\mathbb{E}\|S_{XY}-\Sigma_{XY}\|_{HS}^2$, $\delta^2=\mathbb{E}\|S_{XY}\|^2$. Since $\mathbb{E}[S_{XY}]=\Sigma_{XY}$, it is easy to verify that $\alpha^2+\beta^2=\delta^2$. Substituting and expanding the objective, we get:

$$\begin{aligned}\mathbb{E}\|Z-\Sigma_{XY}\|_{HS}^2 &= \mathbb{E}\|-\rho S_{XY}+(S_{XY}-\Sigma_{XY})\|_{HS}^2\\ &= \rho^2\delta^2+\beta^2-2\rho(\delta^2-\alpha^2)\\ &= \rho^2\alpha^2+(1-\rho)^2\beta^2\end{aligned}$$

Differentiating and equating to zero, gives $\rho^* = \frac{\beta^2}{\delta^2}$.

This ρ^* appears in terms of quantities that depend on the unknown underlying distribution (hence the term *oracle* estimator). We use plugin estimates b, d for β, δ. Let $d^2 = \|S_{XY}\|_{HS}^2 = \frac{1}{n^2}\sum_{i,j=1}^n \widetilde{K}_{ij}\widetilde{L}_{ij} = HSIC$. Since β^2 is the variance of S_{XY}, let b^2 be the sample variance of S_{XY}, i.e. $b^2 = \frac{1}{n}\frac{1}{n}\sum_{k=1}^n \widetilde{\phi}(x_i)\otimes \widetilde{\psi}_{x_i} - S_{XY}{}^2 = \frac{1}{n}\left[\frac{1}{n}\sum_{i=1}^n \widetilde{K}_{ii}\widetilde{L}_{ii} - \frac{1}{n^2}\sum_{i,j=1}^n \widetilde{K}_{ij}\widetilde{L}_{ij}\right]$. Plugging these into S^* and simplifying, we see that HSIC$^* := \|S^*\|_{HS}^2$ is

$$\text{HSIC}^* = \left(1 - \frac{\frac{1}{n}\sum_{i=1}^n \widetilde{K}_{ii}\widetilde{L}_{ii} - \text{HSIC}}{n\text{HSIC}}\right)^2 \text{HSIC} \tag{13.3}$$

Comparing Eq.(13.3) with Eq.(13.2) shows that SCOSE is essentially S^*, up to a factor in the denominator which is of the same order as the bias of the HSIC empirical estimator[2] (see Theorem 1 in [81]). In other words, SCOSE just corresponds to using a slightly different estimator for δ^2 than the simple plugin d^2, which varies on the same order as the bias $\delta^2 - \mathbb{E}d^2$. Hence SCOSE, as estimated via regularization and LOOCV, is (essentially) the optimal linear shrinkage estimator under quadratic risk.

To the best of our knowledge, this is the first such characterization of *optimality of an estimator achieved through leave-one-out cross-validation*. We are only able to prove this because one can explicitly calculate both the oracle linear shrinkage intensity ρ^* as well as the optimal λ^{CV} (as mentioned earlier). This raises a natural open question — can we find other situations where the LOOCV estimator is optimal with respect to some risk measure? (perhaps when explicit calculations are not possible, like ridge regression).

13.4 Experiments

In this section, we run three kinds of experiments: a) to verify that SCOSE has better quadratic risk than FCOSE and original sample estimator, b) detailed synthetic experiments to verify that shrinkage does improve power, across interesting regimes of $\alpha = \{0.01, 0.05, 0.1\}$, and c) real data obtained from MNIST, to show that we shrinkage detect dependence at much lower samples than the original data size.

Quadratic Risk

Figure 13.1 shows that SCOSE is indeed much better than both S_{XY} and FCOSE with respect to quadratic risk. Here, we calculate $\mathbb{E}\|Z - \Sigma_{XY}\|_{HS}^2$ for the distribution given in dataset (A) for $Z \in \{S_{XY}, S_{XY}^S, S_{XY}^F\}$. The expectation is calculated by repeating the experiment 1000 times. Each time Z is calculated according to $N \in \{20, 50, 100\}$ samples and Σ_{XY} is approximated by the empirical cross-covariance matrix on 5,000 samples. The four panels use four different kernels which are linear, polynomial, Laplace and Gaussian from top to bottom. The shrunk estimators are always better than the unshrunk, with a larger difference between SCOSE and FCOSE for finite-dimensional feature spaces (top two). In infinite-dimensional feature spaces (bottom two), SCOSE and FCOSE are much better than the unshrunk estimator but very similar to each other. The differences between all estimators decreases with increasing n, since the sample cross-covariance operator itself becomes very accurate.

Synthetic Data

We perform synthetic experiments in a wide variety of settings to demonstrate that the shrunk test statistics achieve higher power than HSIC in a variety of settings. We follow the schema provided in the introduction

[2] HSIC and HSIC $- 2\frac{\text{HSIC}}{n} - \frac{C}{n^2}$ both converge to population HSIC at same rate determined by the dominant term (HSIC).

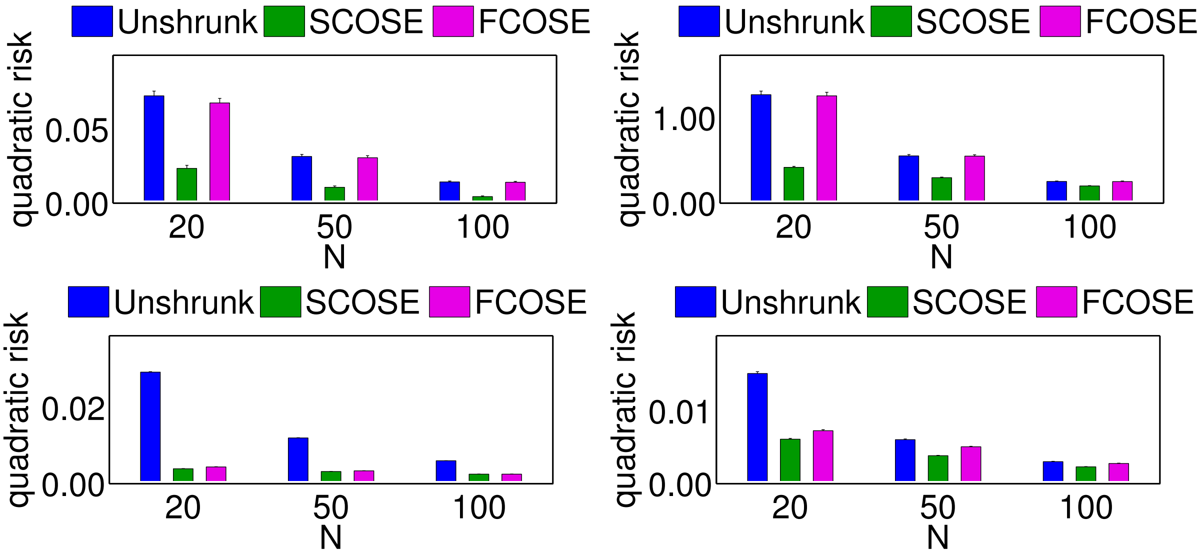


Figure 13.1: All panels show quadratic risk $\mathbb{E}\|X - \Sigma_{XY}\|_{HS}^2$ for $X \in \{S_{XY}, S_{XY}^S, S_{XY}^F\}$. Dataset (A) was used in all four panels, but the kernels were varied - from top to bottom is the linear, quadratic, Gaussian and Laplace kernel.

for independence testing and calculating power. We only consider difficult distributions with nonlinear dependence between X, Y, on which linear methods like correlation are shown to fail to detect dependence (some of them were used in previous papers on independence testing like [86] and [40]).

For all experiments, $\alpha \in \{0.01, 0.05, 0.1\}$ is chosen as the type-1 error (for choosing the threshold level of the null distribution's right tail). For every setting of parameters of each experiment, power is calculated as the percentage of rejection over 200 repetitions (independent trials), with 2000 permutations per repetition (permutation testing to find the null distribution threshold at level α). We use the Gaussian kernel where the bandwidth is chosen by the common median heuristic [184]. Table 13.1 is a representative sample from what we saw on other examples - either large, small or no improvement in power was seen but almost never a worsening of power. The improvements in power may not always be huge, but they are statistically significant - it is difficult to detect such non-linear dependencies at low sample sizes, so *any* increase in power can be important in scientific applications.

Real Data

We use two real datasets - the first is a good example where shrinkage helps a lot, but in the second it does not help (we show it on purpose). Like the synthetic datasets, for most real datasets it either helps or does not hurt (being very rarely worse; see remark in the discussion section).

The first is the Eckerle dataset [60] from the NIST Statistical Reference Datasets (NIST StRD) for Nonlinear Regression, data from a NIST study of circular interference transmittance (n=35, Y is transmittance, X is wavelength). A plot of the data in Figure 13.2 reveals a nonlinear relationship between X, Y (though the correlation is 0.035 with p-value 0.84). We subsample the data to see how often we can detect a relationship at $10\%, 20\%, 30\%$ of the original data size, when the false positive level is always controlled at 0.05. The second is the Aircraft dataset [25] (n=709, X is log(speed), Y is log(span)). Once again, correlation is low, with a p-value of over 0.8, and we subsample the data to $5\%, 10\%, 20\%$ of the original data size.

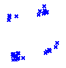
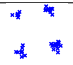
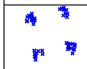
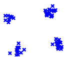
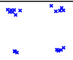
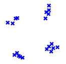
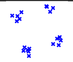
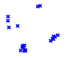
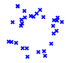
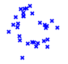
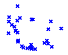
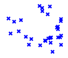
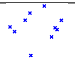
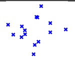
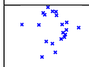
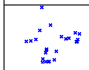
	$\alpha = 0.01$					$\alpha = 0.05$					$\alpha = 0.10$				
	HSIC	HSIC$_S$	HSIC$_F$			HSIC	HSIC$_S$	HSIC$_F$			HSIC	HSIC$_S$	HSIC$_F$		
	0.22 ±0.03	0.21 ±0.03	0.34 ±0.03		✓	0.52 ±0.04	0.52 ±0.04	0.71 ±0.03		✓	0.73 ±0.03	0.72 ±0.03	0.90 ±0.02		✓
	0.41 ±0.03	0.41 ±0.03	0.48 ±0.04		✓	0.68 ±0.03	0.68 ±0.03	0.88 ±0.02		✓	0.85 ±0.03	0.85 ±0.02	0.99 ±0.01		✓
	0.41 ±0.03	0.40 ±0.03	0.52 ±0.04		✓	0.74 ±0.03	0.74 ±0.03	0.94 ±0.02		✓	0.94 ±0.02	0.94 ±0.02	0.99 ±0.01		✓
	0.52 ±0.04	0.52 ±0.04	0.66 ±0.03		✓	0.91 ±0.02	0.91 ±0.02	0.89 ±0.02			0.99 ±0.01	0.99 ±0.01	0.96 ±0.01		✗
	0.04 ±0.01	0.04 ±0.01	0.04 ±0.01			0.12 ±0.02	0.12 ±0.02	0.14 ±0.02			0.23 ±0.03	0.23 ±0.03	0.24 ±0.03		
	0.10 ±0.02	0.10 ±0.02	0.12 ±0.02			0.31 ±0.03	0.31 ±0.03	0.40 ±0.03		✓	0.47 ±0.04	0.47 ±0.04	0.58 ±0.03		✓
	0.33 ±0.03	0.33 ±0.03	0.46 ±0.04		✓	0.77 ±0.03	0.77 ±0.03	0.91 ±0.02		✓	0.95 ±0.01	0.96 ±0.01	0.99 ±0.01		✓
	0.93 ±0.02	0.93 ±0.02	0.96 ±0.01		✓	1.00 ±0.00	1.00 ±0.00	1.00 ±0.00			1.00 ±0.00	1.00 ±0.00	1.00 ±0.00		
	0.07 ±0.02	0.07 ±0.02	0.09 ±0.02			0.24 ±0.03	0.26 ±0.03	0.32 ±0.03		✓	0.44 ±0.04	0.47 ±0.04	0.48 ±0.04		
	0.06 ±0.02	0.07 ±0.02	0.09 ±0.02			0.26 ±0.03	0.28 ±0.03	0.32 ±0.03			0.45 ±0.04	0.47 ±0.04	0.48 ±0.04		
	0.10 ±0.02	0.12 ±0.02	0.14 ±0.02			0.34 ±0.03	0.34 ±0.03	0.39 ±0.03			0.51 ±0.04	0.52 ±0.04	0.53 ±0.04		
	0.07 ±0.02	0.07 ±0.02	0.10 ±0.02		✓	0.30 ±0.03	0.33 ±0.03	0.35 ±0.03			0.53 ±0.04	0.54 ±0.04	0.57 ±0.04		
	0.04 ±0.01	0.05 ±0.02	0.04 ±0.01			0.18 ±0.03	0.27 ±0.03	0.24 ±0.03	✓	✓	0.34 ±0.03	0.45 ±0.04	0.44 ±0.04	✓	✓
	0.16 ±0.03	0.20 ±0.03	0.20 ±0.03			0.45 ±0.04	0.58 ±0.03	0.58 ±0.03	✓	✓	0.67 ±0.03	0.73 ±0.03	0.73 ±0.03	✓	✓
	0.34 ±0.03	0.43 ±0.04	0.43 ±0.04	✓	✓	0.71 ±0.03	0.80 ±0.03	0.79 ±0.03	✓	✓	0.85 ±0.03	0.90 ±0.02	0.89 ±0.02	✓	
	0.63 ±0.03	0.72 ±0.03	0.73 ±0.03	✓	✓	0.91 ±0.02	0.92 ±0.02	0.92 ±0.02			0.95 ±0.01	0.96 ±0.01	0.96 ±0.01		

Table 13.1: The first column shows scatterplots of X vs Y (all having dependence between X, Y). There are 3 sets of 5 columns each - for $\alpha = 0.01, 0.05, 0.1$ (controlled by running 2000 permutations). In eachs set, the first three columns show the power of HSIC, HSIC^S, HSIC^F (with standard deviation over 200 repetitions below). The fourth column shows when HSIC^S is significantly better than HSIC, and the fifth column when HSIC^F has significantly higher power than HSIC. A blank means the powers are not significantly better or worse. In the first dataset (A) (top 4) we show how the power varies with increasing n (becomes easier). In the second dataset (B) (second 4) we show how the power varies with rotation (goes from near-independence to clear dependence). In the third dataset (C) (third 4), we demonstrate a case where shrinkage does *not* help much, which is a circle with a hole. In the last dataset (D) (last 4), we demonstrate a case where HSIC^S does as well as HSIC^F. We tried many more datasets, these are a few representative samples.

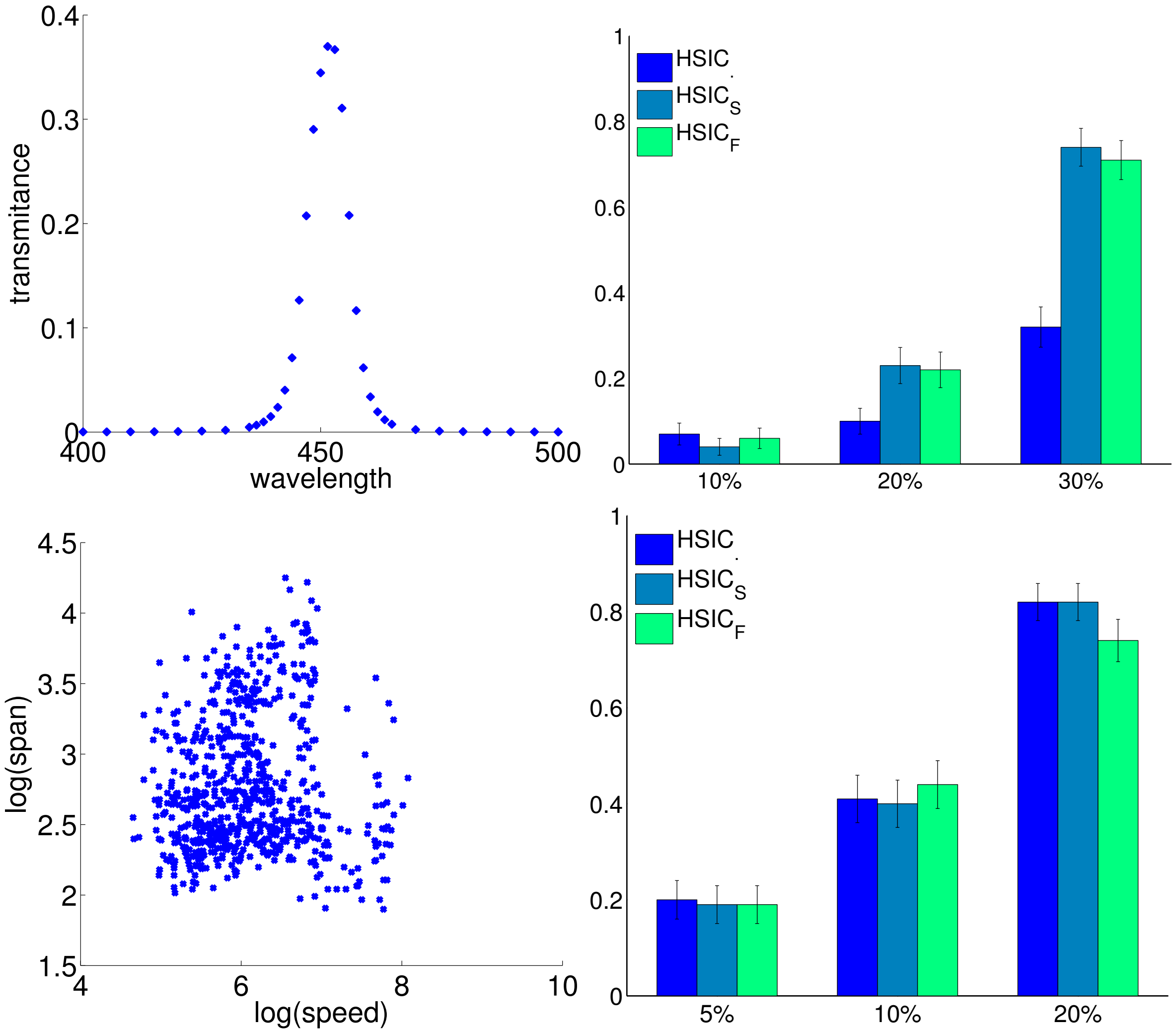


Figure 13.2: Top Row: The left figure shows a plot of wavelength against transmittance. The right figure shows the power of HSIC, $\text{HSIC}^S, \text{HSIC}^F$ when the data are subsampled to $10\%, 20\%, 30\%$ (error bars over 100 repetitions). Bottom Row: The left figure shows a plot of $log(wingspan)$ vs $log(airspeed)$. The right figure shows the power of HSIC, $\text{HSIC}^S, \text{HSIC}^F$ when the data are subsampled to $5\%, 10\%, 20\%$ (error bars over 100 repetitions).

13.5 Discussion

Why might shrinkage improve power? Let us examine the net effect of using shrunk estimators on the value of HSIC, i.e. let us compare HSIC^S and HSIC^F to HSIC by computing these over all the repetitions of the permutation testing procedure described in the introduction. In Fig. 13.3, both estimators are visually similar in transforming the actual test statistic. Perhaps the more interesting phenomenon is that Fig. 13.3 is reminiscent of the graph of a soft-thresholding operator $ST_t(x) = \max\{0, x - t\}$. Intuitively, if the unshrunk HSIC value is small, the shrinkage methods deem it to be "noise" and it is shrunk to zero. Looking at the X-axis scaling of the top and bottom row, the size of the region that gets shrunk to zero decreases with n - as expected, shrinkage has less effect when S_{XY} has low variance). The shrinkage being non-monotone (more so for $n = 20$ than $n = 50$ in Figure 13.3) is key to achieving an improvement in power.

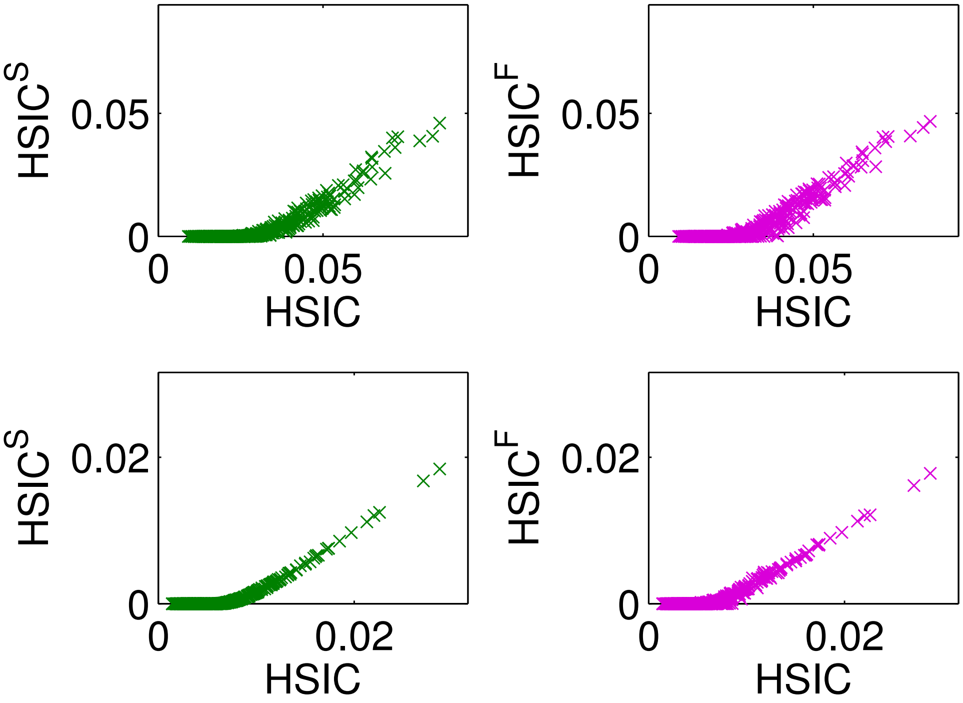


Figure 13.3: The top row corresponds to $n = 20$, and the bottom row has $n = 50$. The left plots compare HSIC^S to HSIC, and the right plots compare HSIC^F to HSIC. Each cross mark corresponds to the shrunk and unshrunk HSIC calculated during a single permutation of a permutation test.

Using the intuition from the above figure, we can finally piece together why shrinkage may yield benefits. A rejection of $\mathcal{H}_0$ occurs when the test statistic stands out in the right tail of its null distribution. Typically, when the alternative is true (this is when rejecting the null improves power) the unshrunk test statistics calculated from the permuted samples is smaller than the unshrunk HSIC calculated on the original sample. However, the effect of shrinking the small statistics towards zero, and setting the smallest ones to zero, is that the unpermuted test statistic under the alternative distribution stands out more in the right tail of the null.

In other words, relative to the unshrunk null distribution and the unshrunk test statistic, the tail of the null distribution is shrunk more towards zero than the unpermuted test statistic, causing the latter to have a higher quantile in the right tail of the former (relative to the quantile before shrinkage). Let us verify this experimentally. In Fig.13.4 we plot for each of the datasets in Table 13.1, the average ratio of unpermuted statistic T to the 95th percentile of the permuted statistics, for $T \in \{\text{HSIC}, \text{HSIC}^S, \text{HSIC}^F\}$. Recall that for dataset (C), we didn't see much of an improvement in power, but for (A),(B),(D) it is clear from Fig. 13.4 that the unpermuted statistic is shrunk less than its null distribution's 95th quantile.

Remark. In our experiments, real and synthetic, shrinkage usually improves (and almost never worsens) power in false-positive regimes that we usually care about. Will shrinkage *always* improve power? Possibly not. Even though shrunk the shrunk S_{XY} dominates S_{XY} for estimation error, it may not be the case that shrunk HSIC always dominates unshrunk HSIC for test power (i.e. the latter may not be

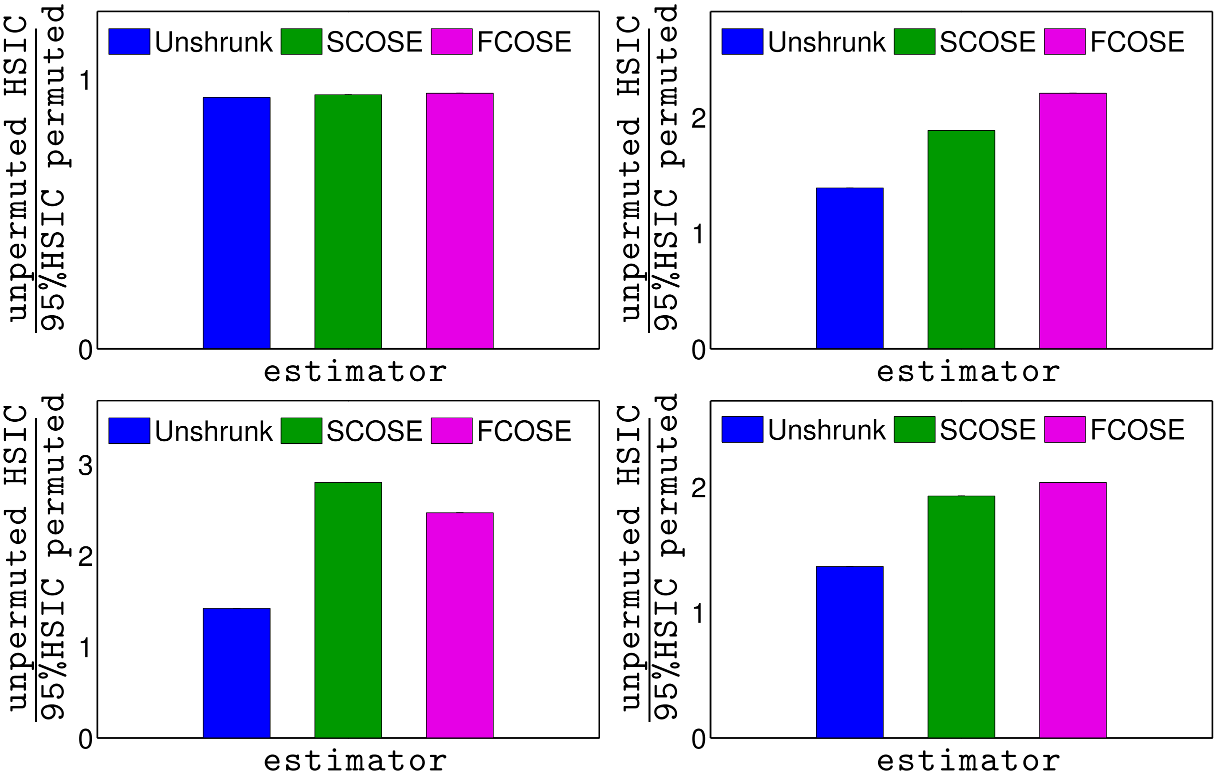


Figure 13.4: All panels show the ratio of the unpermuted HSIC to the 95th percentile of the null distribution based on HSICs calculated from the permuted data. (see Table 13.1) The top row has datasets (C) with radius 2.2, (B) with angle $3 \times \pi/32$, and the bottom row has (D) with $N = 25$, (A) with $N = 40$. These observations were qualitatively the same in all other synthetic data parameter settings, and also for other percentiles than 95th, and since the figures look identical in spirit, they were omitted due to lack of space.

inadmissible). However, just as no single classifier always outperforms another, it is still beneficial to add techniques like shrinkage, that seem to consistently yield benefits in practice, to the practitioner's array of tools.

Conclusion

We presented evidence for an important phenomenon - using biased but lower variance shrunk estimators of cross-covariance operators can often significantly improve test power of HSIC at small sample sizes. This observation (that shrinkage can improve power) has rarely been made in the statistics and machine learning testing literature. We think the reason is that most test statistics for independence testing cannot be immediately expressed as the norm of an empirical operator, making it less obvious *how* to apply shrinkage to improve their power at low sample sizes.

We also showed the optimality (among linear shrinkage estimators) of SCOSE, but observe that the nonlinear shrinkage of FCOSE usually yields higher power. To the best of our knowledge, there seems to be no current literature showing that the choice made by leave-one-out cross-validation (SCOSE) explicitly leads to an estimator that is ”optimal” in some sense (among linear shrinkage estimators). This may be because it is often not possible to explicitly calculate the form of the LOOCV estimator, nor the explicit form of the best linear shrinkage estimator, as can both be done in this simple setting.

Since even the best possible linear shrinkage estimator (as represented by SCOSE) is usually worse than FCOSE, this result indicates that in order to improve upon FCOSE, it will be necessary to further study the class of non-linear shrinkage estimators for our infinite dimensional operators, as done for finite dimensional covariance matrices in [122] and other papers by the same authors.

We ended with a brief investigation into the effect of shrinkage on HSIC and why shrinkage may intuitively improve power. We think that our work will be important for more powerful nonparametric detection of subtle nonlinear dependencies at low sample sizes, a common problem in scientific applications.

Bibliography

[1] A. Agarwal, P.L. Bartlett, P. Ravikumar, and M.J. Wainwright. Information-theoretic lower bounds on the oracle complexity of stochastic convex optimization. *Information Theory, IEEE Transactions on*, 58(5):3235–3249, 2012. 2.1, 2.1

[2] Pedro C Álvarez-Esteban, Eustasio Del Barrio, Juan A Cuesta-Albertos, Carlos Matrán, et al. Similarity of samples and trimming. *Bernoulli*, 18(2):606–634, 2012. 12.1

[3] Pedro César Álvarez-Esteban, Eustasio Del Barrio, Juan Antonio Cuesta-Albertos, and Carlos Matran. Trimmed comparison of distributions. *Journal of the American Statistical Association*, 103 (482), 2008. 12.1

[4] Niall H Anderson, Peter Hall, and D Michael Titterington. Two-sample test statistics for measuring discrepancies between two multivariate probability density functions using kernel-based density estimates. *Journal of Multivariate Analysis*, 50(1):41–54, 1994. 10.3

[5] Theodore W Anderson. *An introduction to multivariate statistical analysis*. Wiley, 1958. 10.1

[6] Theodore W Anderson and Donald A Darling. Asymptotic theory of certain goodness of fit criteria based on stochastic processes. *The annals of mathematical statistics*, pages 193–212, 1952. 10.2

[7] Deanna Needell Anna Ma and Aaditya Ramdas. Code link, 2015. http://www.cmc.edu/pages/faculty/DNeedell/regs.zip. 7.5

[8] Francis J Anscombe. Large-sample theory of sequential estimation. In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 48, pages 600–607. Cambridge Univ Press, 1952. 11.3

[9] Taylor Arnold and Ryan J. Tibshirani. Efficient implementations of the generalized lasso dual path algorithm. arXiv: 1405.3222, 2014. 9.6.1

[10] Jean-Yves Audibert and Alexandre B. Tsybakov. Fast learning rates for plug-in classifiers. *Annals of Statistics*, 35(2):608–633, 2007. 3.1.3

[11] F. Bach and E. Moulines. Non-asymptotic analysis of stochastic approximation algorithms for machine learning. *Advances in Neural Information Processing Systems (NIPS)*, 2011. 3.5

[12] Francis Bach. Duality between subgradient and conditional gradient methods. *arXiv preprint arXiv:1211.6302*, 2012. 5.6

[13] Zhidong D Bai and Hewa Saranadasa. Effect of high dimension: by an example of a two sample problem. *Statistica Sinica*, 6(2):311–329, 1996. 10.1, 10.2, 10.4, 3, 4, 8

[14] Akshay Balsubramani. Sharp uniform martingale concentration bounds. *arXiv preprint arXiv:1405.2639*, 2015. 11.1, 11.2, 11.3, 11.5, 11.5, 11.8, 42, 11.8, 32

[15] Yong Bao and Aman Ullah. Expectation of quadratic forms in normal and nonnormal variables with applications. *Journal of Statistical Planning and Inference*, 140(5):1193–1205, 2010. 10.8,

10.8

[16] L Baringhaus and C Franz. On a new multivariate two-sample test. *Journal of multivariate analysis*, 88(1):190–206, 2004. 10.2, 10.3, 12.1, 12.3.3

[17] Alexandre Belloni and Gustavo Didier. On the behrens-fisher problem: a globally convergent algorithm and a finite-sample study of the wald, lr and lm tests. *The annals of Statistics*, pages 2377–2408, 2008. 10.2

[18] D. Bertsimas and J.N. Tsitsiklis. *Introduction to linear optimization*. Athena Scientific Belmont, MA, 1997. 12.3.1

[19] Peter J Bickel. A distribution free version of the smirnov two sample test in the p-variate case. *The Annals of Mathematical Statistics*, pages 1–23, 1969. 10.2, 12.2.1

[20] Patrick Billingsley. *Convergence of probability measures*. John Wiley & Sons, Inc., New York-London-Sydney, 1968. 12.2.2

[21] HD Block. The perceptron: A model for brain functioning. i. *Reviews of Modern Physics*, 34(1): 123, 1962. 5.5, 6.2.1

[22] Lenore Blum, Felipe Cucker, Michael Shub, and Steve Smale. *Complexity and real computation*. Springer, 1998. 6.7

[23] Jonathan Borwein and Adrian Lewis. *Convex analysis and nonlinear optimization: theory and examples*, volume 3. Springer, 2006. 5.1.1

[24] Stéphane Boucheron, Gábor Lugosi, and Pascal Massart. *Concentration inequalities: A nonasymptotic theory of independence*. Oxford University Press, 2013. 11.5

[25] A. W. Bowman and A. Azzalini. *R package `sm`: nonparametric smoothing methods (version 2.2-5.4)*. University of Glasgow, UK and Università di Padova, Italia, 2014. URL `URLhttp://www.stats.gla.ac.uk/~adrian/sm,http://azzalini.stat.unipd.it/Book_sm`. 13.4

[26] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge Univ. Press, 2004. 9.3

[27] Steve Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternative direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2011. 9.1.1, 9.2, 9.2.4

[28] M.V. Burnashev and K.S. Zigangirov. An interval estimation problem for controlled observations. *Problemy Peredachi Informatsii*, 10(3):51–61, 1974. 3.2.3

[29] C. L. Byrne. *Applied iterative methods*. A K Peters Ltd., Wellesley, MA, 2008. ISBN 978-1-56881-342-4; 1-56881-271-X. 7.2.1, 8.2.1

[30] Tony Cai, Weidong Liu, and Yin Xia. Two-sample test of high dimensional means under dependence. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 76(2):349–372, 2014. 10.2

[31] Raymond J Carroll, David Ruppert, Leonard A Stefanski, and Ciprian M Crainiceanu. *Measurement error in nonlinear models: a modern perspective*, volume 105. Chapman and Hall/CRC, 2010. 4.1

[32] R. Castro and R. Nowak. Minimax bounds for active learning. *IEEE Transactions on Information Theory*, 54(5):2339–2353, 2008. 4.3, 4.3, 4.5

[33] R. Castro and R. Nowak. Active sensing and learning. *Foundations and Applications of Sensor*

Management, pages 177–200, 2009. 3.2.3

[34] R.M. Castro and R.D. Nowak. Minimax bounds for active learning. In *Proceedings of the 20th annual conference on learning theory*, pages 5–19. Springer-Verlag, 2007. 2.1, 2.1, 2.2, 2.2.1, 2.3, 8, 2.6, 2.6, 3.1.3, 3.1.3, 3.2.1, 3.2.3, 4, 4.1.1, 4.2, 4.2, 4.3, 4.5

[35] Y. Censor, P. P. B. Eggermont, and D. Gordon. Strong underrelaxation in Kaczmarz's method for inconsistent systems. *Numerische Mathematik*, 41(1):83–92, 1983. 7.2.2

[36] Song Xi Chen and Ying-Li Qin. A two-sample test for high-dimensional data with applications to gene-set testing. *The Annals of Statistics*, 38(2):808–835, apr 2010. doi: 10.1214/09-aos716. URL `http://dx.doi.org/10.1214/09-aos716`. (document), 10.2, 10.4, 3, 4, 5, 10.4, 10.4, 2, 5, 10.8.2

[37] X. Chen and A. Powell. Almost sure convergence of the Kaczmarz algorithm with random measurements. *J. Fourier Anal. Appl.*, pages 1–20, 2012. ISSN 1069-5869. URL `http://dx.doi.org/10.1007/s00041-012-9237-2`. 10.1007/s00041-012-9237-2. 7.2.1, 8.2.1

[38] Dennis Cheung and Felipe Cucker. A new condition number for linear programming. *Mathematical programming*, 91(1):163–174, 2001. 5.2.1, 5.2.2, 5.2.2, 6.1

[39] Vasek Chvatal. *Linear programming*. Macmillan, 1983. 5.1.1, 6.5

[40] Kacper Chwialkowski and Arthur Gretton. A kernel independence test for random processes. In *Proceedings of The 31st International Conference on Machine Learning*, pages 1422–1430, 2014. 13.4

[41] Kacper Chwialkowski, Aaditya Ramdas, Dino Sejdinovic, and Arthur Gretton. Analytic functions for fast two sample testing. *(in submission)*, 2015. 1.5

[42] Kenneth L Clarkson. Coresets, sparse greedy approximation, and the Frank-Wolfe algorithm. *ACM Transactions on Algorithms (TALG)*, 6(4):63, 2010. 5.2.1, 6.1

[43] Harald Cramér. On the composition of elementary errors: First paper: Mathematical deductions. *Scandinavian Actuarial Journal*, 1928(1):13–74, 1928. 10.2

[44] M. Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in Neural Information Processing Systems 26*, pages 2292–2300, 2013. 12.3.2

[45] George Dantzig. An ϵ-precise feasible solution to a linear program with a convexity constraint in $1/\epsilon^2$ iterations independent of problem size. Technical report, Stanford University, 1992. 5.6, 6.1, 6.5

[46] DA Darling and Herbert Robbins. Iterated logarithm inequalities. *Proceedings of the National Academy of Sciences of the United States of America*, pages 1188–1192, 1967. 11.3

[47] DA Darling and Herbert Robbins. Some nonparametric sequential tests with power one. *Proceedings of the National Academy of Sciences of the United States of America*, 61(3):804, 1968. 11.3

[48] DA Darling and Herbert Robbins. Some further remarks on inequalities for sample sums. *Proceedings of the National Academy of Sciences of the United States of America*, 60(4):1175, 1968. 11.3

[49] P. Laurie Davies and Arne Kovac. Local extremes, runs, strings and multiresolution. *Annals of Statistics*, 29(1):1–65, 2001. 9.1.1, 9.2

[50] Carl de Boor. *A Practical Guide to Splines*. Springer, New York, 1978. 9.1

[51] Eustasio del Barrio. Empirical and quantile processes in the asymptotic theory of goodness-of-fit tests. *Lecture Notes presented at the European Mathematical Society Summer School on Theory and Statistical Applications of Empirical Processes. Laredo, Spain*, 2004. 12.1, 12.2.2

[52] Eustasio del Barrio, Juan A Cuesta-Albertos, Carlos Matrán, et al. Tests of goodness of fit based on the l_2-wasserstein distance. *The Annals of Statistics*, 27(4):1230–1239, 1999. 12.1

[53] Eustasio del Barrio, Juan A Cuesta-Albertos, Carlos Matrán, Sándor Csörgö, Carles M Cuadras, Tertius de Wet, Evarist Giné, Richard Lockhart, Axel Munk, and Winfried Stute. Contributions of empirical and quantile processes to the asymptotic theory of goodness-of-fit tests. *Test*, 9(1):1–96, 2000. 12.1

[54] Eustasio del Barrio, Evarist Giné, Frederic Utzet, et al. Asymptotics for l2 functionals of the empirical quantile process, with applications to tests of fit based on weighted wasserstein distances. *Bernoulli*, 11(1):131–189, 2005. 12.1

[55] Arthur P Dempster. A high dimensional two sample significance test. *The Annals of Mathematical Statistics*, pages 995–1010, 1958. 10.2

[56] L. Devroye, L. Györfi, and G. Lugosi. *A probabilistic theory of pattern recognition*, volume 31. springer, 1996. 3.3.1, 7

[57] Bogdan Dumitrescu. On the relation between the randomized extended kaczmarz algorithm and coordinate descent. *BIT Numerical Mathematics*, pages 1–11, 2014. 7.3.2

[58] John Dunagan and Santosh Vempala. A simple polynomial-time rescaling algorithm for solving linear programs. *Mathematical Programming*, 114(1):101–114, 2008. 6.7

[59] Carl Eckart and Gale Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218, 1936. 5.2.2

[60] K Eckerle. Circular interference transmittance study. *National Institute of Standards and Technology (NIST), US Department of Commerce, USA*, 1979. 13.4

[61] Noureddine El Karoui. The spectrum of kernel random matrices. *The Annals of Statistics*, 38(1): 1–50, 2010. 10.4

[62] Y. C. Eldar and D. Needell. Acceleration of randomized Kaczmarz method via the Johnson-Lindenstrauss lemma. *Numer. Algorithms*, 58(2):163–177, 2011. ISSN 1017-1398. doi: 10.1007/s11075-011-9451-z. URL `http://dx.doi.org/10.1007/s11075-011-9451-z`. 7.2.1, 8.2.1

[63] T. Elfving. Block-iterative methods for consistent and inconsistent linear equations. *Numer. Math.*, 35(1):1–12, 1980. ISSN 0029-599X. doi: 10.1007/BF01396365. URL `http://dx.doi.org/10.1007/BF01396365`. 7.2.1, 8.2.1

[64] Marina Epelman and Robert M Freund. Condition number complexity of an elementary algorithm for computing a reliable solution of a conic linear system. *Mathematical Programming*, 88(3): 451–485, 2000. 5.6, 5.6, 6.1, 6.5

[65] Marina A Epelman, Robert M Freund, et al. *Condition number complexity of an elementary algorithm for resolving a conic linear system.* Citeseer, 1997. 5.6

[66] Moulines Eric, Francis R. Bach, and Zaïd Harchaoui. Testing for homogeneity with kernel fisher discriminant analysis. In J.C. Platt, D. Koller, Y. Singer, and S.T. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 609–616. Curran Associates, Inc., 2008. 13.1

[67] Jianqing Fan. On the optimal rates of convergence for nonparametric deconvolution problems. *The*

Annals of Statistics, pages 1257–1272, 1991. 4.1

[68] Jianqing Fan, Young K Truong, et al. Nonparametric regression with errors in variables. *The Annals of Statistics*, 21(4):1900–1925, 1993. 4.1

[69] Vilim Feller. *An Introduction to Probability Theory and Its Applications: Volume One*. John Wiley & Sons, 1950. 11.1

[70] V Alba Fernández, MD Jiménez Gamero, and J Muñoz García. A test for the two-sample problem based on empirical characteristic functions. *Computational statistics & data analysis*, 52(7):3730–3748, 2008. 10.2, 12.3.4

[71] Tony Finch. Incremental calculation of weighted mean and variance. *University of Cambridge*, 4, 2009. 11.5

[72] Joel Franklin and Jens Lorenz. On the scaling of multidimensional matrices. *Linear Algebra and its applications*, 114:717–735, 1989. 12.3.2

[73] Robert M Freund and Jorge R Vera. Some characterizations and properties of the "distance to ill-posedness" and the condition measure of a conic linear system. *Mathematical Programming*, 86 (2):225–260, 1999. 5.2.1, 6.1

[74] Jerome H Friedman and Lawrence C Rafsky. Multivariate generalizations of the wald-wolfowitz and smirnov two-sample tests. *The Annals of Statistics*, pages 697–717, 1979. 10.2, 12.1

[75] Wayne A Fuller. *Measurement error models*, volume 305. Wiley, 2009. 4.1

[76] Elmer G Gilbert. An iterative procedure for computing the minimum of a quadratic form on a convex set. *SIAM Journal on Control*, 4(1):61–80, 1966. 5.6

[77] Andrew Gilpin, Javier Peña, and Tuomas Sandholm. First-order algorithm with $\mathcal{O}(\ln(1/\epsilon))$ convergence for ϵ-equilibrium in two-person zero-sum games. *Mathematical programming*, 133(1-2): 279–298, 2012. 5.1.1, 5.4

[78] JL Goffin. The relaxation method for solving systems of linear inequalities. *Mathematics of Operations Research*, pages 388–414, 1980. 5.1.1

[79] R. Gordon, R. Bender, and G. T. Herman. Algebraic reconstruction techniques (ART) for three-dimensional electron microscopy and X-ray photography. *J. Theoret. Biol.*, 29:471–481, 1970. 7.2.1, 8.2.1

[80] Thore Graepel, Ralf Herbrich, and Robert C Williamson. From margin to sparsity. *Advances in neural information processing systems*, pages 210–216, 2001. 6.2.2

[81] A. Gretton, O. Bousquet, A. Smola, and B. Schölkopf. Measuring statistical dependence with Hilbert-Schmidt norms. In *Proceedings of Algorithmic Learning Theory*, pages 63–77. Springer, 2005. 13.1, 13.1.1, 13.1.1, 13.1.2, 13.3

[82] A. Gretton, K. Borgwardt, M. Rasch, B. Schoelkopf, and A. Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13:723–773, 2012. 10.2, 10.2, 10.3, 10.3, 10.6, 11.6, 12.1, 12.3.4, 13.1

[83] A. Gretton, B. Sriperumbudur, D. Sejdinovic, H. Strathmann, S. Balakrishnan, M. Pontil, and K. Fukumizu. Optimal kernel choice for large-scale two-sample tests. *Neural Information Processing Systems*, 2012. 10.3

[84] Arthur Gretton, Ralf Herbrich, Alexander Smola, Olivier Bousquet, and Bernhard Schölkopf. Kernel methods for measuring independence. *The Journal of Machine Learning Research*, 6:2075–

2129, 2005. 13.1

[85] Arthur Gretton, Karsten M Borgwardt, Malte Rasch, Bernhard Schölkopf, and Alex J Smola. A kernel method for the two-sample-problem. In *Neural Information Processing Systems*, pages 513–520, 2006. 10.2, 10.3, 12.1, 12.3.4

[86] Arthur Gretton, Kenji Fukumizu, Choon Hui Teo, Le Song, Bernhard Schölkopf, and Alex J Smola. A kernel statistical test of independence. *Neural Information Processing Systems*, 2007. 13.4

[87] Osman Güler, Alan J Hoffman, and Uriel G Rothblum. Approximations to solutions to systems of linear inequalities. *SIAM Journal on Matrix Analysis and Applications*, 16(2):688–696, 1995. 5.1.1, 5.4, 5.4.2

[88] Allan Gut. Anscombe's theorem 60 years later. *Sequential Analysis*, 31(3):368–396, 2012. 11.3

[89] Peter Hall and Christopher C Heyde. *Martingale limit theory and its application*. Academic press, 2014. 8

[90] C. Hamaker and D. C. Solmon. The angles between the null spaces of X-rays. *J. Math. Anal. Appl.*, 62(1):1–23, 1978. ISSN 0022-247x. 7.2.1, 8.2.1

[91] M. Hanke and W. Niethammer. On the acceleration of Kaczmarz's method for inconsistent linear systems. *Linear Algebra and its Applications*, 130:83–98, 1990. 7.2.2

[92] S. Hanneke. Rates of convergence in active learning. *The Annals of Statistics*, 39(1):333–361, 2011. 3.1, 3.1.3, 3.3

[93] Per Christian Hansen. Regularization tools: A matlab package for analysis and solution of discrete ill-posed problems. *Numerical algorithms*, 6(1):1–35, 1994. 7.5

[94] Trevor Hastie and Robert Tibshirani. *Generalized additive models*. Chapman and Hall, London, 1990. 9.1.1

[95] E. Hazan and S. Kale. Beyond the regret minimization barrier: an optimal algorithm for stochastic strongly-convex optimization. In *Proceedings of the 23nd Annual Conference on Learning Theory*, 2011. 2.1, 2.1, 2.1, 2.4, 2.4.1, 2.4.1, 2.4.1, 2.5, 7, 7

[96] Norbert Henze. A multivariate two-sample test based on the number of nearest neighbor type coincidences. *The Annals of Statistics*, pages 772–783, 1988. 10.2, 12.1

[97] G. T. Herman. *Fundamentals of computerized tomography: image reconstruction from projections*. Springer, 2009. 7.2.1, 8.2.1

[98] G.T. Herman and L.B. Meyer. Algebraic reconstruction techniques can be made computationally efficient. *IEEE Trans. Medical Imaging*, 12(3):600–609, 1993. 7.2.1, 8.2.1

[99] Alan J Hoffman. On approximate solutions of systems of linear inequalities. *Journal of Research of the National Bureau of Standards*, 49(4):263–265, 1952. 5.1.1

[100] Mingyi Hong and Zhi-Quan Luo. On the linear convergence of the alternating direction method of multipliers. *arXiv preprint arXiv:1208.3922*, 2012. 5.1.1, 5.4, 5.6

[101] Roger A Horn and Charles R Johnson. *Topics in matrix analysis*. Cambridge Univ. Press Cambridge etc, 1991. 10.8

[102] Harold Hotelling. The generalization of student's ratio. *Annals of Mathematical Statistics*, 2(3): 360–378, aug 1931. doi: 10.1214/aoms/1177732979. URL `http://dx.doi.org/10.1214/aoms/1177732979`. 10.1

[103] Fushing Hsieh and Bruce W. Turnbull. Nonparametric and semiparametric estimation of the

receiver operating characteristic curve. *Ann. Statist.*, 24(1):25–40, 1996. ISSN 0090-5364. doi: 10.1214/aos/1033066197. URL `http://dx.doi.org/10.1214/aos/1033066197`. 12.5.1, 12.5.1

[104] Fushing Hsieh, Bruce W Turnbull, et al. Nonparametric and semiparametric estimation of the receiver operating characteristic curve. *The annals of statistics*, 24(1):25–40, 1996. 12.1

[105] I. A. Ibargimov and R. Z. Hasminskii. *Statistical Estimation. Asymptotic Theory*. Springer-Verlag, 1981. 4.3

[106] Yuri Ingster and Irina A Suslina. *Nonparametric goodness-of-fit testing under Gaussian models*, volume 169. Springer Science & Business Media, 2003. 10.5, 11, 10.5, 10.5

[107] A. Iouditski and Y. Nesterov. Primal-dual subgradient methods for minimizing uniformly convex functions. *Universite Joseph Fourier, Grenoble, France*, 2010. 2.1, 2.2, 2.4, 2.5, 3.1.4, 3.3, 3.5

[108] Andrey Aleksandrovich Ivanov and Aleksandr Ivanovich Zhdanov. Kaczmarz algorithm for tikhonov regularization problem. *Applied Mathematics E-Notes*, 13:270–276, 2013. 8.3.1

[109] K.G. Jamieson, R.D. Nowak, and B. Recht. Query complexity of derivative-free optimization. *arXiv preprint arXiv:1209.2434*, 2012. 2.1, 2.3.2, 5, 3.1.2

[110] Nicholas Johnson. A dynamic programming algorithm for the fused lasso and L_0-segmentation. *Journal of Computational and Graphical Statistics*, 22(2):246–260, 2013. 9.1.1, 9.2, 9.5

[111] S. Kaczmarz. Angenäherte auflösung von systemen linearer gleichungen. *Bull. Int. Acad. Polon. Sci. Lett. Ser. A*, pages 335–357, 1937. 7.2.1, 8.2.1

[112] Takeaki Kariya. A robustness property of hotelling's t2-test. *The Annals of Statistics*, pages 211–214, 1981. 10.1

[113] Maurice Kendall and Alan Stuart. The advanced theory of statistics. vol. 1: Distribution theory. *London: Griffin, 1977, 4th ed.*, 1, 1977. 10.8

[114] A. Ya. Khinchin. über einen satz der wahrscheinlichkeitsrechnung. *Fundamenta Mathematicae*, 6: 9–20, 1924. 38

[115] Seung-Jean Kim, Kwangmoo Koh, Stephen Boyd, and Dimitry Gorinevsky. ℓ_1 trend filtering. *SIAM Review*, 51(2):339–360, 2009. 9.1, 9.1.1, 9.1.2, 9.3, 9.5, 9.5, 9.5, 9.6.1

[116] Andrej N Kolmogorov. *Sulla determinazione empirica di una legge di distribuzione*. na, 1933. 10.2, 12.2.1

[117] J. Komlós, P. Major, and G. Tusnády. An approximation of partial sums of independent RV's, and the sample DF. II. *Z. Wahrscheinlichkeitstheorie und Verw. Gebiete*, 34(1):33–58, 1976. 12.5.1

[118] A. P. Korostelev and A. B. Tsybakov. *Minimax Theory of Image Reconstruction*, volume 82 of *Lecture Notes in Statistics*. Springer, NY, 1993. 1

[119] Tze Leung Lai. Power-one tests based on sample sums. *The Annals of Statistics*, pages 866–880, 1977. 11.3

[120] Tze Leung Lai, Zheng Su, et al. *Sequential nonparametrics and semiparametrics: Theory, implementation and applications to clinical trials*. Institute of Mathematical Statistics, 2008. 11.3

[121] Olivier Ledoit and Michael Wolf. A well-conditioned estimator for large-dimensional covariance matrices. *Journal of Multivariate Analysis*, 88(2):365–411, 2004. 13.3

[122] Olivier Ledoit, Michael Wolf, et al. Nonlinear shrinkage estimation of large-dimensional covariance matrices. *The Annals of Statistics*, 40(2):1024–1060, 2012. 13.5

[123] Yin Tat Lee and Aaron Sidford. Efficient accelerated coordinate descent methods and faster algorithms for solving linear systems. In *Foundations of Computer Science (FOCS), 2013 IEEE 54th Annual Symposium on*, pages 147–156. IEEE, 2013. 8.4

[124] Erich Leo Lehmann and Howard JM D'Abrera. *Nonparametrics: statistical methods based on ranks*. Springer New York, 2006. 10.2, 12.1

[125] Hans Rudolf Lerche. Sequential analysis and the law of the iterated logarithm. *Lecture Notes-Monograph Series*, pages 40–53, 1986. 11.3

[126] D. Leventhal and A. S. Lewis. Randomized methods for linear constraints: convergence rates and conditioning. *Math. Oper. Res.*, 35(3):641–654, 2010. ISSN 0364-765X. doi: 10.1287/moor.1100.0456. URL `http://dx.doi.org/10.1287/moor.1100.0456`. 7.1, 7.1, 7.2.1, 7.2.3, 1, 7.3.1, 26, 8.1, 8.2.1, 8.2.2, 8.4

[127] Dan Li and Tamás Terlaky. The duality between the perceptron algorithm and the von neumann algorithm. In *Modeling and Optimization: Theory and Applications*, pages 113–136. Springer, 2013. 5.3

[128] Nicholas Littlestone. Redundant noisy attributes, attribute errors, and linear-threshold learning using winnow. In *Proceedings of the fourth annual workshop on Computational learning theory*, pages 147–156. Morgan Kaufmann Publishers Inc., 1991. 6.1, 6.2.2

[129] Ji Liu, Stephen J Wright, and Srikrishna Sridhar. An asynchronous parallel randomized kaczmarz algorithm. *arXiv preprint arXiv:1401.4780*, 2014. 7.3

[130] Po-Ling Loh and Sebastian Nowozin. Faster hoeffding racing: Bernstein races via jackknife estimates. In *Algorithmic Learning Theory*, pages 203–217. Springer, 2013. 11.3

[131] M.E. Lopes, L. Jacob, and M.J. Wainwright. A more powerful two-sample test in high dimensions using random projection. In *Advances in Neural Information Processing Systems 24*. MIT Press, 2011. 10.2, 10.3, 10.7.2, 10.7.2

[132] Sébastien Loustau and Clément Marteau. Discriminant analysis with errors in variables. *arXiv preprint arXiv:1201.3283*, 2012. 4.1

[133] R. Lyons. Distance covariance in metric spaces. *Annals of Probability*, 41(5):3284–3305, 2013. 10.2, 12.3.3, 12.3.4

[134] Anna Ma*, Deanna Needell*, and Aaditya Ramdas*. Convergence properties of the randomized extended gauss-seidel and kaczmarz methods. *(in submission) SIAM Journal on Matrix Analysis and Applications*, 2015. 5, 8.1, 8.2.2, 8.3, 8.4.1

[135] Jan R Magnus. The expectation of products of quadratic forms in normal variables: the practice. *Statistica Neerlandica*, 33(3):131–136, 1979. 10.8

[136] Enno Mammen and Sara van de Geer. Locally apadtive regression splines. *Annals of Statistics*, 25 (1):387–413, 1997. 9.1

[137] Andreas Maurer and Massimiliano Pontil. Empirical bernstein bounds and sample variance penalization. *arXiv preprint arXiv:0907.3740*, 2009. 1

[138] Volodymyr Mnih, Csaba Szepesvári, and Jean-Yves Audibert. Empirical bernstein stopping. In *Proceedings of the 25th international conference on Machine learning*, pages 672–679. ACM, 2008. 11.3

[139] Krikamol Muandet, Kenji Fukumizu, Bharath Sriperumbudur, Arthur Gretton, and Bernhard Schoelkopf. Kernel mean estimation and stein effect. In *Proceedings of The 31st International*

Conference on Machine Learning, pages 10–18, 2014. 13.1.2, 13.2, 13.2, 13.2

[140] F. Natterer. *The mathematics of computerized tomography*, volume 32 of *Classics in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2001. ISBN 0-89871-493-1. doi: 10.1137/1.9780898719284. URL `http://dx.doi.org/10.1137/1.9780898719284`. Reprint of the 1986 original. 7.2.1, 8.2.1

[141] D. Needell. Randomized Kaczmarz solver for noisy linear systems. *BIT*, 50(2):395–403, 2010. ISSN 0006-3835. doi: 10.1007/s10543-010-0265-5. URL `http://dx.doi.org/10.1007/s10543-010-0265-5`. 7.1, 7.1, 7.2.1, 7.3, 7.3.2, 8.2.1

[142] D. Needell and J. A. Tropp. Paved with good intentions: Analysis of a randomized block kaczmarz method. *Linear Algebra and its Applications*, 2013. 7.2.1, 8.2.1

[143] D. Needell and R. Ward. Two-subspace projection method for coherent overdetermined linear systems. *Journal of Fourier Analysis and Applications*, 19(2):256–269, 2013. 7.2.1, 8.2.1

[144] D. Needell, N. Sbrero, and R. Ward. Stochastic gradient descent and the randomized kaczmarz algorithm. *Math. Program. Series A*, 2014. to appear. 7.2.1, 8.2.1, 8.4

[145] Arkadi Nemirovski. Prox-method with rate of convergence O(1/t) for variational inequalities with Lipschitz continuous monotone operators and smooth convex-concave saddle point problems. *SIAM Journal on Optimization*, 15(1):229–251, 2004. 6.1

[146] A.S. Nemirovski and D.B. Yudin. *Problem complexity and method efficiency in optimization.* John Wiley & Sons, 1983. 2.1, 2.1, 3.1.1

[147] Y. Nesterov. Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM J. Optimiz.*, 22(2):341–362, 2012. 3.1.2, 7.3, 8.2.2

[148] Yu Nesterov. Excessive gap technique in nonsmooth convex minimization. *SIAM Journal on Optimization*, 16(1):235–249, 2005. 6.1, 6.4, 6.4

[149] Yu Nesterov. Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM Journal on Optimization*, 22(2):341–362, 2012. 8.4

[150] Frank Nielsen and Richard Nock. Approximating smallest enclosing balls with applications to machine learning. *International Journal of Computational Geometry & Applications*, 19(05):389–414, 2009. 5.2.1

[151] R Peto, MC Pike, Philip Armitage, Norman E Breslow, DR Cox, SV Howard, N Mantel, K McPherson, J Peto, and PG Smith. Design and analysis of randomized clinical trials requiring prolonged observation of each patient. ii. analysis and examples. *British journal of cancer*, 35(1):1, 1977. 11.3

[152] *On convergence proofs for perceptrons*, volume XII, 1962. Polytechnic Institute of Brooklyn, Microwave Research Institute. 5.5, 6.2.1

[153] C. Popa. Extensions of block-projections methods with relaxation parameters to inconsistent and rank-deficient least-squares problems. *BIT*, 38(1):151–176, 1998. ISSN 0006-3835. doi: 10.1007/BF02510922. URL `http://dx.doi.org/10.1007/BF02510922`. 7.2.2

[154] C. Popa, T. Preclik, H. Köstler, and U. Rüde. On Kaczmarz's projection iteration as a direct solver for linear least squares problems. *Linear Algebra and Its Applications*, 436(2):389–404, 2012. 7.2.1, 8.2.1

[155] M. Raginsky and A. Rakhlin. Information complexity of black-box convex optimization: A new look via feedback information theory. In *47th Annual Allerton Conference on Communication,*

Control, and Computing, 2009., 2009. 2.1, 2.1, 2.5, 3.1

[156] Alexander Rakhlin, Ohad Shamir, and Karthik Sridharan. Making gradient descent optimal for strongly convex stochastic optimization. *arXiv preprint arXiv:1109.5647*, 2011. 5.5

[157] Aaditya Ramdas and Akshay Balsubramani. Sequential nonparametric testing using the martingale law of the iterated logarithm. *(in submission)*, 2015. 9

[158] Aaditya Ramdas and Javier Peña. Margins, kernels and non-linear smoothed perceptrons. In *Proceedings of The 31st International Conference on Machine Learning*, pages 244–252, 2014. 4, 1, 5.5

[159] Aaditya Ramdas and Javier Pena. Towards a deeper geometric, analytic and algorithmic understanding of margins. *(in submission) Optimization Methods and Software*, 2015. 3

[160] Aaditya Ramdas and Aarti Singh. Optimal rates for stochastic convex optimization under tsybakov noise condition. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, pages 365–373, 2013. 1, 3.1, 3.2.1, 3.2.2, 3.2.3, 7, 3.5

[161] Aaditya Ramdas and Aarti Singh. Algorithmic connections between active learning and stochastic convex optimization. In *Algorithmic Learning Theory*, pages 339–353. Springer, 2013. 1, 4.4, 4.5

[162] Aaditya Ramdas and Ryan J. Tibshirani. Fast and flexible admm algorithms for trend filtering. *Journal of Computational and Graphical Statistics (accepted, in press)*, 2015. 7, 1

[163] Aaditya Ramdas* and Leila Wehbe*. Nonparametric independence testing for small sample sizes. *24th International Joint Conference on Artificial Intelligence*, 2015. 11, 1

[164] Aaditya Ramdas, Barnabas Poczos, Aarti Singh, and Larry Wasserman. An analysis of active learning with uniform feature noise. In *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics*, pages 805–813, 2014. 2

[165] Aaditya Ramdas*, Nicolas Garcia*, and Marco Cuturi. On wasserstein two sample testing and related families of nonparametric tests. *(in submission)*, 2015. 10

[166] Aaditya Ramdas, Deanna Needell, and Ahmed Hefny. Rows vs. columns: Randomized kaczmarz or gauss-seidel for ridge regression. *(to be submitted)*, 2015. 6

[167] Aaditya Ramdas, Sashank Reddi, Barnabás Póczos, Aarti Singh, and Larry Wasserman. On the decreasing power of kernel- and distance-based nonparametric hypothesis tests in high dimensions. *29th AAAI Conference on Aritificial Intelligence*, 2015. 8, 10.2

[168] Aaditya Ramdas, Sashank Reddi, Barnabás Póczos, Aarti Singh, and Larry Wasserman. On the high-dimensional power of linear-time two sample testing against mean-shift alternatives. *(AISTATS '15) Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics*, 2015. 8, 10.3, 10.6, 11.6

[169] Aaditya Ramdas, Sashank J. Reddi, Barnabas Poczos, Aarti Singh, and Larry Wasserman. Adaptivity and computation-statistics tradeoffs for kernel and distance based high dimensional two sample testing. *(in submission))*, 2015. 8

[170] James Renegar. Some perturbation theory for linear programming. *Mathematical Programming*, 65(1):73–91, 1994. 5.1.1

[171] James Renegar. Incorporating condition measures into the complexity theory of linear programming. *SIAM Journal on Optimization*, 5(3):506–524, 1995. 5.1.1, 5.2.2, 6.1

[172] P. Richtárik and M. Takáč. Iteration complexity of randomized block-coordinate descent methods

for minimizing a composite function. *Math. Program.*, pages 1–38, 2012. 7.2.1, 7.3, 8.2.1, 8.2.2, 8.4

[173] Herbert Robbins. Statistical methods related to the law of the iterated logarithm. *The Annals of Mathematical Statistics*, pages 1397–1409, 1970. 11.3

[174] Herbert Robbins. *Herbert Robbins Selected Papers*. Springer, 1985. 11.3

[175] Paul R Rosenbaum. An exact distribution-free test comparing two multivariate distributions based on adjacency. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(4): 515–530, 2005. 10.2, 12.1

[176] Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958. 5.5, 6.1

[177] Leonid I. Rudin, Stanley Osher, and Emad Faterni. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60:259–268, 1992. 9.1

[178] W. Rudin. *Fourier analysis on groups*. Interscience Publishers, New York, 1962. 3

[179] Walter Rudin. *Real and complex analysis*. Tata McGraw-Hill Education, 1987. 11.6

[180] Ankan Saha, SVN Vishwanathan, and Xinhua Zhang. New approximation algorithms for minimum enclosing convex shapes. In *Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1146–1160. SIAM, 2011. 6.1

[181] O.V. Salaevskii. Minimax character of hotellings t2 test. i. In *Investigations in Classical Problems of Probability Theory and Mathematical Statistics*, pages 74–101. Springer, 1971. 10.1

[182] Craig Saunders, Alexander Gammerman, and Volodya Vovk. Ridge regression learning algorithm in dual variables. In *(ICML-1998) Proceedings of the 15th International Conference on Machine Learning*, pages 515–521. Morgan Kaufmann, 1998. 8.5

[183] Mark F Schilling. Multivariate two-sample tests based on nearest neighbors. *Journal of the American Statistical Association*, 81(395):799–806, 1986. 10.2, 12.1

[184] Bernhard Schölkopf and A. J. Smola. *Learning with Kernels*. MIT Press, Cambridge, MA, 2002. 6.3, 8.5, 10.3, 10.7.1, 13.1.1, 13.2, 13.4

[185] Bernhard Schölkopf, Ralf Herbrich, and Alex J Smola. A generalized representer theorem. In *Computational learning theory*, pages 416–426. Springer, 2001. 6.3

[186] E. Schrodinger. Uber die umkehrung der naturgesetze. *Sitzungsberichte Preuss. Akad. Wiss. Berlin. Phys. Math.*, 144:144–153, 1931. 12.3.2

[187] D. Sejdinovic, B. Sriperumbudur, A. Gretton, and K. Fukumizu. Equivalence of distance-based and RKHS-based statistics in hypothesis testing. *The Annals of Statistics*, 41(5):2263–2291, 2013. 10.3, 12.3.4

[188] Robert J Serfling. *Approximation theorems of mathematical statistics*, volume 162. John Wiley & Sons, 2009. 8, 10.8, 15

[189] Yiyuan She and Art B. Owen. Outlier detection using nonconvex penalized regression. *Journal of the American Statistical Association: Theory and Methods*, 106(494):626–639, 2011. 9.5

[190] Galen R. Shorack and Jon A. Wellner. *Empirical processes with applications to statistics*. Wiley Series in Probability and Mathematical Statistics: Probability and Mathematical Statistics. John Wiley & Sons, Inc., New York, 1986. ISBN 0-471-86725-X. 12.2.2

[191] JB Simaika. On an optimum property of two important statistical tests. *Biometrika*, pages 70–80,

1941. 10.1

[192] A. Singh, C. Scott, and R. Nowak. Adaptive hausdorff estimation of density level sets. *Annals of Statistics*, 37(5B):2760–2782, 2009. 2.1

[193] Richard Sinkhorn. Diagonal equivalence to matrices with prescribed row and column sums. *The American Mathematical Monthly*, 74(4):402–405, 1967. 12.3.2

[194] Nickolay Smirnov. Table for estimating the goodness of fit of empirical distributions. *The annals of mathematical statistics*, pages 279–281, 1948. 10.2, 12.2.1

[195] Negar Soheili and Javier Peña. A smooth perceptron algorithm. *SIAM Journal on Optimization*, 22 (2):728–737, 2012. 6.1

[196] Negar Soheili and Javier Peña. A primal–dual smooth perceptron–von Neumann algorithm. In *Discrete Geometry and Optimization*, pages 303–320. Springer, 2013. 5.1.1, 5.5, 6.1

[197] Negar Soheili and Javier Peña. A deterministic rescaled perceptron algorithm. *Mathematical Programming*, pages 1–14, 2013. 6.7

[198] Peter Spirtes, Clark N Glymour, and Richard Scheines. *Causation, prediction, and search*, volume 81. MIT press, 2000. 13.1

[199] K. Sridharan and A. Tewari. Convex games in banach spaces. In *Proceedings of the 23nd Annual Conference on Learning Theory*, 2010. 2.1, 2.3.1

[200] Muni S. Srivastava and Meng Du. A test for the mean vector with fewer observations than the dimension. *Journal of Multivariate Analysis*, 99(3):386–402, mar 2008. doi: 10.1016/j.jmva.2006.11.002. URL `http://dx.doi.org/10.1016/j.jmva.2006.11.002`. 10.2

[201] Muni S Srivastava, Shota Katayama, and Yutaka Kano. A two sample test in high dimensional data. *Journal of Multivariate Analysis*, 114:349–358, 2013. 10.2

[202] Gabriel Steidl, Stephan Didas, and Julia Neumann. Splines in higher order TV regularization. *International Journal of Computer Vision*, 70(3):214–255, 2006. 9.1

[203] Charles Stein. Inadmissibility of the usual estimator for the mean of a multivariate normal distribution. *Proceedings of the Third Berkeley symposium on mathematical statistics and probability*, 1 (399):197–206, 1956. 13.1.2

[204] Ingo Steinwart and Clint Scovel. Fast rates to bayes for kernel machines. In *Advances in neural information processing systems*, pages 1345–1352, 2004. 4.1.1

[205] Quentin Stout. Unimodal regression via prefix isotonic regression. *Computational Statistics and Data Analysis*, 53(2):289–297, 2008. 9.5

[206] T. Strohmer and R. Vershynin. A randomized Kaczmarz algorithm with exponential convergence. *J. Fourier Anal. Appl.*, 15(2):262–278, 2009. ISSN 1069-5869. doi: 10.1007/s00041-008-9030-4. URL `http://dx.doi.org/10.1007/s00041-008-9030-4`. 7.1, 7.1, 7.2.1, 7.3, 1, 7.3.1, 7.4, 8.1, 8.2.1, 8.2.1

[207] Gábor J Székely and Maria L Rizzo. Testing for equal distributions in high dimension. *InterStat*, 5, 2004. 10.2, 10.3, 3, 12.1, 12.3.3

[208] G.J. Székely, M.L. Rizzo, and N.K. Bakirov. Measuring and testing dependence by correlation of distances. *The Annals of Statistics*, 35(6):2769–2794, 2007. 11.6

[209] K. Tanabe. Projection method for solving a singular system of linear equations and its applications. *Numerische Mathematik*, 17(3):203–214, 1971. 7.2.2

[210] Olivier Thas. *Comparing distributions*. Springer, 2010. 12.1, 12.4

[211] Robert Tibshirani, Michael Saunders, Saharon Rosset, Ji Zhu, and Keith Knight. Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society: Series B*, 67(1):91–108, 2005. 9.1

[212] Ryan J. Tibshirani. Adaptive piecewise polynomial estimation via trend filtering. *Annals of Statistics*, 42(1):285–323, 2014. 9.1, 9.4, 9.5, 9.5, 9.5, 9.6.1, 9.6.3

[213] Ryan J. Tibshirani and Jonathan Taylor. The solution path of the generalized lasso. *Annals of Statistics*, 39(3):1335–1371, 2011. 9.1, 9.6.1

[214] Ryan J. Tibshirani, Holger Hoefling, and Robert Tibshirani. Nearly-isotonic regression. *Technometrics*, 53(1):54–61, 2011. 9.5

[215] Paul Tseng. On accelerated proximal gradient methods for convex-concave optimization. *SIAM Journal on Optimization*, 2008. 6.1

[216] A. B. Tsybakov. On nonparametric estimation of density level sets. *Annals of Statistics*, 25(3): 948–969, 1997. 2.1

[217] A.B. Tsybakov. Optimal aggregation of classifiers in statistical learning. *The Annals of Statistics*, 32(1):135–166, 2004. 3.1.3, 4.1.1

[218] A.B. Tsybakov. *Introduction to Nonparametric Estimation*. Springer Series in Statistics. Springer, 2009. ISBN 9780387790510. 2.1, 2.1, 1, 3, 4.3, 7

[219] Aman Ullah. *Finite sample econometrics*. Oxford University Press Oxford, 2004. 10.8

[220] Vladimir N Vapnik. *Statistical learning theory*. Wiley, 1998. 5.1.1

[221] C. Villani. *Optimal transport: old and new*, volume 338. Springer Verlag, 2009. 12.3, 43

[222] Richard Von Mises. Wahrscheinlichkeit statistik und wahrheit. 1928. 10.2

[223] Grace Wahba. *Spline Models for Observational Data*. Society for Industrial and Applied Mathematics, Philadelphia, 1990. 9.1

[224] Abraham Wald. Sequential tests of statistical hypotheses. *The Annals of Mathematical Statistics*, 16(2):117–186, 1945. 11.2

[225] Abraham Wald and Jacob Wolfowitz. On a test whether two samples are from the same population. *The Annals of Mathematical Statistics*, 11(2):147–162, 1940. 10.2, 11.3, 12.1

[226] Abraham Wald and Jacob Wolfowitz. Optimum character of the sequential probability ratio test. *The Annals of Mathematical Statistics*, pages 326–339, 1948. 11.3

[227] Yuxiang Wang, Alex Smola, and Ryan J. Tibshirani. The falling factorial basis and its statistical properties. *International Conference on Machine Learning*, 31, 2014. 9.2.2, 9.4, 9.6.1, 9.6.3

[228] Leila Wehbe, Brian Murphy, Partha Talukdar, Alona Fyshe, Aaditya Ramdas, and Tom Mitchell. Simultaneously uncovering the patterns of brain regions involved in different story reading subprocesses. *PLOS ONE, November Issue*, 2014. 1.5

[229] Leila Wehbe, Aaditya Ramdas, and Tom Mitchell. One-step hypothesis testing for functional neuroimaging. *(in submission)*, 2015. 1.5

[230] Leila Wehbe, Aaditya Ramdas, Rebecca Steorts, and Cosma Shalizi. Regularized brain reading with shrinkage and smoothing. *Annals of Applied Statistics (accepted, in press)*, 2015. 1.5

[231] T. M. Whitney and R. K. Meany. Two algorithms related to the method of steepest descent. *SIAM*

Journal on Numerical Analysis, 4(1):109–118, 1967. 7.2.2

[232] J. Xu and L. Zikatanov. The method of alternating projections and the method of subspace corrections in Hilbert space. *J. Amer. Math. Soc.*, 15(3):573–597, 2002. ISSN 0894-0347. doi: 10.1090/S0894-0347-02-00398-3. URL `http://dx.doi.org/10.1090/S0894-0347-02-00398-3`. 7.2.1, 8.2.1

[233] Wojciech Zaremba, Arthur Gretton, and Matthew Blaschko. B-test: A non-parametric, low variance kernel two-sample test. In *Advances in Neural Information Processing Systems*, pages 755–763, 2013. 10.3, 10.6

[234] Alexander Ivanovich Zhdanov. The method of augmented regularized normal equations. *Computational Mathematics and Mathematical Physics*, 52(2):194–197, 2012. 8.3.1

[235] Anastasios Zouzias and Nikolaos M Freris. Randomized extended kaczmarz for solving least squares. *SIAM Journal on Matrix Analysis and Applications*, 34(2):773–793, 2013. 7.1, 7.1, 7.2.2, 7.2.2, 7.3, 7.3.2, 7.3.2, 1, 7.4, 28, 7.4, 8.2.2