Classification 3: Regularization Wrap-up + Generative Models (LDA)

Siva Balakrishnan
Data Mining: 36-462/36-662

January 29, 2018

Chapter 6 of ISL (for regularization), 4.4 of ISL (LDA)

HW 1, Problem 1 Quick Review

First 0/1 loss: want to identify $\hat{f}(x)$ by min. 0/1 loss (binary)

→ Average loss$(\hat{f}) = \mathbb{E}\, L(\hat{f}(x), y)$

$$= \mathbb{E}_x \Big[\underbrace{\mathbb{E}_y \big[L(\hat{f}(x), y) \,\big|\, X = x \big]}_{\text{just focus on this piece}} \Big]$$

$$\mathbb{E}_y \Big[\mathbb{1}\{\hat{f}(x) = 1\} \mathbb{1}\{y = 0\} + \mathbb{1}\{\hat{f}(x) = 0\} \mathbb{1}\{y = 1\} \,\Big|\, X = x \Big]$$

$$= \mathbb{1}\{\hat{f}(x) = 1\} \mathbb{E}_y \big[\mathbb{1}\{y = 0\} \,|\, X = x \big] + \mathbb{1}\{\hat{f}(x) = 0\} \mathbb{E}_y \big[\mathbb{1}\{y = 1\} \,|\, X = x \big]$$

$$= \mathbb{1}\{\hat{f}(x) = 1\} \mathbb{P}(Y = 0 \,|\, X = x) + \mathbb{1}\{\hat{f}(x) = 0\} \mathbb{P}(Y = 1 \,|\, X = x)$$

So if we choose

$\hat{f}(x)=1$ then loss $\mathbb{P}(Y=0 \mid X=x)$

if $\hat{f}(x)=0$ then loss $\mathbb{P}(Y=1 \mid X=x)$

leads to usual rule.

If loss was imbalanced: (using HW notation)

L_{01} if $\hat{f}=1$, $y=0$

L_{10} if $\hat{f}=0$, $y=1$.

Then we have loss:

$$\mathbb{E}_Y\left[\mathbb{1}\{\hat{f}(x)=1\}\,\mathbb{1}\{y=0\}\times L_{01}+\mathbb{1}\{\hat{f}(x)=0\}\,\mathbb{1}\{y=1\}\,L_{10}\right]$$

If we choose:

$\hat{f}(x)=1$ then loss $L_{01}\times\mathbb{P}(y=0 \mid X=x)$

$\hat{f}(x)=0$ then loss $L_{10}\,\mathbb{P}(y=1 \mid X=x)$

Recap: Regularization Basics

- Regularization broadly is a collection of tools to reduce overfitting.
- Overfitting roughly: $\hat{f}, \tilde{f}$

$$\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{f}(x_i))^2 \ll \frac{1}{n}\sum_{i=1}^{n}(y_i - \tilde{f}(x_i))^2 \text{ but } \mathbb{E}(y-\hat{f})^2 \gg \mathbb{E}(y-\tilde{f})^2.$$

- Complex models might fit the training data well but may not generalize (unless we have large amounts of training data).
- One solution (there are many others) is to trade-off fit for complexity, i.e. find a solution that has low-complexity but fits the training data reasonably well.

Recap: Regularization Continued

- Favoring less complex models can have another benefit beyond reducing overfitting.
- Less complex models might be easier to interpret. Particularly, *sparse* models which use few features can in some cases be easy to interpret.
- This suggests a different way to regularize models – to find models that fit the training data but only use a small number of features.
- The sparsity viewpoint motivates lots of different ideas – best subset fitting, greedily introducing features (forward stepwise algorithms), using regularizers that encourage sparsity. These models can also generalize well.

Recap: Two Popular Regularizers

- Ridge Regularization:

$$\hat{\beta}_{ridge} = \arg\min_{\beta} \frac{1}{2}\sum_{i=1}^{n}(y_i - x_i^T\beta)^2 + \lambda\sum_{j=1}^{p}\beta_j^2$$

- LASSO Regularization:

$$\hat{\beta}_{lasso} = \arg\min_{\beta} \frac{1}{2}\sum_{i=1}^{n}(y_i - x_i^T\beta)^2 + \lambda\sum_{j=1}^{p}|\beta_j|$$

Notice we can regularize logistic regression in the same way. How?

$$\beta = \begin{bmatrix} \beta_0 \\ \hline \beta_1 \\ \vdots \\ \beta_p \end{bmatrix}$$

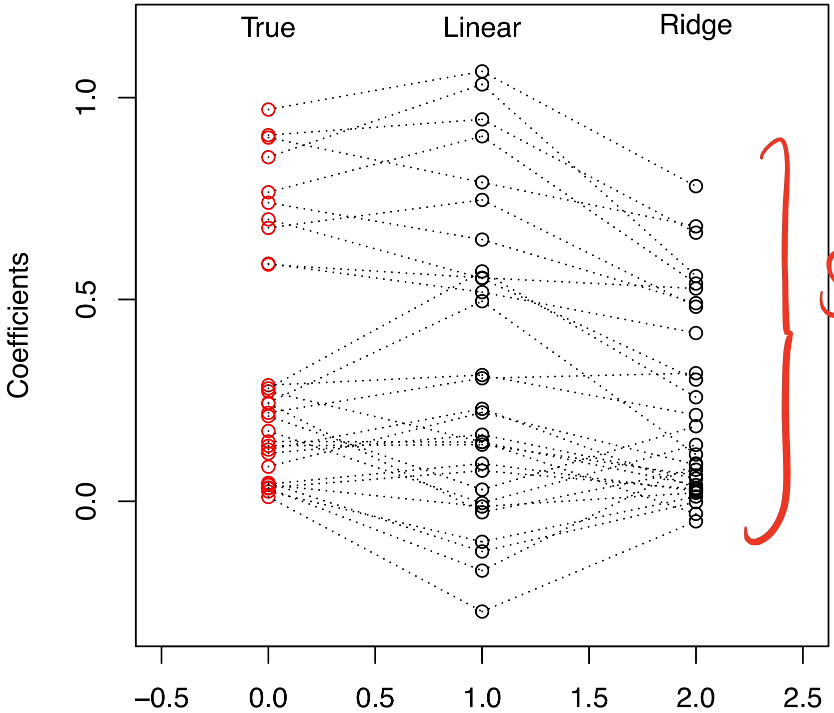
$$\beta_1^2 + \beta_2^2 + \cdots$$

$$|\beta_1| + |\beta_2| \cdots$$

$$\sum_{j=1}^{p} \mathbb{1}\{\beta_j \neq 0\}$$

Example: visual representation of ridge coefficients

A visual representation of the ridge regression coefficients for the same example ($n = 50$, $p = 30$, and $\sigma^2 = 1$; 10 large true coefficients, 20 small) at $\lambda = 25$:

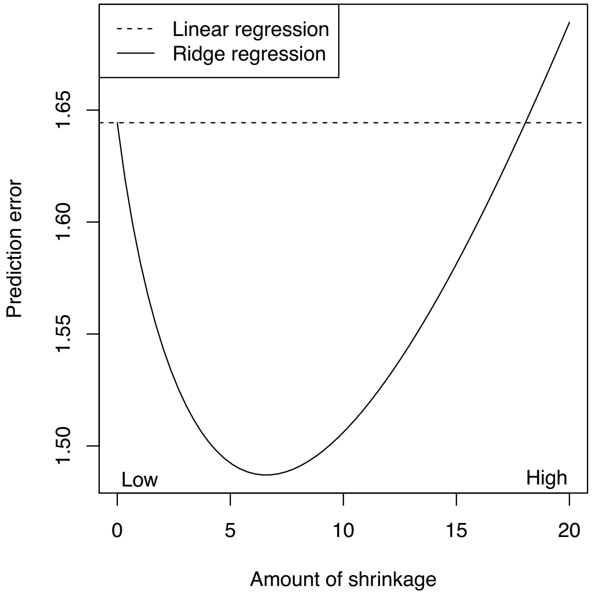


shrinks coefficients toward 0

Does it work?

Recall in regression we can always write:

$$\text{prediction error} = \text{unavoidable error} + \text{bias} + \text{variance}$$



Ridge at best
→ higher bias^2
→ much lower variance

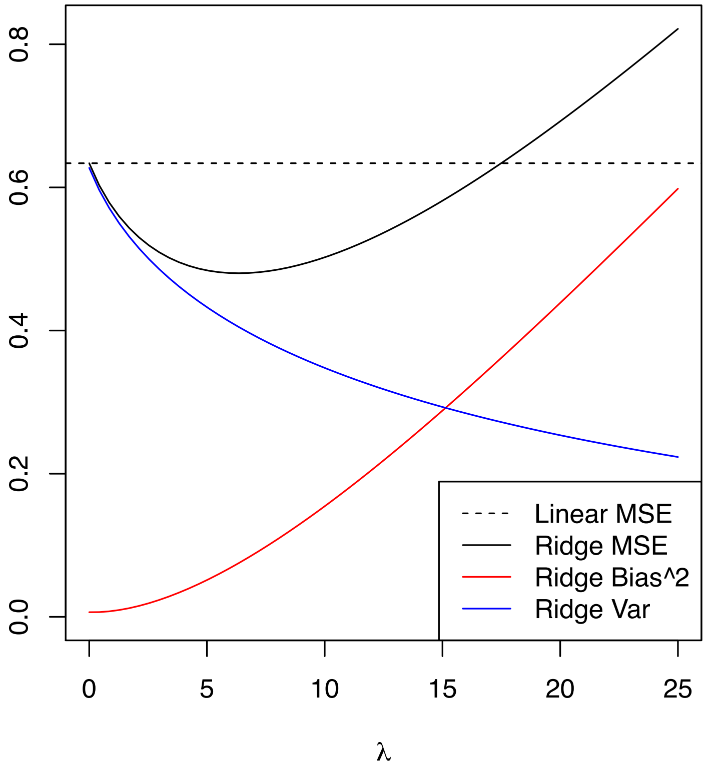
Linear regression: Ridge regression, at its best:

Squared bias ≈ 0.006 $<$ Squared bias ≈ 0.077

Variance ≈ 0.627 $\gg$ Variance ≈ 0.403

Pred. error $\approx 1 + 0.006 + 0.627$ Pred. error $\approx 1 + 0.077 + 0.403$

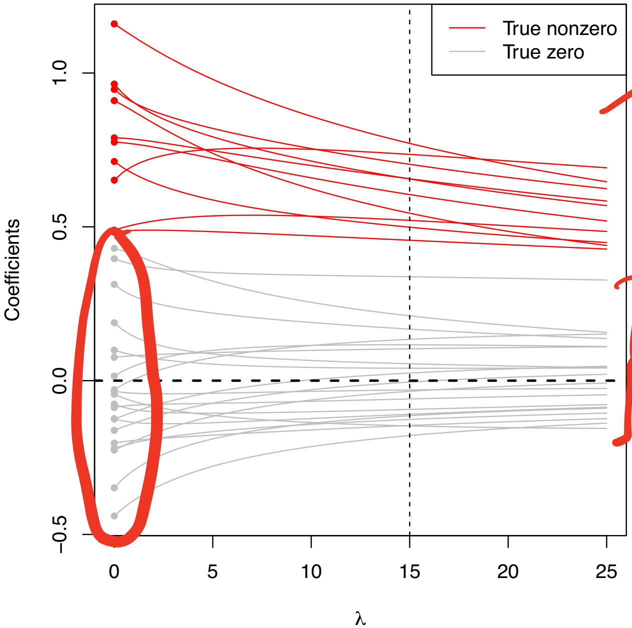
Mean squared error for our last example



Notice that this looks exactly like a model complexity versus test error curve.

Remember that as we vary λ we get different ridge regression coefficients, the larger the λ the more shrunken. Here we plot them again as a function of λ

Suppose many β_j are truly 0.



The red paths correspond to the true nonzero coefficients; the gray paths correspond to true zeros. The vertical dashed line at $\lambda = 15$ marks the point above which ridge regression's MSE starts losing to that of linear regression

An important thing to notice is that the gray coefficient paths are not exactly zero; they are shrunken, but still nonzero

The Lasso

Ridge regression gave better predictions than least squares, but remained uninterpretable.

When p is large, we would like to carry out variable selection at the same time. We do this with the lasso.

The lasso will shrink the estimate, $\widehat{\beta}$, while also carrying out automatic variable selection. As a result, it gives improved predictions *and* interpretable (sparse) models!

The LASSO

The LASSO[1] estimate is defined as

$$\widehat{\beta}^{\text{lasso}} = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \; \|y - X\beta\|_2^2 + \lambda \sum_{j=1}^{p} |\beta_j|$$

$$= \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \; \underbrace{\|y - X\beta\|_2^2}_{\text{Loss}} + \lambda \underbrace{\|\beta\|_1}_{\text{Penalty}}$$

The squared ℓ_2 penalty $\|\beta\|_2^2$ of ridge regression, has been replaced by an ℓ_1 penalty $\|\beta\|_1$. Even though these problems look similar, their solutions behave very differently

Note the name "LASSO" is actually an acronym for: Least Absolute Selection and Shrinkage Operator

[1] Tibshirani (1996), "Regression Shrinkage and Selection via the Lasso"

The LASSO

$$\hat{\beta}^{\text{lasso}} = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \; \|y - X\beta\|_2^2 + \lambda\|\beta\|_1$$

The tuning parameter λ controls the strength of the penalty, and (like ridge regression):

- When $\lambda = 0$, we get: back to least squares
- When $\lambda \to \infty$, we get: $\hat{\beta}^{\text{lasso}} = 0$.

For λ in between these two extremes, we are balancing two ideas: fitting a linear model of y on X, and shrinking the coefficients.

The LASSO

$$\widehat{\beta}^{\text{lasso}} = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \; \|y - X\beta\|_2^2 + \lambda \|\beta\|_1$$

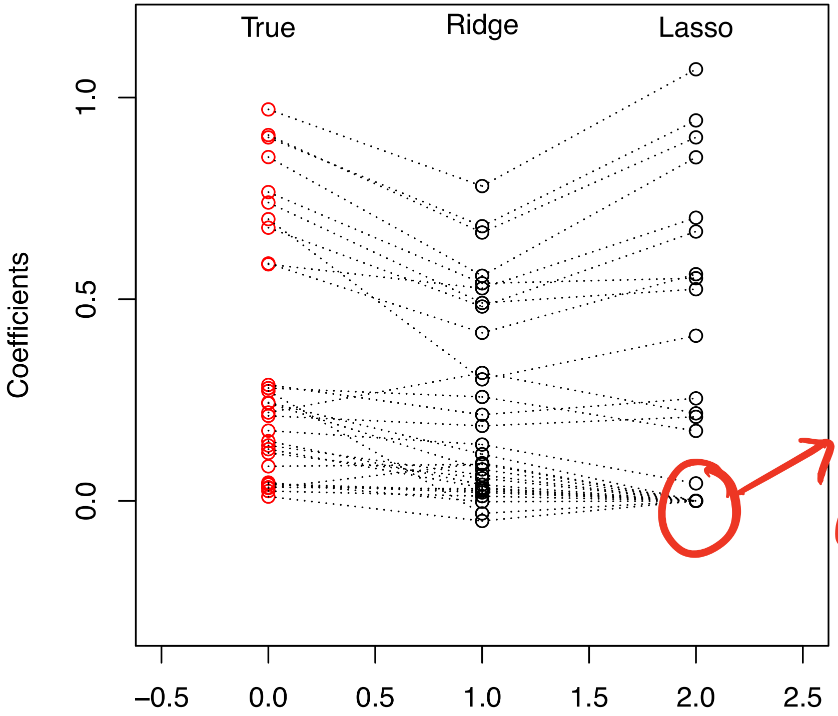
The tuning parameter λ controls the strength of the penalty, and (like ridge regression):

- When $\lambda = 0$, we get:
- When $\lambda \to \infty$, we get:

For λ in between these two extremes, we are balancing two ideas: fitting a linear model of y on X, and shrinking the coefficients.

Example: visual representation of LASSO coefficients

Our running example from last time: $n = 50$, $p = 30$, $\sigma^2 = 1$, 10 large true coefficients, 20 small. Here is a visual representation of LASSO vs. ridge coefficients (with the same degrees of freedom):



many coeffs are shrunk to exactly 0

Advantages of sparsity

- Interpretability: We can understand what the model relies on for prediction (understanding $\widehat{f}$)

- We might gain some insight into the underlying data (though not causally) (helping to understand f)

- If we're building a predictive score, we can measure fewer things in the future (simpler $\widehat{f}$ to apply later)

understand
relevant
&
irrelevant
features

Bias and variance of the lasso

Although we can't write down explicit formulas for the bias and variance of the lasso estimate (e.g., when the true model is linear), we know the general trend. Recall that

$$\hat{\beta}^{\text{lasso}} = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \; \|y - X\beta\|_2^2 + \lambda \|\beta\|_1$$

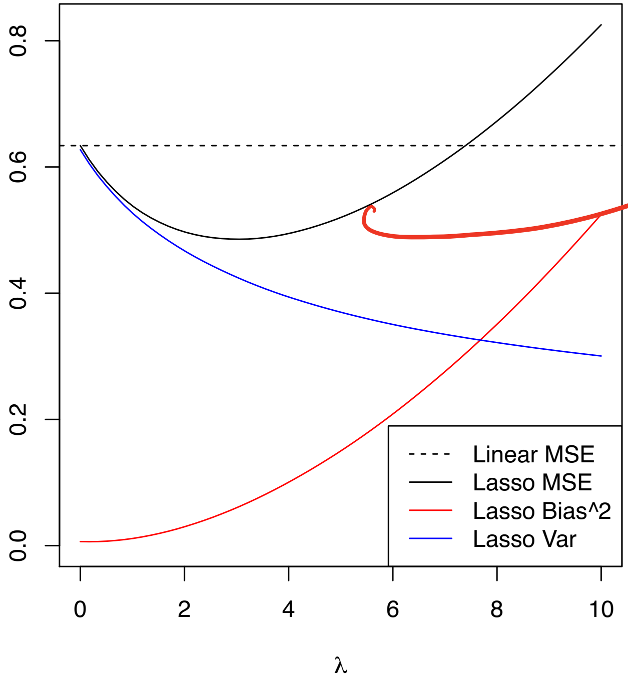
Generally speaking:

- The bias increases as λ (amount of shrinkage)
- The variance decreases as λ (amount of shrinkage)

What is the bias at $\lambda = 0$? The variance at $\lambda = \infty$?

Example: subset of small coefficients

Example: $n = 50$, $p = 30$; true coefficients: 10 large, 20 small

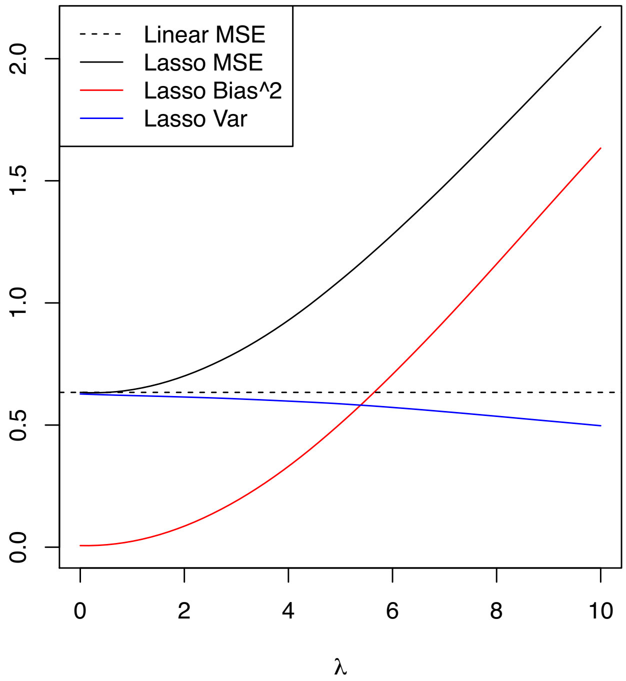


improves on unreg for some λ.

The lasso can also be fit with `glmnet`.

Example: all moderate coefficients

Example: $n = 50$, $p = 30$; true coefficients: 30 moderately large

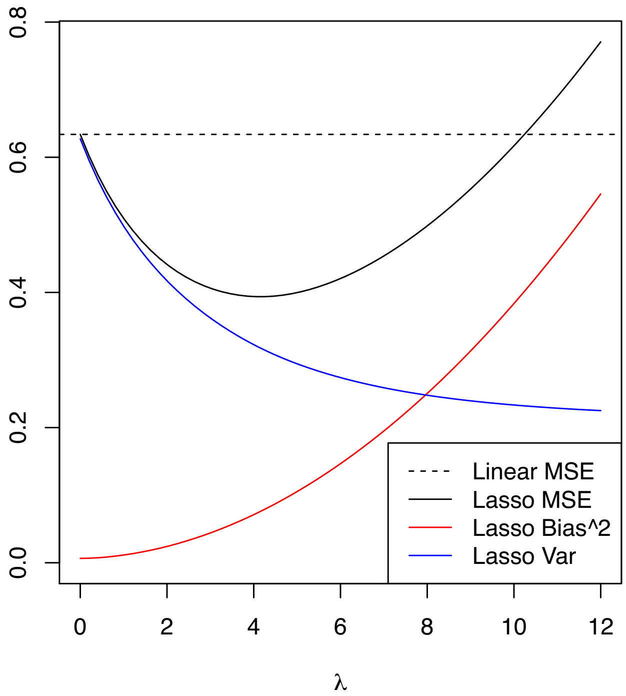


LASSO is perhaps not the right regularizer here.

Note that here, as opposed to ridge regression the variance doesn't decrease fast enough to make the lasso favorable for small λ

Example: subset of zero coefficients

Example: $n = 50$, $p = 30$; true coefficients: 10 large, 20 zero

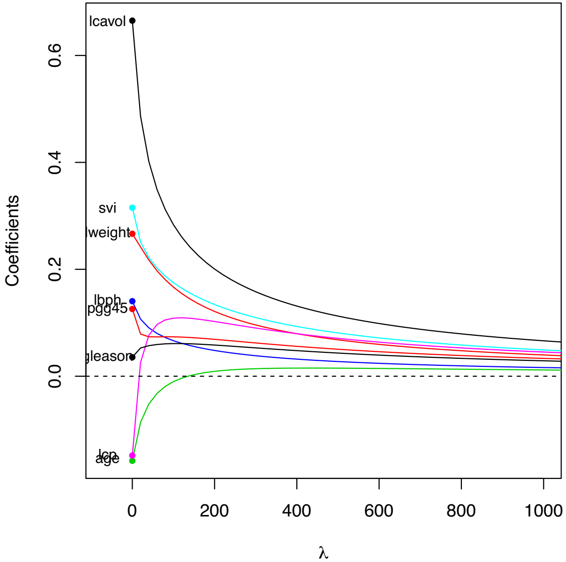


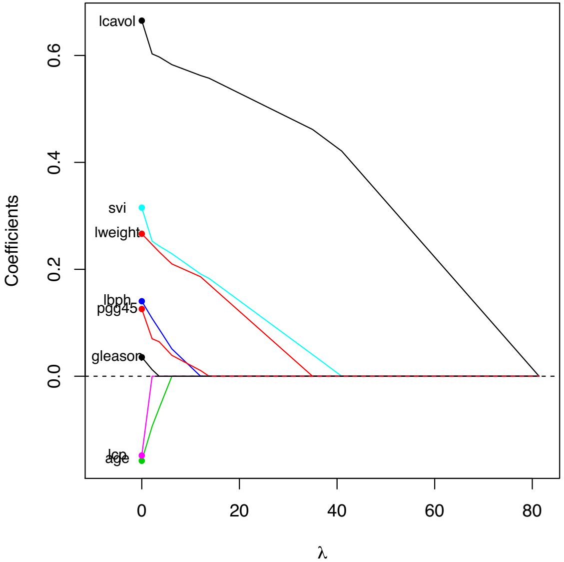
Substantial improvement

Advantage in interpretation

On top the fact that the lasso is competitive with ridge regression in terms of this prediction error, it has a big advantage with respect to interpretation. This is exactly because it sets coefficients exactly to zero, i.e., it performs variable selection in the linear model

For instance here is a picture from ESL – comparing LASSO and Ridge on a prostate cancer dataset.





Why does the lasso give zero coefficients?

- Easier to think about the *constrained form* instead of the *penalized form*.
- **Constrained Form for Ridge:**

$$\hat{\beta}_{ridge} = \arg\min \frac{1}{2}\sum_{i=1}^{n}(y_i - x_i^T\beta)^2$$

subject to: $\left[\sum_{j=1}^{p}\beta_j^2 \leq t\right]$

t small — large bias

t large — low bias.

- **Constrained Form for LASSO:**

$$\hat{\beta}_{LASSO} = \arg\min \frac{1}{2}\sum(y_i - x_i^T\beta)^2$$

$$\sum_{i=1}^{p}|\beta_j| \leq t.$$

Surprisingly, there is an equivalence between the constrained forms and penalized forms.

$$\hat{\beta} = \arg\min \; \underline{f(\beta)} \rightarrow \text{OLS}$$

$$\boxed{\arg\min \; f(\beta) + \lambda \sum \beta_j^2}$$

$$\arg\min \; f(\beta)$$

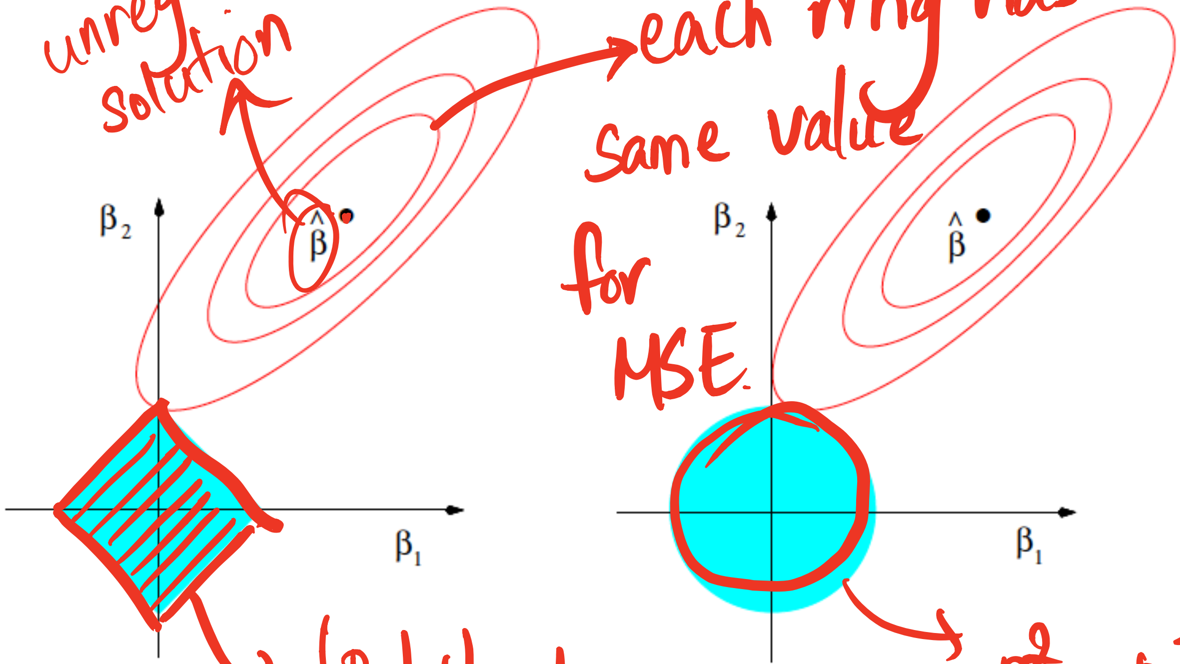
$$\underset{\beta}{\text{over}} \quad \sum_{j=1}^{p} \beta_j^2 \leq t.$$

for every λ there is a t.

Why does the lasso give zero coefficients?

unregularized solution

each ring has same value for MSE



$|\beta_1|+|\beta_2| \leq t$

$\beta_1^2+\beta_2^2 \leq t.$

(From page 71 of ESL)