Understanding, Formally Characterizing, and Robustly Handling Real-World Distribution Shift

Elan Rosenfeld

November 13, 2023

Contents

1 Introduction

Distribution shift remains a significant obstacle to successful and reliable deployment of machine learning (ML) systems. Deep neural networks continue to fail in unexpected ways under small input perturbations, when the data distribution changes, or even when simply being tested on outliers in the heavy tail of natural inputs. It is tempting to approach this problem with the same strategy which has led to great success for many other problems in ML: iteration on an appropriately defined benchmark. While this may work for more traditional "iid" settings, true progress in handling distribution shift comes with the understanding that benchmarks fundamentally cannot capture all possible variation which may occur in the real world. Thus, as in the field of cryptography before it, the problem first requires a precise, workable definition: *what does it mean for a distribution to shift?*

There are any number of reasonable definitions for this term. At first, committing to a specific one may seem like setting our sights too low: ideally, we would like to achieve general robustness to all shifts! Though an appealing idea, no predictor can achieve good performance on all test distributions simultaneously; without a precise characterization of a threat model, robustness guarantees are impossible. Furthermore, each threat model comes with the implicit definition of what *cannot* change—like the concept of a computationally bounded adversary, formally defining distribution shift in a particular setting allows us to narrow our focus by placing constraints on the worst case. Proving that an algorithm or model will remain robust even in the most unfavorable or unlikely settings is only possible if we first define them.

As they are increasingly deployed in important and sensitive contexts, the predictions of deep neural networks have a large and growing influence on real-world outcomes. Systematically developing these models in ways that enable formal guarantees of robustness is the only way to ensure genuine trust in their future behavior, maximizing their widespread applicability and utility. This thesis proposal surveys my past and ongoing work towards the development of trustworthy ML models by (i) working to better understand the effect of heavy tails and distribution shifts, both average- and worst-case; (ii) designing formal, practical characterizations of their structure; and (iii) developing provably correct and efficient algorithms to handle them robustly.

2 Robustness to Adversarial Perturbations

This chapter is based on Cohen et al. [2019] and Rosenfeld et al. [2020].

Introduction. One of the most well-known examples of the surprising failure modes of deep neural networks is their vulnerability to *adversarial examples*: small (i.e., human-imperceptible), targeted perturbations to the input which cause the network to output a very confident incorrect prediction [Biggio et al., 2012, Szegedy et al., 2014]. Not only do such perturbations represent a problematic shift in the distribution of images, it is also insufficient to improve *average* robustness because these attacks are created specifically for the network being evaluated. Many works proposed heuristic methods for training classifiers which were purportedly robust to such attacks; most were subsequently shown to fail against sufficiently powerful adversaries [Carlini and Wagner, 2017, Athalye et al., 2018, Uesato et al., 2018]. In response there began a series of works on *certifiable robustness*, designing classifiers whose prediction at any point x is verifiably constant within some set around x (e.g., Wong and Kolter 2018, Raghunathan et al. 2018). Notably, this does not ensure the *correctness* of such predictions, but only that their prediction could not have been changed by a perturbation from any point within this region.

2.1 Certified Adversarial Robustness via Randomized Smoothing

Unfortunately, these prior approaches are prohibitively expensive, which limits their applicability to modern deep networks. Instead, we consider an approach which borrows ideas from *differential privacy* to derive a scalable alternative. Two previous papers [Lecuyer et al., 2019, Li et al., 2019] showed that an operation we call *randomized smoothing* can transform a black-box classifier f into a new "smoothed classifier" g that is certifiably robust to input perturbations in ℓ_2 norm. Let f be a classifier which maps inputs $x \in \mathbb{R}^d$ to classes $\mathcal{Y}$. Then the smoothed classifier's prediction $g(x)$ is defined to be the class most likely to be predicted by f when the input x is corrupted by isotropic Gaussian noise $\mathcal{N}(0, \sigma^2 I)$. If the base classifier f is most likely to classify $\mathcal{N}(x, \sigma^2 I)$ as x's correct class, then the smoothed classifier g will be correct at x. But the smoothed classifier g will also be provably constant within an ℓ_2 ball around any input x, with a radius dependent on the probabilities with which f classifies $\mathcal{N}(x, \sigma^2 I)$ as each class. The higher the margin, the larger the radius around x in which g provably maintains that prediction.

Formally, when queried at x, the smoothed classifier g returns

$$g(x) = \arg\max_{c \in \mathcal{Y}} \ \mathbb{P}(f(x + \varepsilon) = c), \quad \text{where } \ \varepsilon \sim \mathcal{N}(0, \sigma^2 I)$$

The noise level σ is a hyperparameter of the smoothed classifier g which controls a trade-off between robustness and accuracy. Lecuyer et al. [2019] were the first to apply this idea to certified robustness, although

it took a somewhat more complex form which injected noise into the internal layers of the network. Li et al. [2019] analyzed a functional form essentially matching the definition above. However, both of these guarantees were loose, in the sense that the smoothed classifier g is *provably always* more robust than the guarantee indicates. The present work proves the first tight robustness guarantee for randomized smoothing, enabling much larger radii of certified adversarial robustness.

Main bound. Suppose that when f is evaluated on samples from $\mathcal{N}(x, \sigma^2 I)$, the most probable class c_A is returned with probability p_A, and the "runner-up" class is returned with probability p_B. Our main result is that g is provably robust around x within the ℓ_2 radius $R = \frac{\sigma}{2}(\Phi^{-1}(p_A) - \Phi^{-1}(p_B))$, where Φ^{-1} is the inverse of the standard Gaussian CDF. This result also holds if we replace p_A with a lower bound $\underline{p_A}$ and we replace p_B with an upper bound $\overline{p_B}$.

Theorem 2.1. *Let $f : \mathbb{R}^d \to \mathcal{Y}$ be any deterministic or random function, and let $\varepsilon \sim \mathcal{N}(0, \sigma^2 I)$. Suppose $c_A \in \mathcal{Y}$ and $\underline{p_A}, \overline{p_B} \in [0, 1]$ satisfy:*

$$\mathbb{P}(f(x + \varepsilon) = c_A) \geq \underline{p_A} \geq \overline{p_B} \geq \max_{c \neq c_A} \mathbb{P}(f(x + \varepsilon) = c). \tag{1}$$

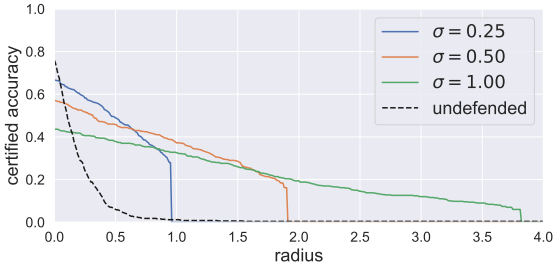
Then $g(x + \delta) = c_A$ for all $\|\delta\|_2 < R$, where $R = \frac{\sigma}{2}(\Phi^{-1}(\underline{p_A}) - \Phi^{-1}(\overline{p_B}))$.

Proof outline. The proof begins with the smoothed distribution $\mathcal{N}(x, \sigma^2 I)$ which is assigned probability p_A by the smoothed f (with runner-up probability p_B) and an arbitrary perturbation δ. We then ask: which base classifier f represents the *worst case* for this perturbation? That is, what hypothetical function f is consistent with the observed probabilities $\mathbb{E}[f(x + \varepsilon)]$ and simultaneously gives the class c_A the smallest probability? A strikingly simple proof—with a similar argument to the Neyman-Pearson Lemma—shows that this worst-case f is precisely a halfspace with boundary orthogonal to δ, and that the resulting g will not change for perturbations along this direction up to a distance of R away from x.

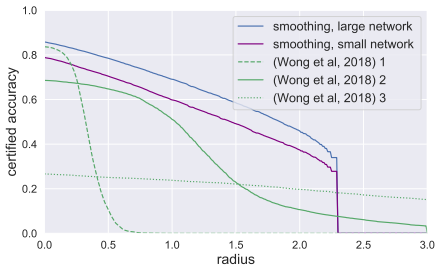
As mentioned above, we further prove that our ℓ_2 robustness guarantee is *tight*: if (1) is all that is known about f, then it is impossible to certify an ℓ_2 ball with radius larger than R. In fact, it is impossible to certify any superset of the ℓ_2 ball with radius R:

Theorem 2.2. *Assume $\underline{p_A} + \overline{p_B} \leq 1$. For any perturbation δ with $\|\delta\|_2 > R$, there exists a base classifier f consistent with the class probabilities (1) for which $g(x + \delta) \neq c_A$.*

Theorem 2.2 shows that Gaussian smoothing naturally induces ℓ_2 robustness: if we make no assumptions on the base classifier beyond the class probabilities (1), then the set of perturbations to which a Gaussian-smoothed classifier is provably robust is *exactly* an ℓ_2 ball. The advantage to this guarantee is that it is completely agnostic to the form of f, which means that it applies to arbitrary architectures and easily scales. However, if f is a neural network it is not technically possible to evaluate the exact integral $\mathbb{E}[f(x+\varepsilon)]$, and therefore we cannot precisely know g, p_A, or p_B. Instead, we can derive high probability bounds on these quantities via Monte Carlo integration—i.e., repeatedly sample $\varepsilon \sim \mathcal{N}(0, \sigma^2 I)$ and evaluate $f(x + \varepsilon)$, then take the empirical expectation. This is the reason that Theorem 2.1 uses bounds on the class probabilities rather than the true values.



(a) Experiments with randomized smoothing on ImageNet for varying settings of the noise σ.



(b) Randomized smoothing compared to a baseline approach.

Results. Figure 1a plots the certified accuracy of a ResNet-50 [He et al., 2016] on ImageNet [Deng et al., 2009] attained by smoothing with varying levels of σ. The dashed black line is the empirical upper bound on the robust accuracy of the base classifier architecture; observe that smoothing improves substantially upon the robustness of the undefended base classifier architecture. We see that σ controls a trade-off between robustness and accuracy: greater noise enables larger certified radii but decreases the average accuracy of the underlying classifier. Figure 1b compares the largest publicly released model from Wong et al. [2018], a small ResNet, to two randomized smoothing classifiers: one which used the same architecture for its base classifier, and one which uses a larger 110-layer ResNet. Observe that smoothing with the larger ResNet substantially outperforms the baseline at all radii, and that smoothing with the small resnet also outperforms the method of Wong et al. [2018] at all but the smallest radii. This emphasizes the advantage enjoyed by methods which can be applied to modern deep networks without needing to modify the training procedure.

2.2 Certifying Robustness to Arbitrary Input Perturbations

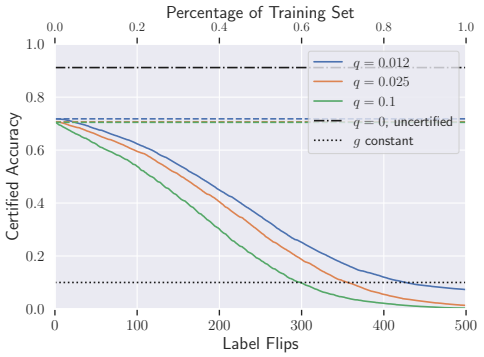
Introduction. The original instantiation of randomized smoothing convolved the input with pixel-wise Gaussian noise, inducing robustness to pixel perturbations with bounded ℓ_2-norm. Functionally, this approach is simply convolving a function's input with noise, thresholding the output, and taking the expectation; the resulting function is then provably Lipschitz [Salman et al., 2020]. Observe that this does not in any way require that the function is a *classifier*. More generally, we can apply this idea to any function with a categorical output. This also means that the input to the function does not need to be an image—by defining an appropriate metric and noise distribution, we can use more generic randomized smoothing procedures [Lee et al., 2019, Dvijotham et al., 2020] to certify robustness of black-box functions to arbitrary input perturbations.

Using this observation for general robustness. An immediate application of this framework is certifying robustness to adversarial attacks on the training data, such as data poisoning. Specifically, the function to be smoothed will include *the entire training procedure* of the classifier, meaning it takes as input the training set in addition to the test input to be classified. Then, analogous to injecting pixel noise at test time, we can derive robustness guarantees by injecting noise into the training set at train time. As a specific example, we can add label noise to the training set to give robustness to label-flipping attacks. However, we still need to evaluate the integral—as our function now includes the training procedure, the previous monte carlo approximation would entail training a new classifier for each noise sample, which is clearly infeasible.

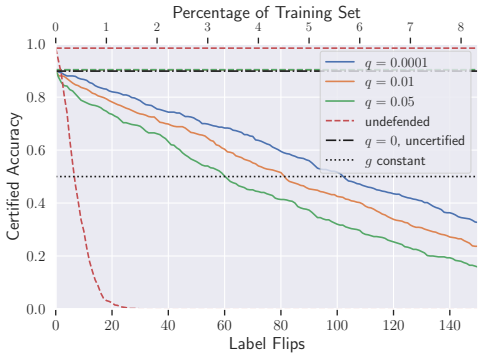
Instead, we can take advantage of the incredible flexibility of pretrained feature embedders and then learn to separate the data in feature space via least-squares classification. The advantage of the least-squares approach is that it reduces the prediction to a linear function of the labels and thus randomizing over the labels is straightforward. Specifically, letting $\mathbf{X} \in \mathbb{R}^{n \times d}$ be the embedded training points, $\mathbf{y}$ the labels, and x_{n+1} the embedded test point, note that the least squares prediction would be $f(x_{n+1}) = x_{n+1}^\top (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}$. Therefore, defining $\alpha = \mathbf{X} \left(\mathbf{X}^T \mathbf{X}\right)^{-1} x_{n+1}^T$, the prediction $f(x_{n+1})$ can be equivalently expressed as $\alpha^T \mathbf{y}$ (this is effectively the kernel representation of the linear classifier). Thus, for any test input x_{n+1} we can compute α one time and then randomly sample many different sets of labels to tractably derive a robustness certificate. Even more compelling, due to the linear structure of this prediction, we can forego sampling entirely and directly bound the tail probabilities using Chernoff bounds. Using binary classification as an example, because the sampled prediction will be 1 whenever $\boldsymbol{\alpha}^T \mathbf{y} \geq 1/2$ and 0 otherwise, we can derive analytical upper and lower bounds on the probability of a given label via the Chernoff bound. Concretely, we can upper bound the probability that the classifier outputs the label 0 by

$$P(\alpha^T \mathbf{y} \leq 1/2) \leq \min_{t>0} \left\{ e^{t/2} \prod_{i=1}^{n} E[e^{-t\alpha_i y_i}] \right\}, \tag{2}$$

which can be evaluated easily since each $\mathbf{y}_i$ is either 0 or 1 with a known probability depending on the noise distribution. Conversely, the probability that the classifier outputs the label 1 is upper bounded by Eq. (2) but evaluated at $-t$. Thus, we can solve the minimization problem unconstrained over t (e.g., with Newton's method), and then let the sign of t dictate which label to predict and the value of t determine the bound.



(a) Certified Accuracy under Label-Flipping Attacks on CIFAR-10



(b) Certified Accuracy under Label-Flipping Attacks on Dogfish

Results. Fig. 2a depicts the results of our method on CIFAR-10 [Krizhevsky and Hinton, 2009]. We use advances in unsupervised representation learning [Chen et al., 2020a] to learn high-quality features without relying on potentially poisoned labels. The hyperparameter q denotes the probability with which each training label was independently flipped for the noise injection process. As with traditional randomized smoothing, this presents a robustness/accuracy trade-off. For example, our classifier with $q = 0.12$ achieves 50% certified accuracy up to 175 labels flips and decays gracefully (random prediction would be infinitely robust but get only 10% accuracy). Note that this represents the number of label flips *separately* per test example—that is, we are certifying robustness to an adversary that could flip the labels of 175 different training points specifically selected for each test example. Fig. 2b plots our results on Dogfish [Koh and Liang, 2017], a binary classification task constructed from the Dog and Fish synsets of ImageNet. For comparison we evaluate an empirical upper bound on the accuracy of an undefended network under an influence-based attack. Here we also see significant certified robustness of this method, enabling better-than-random certified accuracy for attacks which flip as much as 6% of the training labels for each test point. Furthermore, the uncertified accuracy of our least-squares classifier remains quite high for both datasets.

3 Learning Robust Classifiers via Invariance

This chapter is based on Rosenfeld et al. [2021], Rosenfeld et al. [2022b], Chen et al. [2022], and Rosenfeld et al. [2022a].

Introduction. In trying to learn more robust models, a common theme which has repeatedly resurfaced is that of *invariance*. Since generalization to all distributions is impossible, clearly there must be some shared statistical components between the train and test distributions. If we can characterize these components, it is natural to expect that a predictor which is invariant to everything which is *not* shared will generalize broadly. That is, our goal will be to learn a predictor which identifies and uses only information which we expect will give consistent signal, regardless of distribution, and to intentionally ignore any signal which may imply different things in different contexts. It is tempting to begin with this intuition and immediately set out to enforce such an invariance. However, particularly for deep learning, such an approach requires care. In this section I describe work which formally demonstrates the significant failure modes of a large body of work in this direction, and I explore how this insight has informed more rigorous approaches to the problem of invariant generalization.

3.1 A Rigorous Analysis of Invariant Prediction with Latent Features

One strategy which seems promising for achieving robustness is Invariant Causal Prediction (ICP; Peters et al. 2016), which views the task of OOD generalization through the lens of causality. This framework assumes that the data are generated according to a Structural Equation Model (SEM; Bollen 2005), which

consists of a set of structural equations that specify (the distribution over) each variable given its parents. ICP further assumes that the training data can be partitioned into *environments*, where each environment corresponds to interventions on the SEM [Pearl, 2009], but where the equations are unaffected. Since the causal mechanism of each variable is unchanging (unless it is directly intervened upon), learning mechanisms that are the same across environments ensures recovery of invariant features which generalize under arbitrary interventions. This naturally leads to the idea of predicting a target variable using only its direct parents in the causal DAG, which is *minimax-optimal* with respect to all possible environments.

Learning and predicting with deep invariant features. With the rise of deep learning, it is natural to imagine that the complex data typically processed by neural networks follows this causal structure in an unobserved *latent space*; the goal is then to learn a deep representation $\Phi : \mathcal{X} \to \mathbb{R}^d$ (i.e., a neural feature embedder), which will uncover these latent features, to be combined with a linear classifier $\beta \in \mathbb{R}^d$. We refer to this ideal combination as the "optimal invariant predictor"—if such a predictor does exist, the first step for any proposed method is to establish that it will succeed in learning it. In particular, we focus our analysis on the seminal Invariant Risk Minimization objective (IRM; Arjovsky et al. 2019).

Applying the core idea of ICP to neural networks, the IRM objective aims to learn a set of features such that the optimal linear predictor on top of these features is the same for every environment. This follows naturally from the structure of a causal SEM—with enough variation, only the direct parents of the target will enable such a predictor—and in fact this was *proven* to be the correct constraint for observed covariates by [Peters et al., 2016]. Unfortunately, though intuitive, it is not immediately clear whether such an approach will still work when the causal factors are latent. Furthermore, until the present work there were no formal guarantees for the IRM objective and experimental evidence of its success remains non-existent. To gain a deeper understanding of when this type of approach can or cannot be expected to succeed, we design a latent variable model of invariant features which carefully formalizes the implicit assumptions of that work. Under this model, we show that despite being inspired by ICP, the IRM objective can frequently be expected to perform no better than ERM under large shift, particularly for deep learning. However, we also derive the first *positive* results for this algorithm, highlighting some ways in which future methods can provably do better (including work discussed later in this Section).

A model of latent invariant features. We consider an SEM with explicit separation of invariant features z_c, whose joint distribution with the label is fixed for all environments, and environmental features z_e ("non-invariant"), whose distribution can vary, thus formalizing the intuition behind invariant prediction techniques such as IRM. We assume that data are drawn from a set of E training environments $\mathcal{E} = \{e_1, e_2, \ldots, e_E\}$ and that we know from which environment each sample is drawn. For a given environment e, the data are defined by the following process: first, a label $y \in \{\pm 1\}$ is drawn according to a fixed probability: $y = 1$ with probability η, and 0 otherwise. Next, both invariant and environmental features are drawn according to a Gaussian: $z_c \sim \mathcal{N}(y \cdot \mu_c, \sigma_c^2 I)$, $z_e \sim \mathcal{N}(y \cdot \mu_e, \sigma_e^2 I)$, with $\mu_c \in \mathbb{R}^{d_c}, \mu_e \in \mathbb{R}^{d_e}$. Finally, the observation x is generated as via a mixing function applied to the latent features: $x = f(z_c, z_e)$. The complete data generating process is displayed in Fig. 3. We assume infinite samples from each environment; this allows us to isolate the impact of different environments, ignoring finite-sample effects within each one. Our only assumption on f is that it is injective, so that it is in principle possible to recover the latent features from the observations. We write the joint and marginal distributions as $p^e(x, y, z_c, z_e)$. Note also that although we have framed the model as y causing z_c, the causation can just as easily be viewed in the other direction: since the z_c are latent the only restriction is that they satisfy $z_c \perp\!\!\!\perp z_e \mid y$. This means the latent variables also obey the DAG $z_c \to y \to z_e \leftarrow e$ with a bottleneck at y, making z_c the *cause* of y and z_e the *anticause*. Our analysis therefore also considers the task of *Causal Representation Learning* for a particular causal structure. We can now precisely identify the specific learning goal of objectives such as IRM, allowing us to determine when it will succeed.

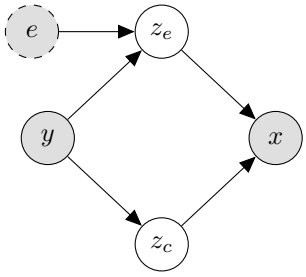


Figure 3: A DAG depicting our model. Shading indicates the variable is observed. Equivalently, z_c could be considered to cause y.

Definition 3.1. Under the above model, the *optimal invariant predictor* is the predictor defined by the composition of a) the featurizer which recovers the invariant features and b) the classifier which is optimal with respect to those features:

$$\Phi^*(x) := \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \circ f^{-1}(x) = [z_c], \quad \beta^* := \begin{bmatrix} \beta_c \\ \beta_0 \end{bmatrix} := \begin{bmatrix} 2\mu_c/\sigma_c^2 \\ \log \frac{\eta}{1-\eta} \end{bmatrix}.$$

The IRM objective. Using the causal DAG intuition from ICP, the IRM objective aims to identify the optimal invariant predictor by solving the constrained optimization problem:

$$\min_{\Phi, \hat{\beta}} \quad \frac{1}{|\mathcal{E}|} \sum_{e \in \mathcal{E}} \mathcal{R}^e(\Phi, \hat{\beta}) \qquad \text{s.t.} \quad \hat{\beta} \in \arg\min_{\beta} \mathcal{R}^e(\Phi, \beta) \quad \forall e \in \mathcal{E}, \tag{3}$$

where $\mathcal{R}^e$ denotes the risk of a predictor in environment e (e.g., expected logistic loss). This bilevel program is highly non-convex and difficult to solve. To find an approximate solution, the authors consider a Langrangian form, whereby the sub-optimality with respect to the constraint is expressed as the squared norm of the gradients of each of the inner optimization problems: $\min_{\Phi, \hat{\beta}} \frac{1}{|\mathcal{E}|} \sum_{e \in \mathcal{E}} \left[\mathcal{R}^e(\Phi, \hat{\beta}) + \lambda \|\nabla_{\hat{\beta}} \mathcal{R}^e(\Phi, \hat{\beta})\|^2 \right]$.

The difficulties of IRM in the linear regime. It remains to establish whether or not the above objective will result in a more robust predictor. Our first result presents matching upper and lower bounds in the setting where f is linear: we demonstrate that observing a large number of environments—equal to number of environmental features d_e—is necessary and sufficient for generalization in the linear regime.

Theorem 3.2 (Linear case)**.** *Assume f is linear. Suppose we observe E training environments. Then the following hold:*

1. *Suppose $E > d_e$. Consider any linear featurizer Φ which is feasible under the IRM objective, with invariant optimal classifier $\hat{\beta} \neq 0$, and write $\Phi(f(z_c, z_e)) = Az_c + Bz_e$. Then under mild non-degeneracy conditions, it holds that $B = 0$. Consequently, $\hat{\beta}$ is the optimal classifier for all possible environments.*

2. *If $E \leq d_e$ and the environmental means μ_e are linearly independent, then there exists a linear Φ—where $\Phi(f(z_c, z_e)) = Az_c + Bz_e$ with rank$(B) = d_e + 1 - E$—which is feasible under the IRM objective. Further, both the logistic and 0-1 risks of this Φ and its corresponding optimal $\hat{\beta}$ are strictly lower than those of the optimal invariant predictor.*

Theorem 3.2 says that when $E \leq d_e$, the global minimum necessarily uses these non-invariant features and therefore will not universally generalize to unseen environments. On the other hand, in the (perhaps unlikely) case that $E > d_e$, any feasible solution will generalize, and the optimal invariant predictor has the minimum (and minimax) risk of all such predictors:

Corollary 3.3. *For both logistic and 0-1 loss, the optimal invariant predictor is the global minimum of the IRM objective if and only if $E > d_e$.*

The failure of IRM in the non-linear regime. We've demonstrated that OOD generalization is difficult in the linear case, but it is achievable given enough training environments. However, the task becomes substantially more difficult when observed x is a non-linear function of the latent causal factors. As this is almost universally presumed to be the case in deep learning, it is just as important to understand how this objective will perform for non-linear f. Unfortunately, we show that in this setting the global minimum of the IRM objective generalizes only to test environments that are sufficiently similar to the training environments. Thus, this objective present no real improvement over ERM under distribution shift. More precisely, we prove that unless we observe enough environments to "cover" the space of non-invariant features, a solution that appears to be invariant can still wildly underperform on a new test distribution. We make use of two constants in the following theorem—the average squared norm of the environmental means, $\overline{\|\mu\|^2} := \frac{1}{E} \sum_{e \in \mathcal{E}} \|\mu_e\|^2$; and the standard deviation of the response variable of the ERM-optimal classifier, $\sigma_{\text{ERM}} := \sqrt{\|\beta_c\|^2 \sigma_c^2 + \|\beta_{e;\text{ERM}}\|^2 \sigma_e^2}$.

Theorem 3.4 (Non-linear case)**.** *Suppose we observe E environments $\mathcal{E} = \{e_1, \ldots, e_E\}$, where $\sigma_e^2 = 1\ \forall e$. Then, for any $\epsilon > 1$, there exists a featurizer Φ_ϵ which, combined with the ERM-optimal classifier $\hat{\beta} = [\beta_c, \beta_{e;ERM}, \beta_0]^T$, satisfies the following properties, where we define $p_\epsilon := \exp\{-d_e \min(\epsilon - 1, (\epsilon - 1)^2)/8\}$:*

1. *The regularization term of $\Phi_\epsilon, \hat{\beta}$ is bounded as $\frac{1}{E}\sum_{e\in\mathcal{E}} \|\nabla_{\hat{\beta}} \mathcal{R}^e(\Phi_\epsilon, \hat{\beta})\|^2 \in \mathcal{O}\left(p_\epsilon^2 \left(c_\epsilon d_e + \overline{\|\mu\|^2}\right)\right)$ for some constant c_ϵ that depends only on ϵ.*
2. *$\Phi_\epsilon, \hat{\beta}$ exactly matches the optimal invariant predictor on at least a $1 - p_\epsilon$ fraction of the training set. On the remaining inputs, it matches the ERM-optimal solution.*

Further, for any test distribution, suppose its environmental mean μ_{E+1} is sufficiently far from the training means: $\forall e \in \mathcal{E}, \min_{y\in\{\pm 1\}} \|\mu_{E+1} - y \cdot \mu_e\| \geq (\sqrt{\epsilon} + \delta)\sqrt{d_e}$ for some $\delta > 0$, and define $q := \frac{2E}{\sqrt{\pi}\delta} \exp\{-\delta^2\}$. Then $\Phi_\epsilon, \hat{\beta}$ is equivalent to the ERM-optimal predictor on at least a $1 - q$ fraction of the test distribution.

The proof proceeds by constructing a "decoy" predictor which is effectively indistinguishable from the optimal invariant predictor but behaves almost identically to the standard ERM solution. We give a brief intuition for each of the claims made above:

1. The first claim says that the predictor will have a penalty term which is exponentially small in d_e. Thus, in high dimensions, it will appear as a perfectly reasonable solution to the objective.
2. The second claim says that this predictor is identical to the invariant optimal predictor on all but an exponentially small fraction of the training data; on the remaining fraction, it matches the ERM-optimal solution, implying that for large enough d_e, the "decoy" predictor will often be a preferred solution. In the finite-sample setting, we would need exponentially many samples to even distinguish between the two!
3. The third claim is the crux of the theorem; it says that this predictor will completely fail to use invariant prediction on most environments. Recall, the intent of IRM is to be robust precisely when ERM breaks down: when the test distribution differs greatly from the training distribution. If we expect the new environments to be similar, ERM already ensures reasonable test performance; thus, IRM fundamentally *does not improve* over ERM in this regime.

Proof Technique. The general idea of the proof is to construct a feature embedder Φ which perfectly recovers the z_c within the high-density regions of the training data but includes z_e everywhere else, and a classifier β which uses both. Thus this predictor's behavior on the training data appears to be exactly that of the optimal invariant predictor, but the fact that the training data has extremely low likelihood everywhere else obscures the fact that it is no more robust than ERM under shift.

Prior analyses of Domain Generalization. Though a few theoretical results of this flavor already exist for learning from multiple domains [Blanchard et al., 2011, Muandet et al., 2013], **all of them assume a prior over domains**—that is, they assume that the train and test distributions are drawn i.i.d. from a fixed meta-distribution. In some sense this merely "kicks the can down the road": it still only gives guarantees *in expectation* via concentration. This is not to downplay these works' contribution: they show enormous foresight in leveraging variation across distributions to improve generalization. However, we argue that this approach does not substantially differ from simply minimizing joint empirical risk. Indeed, we observe that the methods analyzed by those works use essentially that strategy. Instead, our analyses throughout this section emphasizes robustness without a prior (which has since become the more common usage of the term "Domain Generalization"). Here it is not sufficient to have good performance on average, nor can we afford to completely forgo any structural assumptions on how the distribution shifts.

Environment complexity. This work introduced the concept of *environment complexity* in deep learning, which measures the error of a predictor or identifiability of relevant latent features as the number of training environments E grows. This measure has since become a key statistical quantity in the analysis of these algorithms, serving as a standardized framework for comparison. It has also recently been further explored in the field of Causal Representation Learning (e.g., Seigal et al. 2022, Buchholz et al. 2023), which is appropriate as it was inspired by causal discovery via the seminal work of Peters et al. [2016].

Conclusion. This work represents the first analysis of invariant causal prediction with latent factors and the first lower bounds for deep DG. The model we devloped has since been used extensively for the development of improved DG algorithms based on invariance, along with the notion of environment complexity which continues to serve as an important quantity for representation learning algorithms.

3.2 Using Models of Distribution Shift to Develop Provably Robust Predictors

3.2.1 Iterative Feature Matching

Designing more general models of latent distribution shift will allow us to study what kinds of approaches should be expected to work in a given setting and derive precise conditions for them to be provably robust. In particular, another popular approach to DG is *feature matching*, which learns useful representations which also match across the different training distributions. This is typically enforced via a moment-matching constraint or by penalizing the ability of an adversary to discriminate between domains given the embedded features. This idea has been revisited many times [Ganin et al., 2016, Sun and Saenko, 2016], with varying degrees of success, but there was no prior theoretical understanding of why they might work.

Performing a smoothed analysis. To investigate this approach more formally, we begin by studying a slightly more general model than the one in the previous section: we allow for arbitrary covariances of the z, but more importantly, we perform a *smoothed* analysis by modeling the covariance of the z_e to be a random perturbation of an arbitrary (possibly adversarially chosen) PSD matrix. This approach is common for deriving *almost* worst-case performance guarantees by showing that the property of being worst-case is somewhat "brittle"—essentially, it allows us to show that problem instances which suffer worst case performance are sufficiently rare for our purposes. Specifically, with covariance $z_e \sim \mathcal{N}(y \cdot \mu_e, \Sigma_e)$, we suppose that $\Sigma_e := \overline{\Sigma}_e + G_e G_e^\top$, where $\overline{\Sigma}_e$ is an arbitrary PSD matrix and each entry of G_e is drawn from $\mathcal{N}(0, 1)$. This captures more realistic settings where the true data is not chosen to be specifically adversarial with respect to our particular training samples and algorithm.

In this modified setting, we show that moment-matching can be equivalently viewed as an alternative algorithm for learning invariant features and it is therefore a promising candidate for robust Domain Generalization via invariant prediction. Furthermore, our analysis highlights the potential of *iterative* approaches which commit to a particular representation subspace using only a subset of domains before using the next subset. The advantage to this approach is that the training domains cannot "collude": with high probability, if there is a representation that looks ideal for one subset of domains, it will violate our assumed invariance on a different subset. Because of this idea, we name the approach *Iterative Feature Matching* (IFM).

Theoretical results. With our smoothed analysis, we succeed in proving an *exponential* improvement upon the previous linear lower bound for identifying the optimal invariant predictor (we also show the previous lower bounds still apply to IRM and ERM under this new model). Our updated result shows that with high probability, iteratively matching the feature moments enjoys logarithmic environment complexity:

Theorem 3.5 (IFM upper bound)**.** *Under the smoothed model, suppose at each round IFM uses* $|\mathcal{E}| = \tilde{\Omega}(1)$ *training environments. Then with probability* $1 - \exp(-\Omega(d_e))$ *over the randomness in* G_e*, IFM terminates in* $O(\log d_s)$ *rounds and outputs the optimal invariant predictor.*

This result serves as the first formal justification for the observed success of feature matching methods. Furthermore, we show experimentally that updating prior feature matching methods to use the training data iteratively as suggested by our analysis leads to substantial gains to empirical environment complexity, enabling robust generalization with many fewer training domains.

3.2.2 Online Domain Generalization and Subpopulation Shift

Another common way to model shift with multiple training environments is *subpopulation/group shift*, which assumes that the test distribution will lie in the convex hull of the train distribution likelihoods [Duchi et al., 2019, Albuquerque et al., 2020]. Formally, an interpolation of the domains in $\mathcal{E}$ is any distribution which is written $p^\lambda := \sum_{e \in \mathcal{E}} \lambda_e p^e$, where $\lambda \in \Delta_E$ is a vector of convex coefficients. Such a shift is plausible

for many reasonable settings—it captures the existence of fixed subpopulations whose *proportions* in the overall distribution change but whose density does not (e.g., hospitals in different locations). Solving for the minimax-optimal predictor here can be achieved via mirror descent [Nemirovski et al., 2009, Sagawa et al., 2020]. However, this solution is simply equivalent to the predictor which achieves the best worst-population performance. This means such an approach cannot account for *distributions over distributions*, where the proportions may be random, nor can it account for changing proportions over time, which commonly occurs in the real world.

It therefore seems helpful lift this problem to an online game, with the goal of ensuring *asymptotic* robustness which accounts for shifting or random subpopulations. Here we imagine a player deploying classifiers on each timestep t, followed by an adversary presenting a new distribution in the convex hull of populations—the goal of the player is to minimize cumulative regret, defined as the total excess risk suffered over the best fixed predictor in hindsight. By analyzing the statistical and computational properties of this task, we can better understand the nuances of generalization under subpopulation shift. It will also help us develop strategies which are provably robust over long time-horizons.

Results. In our work exploring this idea [**?**], we begin by studying the *value* of this game V_t, defined as the fixed hindsight regret suffered in t rounds under perfect play by both player and adversary. Our first result shows that the structure of subpopulation shift leads to a value which is equivalent to that of online convex optimization. The precise quantity depends on several model-specific constants which are unimportant for the present exposition, so we absorb them into a generic constant C.

Theorem 3.6. *For all $t \in \mathbb{N}$ it holds that $V_t > C \log t$.*

Note that this $O(\log t)$ rate is achieved by the well-known algorithm Follow-The-Leader (FTL), which just plays the minimizer of the sum of all previously seen functions [Hazan et al., 2007]. Observe that this strategy is precisely ERM! In other words, ERM is provably minimax-optimal for long-term regret minimization under subpopulation shift. On a positive note, this implies that if the goal is long-term robustness to subpopulation shift, our existing approach with appropriate regularization is optimal. However, existing methods frequently do *not* succeed in such settings, which suggests that subpopulation shift may be an inadequate model of distribution shift even when it seems intuitively appropriate. The second observation we make is that Theorem 3.6 also implies a (big-O) *lower* bound on the additional regret suffered per round—that is, our analysis uncovers a fundamental barrier to single-round regret as a function of the number of environments already observed. Furthermore, this bound is tight:

Corollary 3.7. *Suppose we've seen E environments. Then the additional regret suffered due to one more round is $\Omega\left(\frac{1}{E}\right)$. This lower bound is attained by ERM.*

3.2.3 Domain-Adjusted Regression

Our earlier study of causality-inspired models of distribution shift targeted full invariance, under the expectation that an adversary could induce a shift via arbitrary interventions. However, it seems unrealistic to expect changes to such a degree in reality, which means that true minimaxity is overly conservative. Instead, we shift our attention to a more realistic latent shift with large but bounded magnitude.

To succeed in this setting, we reframe the problem of DG by thinking of each training domain as a distinct transformation from a shared canonical representation space. In this framing, we can "adjust" each domain in order to undo these transformations, aligning the representations to learn a single robust predictor in this unified space. Thus we name this method *Domain-Adjusted Regression* (DARE). The advantage here is that this adjustment can uncover representations which *previously* varied—and would therefore be discarded—but are now shared among domains and can improve performance. Though there are many possibilities for such an alignment, we study the case of whitening. Formally, given observations x from distributions p^e with mean μ_e and covariance Σ_e, the DARE objective solves

$$\min_{\beta} \sum_{e \in \mathcal{E}} \mathbb{E}_{p^e}[\ell(\beta^T \Sigma_e^{-1/2} x,\ y)] \quad \text{subject to softmax}\left(\beta^T \Sigma_e^{-1/2} \mu_e\right) = \frac{1}{k}\mathbf{1}.\ \forall e \in \mathcal{E}. \tag{4}$$

Analysis. To study the quantifiable benefits of this pre-stage alignment, we consider a model of distribution shift which faithfully captures this idea: $\varepsilon = \varepsilon_0 + b_e,\ y = \mathbf{1}\{\beta^{*T}\varepsilon + \eta \geq 0\},\ x = A_e\varepsilon$. Here, $\varepsilon_0 \sim p^e(\varepsilon_0)$, which we allow to be *any domain-specific distribution*; we assume only that its mean is zero (such that $\mathbb{E}[\varepsilon] = b_e$) and that its covariance exists. We fix $\beta^* \in \mathbb{R}^d$ for all domains and model η as logistic noise. Finally, $A_e \in \mathbb{R}^{d\times d}, b_e \in \mathbb{R}^d$ are domain-specific. Under this model, we are able to show that the solution to DARE is minimax over the set of *bounded* shifts. This bound is quantified as the magnitude $B \in \mathbb{R}$ of test variation which can occur in latent subspaces in which such variation does *not* occur in the training distributions. Such a bound is clearly necessary: if a particular statistical relationship is invariant during training but can change arbitrarily at test time, generalization is impossible. In essence, prior analyses considered only $B = 0$ and $B = \infty$—our work relaxes this to allow for the entire interval:

Theorem 3.8 (DARE risk and minimaxity)**.** *For any $\rho \geq 0$, denote the set of possible test environments $\mathcal{A}_\rho$ which contains all parameters (A'_e, b'_e) subject to the "new variation" magnitude bound B a bound on the mean: $\|b'_e\| \leq \rho$. Let $\hat{\beta}$ be the solution to the* DARE *objective. Then,*

$$\sup_{(A'_e,b'_e)\in\mathcal{A}_\rho} \mathcal{R}_{e'}(\hat{\beta}) = (1+\rho^2)(\|\beta^*\|^2 + 2B\|\beta^*_{\hat{\Pi}}\|\|\beta^*_{I-\hat{\Pi}}\|). \tag{5}$$

Furthermore, the DARE *solution is* minimax*: $\hat{\beta} \in \arg\min_{\beta\in\mathbb{R}^d} \sup_{(A'_e,b'_e)\in\mathcal{A}_\rho} \mathcal{R}_{e'}(\beta)$.*

4 Understanding Heavy-Tailed Data in the Real World, and How to Use it

This chapter is based on Rosenfeld et al. [2022a], Rosenfeld and Garg [2023], and Rosenfeld and Risteski [2023].

Introduction. Though inspired by our understanding of what causes real-world shift (e.g., causal interventions or subpopulation shift), the methods in the previous section use no information about the test distribution we will actually be evaluated on. Instead, many of them generically assume a "meta-model" of shift which describes the limits of data variation and then infer these limits from the training data. Though this approach makes formal guarantees much more tractable, it can also sometimes detach the subsequent analysis from practical implications. To keep these efforts grounded, it is therefore also important to work to understand heavy tails and distribution shift as they occur in natural data. Such exploration leads to robustness guarantees of a less abstract nature; it also helps us to understand the nature of real-world shift, giving actionable insight towards the best way to reason about it and address it in practice.

4.1 Relaxing the Access Model

Introduction. The above works primarily emphasize minimaxity, aiming to learn a single robust predictor which must generalize equally well to all shifts of a given structure. Though this is a well-studied statistical framework, it is both incredibly restrictive and not entirely realistic. First, it requires ignoring *all* non-invariant sources of information, even those which would be helpful in almost all settings of practical interest. This is because of the "technically possible" worst case where using this information could in principle be harmful, even if such a case is extremely unlikely to occur. In other words, though robust, minimaxity is highly sub-optimal on average in most settings. Furthermore, these approaches are based on the premise that one has no knowledge whatsoever of the test environment before deploying the learned predictor. In practice, this is often not the case—at the bare minimum, simply *using* the predictor requires access to unlabeled inputs from the test distribution.

A natural next question is whether we can relax this restrictive access model, robustly (and provably) improve our predictor with this data. Notably, it is often unrealistic to expect to be able retrain or finetune a large neural network deployed in the wild; sometimes it is completely impossible (e.g., when the model is too large to evaluate locally or proprietary so the embeddings come from an API). Though methods like self-training [Chen et al., 2020b] or "test-time training" [Wang et al., 2020] show promise for adapting to new

distributions with only unlabeled data, they cannot be used in such settings. Thus, we focus on methods which can efficiently leverage test data to the extent possible without expensive training.

4.1.1 Standard Finetuning Already Learns Features Sufficient for Robust Prediction

In Rosenfeld et al. [2022a], we first ask: to what degree do we need to modify existing embedding backbones *at all?* It is commonly assumed that the lack of robustness in neural networks arises from the use of "shortcuts" or "spurious features" and that the only way to correct this is to train the network with a modified objective which penalizes the dependence on such features. Indeed, this is the intent of many of the invariance based approaches described above. However, these objectives are difficult to tune and optimize and expensive to iterate. This motivates us to study whether the observed failures under shift are primarily because of (i) learning the "wrong" features or (ii) failing to robustly predict using the features we have.

We begin with a simple experiment (Fig. 4) to try to distinguish between these two possibilities: we finetune a deep network with standard training on several domain generalization benchmarks from DomainBed [Gulrajani and Lopez-Paz, 2021]. After training, we freeze the features and separately retrain just the last linear layer. Crucially, when doing this, we give it an unreasonable advantage by optimizing on both the train and test domains—henceforth we refer to this as "cheating". Since we use just a linear classifier, this process establishes a lower bound on what performance we could plausibly achieve using standard features, with access to data from the target distribution. We then separately cheat while training the full network end-to-end, simulating the idealized setting with no distribution shift.

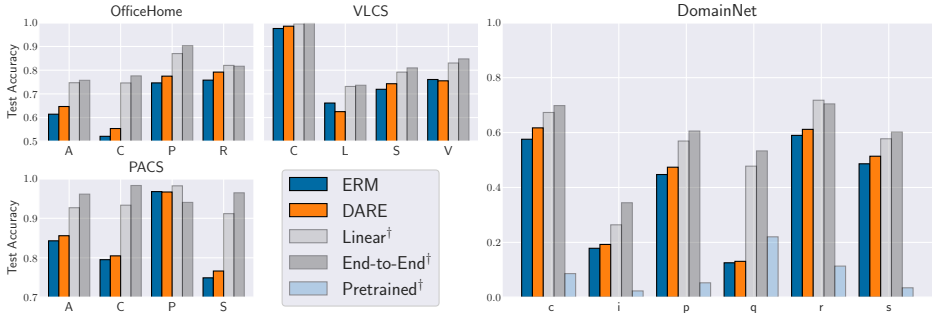


Figure 4: Comparing standard training performance on several datasets from the DomainBed benchmark to *potential* performance were we allowed to retrain the linear head with access to test domain data. Simply retraining the last linear layer substantially improves over the baseline and is usually almost as good as retraining the whole network end-to-end.

Surprisingly, we find that simple (cheating) logistic regression on features learned via standard finetuning results in enormous improvements over current state of the art, on the order of 10-15%. In fact, it usually performs comparably to the full cheating method—which trains the network end-to-end with test domain access—sometimes even outperforming it. Put another way, cheating while training the entire network rarely does significantly better than cheating while retraining just the last linear layer. This suggests that finetuning modern deep architectures with established training and regularization practices may be "good enough" for learning features which generalize out-of-distribution and that the current bottleneck lies primarily in learning a simple, robust predictor.

4.1.2 Using Unlabeled Data to Provably Adapt Just-in-Time or Bound Test Error

An obvious implication of the previous finding is that with a small amount of labeled test data, simply retraining the last layer on this data will significantly improve on current methods and is likely to be preferable to retraining the entire network. However, it is equally important to note that this observation can be used even *without* labeled test data. To explore this, we first consider a simple extension of our DARE objective to the task of unsupervised domain adaptation "just-in-time". Here, our method uses unlabeled test data to rapidly estimate a better adjustment for the test domain without any form of retraining. This results in quantifiable benefits to sample complexity under a simple Gaussian model:

Theorem 4.1. *Suppose we observe $n_s = \Omega(m(\Sigma_S)d^2)$ samples in $\mathbb{R}^d$ from source distribution $\mathcal{N}(\mu_S, \Sigma_S)$ and $n_T = \Omega(m(\Sigma_T)d^2)$ samples from target distribution $\mathcal{N}(\mu_T, \Sigma_T)$. Next we solve the unconstrained DARE objective over the source data and predict $\hat{\beta}^T \hat{\Sigma}_T^{-1/2} x$ on the target data. Then with probability at least $1 - 3d^{-1}$, the excess squared risk of our predictor on this new environment is bounded as*

$$\mathcal{R}_T = \mathcal{O}\left(d^2 \|\mu_T\|^2 \left(\frac{m(\Sigma_S)}{n_S} + \frac{m(\Sigma_T)}{n_T}\right)\right).$$

Bounding test error. Next, we consider the possibility that knowledge of the *existence* of a robust linear classifier can benefit us, even if we have no way of identifying it. In Rosenfeld and Garg [2023] we show how it can lead to (almost) provable bounds on the test error of deep neural networks under distribution shift. Previous algorithms for estimating error with unlabeled data focus only on the average case; while they achieve good accuracy, they consistently *under*estimate error, particularly on the heavy tail of large distribution shift—this is precisely when an accurate estimation is most important. We instead aim to derive reliable (conservative) test error bounds. Notably, all previous bounds of this kind are vacuous when applied to deep neural networks, but we also observe that they typically rely on uniform convergence. On the other hand, our experimental finding above highlights the fact that **real-world distribution shift is never truly worst-case**, suggesting that uniform convergence results may not be necessary in practice. This is because we know that a linear classifier exists which gets very high accuracy on both distributions, so if all linear functions which agree with the training labels would imply our classifier has low error on the target data, we should consider it more likely that our classifier indeed has low error than that the true labeling function is worst-case. We formalize this notion into a simple, intuitive assumption, from which we then derive a test error bound using unlabeled data.

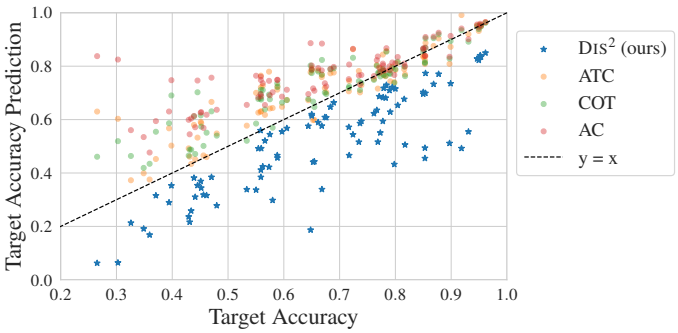


Figure 5: Comparison of our method DIS^2, which uses unlabeled data to provide non-vacuous error bounds under shift, to prior methods for error estimation [Garg et al., 2022, Lu et al., 2023]. Though slightly more accurate on average, they consistently overestimate accuracy, particularly when the distribution shift is large.

We omit the precise form of the bound for space, but Fig. 5 depicts its performance. We see that unlike the prior SOTA, it is consistently conservative in that it never overestimates test accuracy; i.e., it gives a reliable error upper bound. Furthermore, its average accuracy is extremely competitive with these baselines.

4.2 Exploring the Effect of Outliers on Neural Network Optimization

In my most recent work [Rosenfeld and Risteski, 2023], I shift focus from distribution shift to a related question: how do heavy tails in the training data affect the behavior (e.g. generalization) of a learned model? During this exploration, I uncovered the surprisingly large influence of paired groups of outliers on the training dynamics of neural networks (and, consequently, the learned model). These groups are characterized by the inclusion of one or more large magnitude features that dominate the network's output, providing large, consistent, and *opposing* gradients; because of this structure, we refer to them as *Opposing Signals*. Many of these features are largely irrelevant to the target task but, crucially, they do meaningfully correlate with it. In fact, in many cases these features perfectly encapsulate the classic statistical conundrum of "correlation vs. causation" which plays a large role in our understanding of the failure of deep networks

to generalize. For example, a bright blue sky background does not determine the label of a CIFAR image, but it does most often occur in images of planes. Other features *are* relevant, such as the presence of wheels and headlights on trucks and cars, or whether punctuation comes before after the end of a quotation.

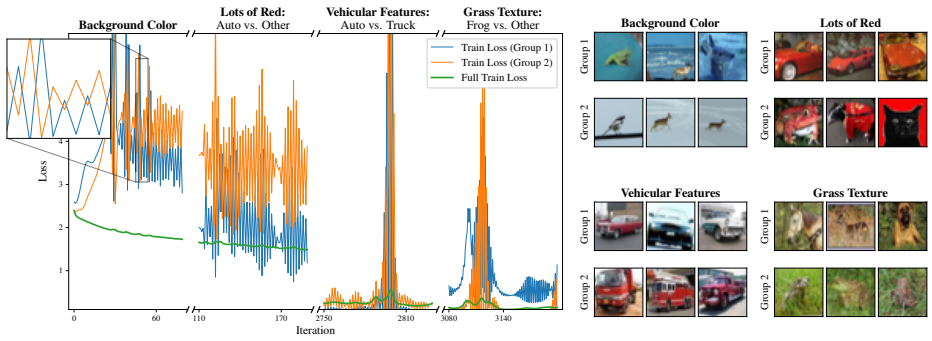


Figure 6: We plot the overall loss of a ResNet-18 trained with GD on CIFAR-10, plus the losses of a small but representative subset of outlier groups. These groups have consistent *opposing signals* (e.g., wheels and headlights sometimes means `car` and sometimes `truck`). Throughout training, losses on these groups oscillate with growing amplitude—this oscillation has an obvious correspondence to the short term spikes in overall training loss and, in particular, appear to be the direct cause of the "edge-of-stability" phenomenon.

Fig. 6 depicts the training loss of a ResNet-18 trained with full-batch gradient descent on CIFAR-10, along with a few dominant outlier groups and their respective losses. Though this proposal omits the details of the precise dynamics which lead to the depicted behavior, we emphasize the natural opposing structure of the visualized outliers, as well as the enormous influence they appear to have on the overall train loss—in particular, they appear to be the direct cause of the "edge-of-stability" phenomenon [Cohen et al., 2021]. In the paper we present extensive experimental results further exploring this finding, and we identify possible connections to a wide variety of other observations, including grokking, simplicity bias, and the effectiveness of training techniques such as Adam, and Sharpness-Aware Minimization. We conjecture that further exploration of these opposing signals will give a much deeper understanding of how training data (and outliers in particular) influence the optimization and learned behavior of neural networks. In the long run we expect this to enable improved methods for robust training of deep networks.

5 Future Work

5.1 Finishing Work on Robust One-Shot Strategic Classification

I am close to completing a new work on strategic classification when the agents' cost function is unknown [Rosenfeld and Rosenfeld, 2023]. While prior works study this setting through the lens of online learning, the repeated deployment of predictors this requires is infeasible in many settings. Furthermore, these analyses come only with asymptotic regret bounds; this somewhat abstract quantity does not account for real short-term costs. Thus we argue the importance for *one-shot* strategic classification, which requires committing to a single robust classifier, once. Instead, we assume an uncertainty set over cost functions and derive efficient algorithms which provably converge to the minimax-optimal predictor under strategic repsonse. It remains to prove a final lower bound and write the rest of the paper—I anticipate completing this by the late October/early November, to be submitted for publication shortly after.

5.2 Further Use of Unlabeled Data for "Just-in-Time" Adaptation

I am currently working with two other students to understand the effect of BatchNorm (BN) [Ioffe and Szegedy, 2015] on the performance of neural networks under distribution shift. Prior work [Nado et al., 2020] has shown that simply allowing BN to use the test data statistics can improve performance substantially—this is functionally similar to unsupervised adaptation with DARE. We have observed empirically that (i)

this effect is much more pronounced for synthetic shifts such as blurring or noise, and (ii) the benefit of this modification can be achieved (or sometimes surpassed) by updating the statistics of only a single layer. As synthetic benchmarks have limited value for practical purposes, we are currently investigating what kind of modifications would be necessary for this approach to work on realistic distribution shift, what conditions would enable provable guarantees, and what our current findings may imply about how real-world distribution shift differs from synthetic benchmarks. I plan to continue working on this through the fall, and I expect we will be able to modify this method to address realistic shift by early next semester.

5.3 Improving on Domain-Adversarial Representations with a Corrected Loss

In Rosenfeld and Garg [2023] we observed that existing methods which optimize *dis*agreement use incorrect proxy losses which do not capture the true objective of interest; replacing these with a theoretically justified loss led to substantial downstream improvements. I have recently begun advising another student on a project which investigates how this observation can be applied to other adversarial objectives. These objectives similarly use an ill-suited loss and our initial experiments indicate that these methods can be made much more effective and robust to hyperparameter selection by making a similar correction. I plan to complete this project by the middle of next semester.

5.4 Identifying and Addressing Meaningful Threat Models for LLMs

I have recently begun to study what kinds of guarantees are feasible for massive, complex models such as LLMs. It would ideally be possible to treat these models as we do cyberpysical systems [Platzer, 2010], giving guarantees on the boundaries of their reachable states according to a set of rules which govern their complex dynamics. Our limited understanding of their inner workings, combined with the massive space of possible behaviors, makes this problem exceedingly complex.

A current goal of mine which is a bit more tractable is to derive conditions which will allow a practitioner to verify that in the course of both normal and adversarial interactions, an LLM will never output a *specific phrase* which appears in its training set—such a guarantee would make great progress towards LLMs with discretion which can be trained on sensitive data yet trusted to interact with agents who should not have access to it. Though ensuring this in general may not be possible, an alternative is to guarantee an upper bound on the excess probability that an LLM will do this over a baseline model which does not observe this phrase during training. I have made some exciting initial progress towards training methods which enable such guarantees, though the settings of normal and adversarial interactions require separate treatments. I plan to continue to work towards this goal; though this idea is still in its early stages, I expect to have preliminary experimental results by the end of the semester and I intend to complete this project before graduating next semester.

I am also exploring new ideas for a collaborative project along similar lines, thought with more of a focus on adversarial prompts. We will further flesh this direction out in the coming months but we don't have anything concrete as of now.

5.5 Better Understanding the Downstream Effects of Outliers with Opposing Signals

I am very excited about my most recent work which identifies a possible common factor underlying several prior observations of neural network training dynamics—this finding offers a new lens through which to study modern stochastic optimization and understand how different training choices affect optimization and generalization. I have some concrete initial ideas for demonstrating how, under certain conditions, this observation would provably explain the improved generalization capabilities of Sharpness-Aware Minimization [Foret et al., 2021] and how this might inform development of future algorithmic improvements. I will continue to develop this idea through this semester, with the goal of completing it before I graduate next semester. Furthermore, this finding arose from my investigation into a new approach to deriving generalization bounds for neural networks. I am looking forward to exploring this direction further—though I don't have anything too concrete, I have several ideas for followup experiments and with luck it may give rise to additional results of a similar nature.

References

Isabela Albuquerque, João Monteiro, Mohammad Darvishi, Tiago H. Falk, and Ioannis Mitliagkas. Generalizing to unseen domains via distribution matching. *arXiv preprint arXiv:1911.00804*, 2020.

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018*, July 2018.

Battista Biggio, Blaine Nelson, and Pavel Laskov. Poisoning attacks against support vector machines. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*, pages 1467–1474, USA, 2012.

Gilles Blanchard, Gyemin Lee, and Clayton Scott. Generalizing from several related classification tasks to a new unlabeled sample. *Advances in neural information processing systems*, 24, 2011.

Kenneth A Bollen. Structural equation models. *Encyclopedia of biostatistics*, 7, 2005.

Simon Buchholz, Goutham Rajendran, Elan Rosenfeld, Bryon Aragam, Bernhard Schölkopf, and Pradeep Ravikumar. Learning linear causal representations from interventions under general nonlinear mixing. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Nicholas Carlini and David Wagner. Adversarial examples are not easily detected: Bypassing ten detection methods. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, pages 3–14, New York, NY, USA, 2017.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020a.

Xuxi Chen, Wuyang Chen, Tianlong Chen, Ye Yuan, Chen Gong, Kewei Chen, and Zhangyang Wang. Self-pu: Self boosted and calibrated positive-unlabeled training. In *International Conference on Machine Learning*, pages 1510–1519. PMLR, 2020b.

Yining Chen, Elan Rosenfeld, Mark Sellke, Tengyu Ma, and Andrej Risteski. Iterative feature matching: Toward provable domain generalization with logarithmic environments. In *Advances in Neural Information Processing Systems*, volume 35, pages 1725–1736. Curran Associates, Inc., 2022.

Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 1310–1320. PMLR, 09–15 Jun 2019.

Jeremy Cohen, Simran Kaur, Yuanzhi Li, J Zico Kolter, and Ameet Talwalkar. Gradient descent on neural networks typically occurs at the edge of stability. In *International Conference on Learning Representations*, 2021. URL `https://openreview.net/forum?id=jh-rTtvkGeM`.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *Computer Vision and Pattern Recognition (CVPR)*, 2009.

John Duchi, Tatsunori Hashimoto, and Hongseok Namkoong. Distributionally robust losses for latent covariate mixtures. *arXiv preprint arXiv:2007.13982*, 2019.

Krishnamurthy Dvijotham, Jamie Hayes, Borja Balle, Zico Kolter, Chongli Qin, Andras Gyorgy, Kai Xiao, Sven Gowal, and Pushmeet Kohli. A framework for robustness certification of smoothed classifiers using f-divergences. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020, Conference Track Proceedings*, 2020.

Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=6Tm1mposlrM.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *J. Mach. Learn. Res.*, 17(1):2096–2030, jan 2016. ISSN 1532-4435.

Saurabh Garg, Sivaraman Balakrishnan, Zachary Chase Lipton, Behnam Neyshabur, and Hanie Sedghi. Leveraging unlabeled data to predict out-of-distribution performance. In *International Conference on Learning Representations*, 2022.

Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. In *International Conference on Learning Representations*, 2021.

Elad Hazan, Amit Agarwal, and Satyen Kale. Logarithmic regret algorithms for online convex optimization. *Machine Learning*, 69(2):169–192, 2007. URL https://doi.org/10.1007/s10994-007-5016-8.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *Computer Vision and Pattern Recognition (CVPR)*, 2016.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. pmlr, 2015.

Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, pages 1885–1894, 2017.

Alex Krizhevsky and Geoffrey Hinton. Learning Multiple Layers of Features from Tiny Images. Technical report, Citeseer, 2009.

Mathias Lecuyer, Vaggelis Atlidakis, Roxana Geambasu, Daniel Hsu, and Suman Jana. Certified robustness to adversarial examples with differential privacy. In *2019 IEEE symposium on security and privacy (SP)*, pages 656–672. IEEE, 2019.

Guang-He Lee, Yang Yuan, Shiyu Chang, and Tommi S. Jaakkola. Tight certificates of adversarial robustness for randomly smoothed classifiers. In *Advances in Neural Information Processing Systems 31*, 2019.

Bai Li, Changyou Chen, Wenlin Wang, and Lawrence Carin. Certified adversarial robustness with additive noise. *Advances in neural information processing systems*, 32, 2019.

Yuzhe Lu, Zhenlin Wang, Runtian Zhai, Soheil Kolouri, Joseph Campbell, and Katia Sycara. Predicting out-of-distribution error with confidence optimal transport. In *ICLR 2023 Workshop on Pitfalls of limited data and computation for Trustworthy ML*, 2023.

Krikamol Muandet, David Balduzzi, and Bernhard Schölkopf. Domain generalization via invariant feature representation. In *International Conference on Machine Learning*, pages 10–18. PMLR, 2013.

Zachary Nado, Shreyas Padhy, D Sculley, Alexander D'Amour, Balaji Lakshminarayanan, and Jasper Snoek. Evaluating prediction-time batch normalization for robustness under covariate shift. *arXiv preprint arXiv:2006.10963*, 2020.

Arkadi Nemirovski, Anatoli Juditsky, Guanghui Lan, and Alexander Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on optimization*, 19(4):1574–1609, 2009.

Judea Pearl. *Causality*. Cambridge university press, 2009.

Jonas Peters, Peter Bühlmann, and Nicolai Meinshausen. Causal inference by using invariant prediction: identification and confidence intervals. *Journal of the Royal Statistical Society*, 2016.

André Platzer. *Logical analysis of hybrid systems: proving theorems for complex dynamics*. Springer Science & Business Media, 2010.

Aditi Raghunathan, Jacob Steinhardt, and Percy Liang. Certified defenses against adversarial examples. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=Bys4ob-Rb.

Elan Rosenfeld and Saurabh Garg. (Almost) provable error bounds under distribution shift via disagreement discrepancy. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Elan Rosenfeld and Andrej Risteski. Outliers with opposing signals have an outsized effect on neural network optimization. *(In Submission) arXiv preprint arXiv:2311.04163*, 2023.

Elan Rosenfeld and Nir Rosenfeld. One-shot strategic classification under unknown costs. *(In Preparation), arXiv preprint arXiv:2311.02761*, 2023.

Elan Rosenfeld, Ezra Winston, Pradeep Ravikumar, and Zico Kolter. Certified robustness to label-flipping attacks via randomized smoothing. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 8230–8241. PMLR, 13–18 Jul 2020.

Elan Rosenfeld, Pradeep Kumar Ravikumar, and Andrej Risteski. The risks of invariant risk minimization. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=BbNIbVPJ-42.

Elan Rosenfeld, Pradeep Ravikumar, and Andrej Risteski. Domain-adjusted regression or: Erm may already learn features sufficient for out-of-distribution generalization. *arXiv preprint arXiv:2202.06856*, 2022a.

Elan Rosenfeld, Pradeep Ravikumar, and Andrej Risteski. An online learning approach to interpolation and extrapolation in domain generalization. In *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pages 2641–2657. PMLR, 28–30 Mar 2022b.

Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=ryxGuJrFvS.

Hadi Salman, Andrew Ilyas, Logan Engstrom, Ashish Kapoor, and Aleksander Madry. Do adversarially robust imagenet models transfer better? *Advances in Neural Information Processing Systems*, 33:3533–3545, 2020.

Anna Seigal, Chandler Squires, and Caroline Uhler. Linear causal disentanglement via interventions. *arXiv preprint arXiv:2211.16467*, 2022.

Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *Computer Vision – ECCV 2016 Workshops*, pages 443–450, Cham, 2016. Springer International Publishing. ISBN 978-3-319-49409-8.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing Properties of Neural Networks. In *International Conference on Learning Representations (ICLR)*, 2014.

Jonathan Uesato, Brendan O'donoghue, Pushmeet Kohli, and Aaron Oord. Adversarial risk and the dangers of evaluating against weak attacks. In *International Conference on Machine Learning*, pages 5025–5034. PMLR, 2018.

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. *arXiv preprint arXiv:2006.10726*, 2020.

Eric Wong and Zico Kolter. Provable defenses against adversarial examples via the convex outer adversarial polytope. In *International conference on machine learning*, pages 5286–5295. PMLR, 2018.

Eric Wong, Frank Schmidt, Jan Hendrik Metzen, and J Zico Kolter. Scaling provable adversarial defenses. *Advances in Neural Information Processing Systems*, 31, 2018.