10-601 Recitation #10
PCA, Clustering, and Constrained Optimization

November 15th, 2011

Shing-hon Lau

Office hours: Friday 3-4 PM

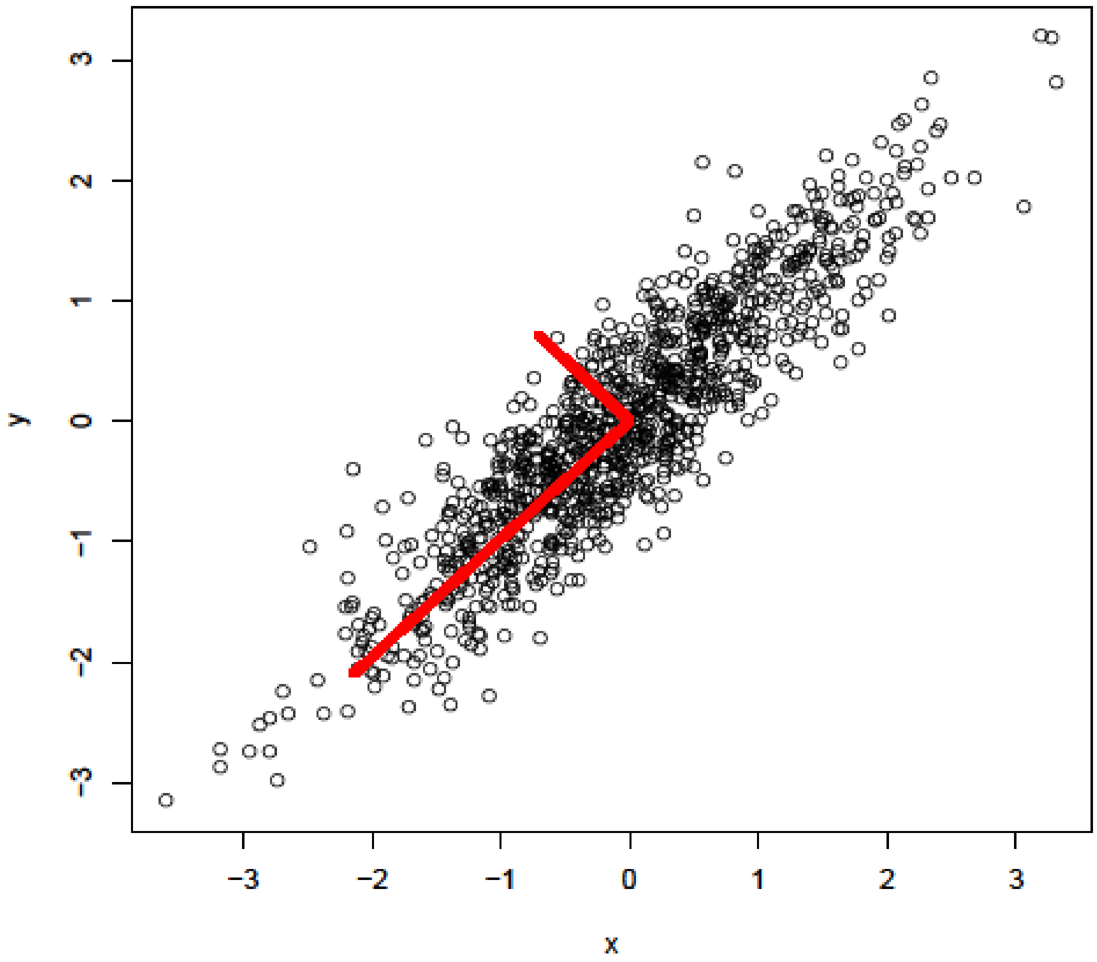
Agenda

- HW #5 due Monday 5 PM
  - Submit written copy
- Keep working on projects!
- PCA/ICA
- Clustering (k-means and spectral)
- Constrained optimization

Principal Component Analysis (PCA)

- Want to find the *k* most important components in the data
- Choose only orthogonal components
- Maximize variance captured or minimize reconstruction error

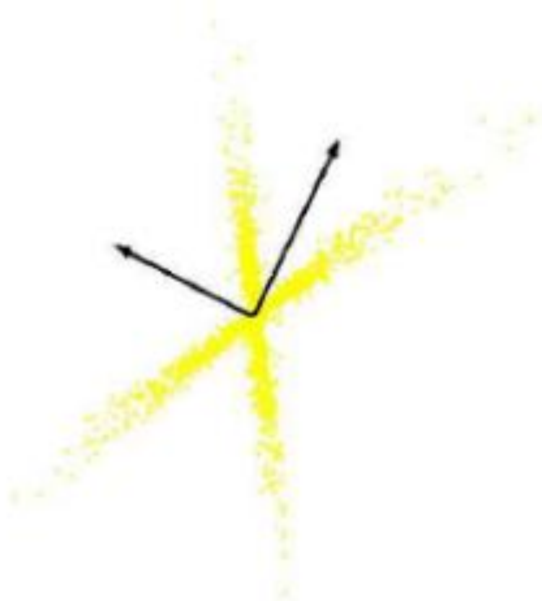
Principal Component Analysis (PCA)

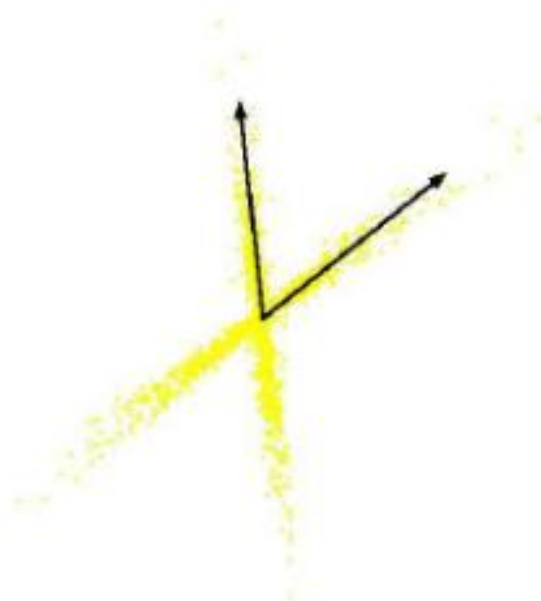


Principal Component Analysis (PCA)

- Principal components are the eigenvectors of XX^T
- Eigenvector tells you the direction
- Eigenvalue tells you the importance
- Preserving the top *k* components performs dimensionality reduction
- Project data onto the *k* components

Independent Component Analysis (ICA)





PCA
(orthogonal coordinate)

ICA
(non-orthogonal coordinate)

Clustering

- Why cluster?
  - Have a bunch of data and want to see if there are natural groupings
- Generally try to minimize $\sum_{j=1}^{m} d(\mu_{C(j)}, x_j)$
- Main issues are:
  - How many clusters do we want? What is *k*?
  - How do we measure close?

K-means Clustering

- User specifies *k*
- Algorithm is **NOT** capable of learning *k*
- Measure of closeness is Euclidean distance (in classical *k*-means)
- Learning procedure is EM (!)

K-means Clustering

Algorithm

Input – Desired number of clusters, *k*

Initialize – the *k* cluster centers (randomly if necessary)

Iterate –

1. Assign the objects to the nearest cluster centers

2. Re-estimate the *k* cluster centers (aka the centroid or mean) based on current assignment

$$\vec{\mu}_k = \frac{1}{\mathcal{C}_k} \sum_{i \in \mathcal{C}_k} \vec{x}_i$$

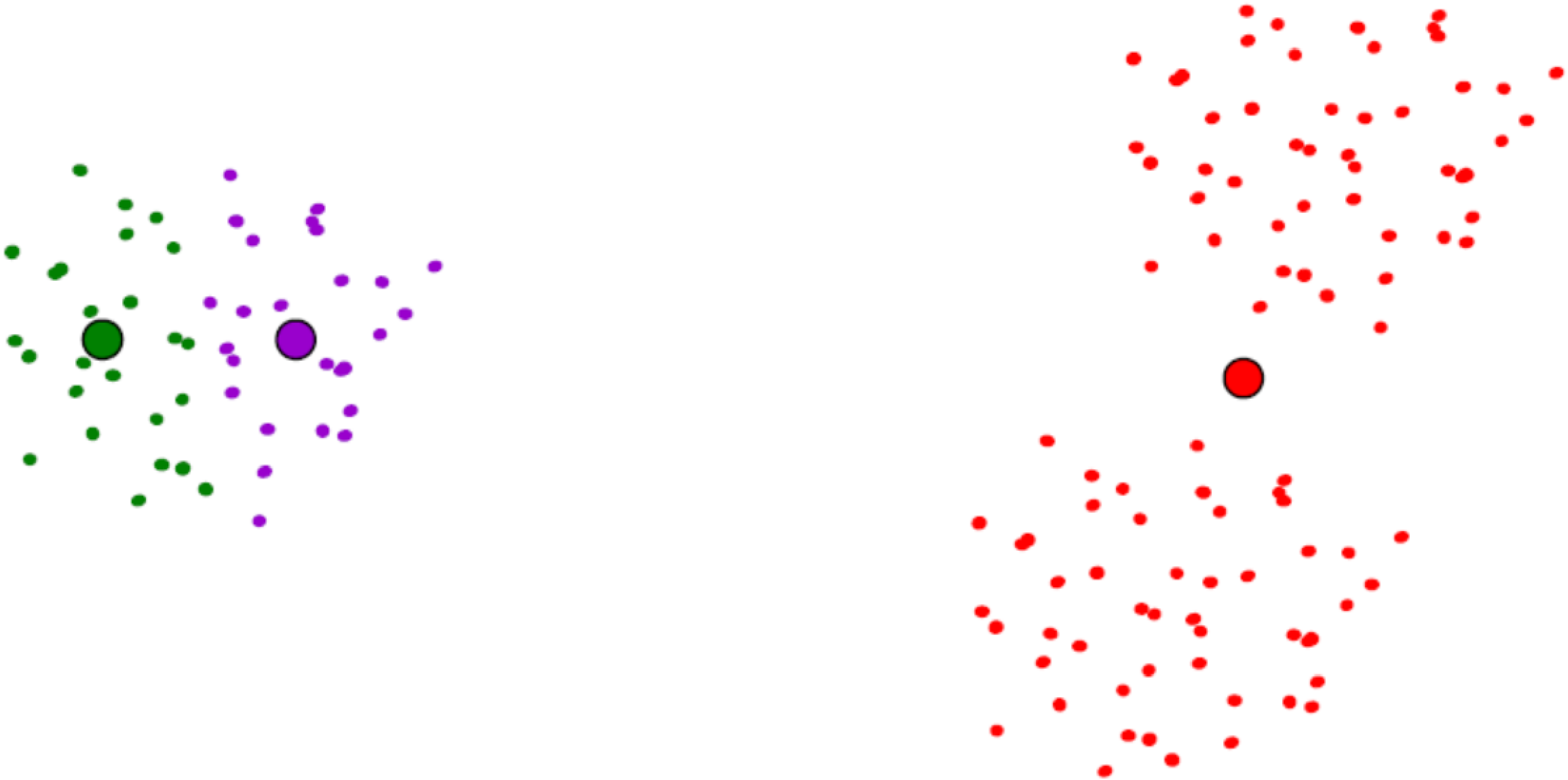
Termination –

If none of the assignments changed in the last iteration, exit. Otherwise go to 1.

K-means Clustering

- Note that the objective function may not be convex! $\sum_{j=1}^{m} d(\mu_{C(j)}, x_j)$
- That means we can fall into local optima
- In practice, this means bad clusters!

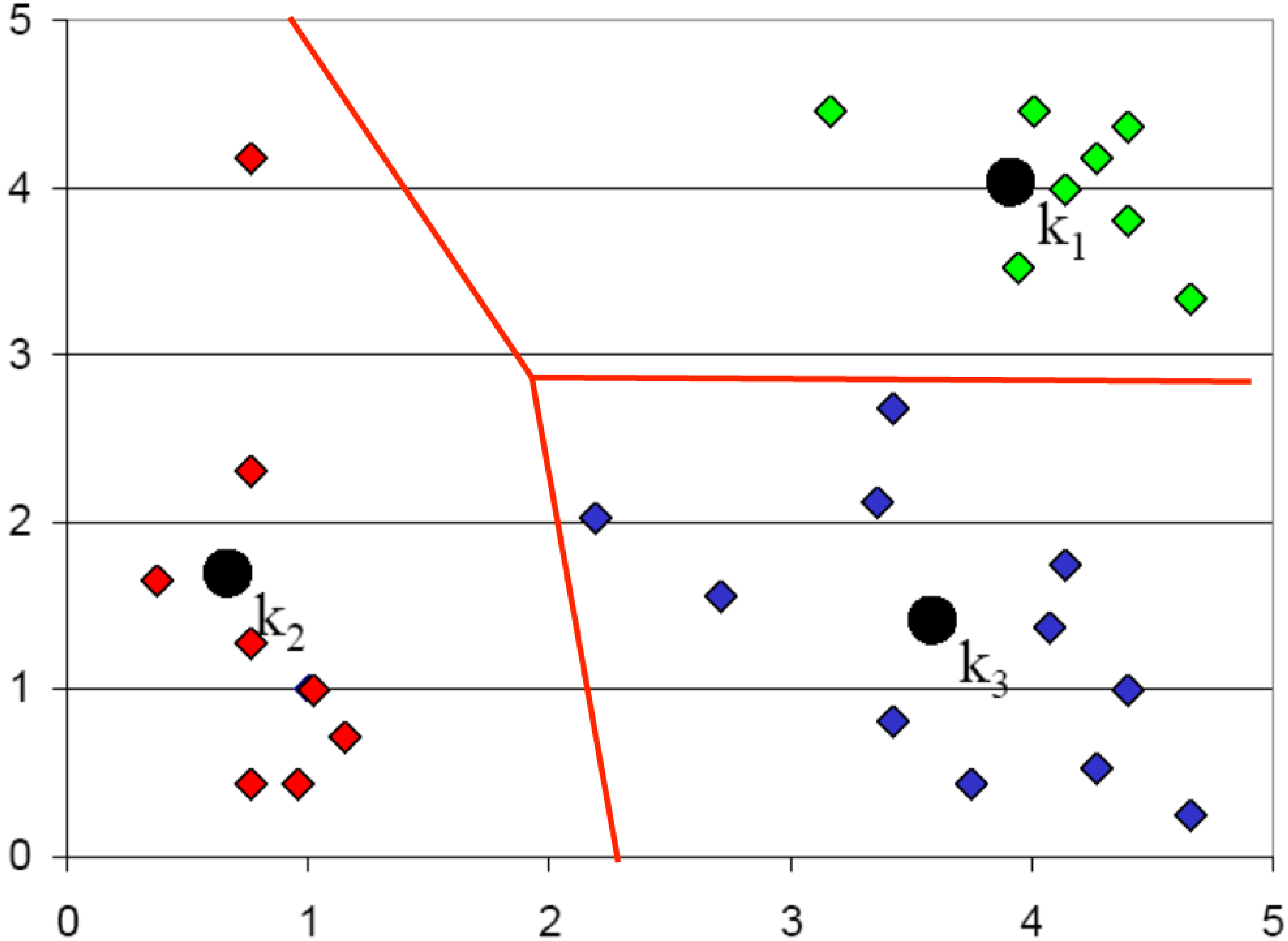
K-means Clustering



K-means Clustering

- Note that the objective function may not be convex! $\sum_{j=1}^{m} ||\mu_{C(j)} - x_j||^2$
- That means we can fall into local optima
- In practice, this means bad clusters!
- Can only find convex cluster boundaries

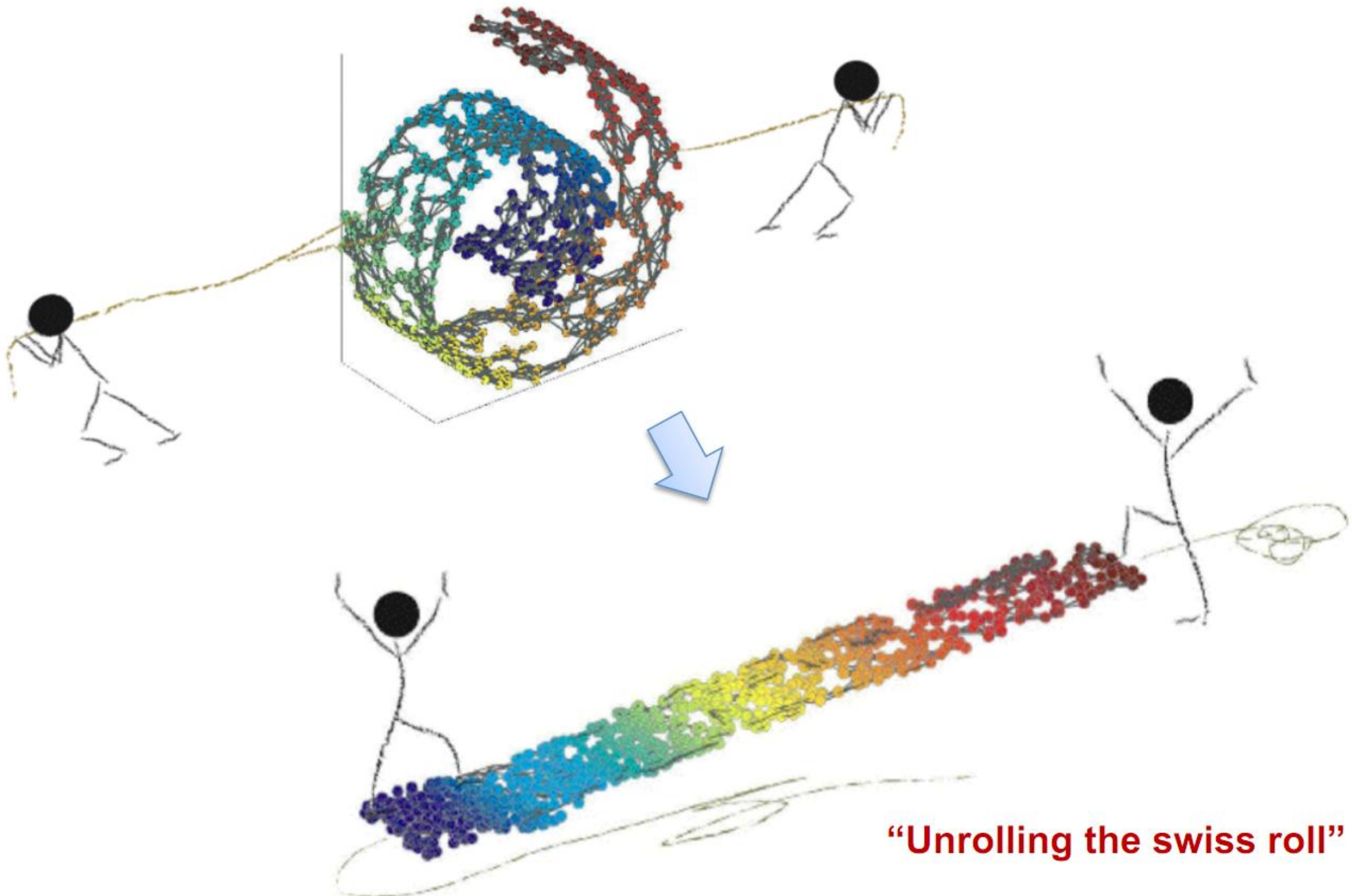
K-means Clustering



K-means Clustering

- Note that the objective function may not be convex! $\sum_{j=1}^{m} ||\mu_{C(j)} - x_j||^2$
- That means we can fall into local optima
- In practice, this means bad clusters!
- Can only find convex clusters
- Can use spectral clustering to get non-convex clusters

Spectral Clustering



Constrained Optimization

- So far in this class we have only seen unconstrained optimization

$$\hat{\theta} = argmax_{\theta} P(D|\theta)$$

$$W_{MCLE} = argmax_W \prod_l P(Y^l|W, X^l)$$

$$argmin_C \sum_{j=1}^{m} ||\mu_{C(j)} - x_j||^2$$

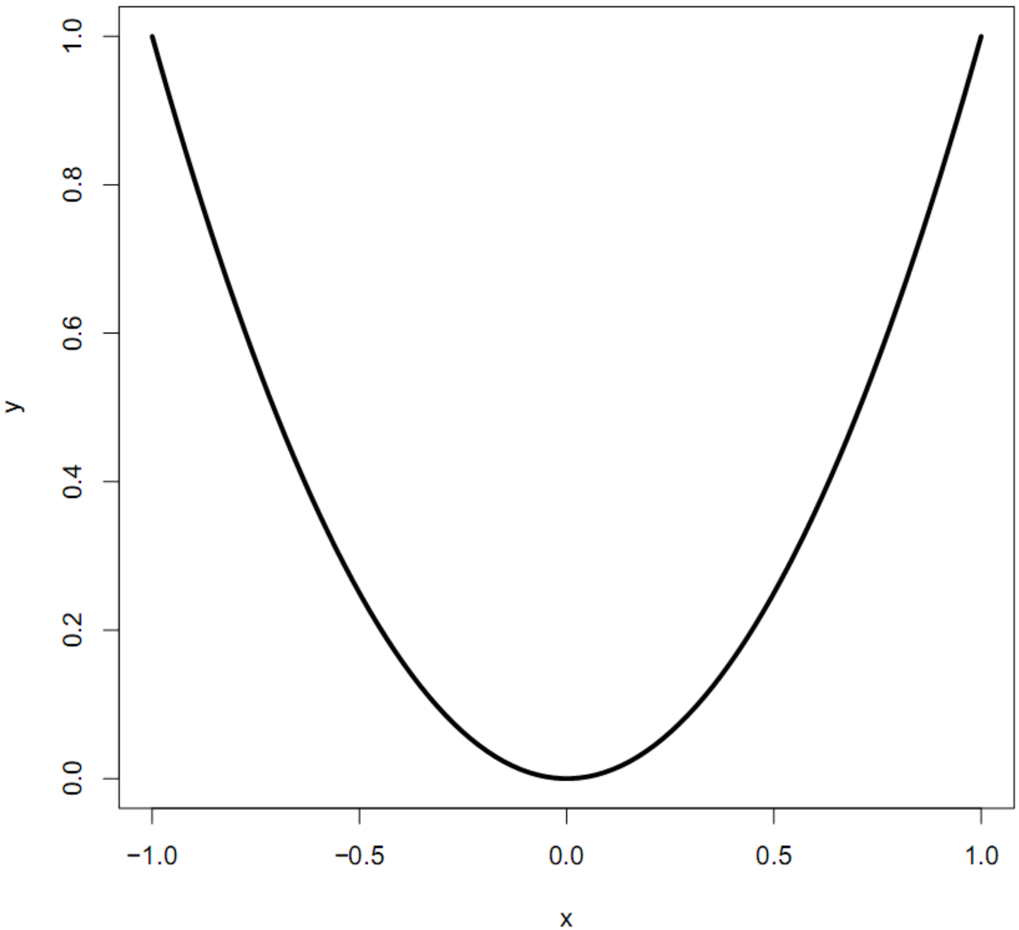
Constrained Optimization

- Unconstrained problems are generally easy to solve
  - Take the derivative and set it to zero
  - Use gradient descent if you can't get an analytical solution
- But what if we want to restrict the values that our parameter(s) can take?

Constrained Optimization

- Why might we want to restrict our parameter space?
  - Might know that the parameter MUST lie within some region (remember the midterm question?)
  - Might have a preference for smaller or larger values for some parameters
  - In some cases, it is crucial to the problem
    - Probabilities must sum to 1
    - SVM margins constraints must be met

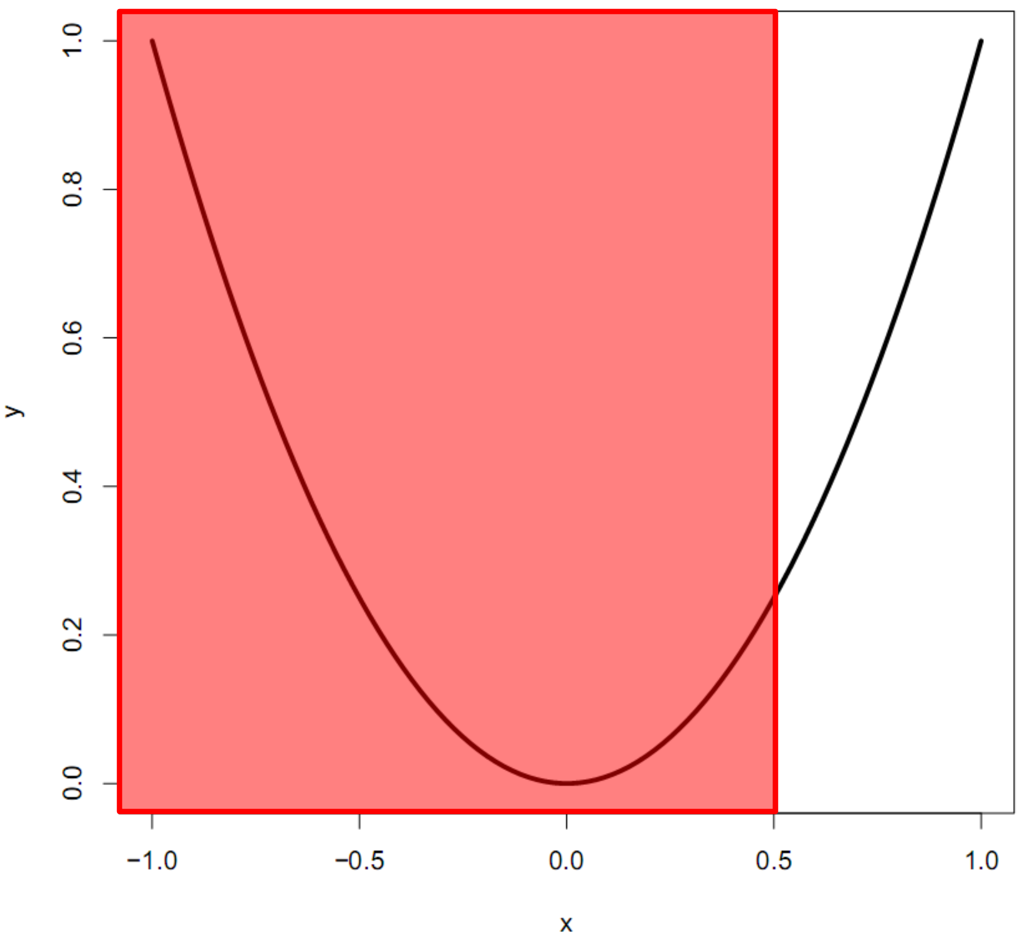
Constrained Optimization



- Start with an unconstrained problem

min x^2

Constrained Optimization

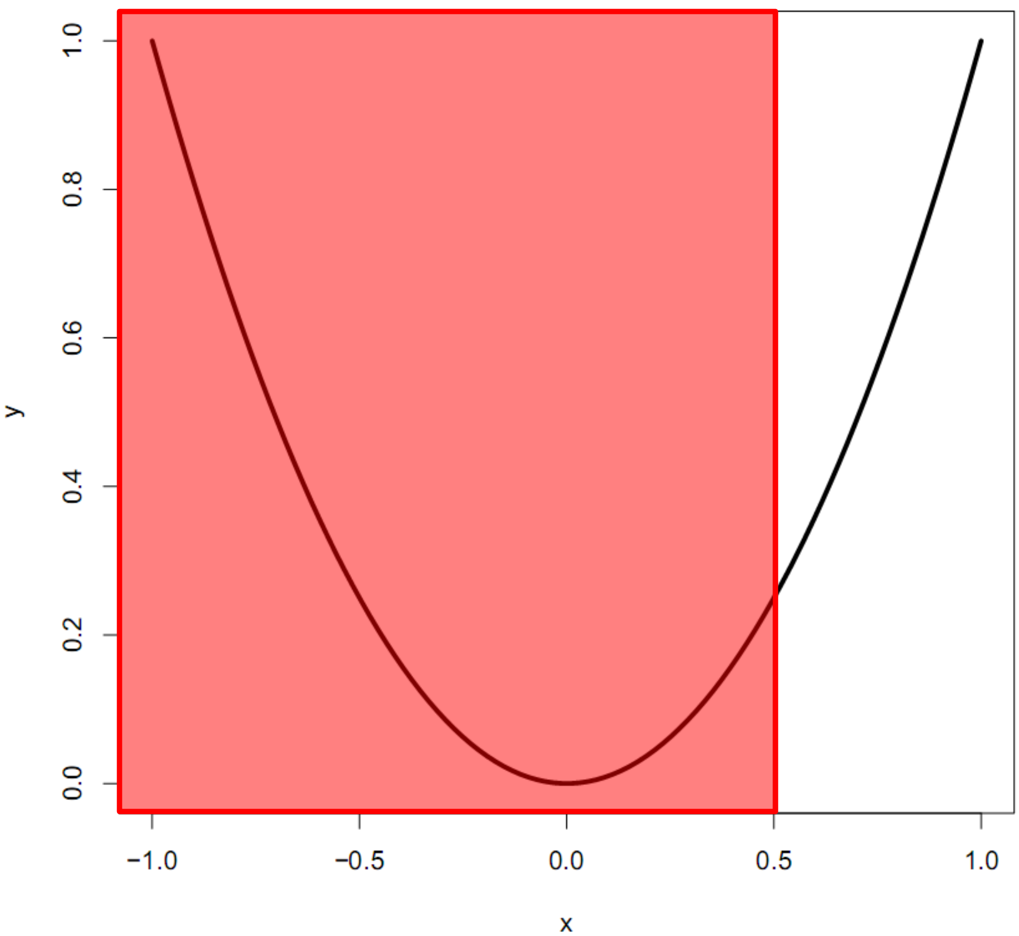


- Add a constraint
- Does it matter?

min x^2

s.t. $x \leq 0.5$

Constrained Optimization

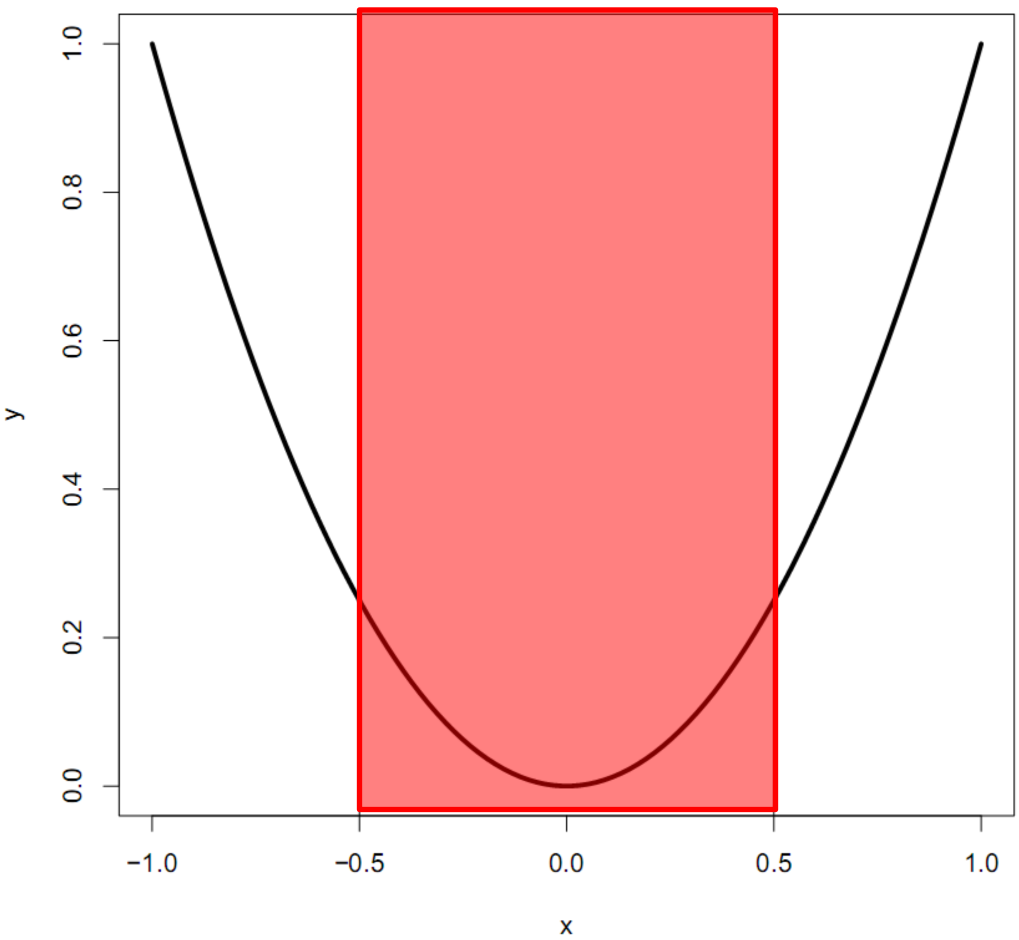


- Add a constraint
- Does it matter?
- No! Not active!

min x^2

s.t. $x \leq 0.5$

Constrained Optimization



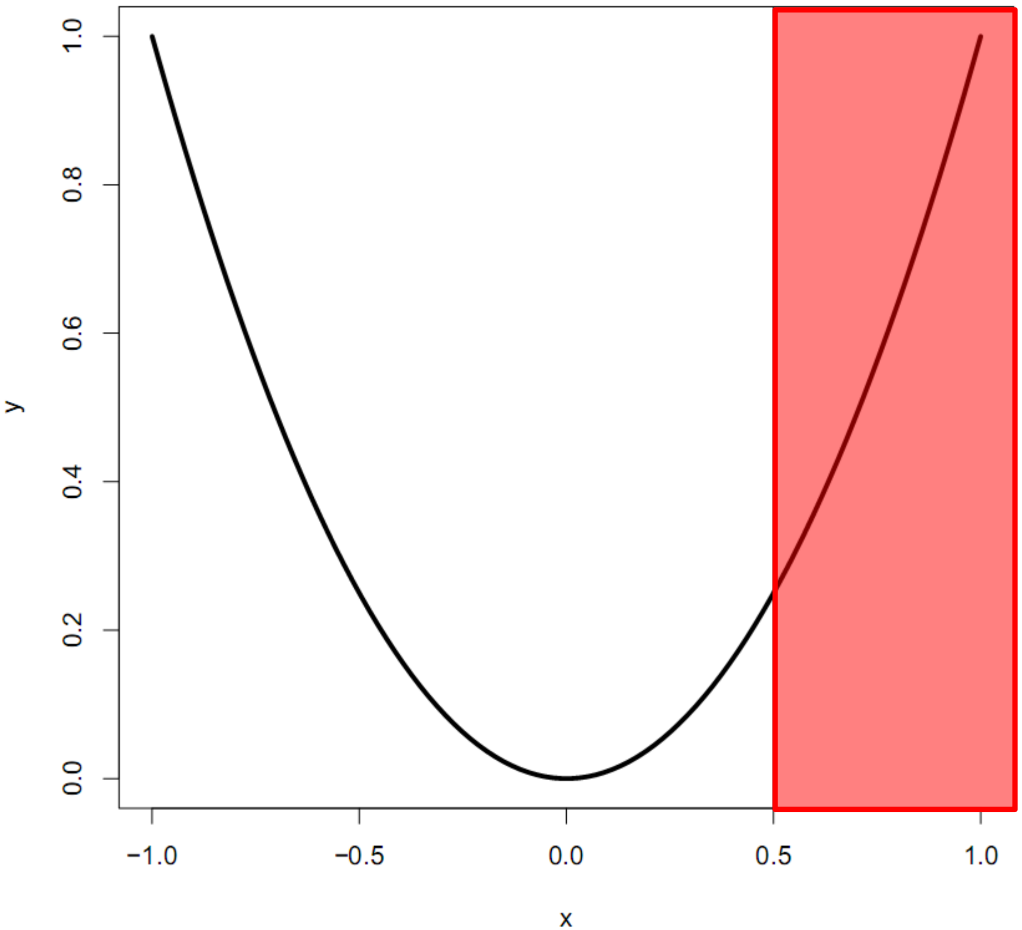
- Add another constraint
- Both are not active

min x^2

s.t. $x \leq 0.5$,

$x \geq -0.5$

Constrained Optimization

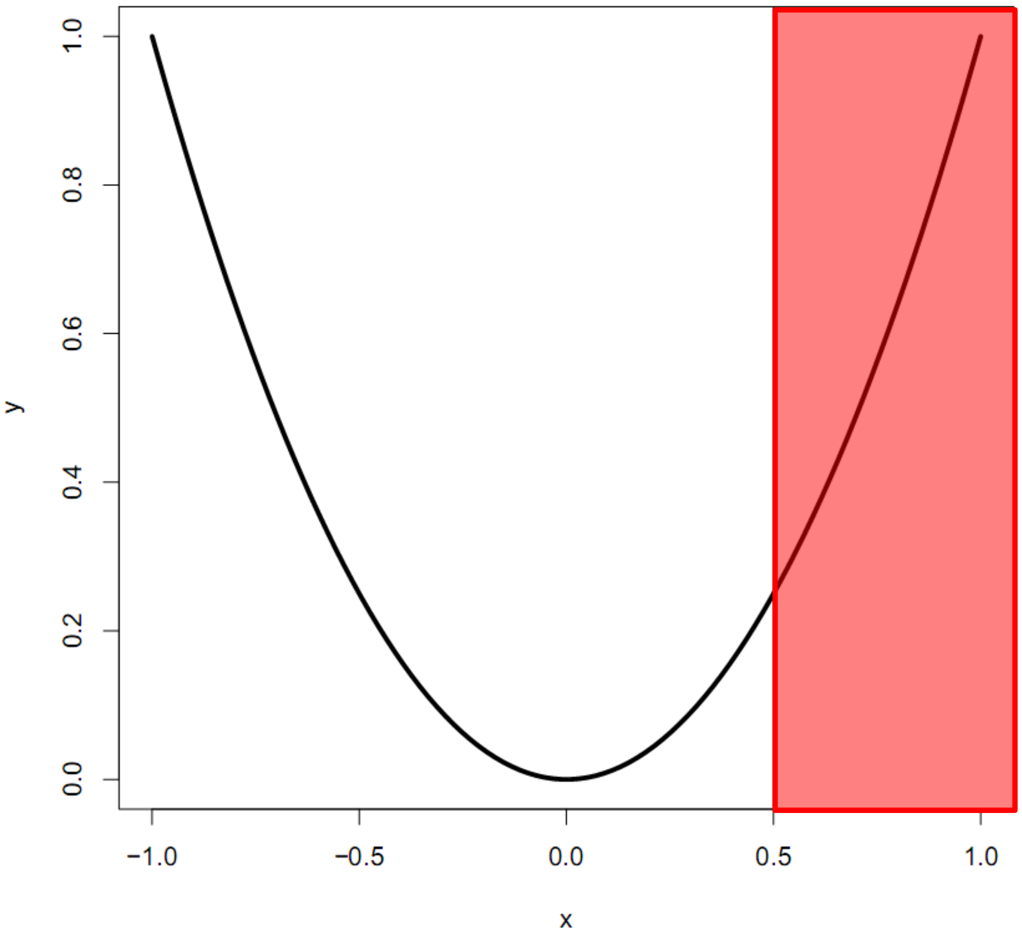


- Try a new constraint
- Does it matter?

min x^2

s.t. $x \geq 0.5$

Constrained Optimization

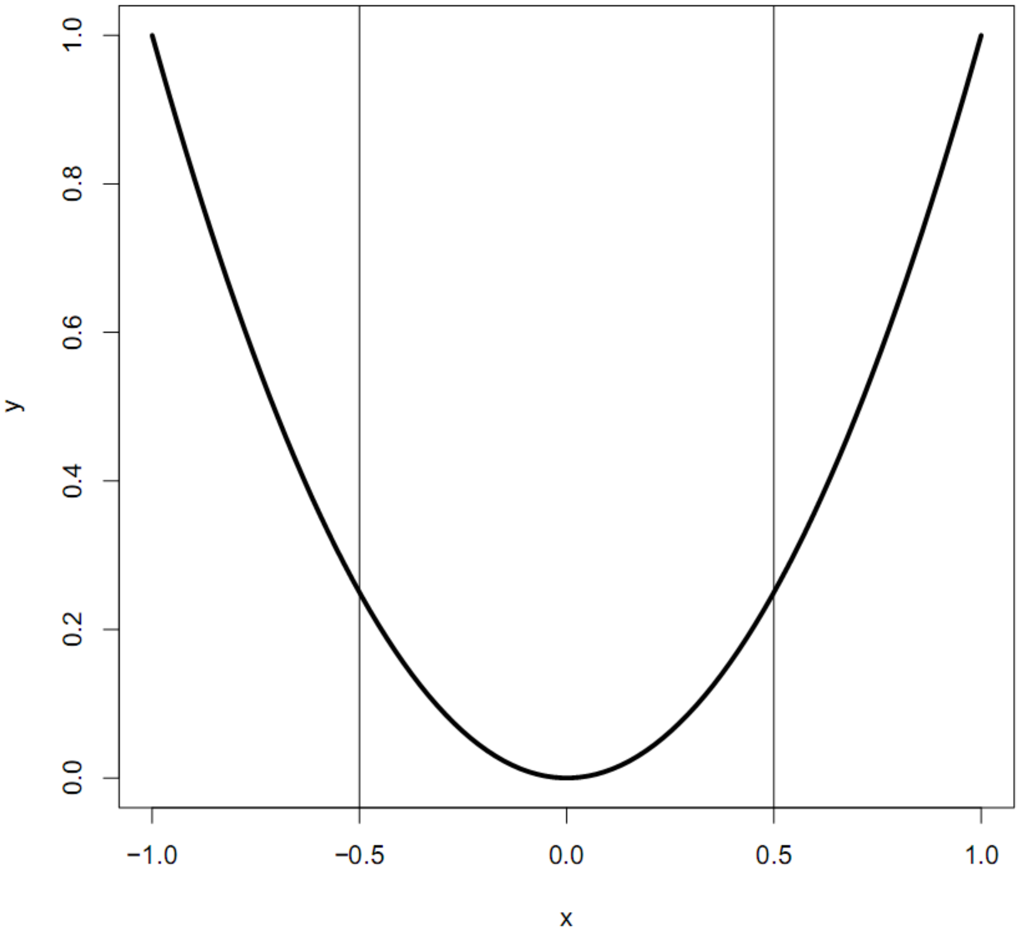


- Add a constraint
- Does it matter?
- Yes!

min x^2

s.t. $x \geq 0.5$

Constrained Optimization



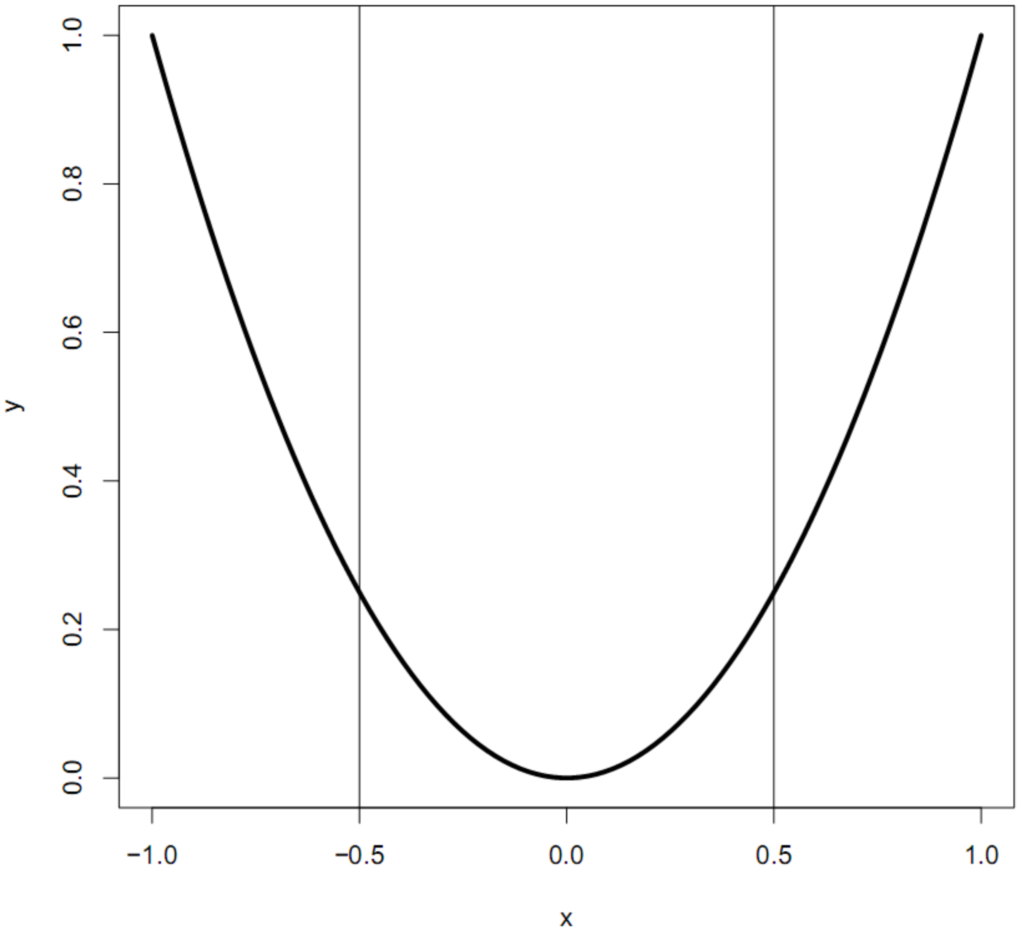
- What about now?

min x^2

s.t. $x \geq 0.5$,

$x \leq -0.5$

Constrained Optimization



- What about now?
- Infeasible!

min x^2

s.t. $x \geq 0.5$,

$x \leq -0.5$

Constrained Optimization

- In the general setting, we have:

min	$f(x)$	(objective)
s.t.	$g_1(x) \geq 0$	(inequality
	$g_2(x) \leq 0$	constraints)
	$h(x) = 0$	(equality
		constraints)

Linear Programs (LP)

$$\begin{array}{ll} \min & f(x) \\ \text{s.t.} & g_1(x) \geq 0 \\ & g_2(x) \leq 0 \\ & h(x) = 0 \end{array}$$

- If f(x), g(x), h(x) all linear, then we have a linear program
- Typically solved using Simplex methods

Quadratic Programs (QP)

$$
\begin{aligned}
\min \quad & f(x) \\
\text{s.t.} \quad & g_1(x) \geq 0 \\
& g_2(x) \leq 0 \\
& h(x) = 0
\end{aligned}
$$

- If f(x) quadratic and g(x) and h(x) linear, then we have a quadratic program
- SVMs are a quadratic program

Lagrangian

- Basic idea:
  - Constraints are hard to deal with
  - But we know how to handle unconstrained problems
  - Change a constrained problem into an unconstrained problem
- Leads to the “dual” formulation of the original “primal” problem
- Dual can be easier to work with

Lagrangian Example

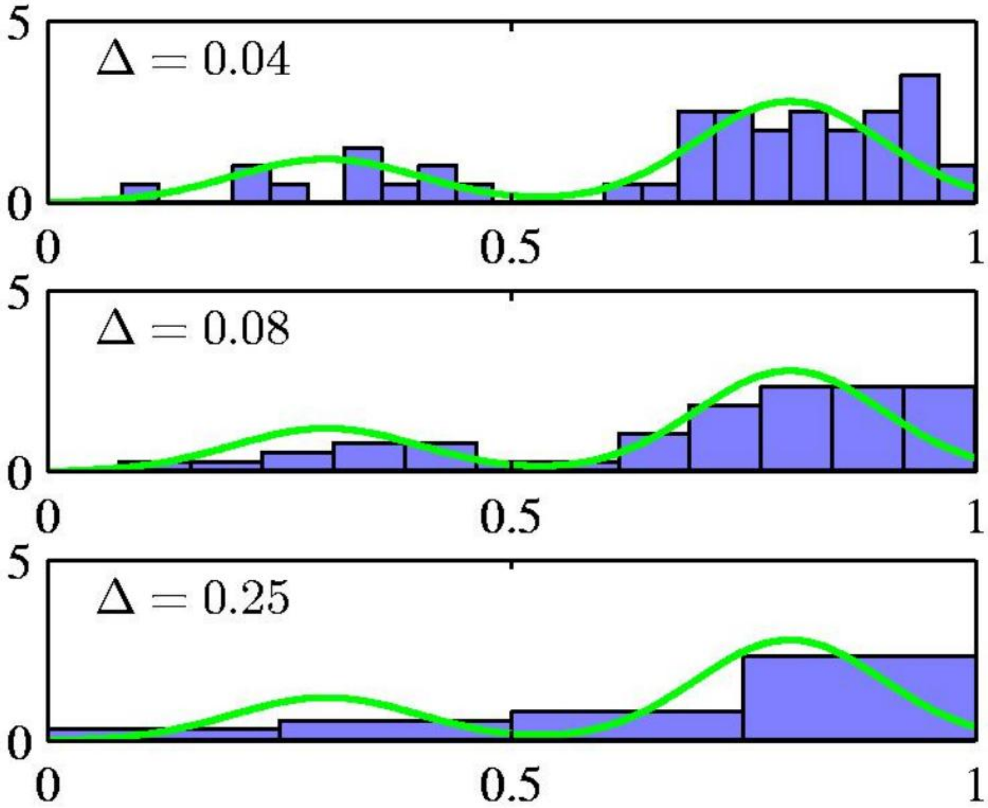


Image src: Bishop book

- Recall the histogram setting in class
- How to estimate the probability of being in a bin?

Lagrangian Example

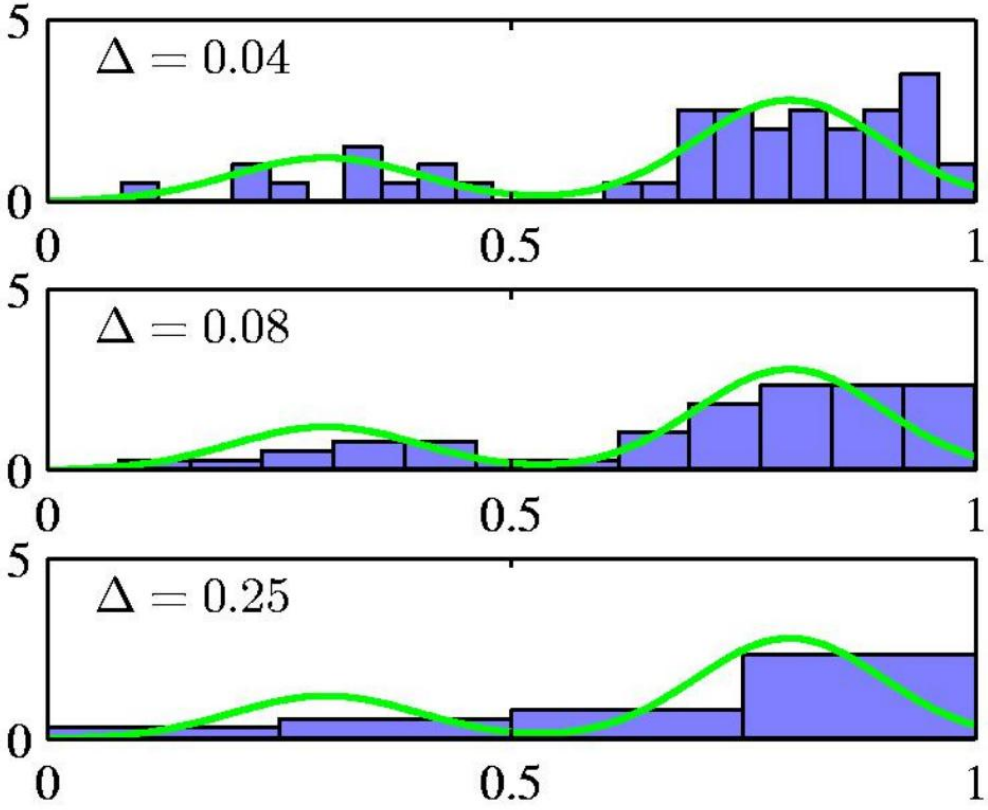


Image src: Bishop book

- Let's say we want the MLE estimator for the probs.
- Say I have *m* bins
- What is the likelihood?

Lagrangian Example

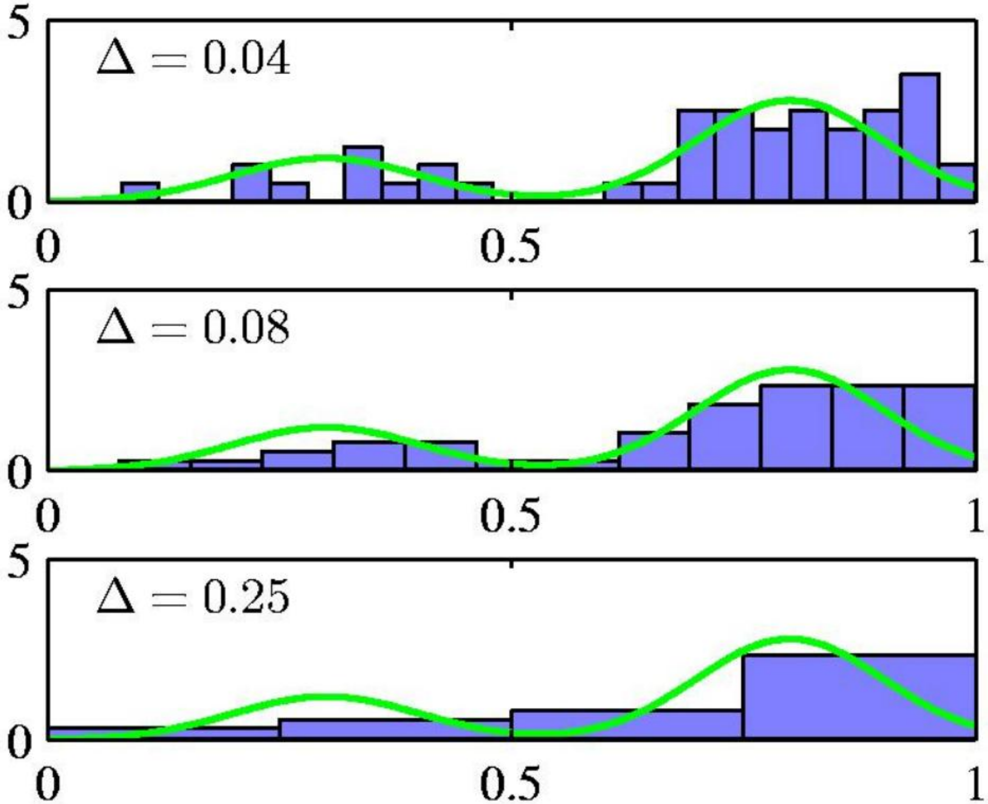


Image src: Bishop book

- Let's say we want the MLE estimator for the probs.
- Say I have *m* bins

$$L = \prod_{j=1}^{m} p_j^{n_j}$$

$$l = \sum_{j=1}^{m} n_j log(p_j)$$

Lagrangian Example

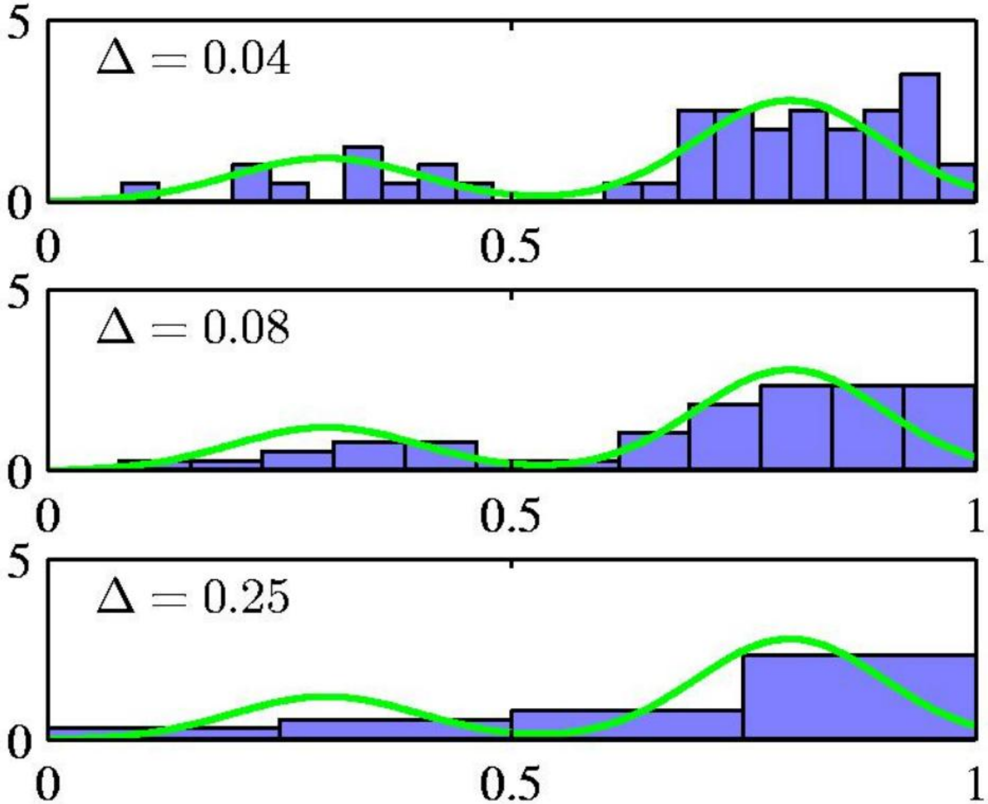


Image src: Bishop book

- Let's say we want the MLE estimator for the probs.
- Say I have *m* bins

$$L = \prod_{j=1}^{m} p_j^{n_j}$$

$$l = \sum_{j=1}^{m} n_j log(p_j)$$

$$\sum_{j=1}^{m} p_j = 1$$

Lagrangian Example

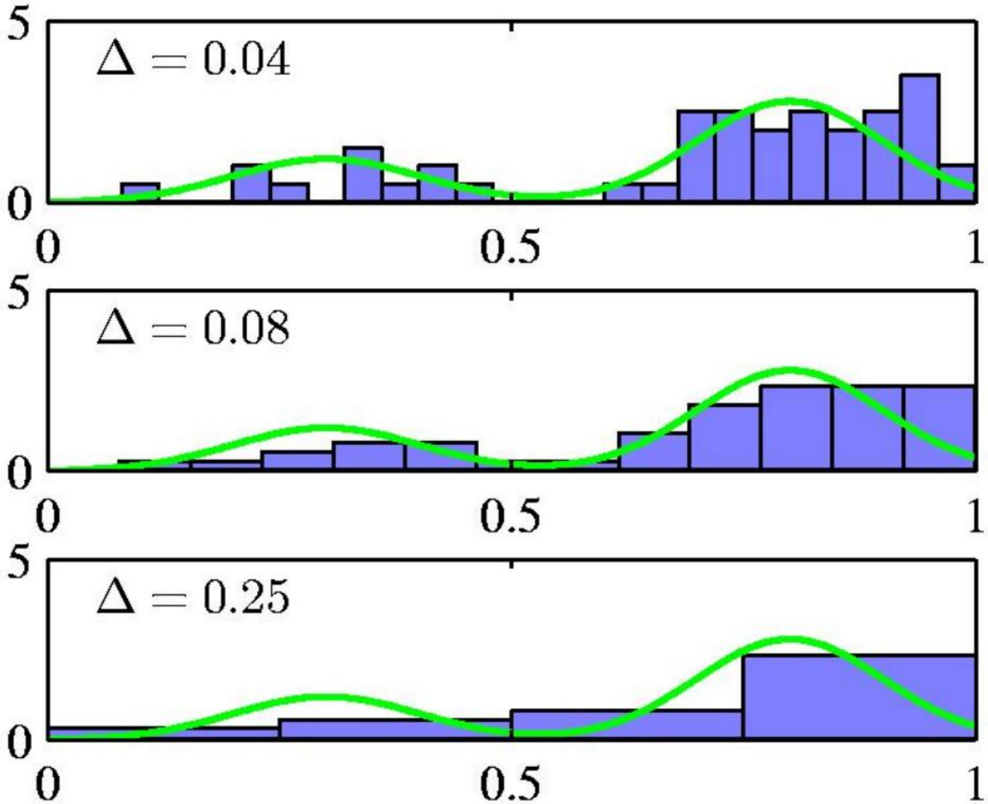


Image src: Bishop book

$$l = \sum_{j=1}^{m} n_j log(p_j)$$

$$\sum_{j=1}^{m} p_j = 1$$

- Exactly a multinomial dist.
- Solve it like the binomial case?

Lagrangian Example

$$l = \sum_{j=1}^{m} n_j log(p_j)$$

$$\sum_{j=1}^{m} p_j = 1$$

- Note that I'm trying to maximize the likelihood
- If I fail to meet the constraint, objective function should be -Inf

Lagrangian Example

$$L(p,\lambda) = \sum_{j=1}^{m} n_j log(p_j) - \lambda(\sum_{j=1}^{m} p_j - 1)$$

$$\frac{\partial L}{\partial \lambda} = \sum_{j=1}^{m} p_j - 1 = 0$$

$$\frac{\partial L}{\partial p_j} = \frac{n_j}{p_j} - \lambda = 0$$

Lagrangian Example

$$L(p, \lambda) = \sum_{j=1}^{m} n_j log(p_j) - \lambda(\sum_{j=1}^{m} p_j - 1)$$

$$\frac{\partial L}{\partial \lambda} = \sum_{j=1}^{m} p_j = 1$$

$$\frac{n_j}{p_j} = \lambda$$

$$\frac{n_j}{p_j} = \frac{N}{1}$$

$$p_j = \frac{n_j}{N}$$

(total of N points, and

all probs must sum to 1)

Duality

- Idea: Instead of solving the optimization problem directly, can we find a bound for the objective value?
- While we're at it, can we find the "best" bound?
- Can show that under some conditions (KKT), the best bound is the value of the objective function at the optimum

Duality

$$\min \quad x + y$$
$$\text{s.t.} \quad x + y \geq 1$$
$$x, y \geq 0$$

- Best bound is 1 (just take 1^{st} constraint)

Slides inspired by Geoff Gordon's 10-725 slides

Duality

$$\begin{aligned} \min \quad & x + 3y \\ \text{s.t.} \quad & x + y \geq 1 \\ & x, y \geq 0 \end{aligned}$$

- Best bound is 1 (just take 1st constraint + 2 times $y \geq 0$ constraint)

Slides inspired by Geoff Gordon's 10-725 slides

Duality

$$\begin{aligned} \min \quad & px + qy \\ \text{s.t.} \quad & x + y \geq 1 \\ & x, y \geq 0 \end{aligned}$$

- Want to take a times 1st constraint, b times 2nd constraint, c times 3rd constraint and have them add up to the objective function

Slides inspired by Geoff Gordon's 10-725 slides

Duality

$$\min \; px + qy$$
$$\text{s.t.} \; x + y \geq 1$$
$$x, y \geq 0$$

$$a(x+y-1) + b(x) + c(y) \geq 0$$

$$\Rightarrow \; (a+b)x + (a+c)y \geq a$$

Slides inspired by Geoff Gordon's 10-725 slides

Duality

$$\min\ px + qy \qquad\qquad \max\ a$$

$$\text{s.t.}\ \ x + y \geq 1 \qquad\qquad \text{s.t.}\ \ a + b = p$$

$$x, y \geq 0 \qquad\qquad a + c = q$$

Primal form on the left

Dual form on the right

Slides inspired by Geoff Gordon's 10-725 slides

Duality

- We'll see the dual applied to a QP when we talk about SVMs
- Dual form in SVMs opens allows us to use the "kernel trick"
- Kernels allow us to automatically "project" the data into a space where classification is easier