# Regularization
# Ridge Regression and LASSO

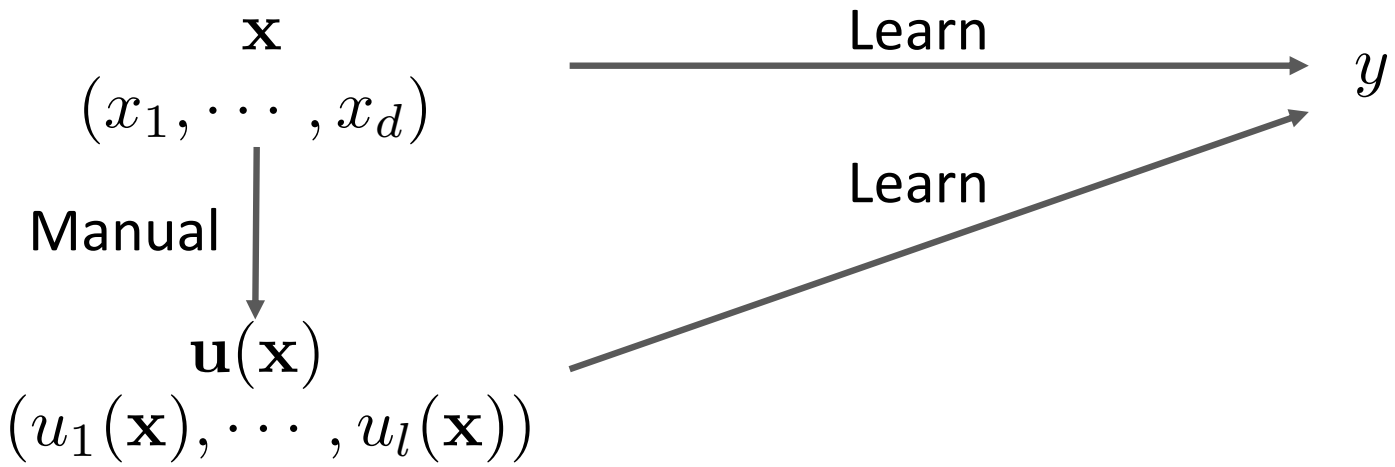
CSE512 – Machine Learning, Spring 2018, Stony Brook University

Instructor: Minh Hoai Nguyen (minhhoai@cs.stonybrook.edu)

Date: 5 Feb 2018

Many slides are by Carlos Guestrin @ University of Washington

# Review of Model Complexity

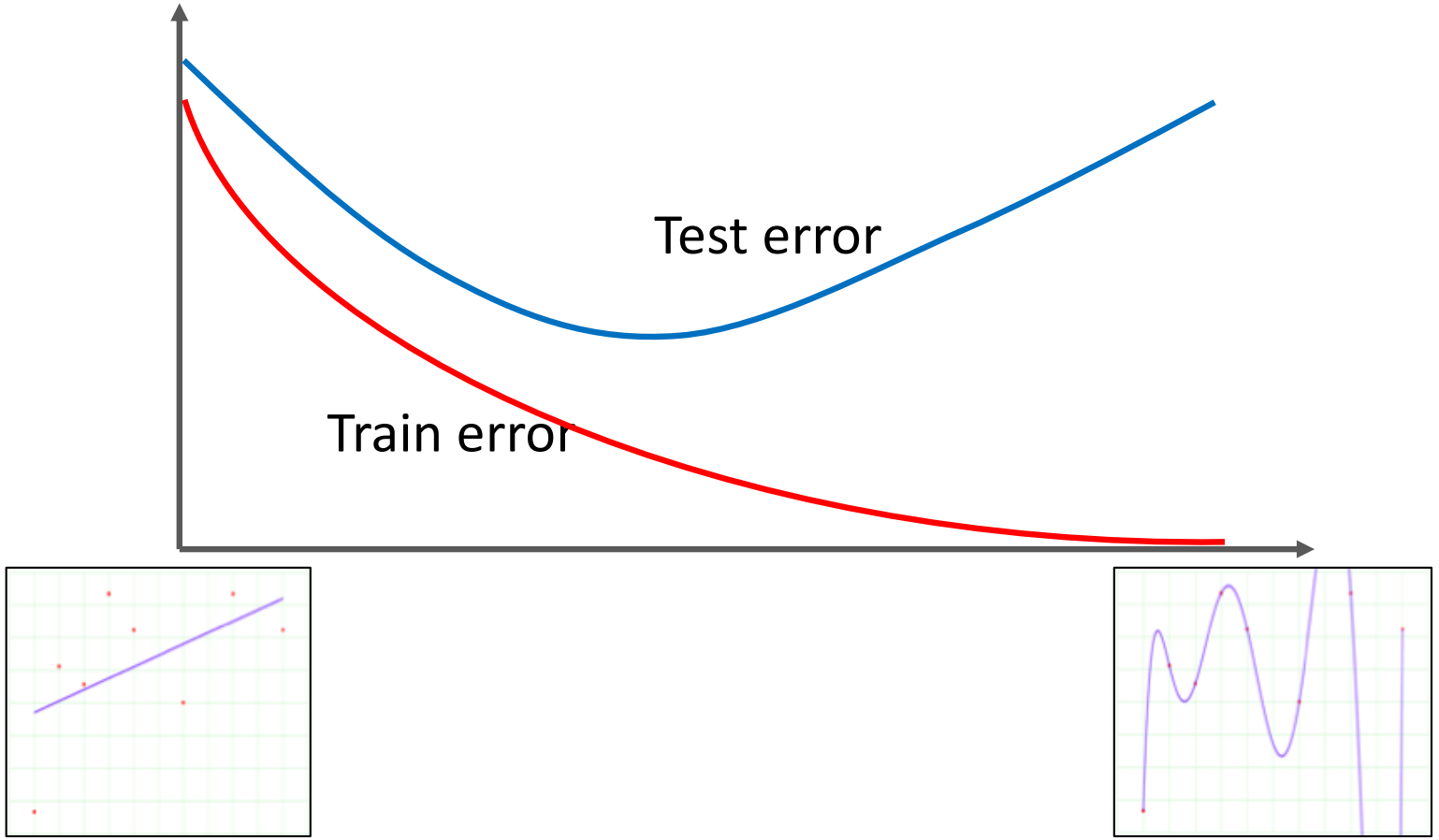



Examples

$$x \to (x, x^2, x^3)$$

$$(x_1, x_2) \to (x_1^2, x_2^2, x_1 x_2, x_1, x_2)$$

# Model Complexity and Errors



# What if Data is Complex to Start with?

$\mathbf{x}$

$(x_1, \cdots, x_d)$

Very complex and high dimension

Learn

$y$

$\mathbf{u}(\mathbf{x})$

$(u_1(\mathbf{x}), \cdots, u_l(\mathbf{x}))$

May even be more complex

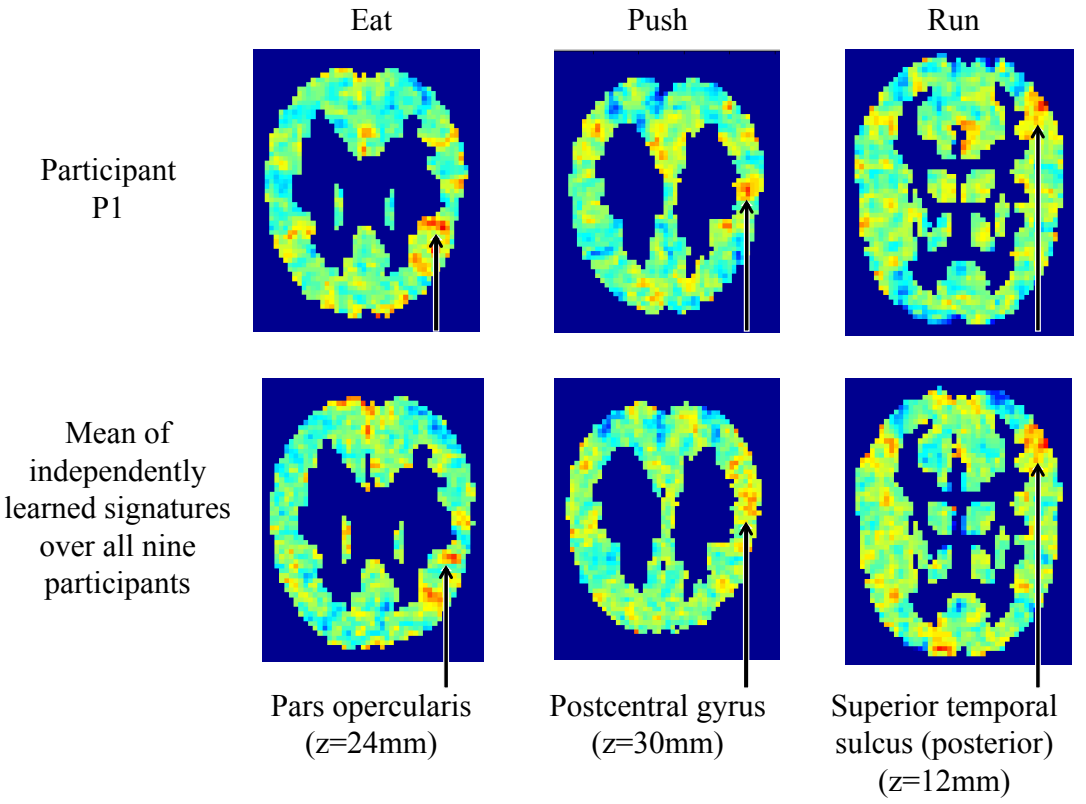



Figure from Tom Mitchell

# Regularization on Parameters

- Aim to impose a “complexity” penalty by penalizing large weights

# Ridge Regression

Linear Regression Objective

$$\sum_{i=1}^{n}(y^i - (\sum_{j=1}^{d}\theta_j x_j^i + \theta_{d+1}))^2$$

Ridge Regression Objective

$$\sum_{i=1}^{n}(y^i - (\sum_{j=1}^{d}\theta_j x_j^i + \theta_{d+1}))^2 + \lambda\sum_{j=1}^{d}\theta_j^2$$

- Encourage coefficients to be small
- Tradeoff between error train and magnitude of coefficients
- Don’t penalize the bias

# Ridge Regression in Matrix Notation

# Ridge Regression Solution

# Ridge Regression: Effect of Regularization

- Solution is indexed by the regularization parameter $\lambda$
- Larger $\lambda$
- Smaller $\lambda$
- As $\lambda \rightarrow 0$
- As $\lambda \rightarrow \infty$

# Ridge Coefficients as a function of λ

Prostate Cancer Dataset

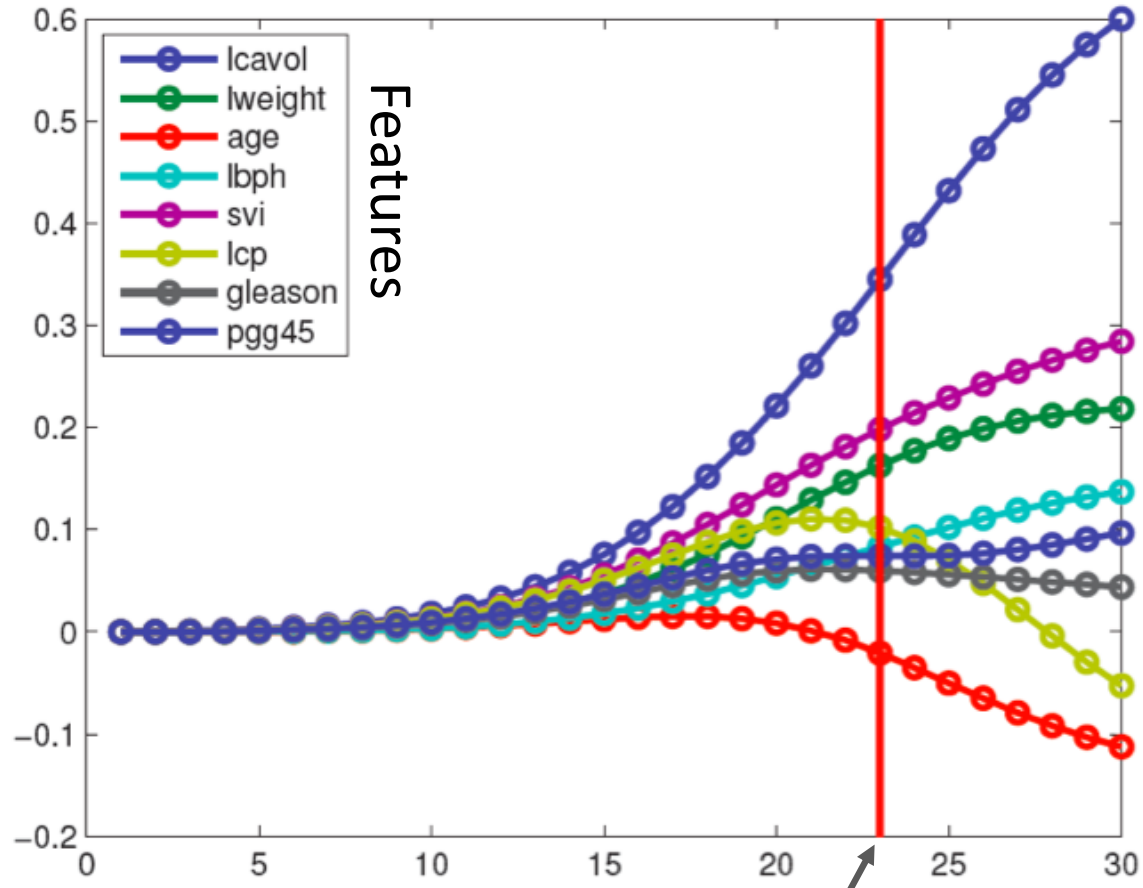


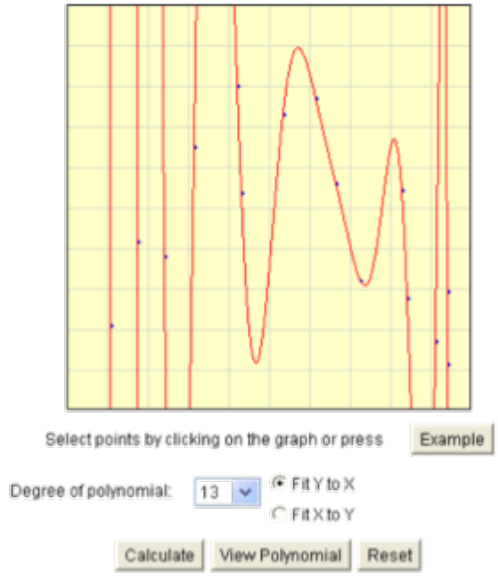

Fig 13.7
Kevin Murphy
textbook

Large λ

Small λ

Typical approach: select λ using cross validation

# Error as a function of regularization parameter for a fixed model complexity




$\lambda=\infty$ $\lambda=0$

# Run-time Complexity

$$\mathbf{x} \quad \xrightarrow{\text{Learn}} \quad y$$
$$(x_1, \cdots, x_d)$$

$$\hat{y} = \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_d x_d + \theta_{d+1}$$

- Using Ridge regression many coefficients will be small
- However, small is different from 0
- It might be expensive for classification if d is large

# Sparsity

- Vector **w** is sparse, if many entries are zero:

- Very is desirable in many cases:
  - **Efficiency**: If size(**w**) = 100B, each prediction is expensive:
    - If part of an online system, too slow
    - If **w** is sparse, prediction computa
  - **Interpretability**: What are the relevant dimension to make a prediction?
    - E.g., what are the parts of the brain associated with particular words?

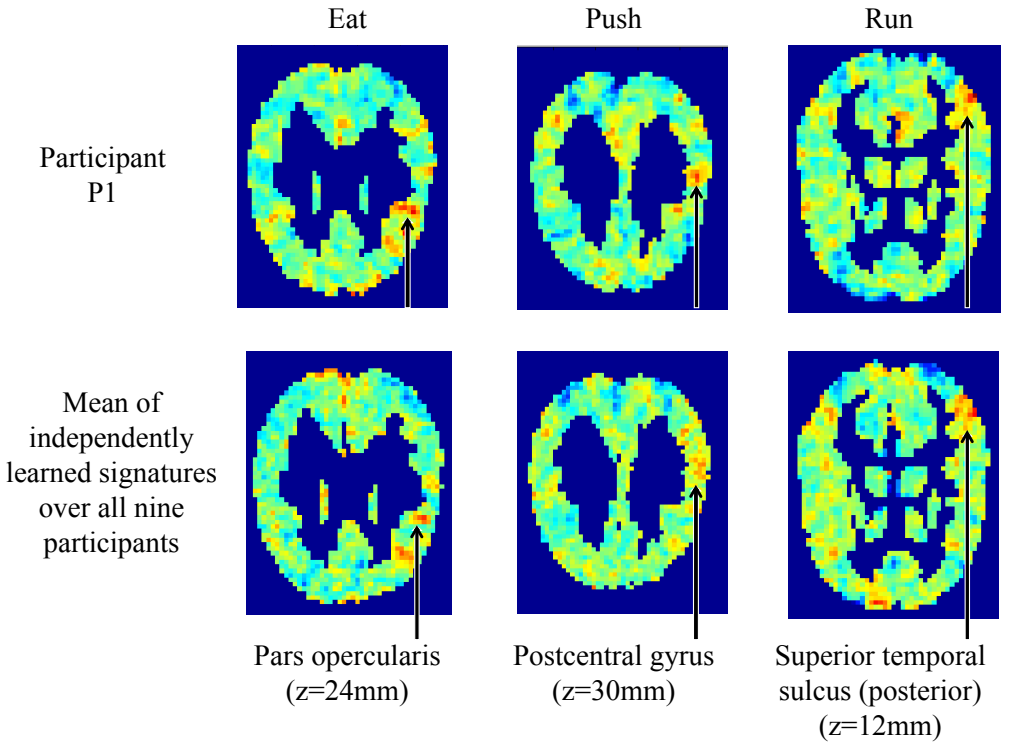



Figure from Tom Mitchell

# Feature Selection Problem

Least Square Linear Regression:

$$\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} (y^i - (\sum_{j=1}^{d} \theta_j x_j^i + \theta_{d+1}))^2$$

Feature Selection Problem for Sparsity:

$$\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} (y^i - (\sum_{j=1}^{d} \theta_j x_j^i + \theta_{d+1}))^2$$

$$\text{Subject to } \sum_{j=1}^{d} \delta(\theta_j \neq 0) \leq k$$

But this function is combinatorial and very difficult to optimize!

# Machine Learning and Optimization

- Go hand-in-hand
- Machine Learning
  - Function Approximation
  - Optimize a loss function
- Loss function in Machine Learning
  - What we want/need to optimize might be difficult to optimize
  - Approximate optimization algorithm
  - Change the loss function!

# Machine Learning and Optimization

- Message from Machine Learning to Optimization:
  - I have this loss function to optimize. Help!
- Message from Optimization to Machine Learning:
  - Here are the list of things I can optimize, don't come up some nonsense that I cannot do.
- Many Machine Learning Papers:
  - A new optimization algorithm for an old loss function
  - A new loss function that can be easily optimized

# LASSO Regression

- LASSO: Least Absolute Shrinkage and Selection Operator
- New Objective

$$\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} (y^i - (\sum_{j=1}^{d} \theta_j x_j^i + \theta_{d+1}))^2 + \lambda \sum_{j=1}^{d} |\theta_j|$$

- LASSO Regression leads to a sparse solution!!!
- This simple approach has changed statistics, machine learning, and electrical engineering

# LASSO Equivalence

LASSO objective

$$\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} (y^i - (\sum_{j=1}^{d} \theta_j x_j^i + \theta_{d+1}))^2 + \lambda \sum_{j=1}^{d} |\theta_j|$$

Equivalent

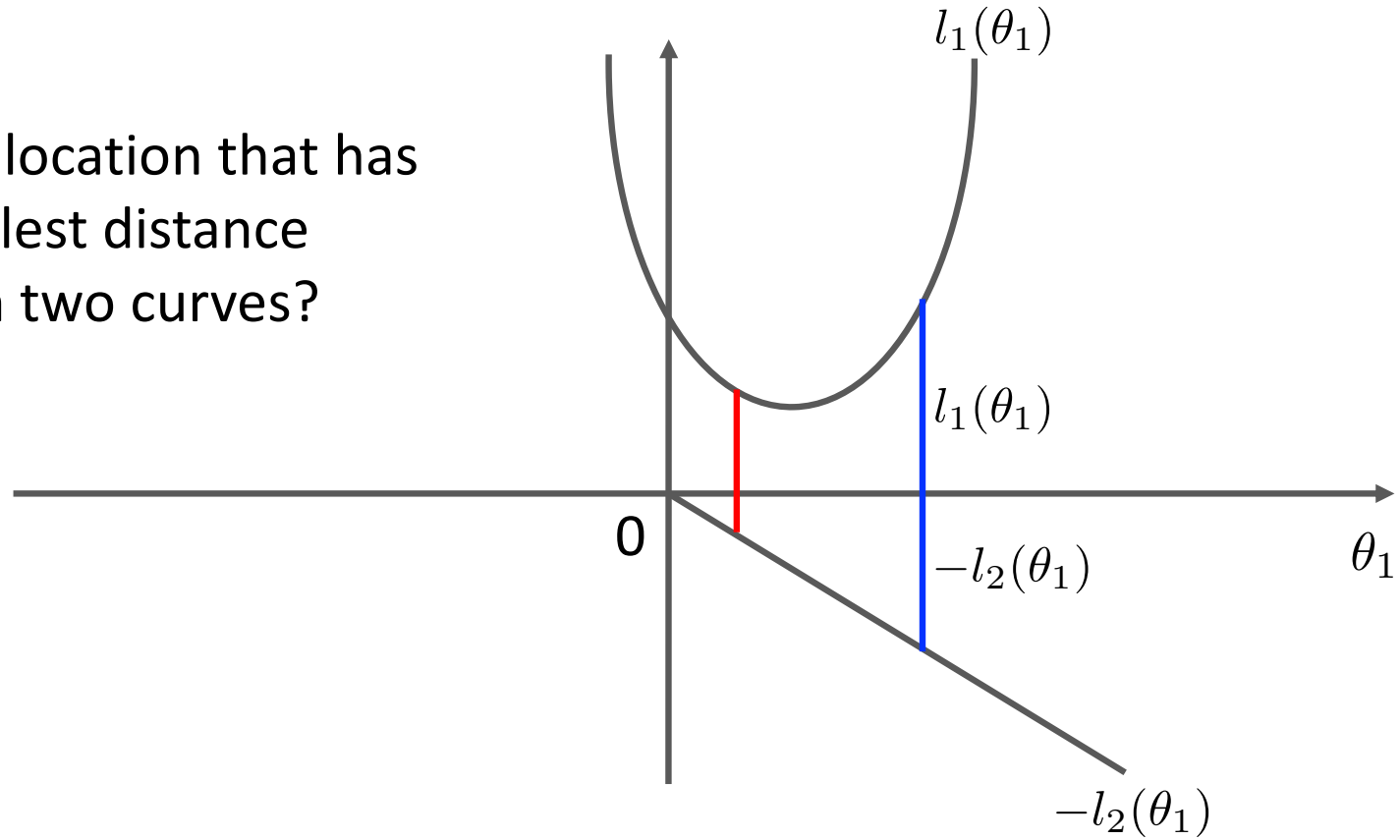
$$\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} (y^i - (\sum_{j=1}^{d} \theta_j x_j^i + \theta_{d+1}))^2$$

$$\text{s.t.} \sum_{j=1}^{d} |\theta_j| \leq \lambda'$$

# Geometric Intuition for 1D

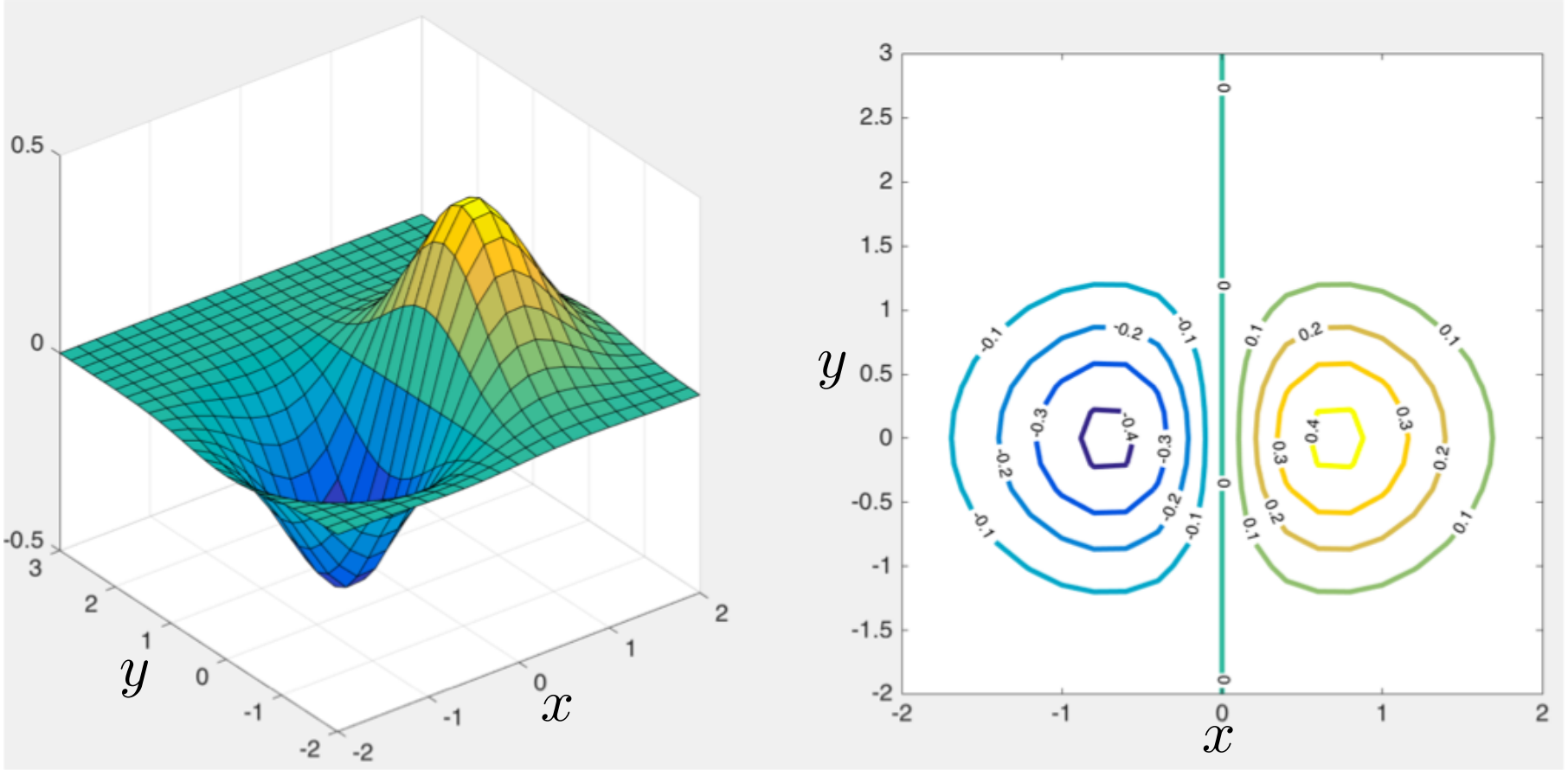
$$\min_{\boldsymbol{\theta}} \underbrace{\sum_{i=1}^{n}(y^i - (\sum_{j=1}^{d}\theta_j x_j^i + \theta_{d+1}))^2}_{l_1(\theta_1)} + \underbrace{\lambda \sum_{j=1}^{d}|\theta_j|}_{l_2(\theta_1)}$$

Find the location that has the smallest distance between two curves?

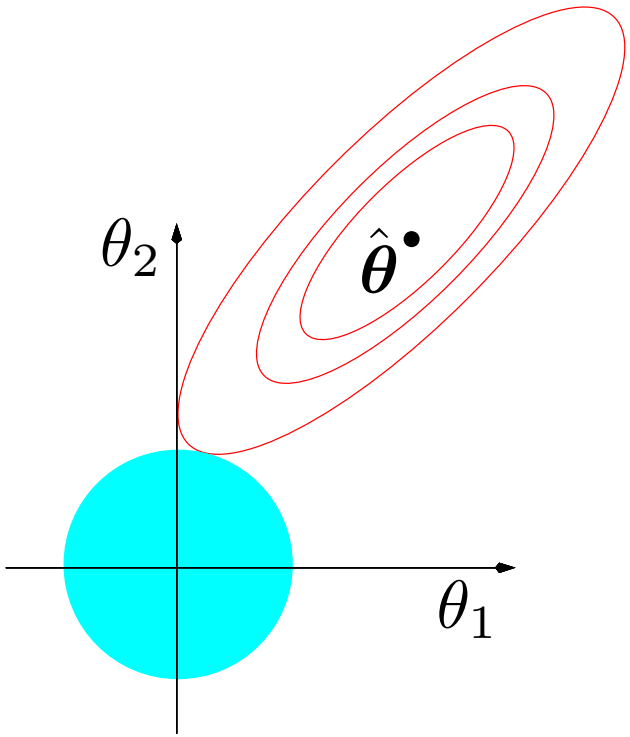


# Function Contour

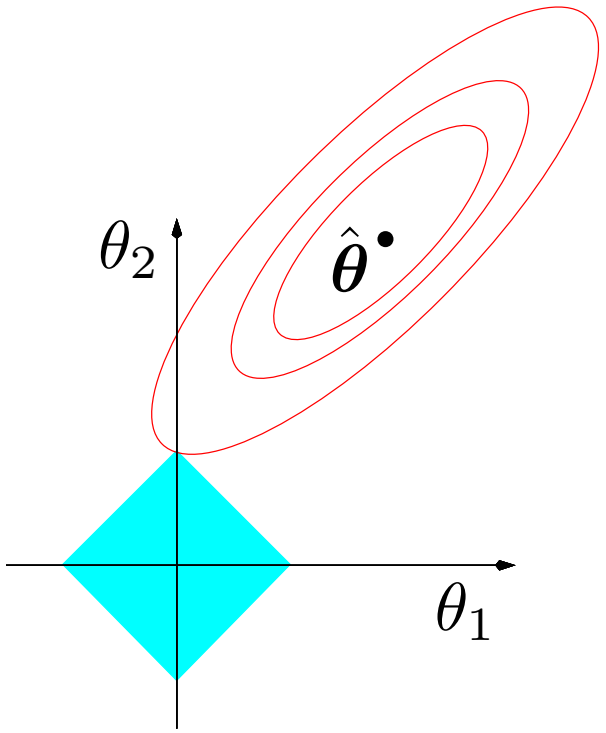
$$f(x, y) = x * e^{-x^2 - y^2}$$



# Geometric Intuition for Sparsity




Ridge Regression

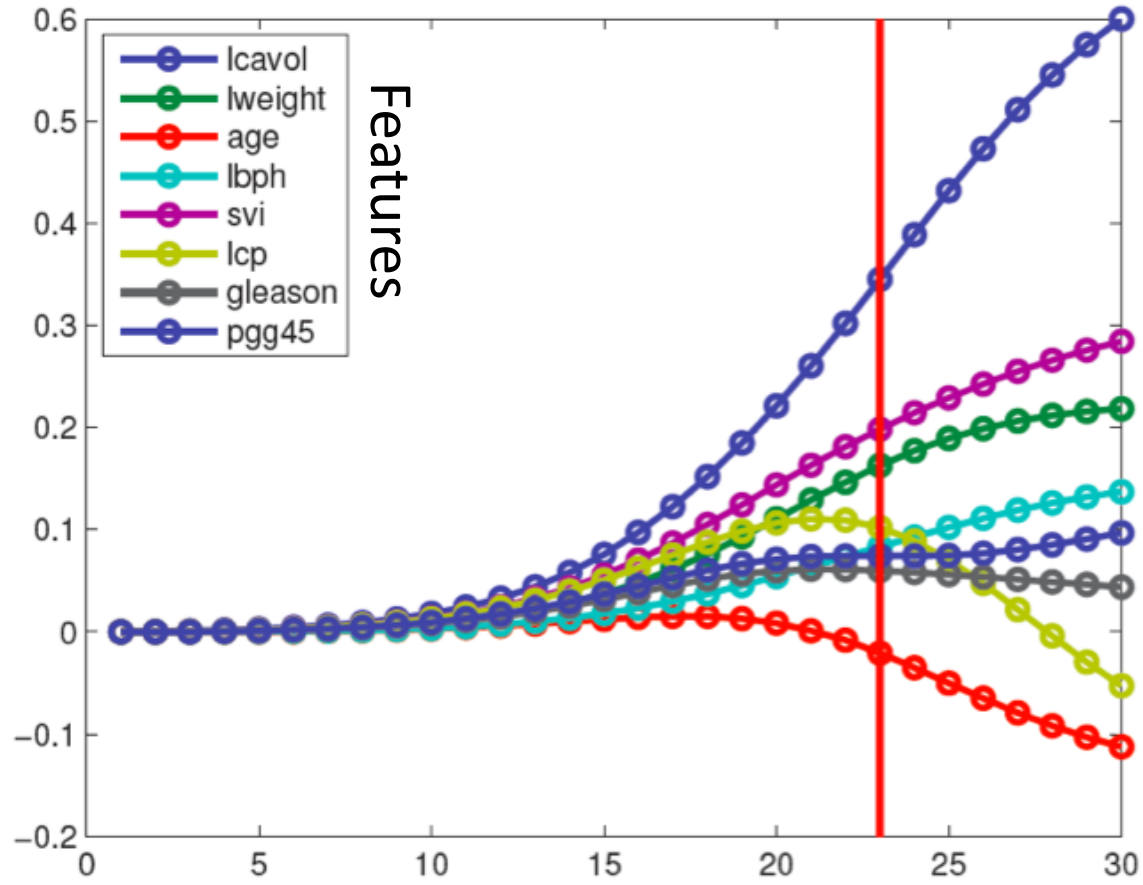



Lasso

From
Rob Tibshirani
slides

# Recall: Ridge Coefficient Path

Prostate Cancer Dataset




Large λ

Small λ

Fig 13.7
Kevin Murphy
textbook

# LASSO Coefficient Path

Prostate Cancer Dataset

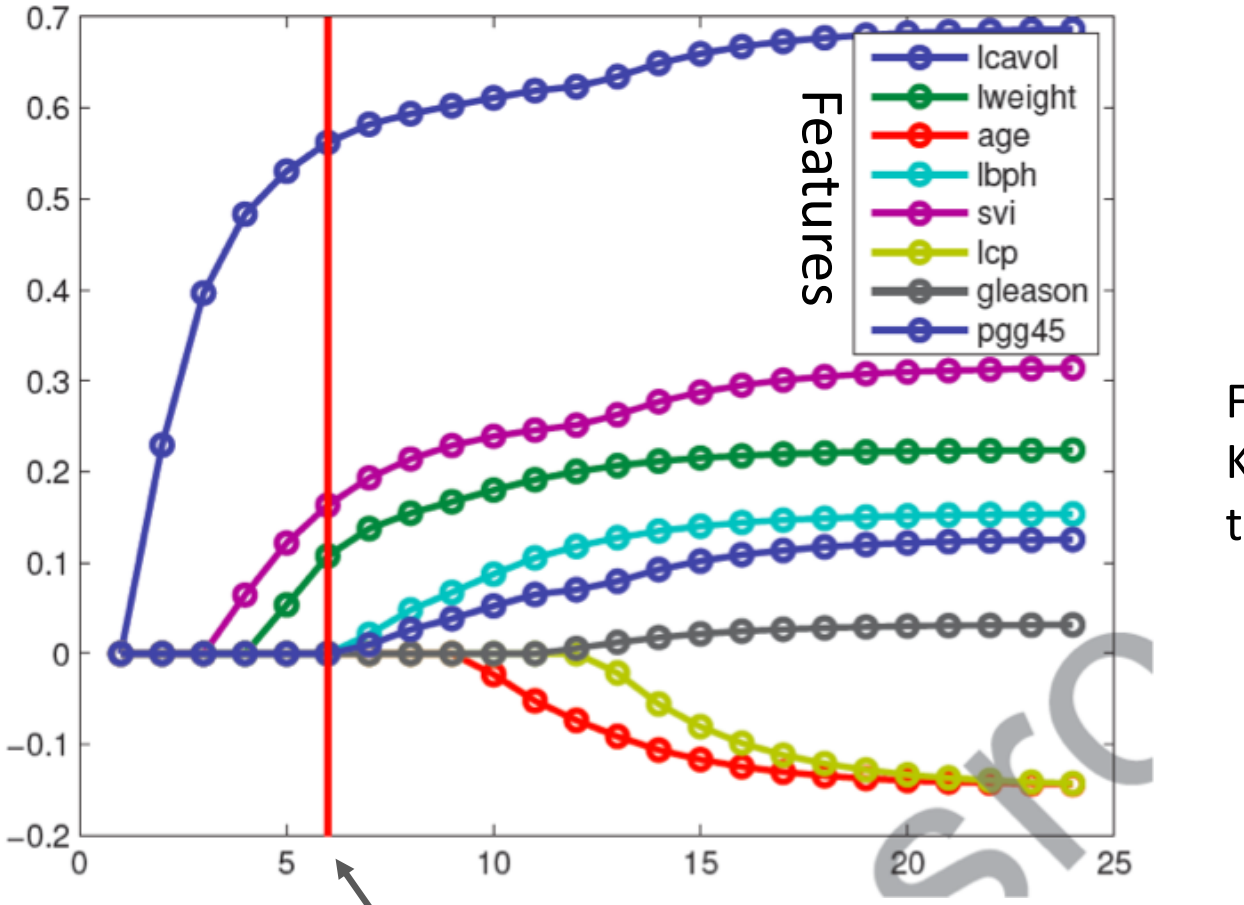



Fig 13.7
Kevin Murphy
textbook

# LASSO Example

| Term | Least Squares | Ridge | Lasso |
|---|---|---|---|
| Intercept | 2.465 | 2.452 | 2.468 |
| lcavol | 0.680 | 0.420 | 0.533 |
| lweight | 0.263 | 0.238 | 0.169 |
| age | −0.141 | −0.046 | |
| lbph | 0.210 | 0.162 | 0.002 |
| svi | 0.305 | 0.227 | 0.094 |
| lcp | −0.288 | 0.000 | |
| gleason | −0.021 | 0.040 | |
| pgg45 | 0.267 | 0.133 | |

From Rob Tibshirani slides

# Optimizing the LASSO Objective

LASSO objective: $\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} (y^i - (\sum_{j=1}^{d} \theta_j x_j^i + \theta_{d+1}))^2 + \lambda \sum_{j=1}^{d} |\theta_j|$

Simple approach: take the derivative and set to 0???

But, the function is not differentiable

Derivative of $|\theta_i|$

Even when it is differentiable, there is no closed-form solution

# Coordinate Descent

- Given a function f, find minimum wrt to a set of parameters

- Often hard to find minimum for all coordinates, but easy for one coordinate

# Coordinate Descent

- Optimize One coordinate at a time
- How to pick the next coordinate
- Very useful for many problems
  - Often converges to a local minimum
  - Converges to the global minimum in some cases

# Taking the Derivative

$$\min_{\boldsymbol{\theta}} \underbrace{\sum_{i=1}^{n}(y^i - (\sum_{j=1}^{d}\theta_j x_j^i + \theta_{d+1}))^2}_{E(\boldsymbol{\theta})} + \underbrace{\lambda \sum_{j=1}^{d}|\theta_j|}_{R(\boldsymbol{\theta})}$$

$$\frac{\partial E(\boldsymbol{\theta})}{\partial \theta_l} =$$

Rewrite in this form $\frac{\partial E(\boldsymbol{\theta})}{\partial \theta_l} = a_l\theta_l - c_l$

# Gradient of Sum of Squared Residuals

$$\frac{\partial E(\boldsymbol{\theta})}{\partial \theta_l} = a_l \theta_l - c_l$$

$a_l =$

$c_l =$

# Gradient of Regularization Term

$$\min_{\boldsymbol{\theta}} \underbrace{\sum_{i=1}^{n}(y^i - (\sum_{j=1}^{d}\theta_j x_j^i + \theta_{d+1}))^2}_{E(\boldsymbol{\theta})} + \underbrace{\lambda \sum_{j=1}^{d}|\theta_j|}_{R(\boldsymbol{\theta})}$$

$$\frac{\partial R(\boldsymbol{\theta})}{\partial \theta_l} =$$

# Subgradient of Convex Function

- Gradients lower bound convex functions

- Gradients are unique at $\theta$ iff function differentiable at $\theta$
- Subgradients: generalize gradients to non-differentiable points:
  - Any plane that lowers bounds the function

# Subgradient of Regularization Term

$$\min_{\boldsymbol{\theta}} \underbrace{\sum_{i=1}^{n}(y^i - (\sum_{j=1}^{d}\theta_j x_j^i + \theta_{d+1}))^2}_{E(\boldsymbol{\theta})} + \underbrace{\lambda \sum_{j=1}^{d}|\theta_j|}_{R(\boldsymbol{\theta})}$$

$$\frac{\partial R(\boldsymbol{\theta})}{\partial \theta_l} =$$

# Subgradient of the entire loss function

$$\partial_{\theta_l} L(\boldsymbol{\theta}) = \begin{cases} a_l\theta_l - c_l - \lambda & \text{if } \theta_l < 0 \\ [-c_l - \lambda, -c_l + \lambda] & \text{if } \theta_l = 0 \\ a_l\theta_l - c_l + \lambda & \text{if } \theta_l > 0 \end{cases}$$

$$a_l = 2\sum_{i=1}^{n}(x_l^i)^2 \qquad c_l = 2\sum_{i=1}^{n} x_l^i(y^i - (\sum_{j\neq l}\theta_j x_j^i + \theta_{d+1}))^2$$

Conditions for optimality:

The gradient is 0 at the differentiable point

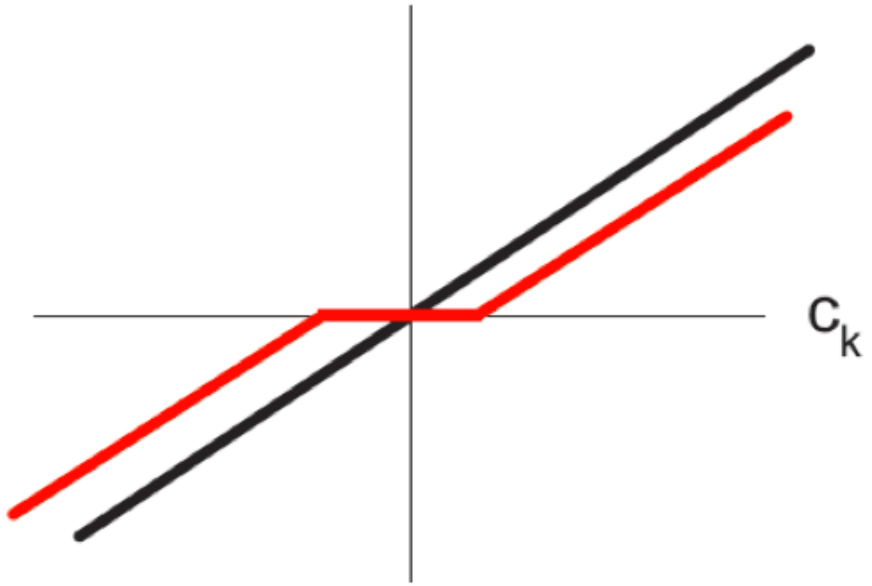
The subgradient contains 0 at a non-differentiable point

# Setting the Subgradient to 0

Coordinate descent optimization rule:

$$\hat{\theta}_l = \begin{cases} (c_l + \lambda)/a_l & \text{if } c_l < -\lambda \\ 0 & \text{if } c_l \in [-\lambda, \lambda] \\ (c_l - \lambda)/a_l & \text{if } c_l > \lambda \end{cases}$$

This is soft thresholding:




From
Kevin Murphy
textbook

# Coordinate Descent for LASSO (aka Shooting Algorithm)

Repeat until convergence:

- Pick a coordinate (random or sequentially)
- Update:

- Where

$$a_l = 2\sum_{i=1}^{n}(x_l^i)^2 \qquad c_l = 2\sum_{i=1}^{n} x_l^i(y^i - (\sum_{j\neq l}\theta_j x_j^i + \theta_{d+1}))^2$$

- For convergence rate: see Shalev-Shwartz and Tewari 2009

# Regularization and Bayesian View

| Obj. Func | Regularization | Likelihood | Prior | Name | Murphy's book |
|---|---|---|---|---|---|
| L2 | No | Gaussian | Uniform | Least Squares | 7.3 |
| L2 | L2 | Gaussian | Gaussian | Ridge | 7.5 |
| L2 | L1 | Gaussian | Laplace | LASSO | 13.3 |
| L1 | No | Laplace | Uniform | Robust Regression | 7.4 |

$$Lap(y|\boldsymbol{\theta}^T\mathbf{x}, b) \propto \exp(-\frac{1}{b}|y - \boldsymbol{\theta}^T\mathbf{x}|)$$

# What you need to know

- Regularization: penalize complex model
- L2 regularization:
  - Ridge regression
  - Lead to small but non-zero weights
- LASSO L1 regularization:
  - Sparse solution
  - Non-differentiable but convex -> Subgradient
- Coordinate descent algorithm
  - Shooting algorithm is a simple approach for LASSO