CSCE 421: Machine Learning

Lecture 8: Regularization

Texas A&M University
Bobak Mortazavi
Ryan King
Zhale Nowroozilarki

Goals

- Understanding how to tune models with lots of features!
- Regularization
- Ridge Regression
- Lasso
- Note: Be careful with notation and the interchange between w and β

What kind of Features can Data Have?

Feature Selection: What is it?

Forward Selection

- A “top down” view: Start with a model that includes all input features!

Forward Selection

- A “top down” view: Start with a model that includes all input features!
- Gradually remove features of less importance

Forward Selection

- A “top down” view: Start with a model that includes all input features!
- Gradually remove features of less importance
- Earlier, we did this with p-values

Forward Selection

- A “top down” view: Start with a model that includes all input features!
- Gradually remove features of less importance
- Earlier, we did this with p-values
- Now we can teach the model to learn the importance while it trains

Forward Selection

- A “top down” view: Start with a model that includes all input features!
- Gradually remove features of less importance
- Earlier, we did this with p-values
- Now we can teach the model to learn the importance while it trains
- This is called a “regularizer” – a term we add to our cost/loss function to help train models

Forward Selection

- A “top down” view: Start with a model that includes all input features!
- Gradually remove features of less importance
- Earlier, we did this with p-values
- Now we can teach the model to learn the importance while it trains
- This is called a “regularizer” – a term we add to our cost/loss function to help train models
- This regularizer penalizes the selection of too many parameters – so model learns to eliminate features that are less important

Loss Function Regularization

- Let's assume we have the following loss function:

$$F(w) = f_1(w)$$

- Regularization is then achieved by adding to the cost as:

$$F(w) = f_1(w) + \lambda f_2(w)$$

Loss Function Regularization

- Let's assume we have the following loss function:

$$F(w) = f_1(w)$$

- Regularization is then achieved by adding to the cost as:

$$F(w) = f_1(w) + \lambda f_2(w)$$

- λ is known as the regularization parameter, and is always ≥ 0, where 0 is no regularization.

Loss Function Regularization

- Let's assume we have the following loss function:

$$F(w) = f_1(w)$$

- Regularization is then achieved by adding to the cost as:

$$F(w) = f_1(w) + \lambda f_2(w)$$

- λ is known as the regularization parameter, and is always ≥ 0, where 0 is no regularization.
- So what does a larger λ mean?

Loss Function Regularization

$$F(w) = f_1(w) + \lambda f_2(w)$$

- $\lambda \geq 0$, where $\lambda = 0$ is no regularization.
- So what does a larger λ mean?
  - More dominance by f_2 in the overall cost function
  - Higher regularization
- In practice, λ needs to be tuned so that:
  - F(w) still retains the error of the model through training data $f_1(w)$
  - The altered minima of F(w) reflect the most relevant input features
  - Most popular choice is through vector norms

Ridge Regression

- Let’s return to Linear Regression
- Our Cost Function is:

Ridge Regression

- Let's return to Linear Regression
- Our Cost Function is:

$$RSS = \sum_{p=1}^{P}(y_p - f(x))^2 = \sum_{p=1}^{P}(y_p - w_0 - \sum_{j=1}^{N} w_j x_{pj})^2$$

Ridge Regression

- Let's return to Linear Regression
- Our Cost Function is:

$$RSS = \sum_{p=1}^{P}(y_p - f(x))^2 = \sum_{p=1}^{P}(y_p - w_0 - \sum_{j=1}^{N} w_j x_{pj})^2$$

- With Ridge Regression, we are going to modify this equation by adding a penalty (paying a price) for using too many predictors!

Ridge Regression

- Let's return to Linear Regression
- Our Cost Function is:

$$RSS = \sum_{p=1}^{P} (y_p - f(x))^2 = \sum_{p=1}^{P} (y_p - w_0 - \sum_{j=1}^{N} w_j x_{pj})^2$$

- With Ridge Regression, we are going to modify this equation by adding a penalty (paying a price) for using too many predictors!

$$L(w) = RSS + \lambda \sum_{j=1}^{N} w_j^2 = \sum_{p=1}^{P} (y_p - w_0 - \sum_{j=1}^{N} w_j x_{pj})^2 + \lambda \sum_{j=1}^{N} w_j^2$$

Ridge Regression

$$L(w) = RSS + \lambda \sum_{j=1}^{N} w_j^2 = \sum_{p=1}^{P} (y_p \; - w_0 \;\; - \sum_{j=1}^{N} w_j x_{pj})^2 + \lambda \sum_{j=1}^{N} w_j^2$$

- Ridge Regression creates a tradeoff. You want coefficients that reduce RSS, but now you have a shrinkage penalty.
- This penalty is small if the w are close to 0
- Where least squares creates a single set of coefficients, Ridge Regression now creates a set w_λ^R for each λ

Ridge Regression

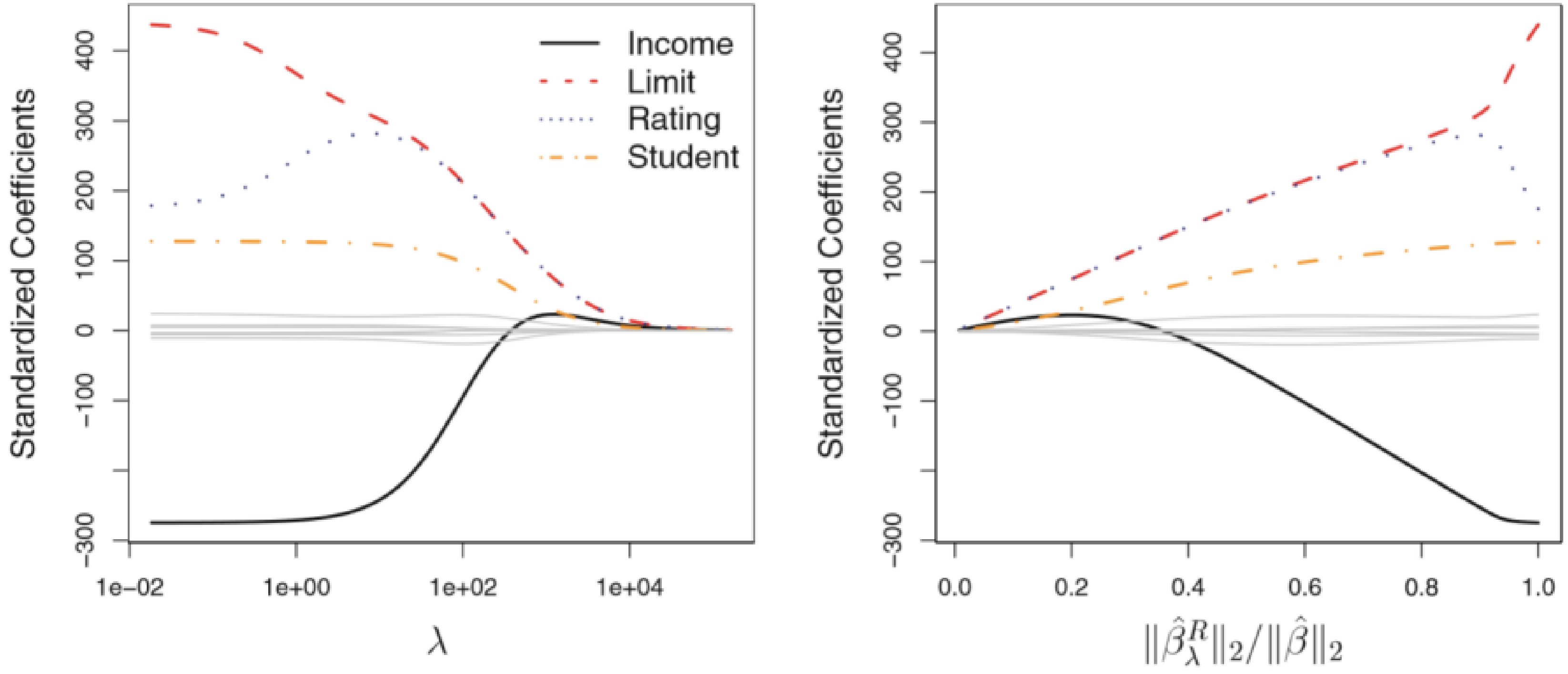
$$L(w) = RSS + \lambda \sum_{j=1}^{N} w_j^2 = \sum_{p=1}^{P} (y_p - w_0 - \sum_{j=1}^{N} w_j x_{pj})^2 + \lambda \sum_{j=1}^{N} w_j^2$$

- Selecting the right λ is key
- Note that the penalty is not assigned to the intercept, since that intercept is the mean value of response when all other factors are 0.
- If we assume all the columns of X have been centered (meaning each has a column mean of 0) then the intercept is the sample mean.

An Example: Credit Default Prediction

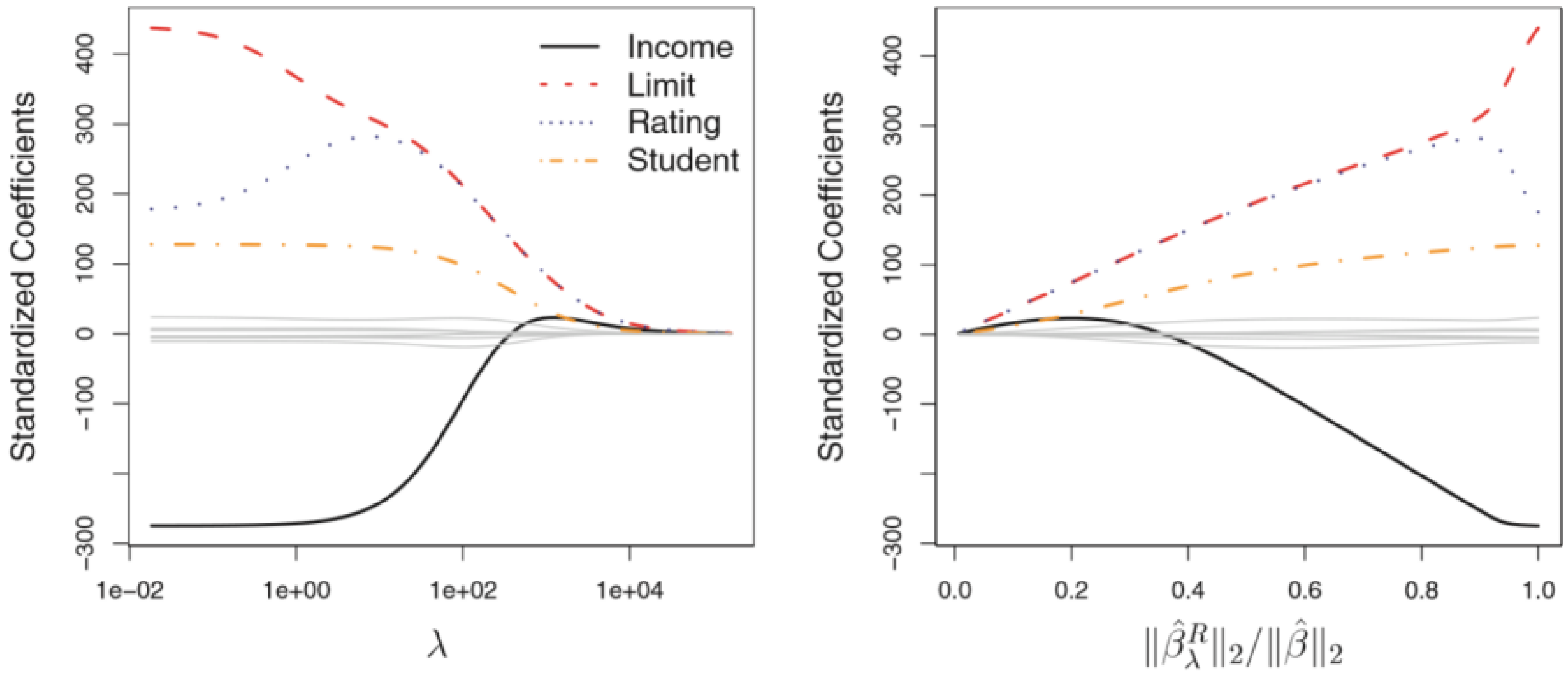
```
      ID            Income           Limit           Rating     
 Min.   :  1.0   Min.   : 10.35   Min.   :  855   Min.   : 93.0  
 1st Qu.:100.8   1st Qu.: 21.01   1st Qu.: 3088   1st Qu.:247.2  
 Median :200.5   Median : 33.12   Median : 4622   Median :344.0  
 Mean   :200.5   Mean   : 45.22   Mean   : 4736   Mean   :354.9  
 3rd Qu.:300.2   3rd Qu.: 57.47   3rd Qu.: 5873   3rd Qu.:437.2  
 Max.   :400.0   Max.   :186.63   Max.   :13913   Max.   :982.0  
     Cards            Age          Education        Gender    Student 
 Min.   :1.000   Min.   :23.00   Min.   : 5.00    Male :193   No :360  
 1st Qu.:2.000   1st Qu.:41.75   1st Qu.:11.00   Female:207   Yes: 40  
 Median :3.000   Median :56.00   Median :14.00                         
 Mean   :2.958   Mean   :55.67   Mean   :13.45                         
 3rd Qu.:4.000   3rd Qu.:70.00   3rd Qu.:16.00                         
 Max.   :9.000   Max.   :98.00   Max.   :20.00                         
 Married              Ethnicity     Balance       
 No :155   African American: 99   Min.   :   0.00  
 Yes:245   Asian           :102   1st Qu.:  68.75  
           Caucasian       :199   Median : 459.50  
                                  Mean   : 520.01  
                                  3rd Qu.: 863.00  
                                  Max.   :1999.00  
```

Ridge Regression and Credit Data



- Each line is one of ten variables as a function of λ
- We can see when λ = 0 we get the standard least squares model
- When λ approaches infinity, we have the null model

Ridge Regression and Credit Data



- Income, limit, rating, and student have the largest coefficients
- Note, in some steps, individual estimates might actually grow because of relative importance!
- What is the right hand figure showing?
- The amount coefficient estimates have been shrunk to 0 as λ increases

Data Scaling

- Scaling is now going to be an important part of our consideration
- In Least Squares, if X was scaled by some constant c, then the least squares solution would be scaled by 1/c – this is no longer going to be the case
- $x_j w_{j,\lambda}^R$ will depend on λ and scaling of x_j
- To avoid scaling issues, we need to standardize predictors

$$\tilde{x}_{pj} = \frac{x_{pj}}{\sqrt{\frac{1}{p}\sum_{p=1}^{P}(x_{pj} - \bar{x}_{pj})^2}}$$

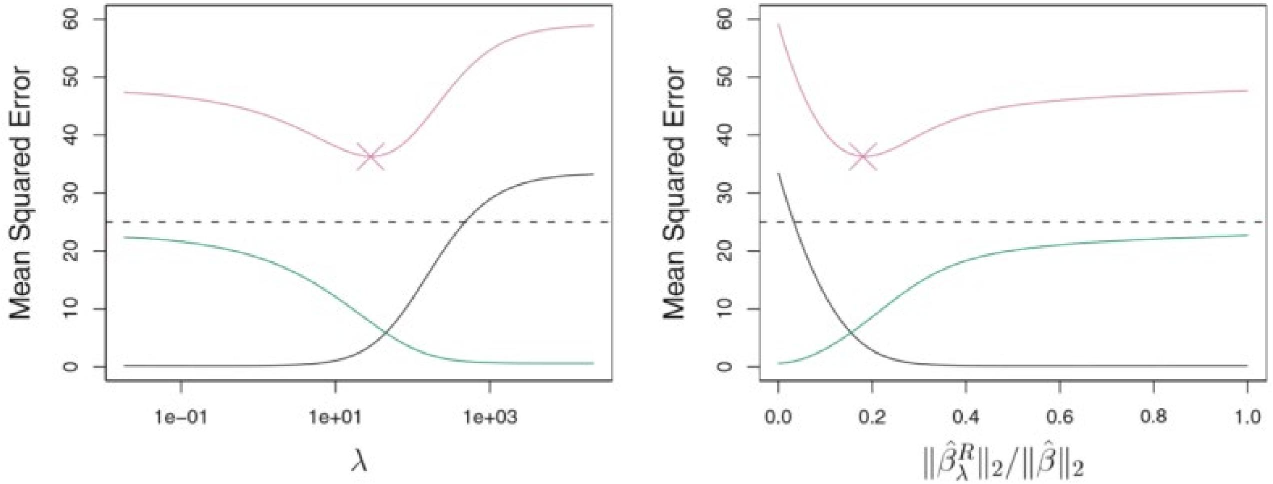
Data Centering (Normalization)

- Normalizing Data is an important step to helping techniques consider only features that provide explanations of variance
- A common technique is to scale and center each predictor – resulting in a mean of 0 and standard deviation of 1

$$\tilde{x}_j = \frac{x_j - \bar{x}_j}{\sigma_j^2}$$

Why does this help?

- Rooted in the bias-variance trade off of models
- As λ increases, flexibility of ridge regression fit decreases, decreasing variance but increasing bias



- Simulated data of p = 45, N = 50, black is bias, green is variance, purple is test error
- λ = 30 is the optimal solution and mean squared error of least squares is almost as high as the null-model!

LASSO

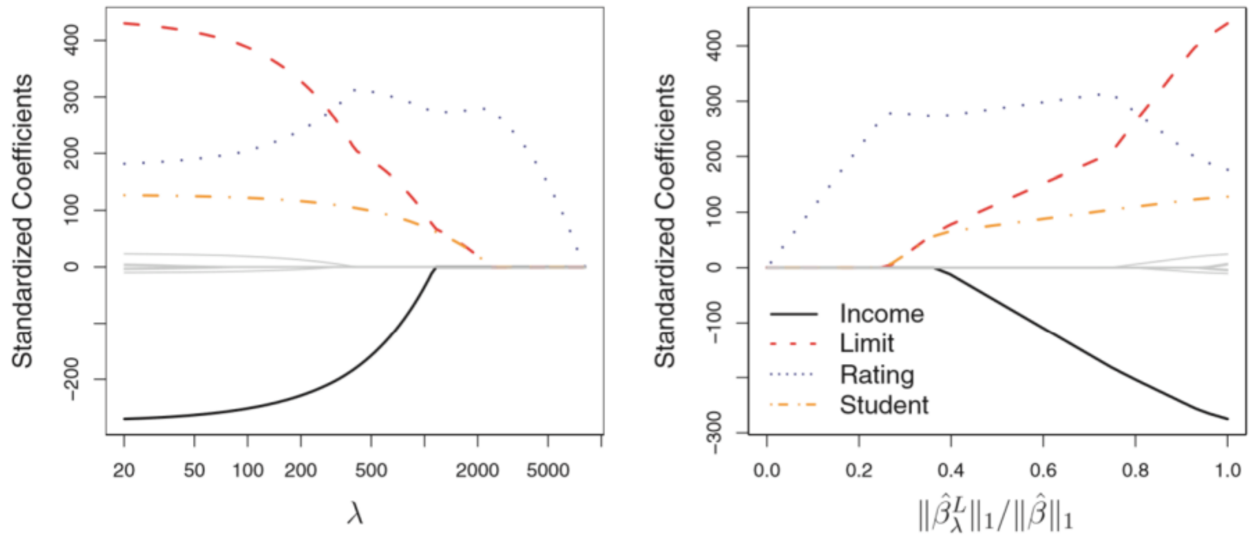
- Ridge Regression has one obvious disadvantage. It will still fit all the predictors.
- The penalty $\lambda \sum_j w_j^2$ will shrink all coefficients but none will hit 0 exactly
- This may no be a problem for accuracy, but it is for interpretability and feature importance
- For example, with the credit data set, the ridge regression will still use all 10 predictors, even if it finds that income, limit, rating, and student are the most important.
- So, what else can we do?

L1 regularization (LASSO)

$$L(w) = RSS + \lambda \sum_{j=1}^{N} |w_j| = \sum_{p=1}^{P} (y_p - w_0 - \sum_{j=1}^{N} w_j x_{pj})^2 + \lambda \sum_{j=1}^{N} |w_j|$$

- If we now create a set of w_λ^L for each λ
- We can use the l1 norm instead of the l2 norm
- Lasso will shrink coefficients, but the l1 penalty will result in coefficients actually reaching 0 with λ sufficiently large
- This means LASSO actually performs variable selection!

LASSO and the credit data



- Lasso picks rating, then student and limit together, then income. Eventually all others would enter as you approach least squares fit
- Where ridge selects coefficients/shrinkage, lasso produces models with any number of variables

Another Formulation

$$\min_{w} \sum_{p=1}^{P} (y_p - w_0 - \sum_{j=1}^{N} w_j x_{pj})^2$$

Subject to $\sum_{j=1}^{N} |w_j| \leq s$ for LASSO

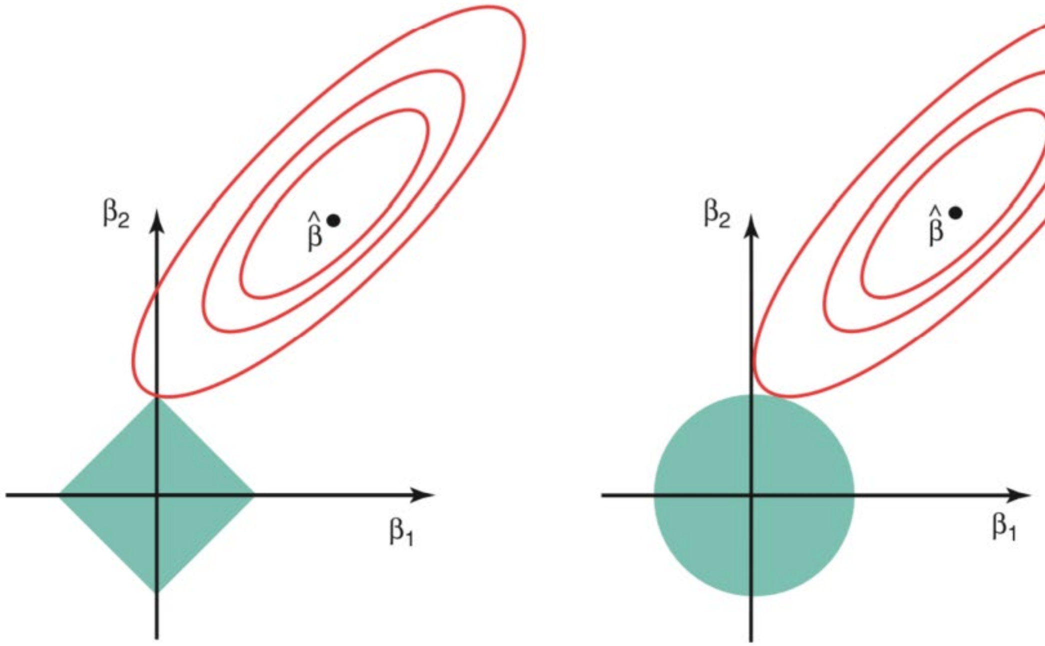
And Subject to $\sum_{j=1}^{N} w_j^2 \leq s$ for LASSO

- If we then consider the p = 2 solution for simplicity

The LASSO solution falls within the diamond $|w_1| + |w_2| \leq s$

The Ridge solution falls within the circle $w_1^2 + w_2^2 \leq s$

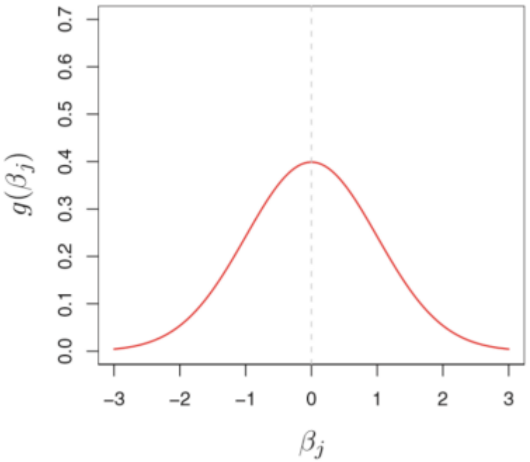
Visualizing the Concept

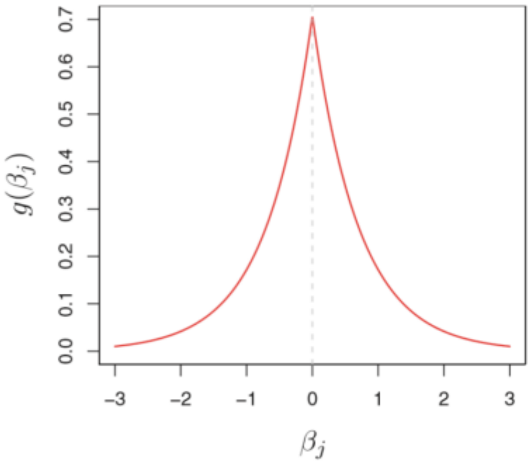


- Ellipses are increasing RSS from the least squares solution
- If the λ allows enough to include RSS that is the fit found
- Because LASSO will intersect at a corner, while Ridge just somewhere on the circle – LASSO sets coefficients to 0 while Ridge just shrinks them

Distributions of Coefficients

- Lasso is better if small set of predictors dominates response
- Ridge is better if all predictors contribute somewhat equally
- Cannot tell in advance, need cross-validation to give us an idea
- Lasso shrinks very differently than Ridge, known as soft thresholding
- Ridge assumes the density function of the posterior probabilities of w are Gaussian (most coefficients are somewhere near 0), while Lasso assumes Laplacian (most coefficients centered at 0)





How to Solve LASSO

Rewrite the optimization problem:

$$\min_{w} \frac{1}{2}\|y - Xw\|_2^2 + \lambda\|w\|_1$$

Challenges:

- The optimization if non-smooth.
- Subgradient Method
  - Subgradients are easy to derive and implement
  - Convergence needs carefully chosen step sizes
  - Convergence is weak theoretically

LASSO algorithms

- Fast l1 minimization algorithms:
  - Iterative Shrinking Thresholding Algorithm (ISTA)
  - Proximal Gradient Method (PGM)
  - Alternating Direction Methods of Multipliers

Iterative Shrinking Thresholding Algorithm (ISTA)

ISTA considers the LASSO model as a special case of the composite objective function:

$$\min_{w} f(w) = f_1(w) + f_2(w),$$

where f_1 is a smooth and convex function, and f_2 is the regularization term that is not necessarily smooth nor convex. Here $f_1(w) = \frac{1}{2}\|y - Xw\|_2^2$.

- If $f_2(w) = \lambda\|w\|_2^2$: Ridge regression.
- If $f_2(w) = \lambda\|w\|_1$: LASSO.

Solving ISTA using Hessian Matrix Approximation

Estimate $f_1(w)$ using its Taylor expansion to the second order around w^k:

$$w^{k+1} = argmin_w\{f_1(w^k) + \nabla f_1(w^k)^T(w - w^k) + \frac{1}{2}(w - w^k)^T\nabla^2 f_1(w^k)(w - w^k) + f_2(w)\}$$

$$\approx argmin_w\{\nabla f_1(w^k)^T(w - w^k) + \frac{\alpha^k}{2}\|w - w^k\|_2^2 + f_2(w)\} \qquad = argmin_w\{\frac{\alpha^k}{2}\|w - \gamma^k\|_2^2 + f_2(w)\}$$

where $\gamma^k = w^k - \frac{1}{\alpha^k}\nabla f_1(w^k)$

Solving ISTA using Hessian Matrix Approximation

Specifically for LASSO, where $f_2(w) = \lambda\|w\|_1$, the last optimization step is separable:

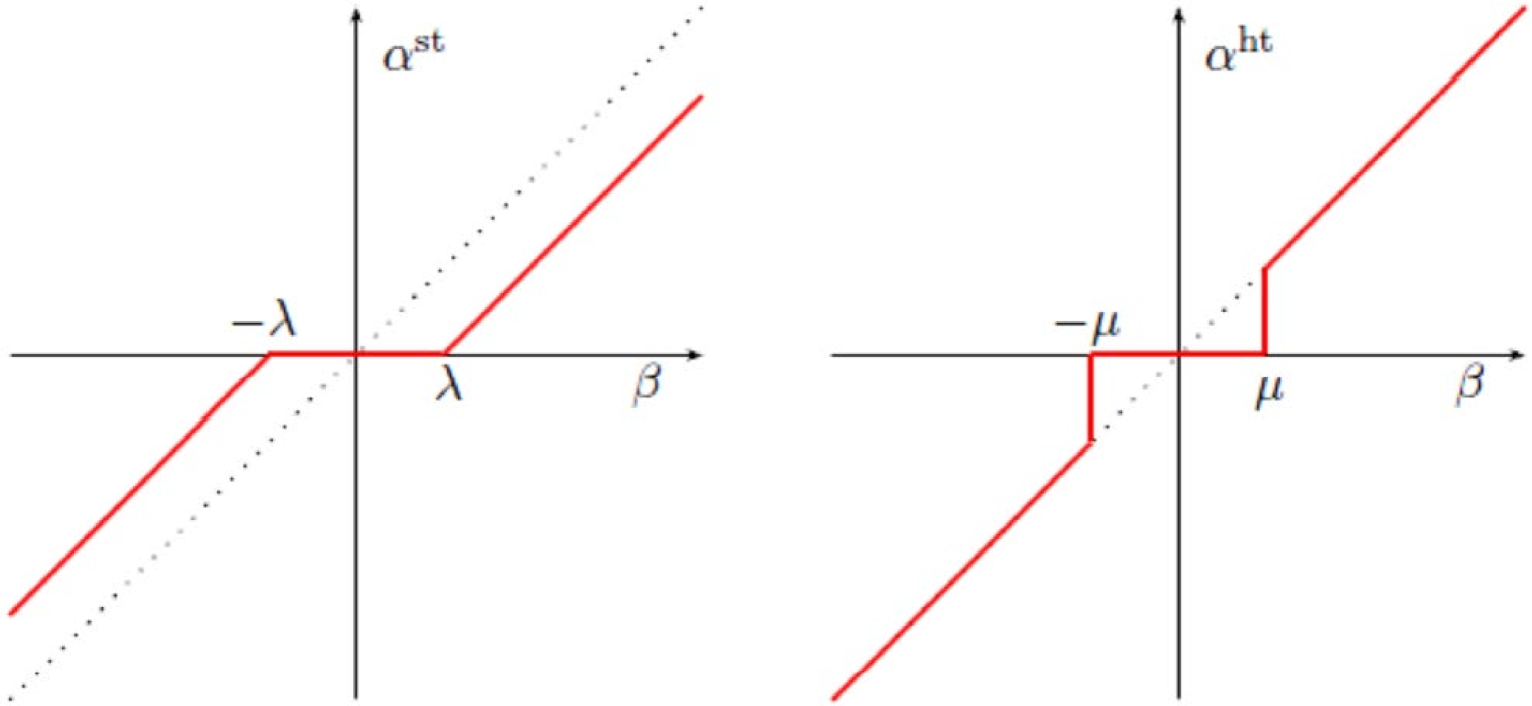
$$w^{k+1} = argmin_w\{\sum_i \frac{\alpha^k}{2}(w_i - \gamma_i^k)_2^2 + \lambda|w_i|\}\}$$

The problem consists of multiple independent 1-D problems that have explicit solution:

$$w_i^{k+1} = soft(\gamma_i^k, \frac{\lambda}{\alpha^k})$$

$$\begin{aligned} \text{soft}(u,a) &\doteq \text{sgn}(u)\max\{|u|-a,0\} \\ &= \begin{cases} \text{sgn}(u)(|u|-a) & \text{if } |u|>a \\ 0 & \text{otherwise} \end{cases} \end{aligned}$$

Visualizing Soft Thresholding



(a) Soft-thresholding operator, $\alpha^{st} = \text{sign}(\beta)\max(|\beta| - \lambda, 0)$.

(b) Hard-thresholding operator $\alpha^{ht} = \mathbf{1}_{|\beta| \geq \mu}\beta$.

ISTA using Proximal Gradient Method

At each step, perform a gradient descent step on $f_1(w)$ without considering the non-smooth regularization:

$$\gamma^k = w^k - \alpha^k \nabla f_1(w^k) = w^k + \alpha^k X^T(y - Xw^k)$$

Then combine the regularization by solving the following proximity problem:

$$w^{k+1} = argmin_w \{ \frac{1}{2\alpha^k} \|w - \gamma^k\|_2^2 + \lambda \|w\|_1 \}$$

which will induce exactly the same solution:

$$w^{k+1} = soft(w^k + \alpha^k X^T(y - Xw^k), \alpha^k \lambda).$$

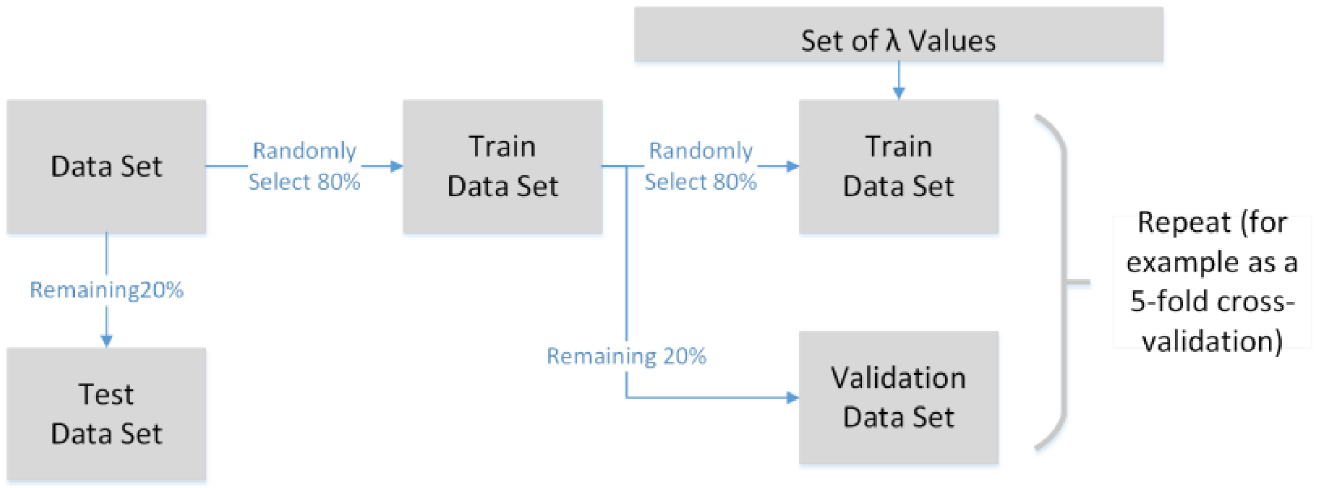
Need select a small enough step size α^k.

LASSO In Practice: Picking λ

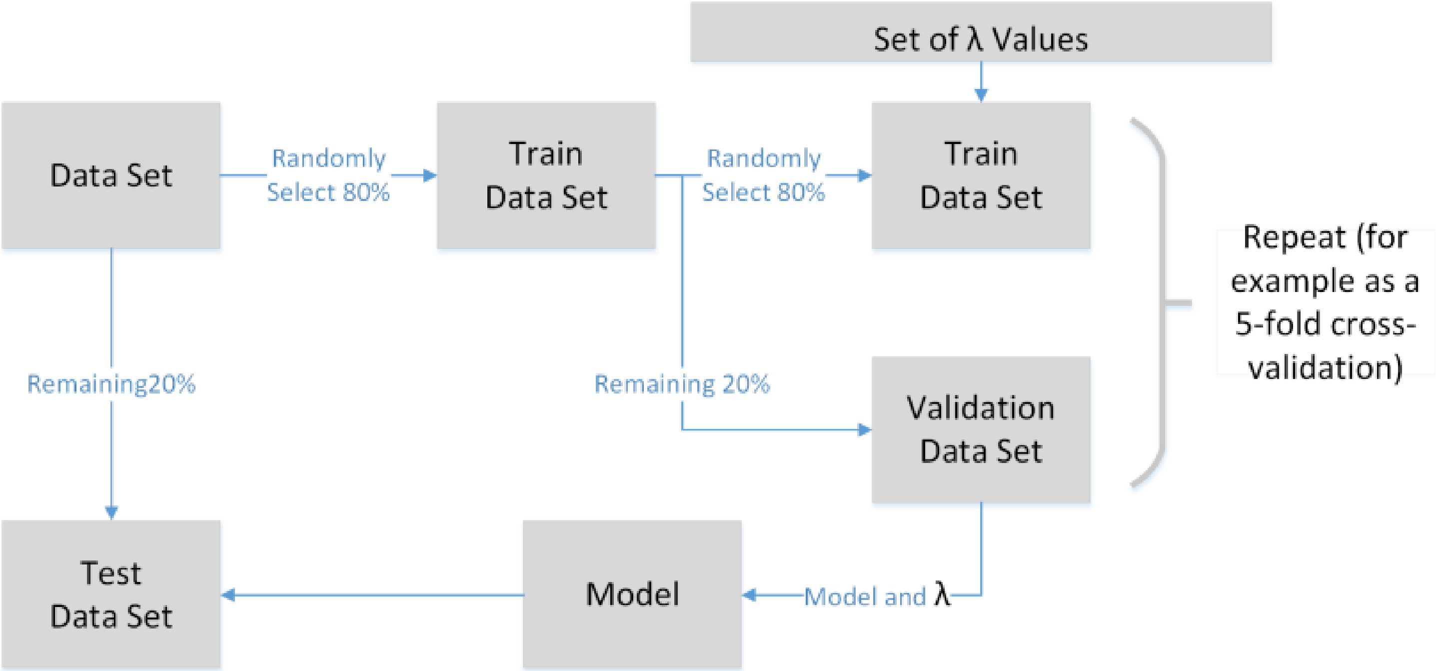
- Need to pick best λ (or s in the alternative formulation) for best estimation
- We can run a cross-validation over a grid of λ values
- We pick the *lambda* with the smallest error

LASSO In Practice: Picking λ

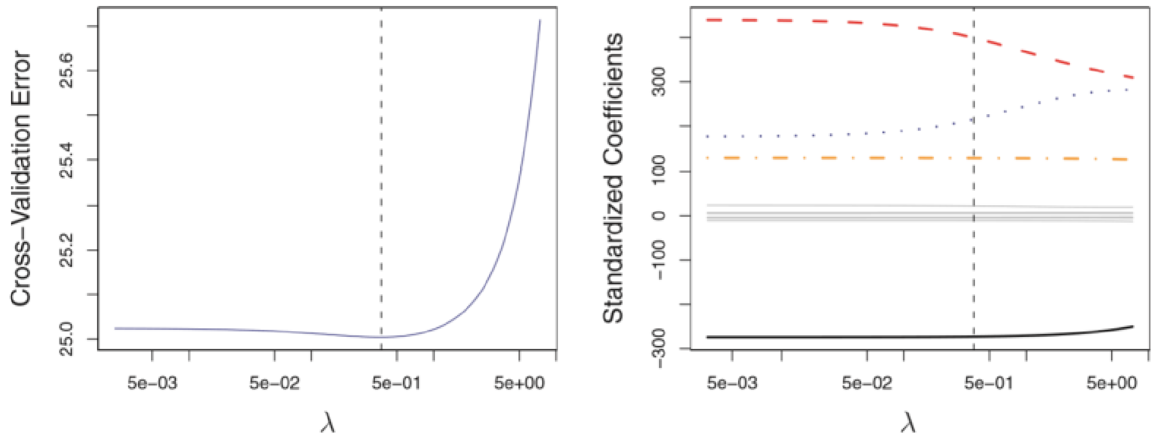
LASSO In Practice: Picking λ



LASSO In Practice: Picking λ

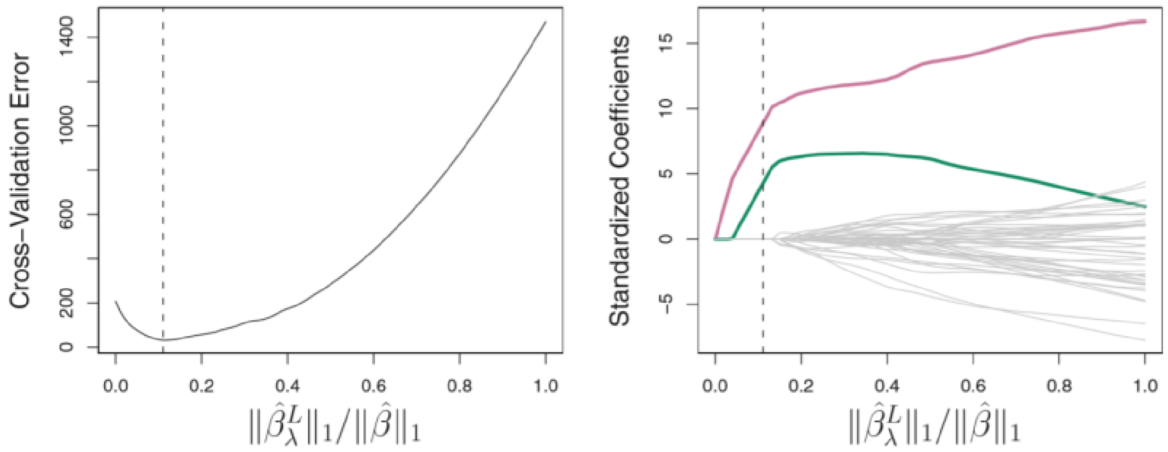


LASSO Examples



- Sometimes Lasso does not do better than Least Squares Solution
- Small λ selected here

LASSO: A Synthetic Example



- Sometimes Lasso does a lot better than Least Squares Solution

Elastic Net: Best of Both Worlds!

- It is not immediately obvious which is better – sometimes need cross-validation to pick between ridge and lasso
- If P > N, but variables are correlated, ridge will empirically do better than lasso
- If N > P lasso cannot select more than P variables before it saturates
- A mix then would be beneficial: Elastic Net

Vanilla Elastic Net

New Objective Function is

$$J(w, \lambda_1, \lambda_2) = \|y - Xw\|^2 + \lambda_2\|w\|_2^2 + \lambda_1\|w\|_1$$

- The objective now has a penalty that is from ridge regression and a penalty that is from lasso
- It turns out this doesn't predict really well, unless the optimal solution is found by ridge or by lasso
- This is because some solution in the middle has coefficients penalized by both λ_1 and λ_2
- To fix it, we adjust the optimal solution. So, first, we solve the vanilla version

LARS-Elastic Net

First we re-write X as

$$\tilde{X} = \frac{1}{\sqrt{1+\lambda_2}} \begin{pmatrix} X \\ \sqrt{\lambda_2} I_p \end{pmatrix}$$

Where I_p is the identity matrix and

$$\tilde{y} = \begin{pmatrix} y \\ 0_{p\times 1} \end{pmatrix}$$

Then we solve for w like a normal lasso problem

$$\tilde{w} = argmin_{\tilde{w}} \|\tilde{y} - \tilde{X}\tilde{w}\|^2 + \frac{\lambda_1}{\sqrt{1+\lambda_2}} \|\tilde{w}\|_1$$

So $w = \frac{\tilde{w}}{\sqrt{1+\lambda_2}}$

Improved Elastic Net

Then we solve for w like a normal lasso problem

$$\tilde{w} = argmin_{\tilde{w}} \|\tilde{y} - \tilde{X}\tilde{w}\|^2 + \frac{\lambda_1}{\sqrt{1+\lambda_2}} \|\tilde{w}\|_1$$

So $w = \frac{\tilde{w}}{\sqrt{1+\lambda_2}}$

- So now we want to undo one of the penalties so coefficients aren't double penalized
- for simplicity we undo the λ_2 penalty (ℓ_2)

$$\hat{w} = \sqrt{1+\lambda_2}\tilde{w}$$

Goals

- Understanding how to tune models with lots of features!
- Regularization
- Ridge Regression
- Lasso
- Takeaways: Linear Models, Regression vs. Classification, Gradient Descent, feature selection, regularization (and modifying loss functions)