CS60050
Machine Learning

Linear Regression
Regularization

Slides from Matt Gormley (CMU), Dr. Yaser S. Abu-Mostafa (CalTech, USA),
Dr. Ambedkar Dukipatti (CSA, IISC)

Somak Aditya
Assistant Professor

Sudeshna Sarkar

Department of CSE, IIT Kharagpur

Jul 31, 2024

Empirical Risk, Overfitting

Performance Measure: Loss Function

- We need a guiding mechanism to tell us how good our predictions are given an input.
- The loss for a given example (x, y) is given by
$$\mathrm{loss}(Y, f(X)\ )$$
- We want to perform well on any test data:
$$(X, Y) \sim P_{XY}$$
- Given an X drawn randomly from a distribution, how well does the predictor perform on average?
$$\boxed{Risk\ R(f) = \mathbb{E}_{XY}[loss(Y, f(X))]}$$

Learning as an Optimization

Objective

Given a loss function ℓ, find f such that

Optimal Predictor:

$$f^* = arg\min_f \mathbb{E}[l(Y, f(X))]$$

Empirical Risk Minimizer:

$$\hat{f}_n = \arg\min_{f \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} l(Y_i, f(X_i))$$

$$\boxed{\frac{1}{n} \sum_{i=1}^{n} l(Y_i, f(X_i)) \xrightarrow{\text{law of large numbers}} \mathbb{E}_{XY}[l(Y, f(X))]}$$

Loss and Risk in Regression

- Square Loss

$$loss(X, Y) = (f_\theta(X) - Y)^2$$

We found $\boldsymbol{\theta}$ which minimize the squared loss on data ***we already have***. This averaged loss is called **empirical risk.**

- Risk $R(f)$: What we really want to do is predict the y values for points x ***we haven't seen yet***. i.e. minimize the expected loss on some new data.

$$\mathbb{E}[(f(X) - Y)^2]$$

Machine learning approximates risk-minimizing models with empirical-risk minimizing ones.

Risk Minimization

Generally minimizing empirical risk (loss on the data) instead of true risk works fine, but it can fail if:

- The **data sample is biased**. e.g. you cant build a (good) classifier with observations of only one class.
- There is **not enough data** to accurately estimate the parameters of the model. Depends on the complexity (number of parameters, variation in gradients, complexity of the loss function, generative vs. discriminative etc.).

Regularization

- What if we have too many features?
- Can linear regression itself solve the feature selection problem?

1. Case 1: we want "$\boldsymbol{\theta}$" such that most of its elements are small
2. Case 2: we want "$\boldsymbol{\theta}$" such that most of its elements are 0

Regularized Loss $= L(y, \hat{y}) + \lambda \, \mathrm{Regularizer}(\theta)$

Regularization is very important when training set is small and the number of features is very large.

Generalization capacity and Regularization

- How well will the learned function work on the unseen data?
- Occam's principle: A simple f can generalize better
- Regularization to keep the function from overfitting the training data. (More on overfitting later in the class)

$$\mathrm{f} = \arg\min_{f \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} l\big(Y_i, f(X_i)\big) + \lambda\, \mathrm{Complexity}(f)$$

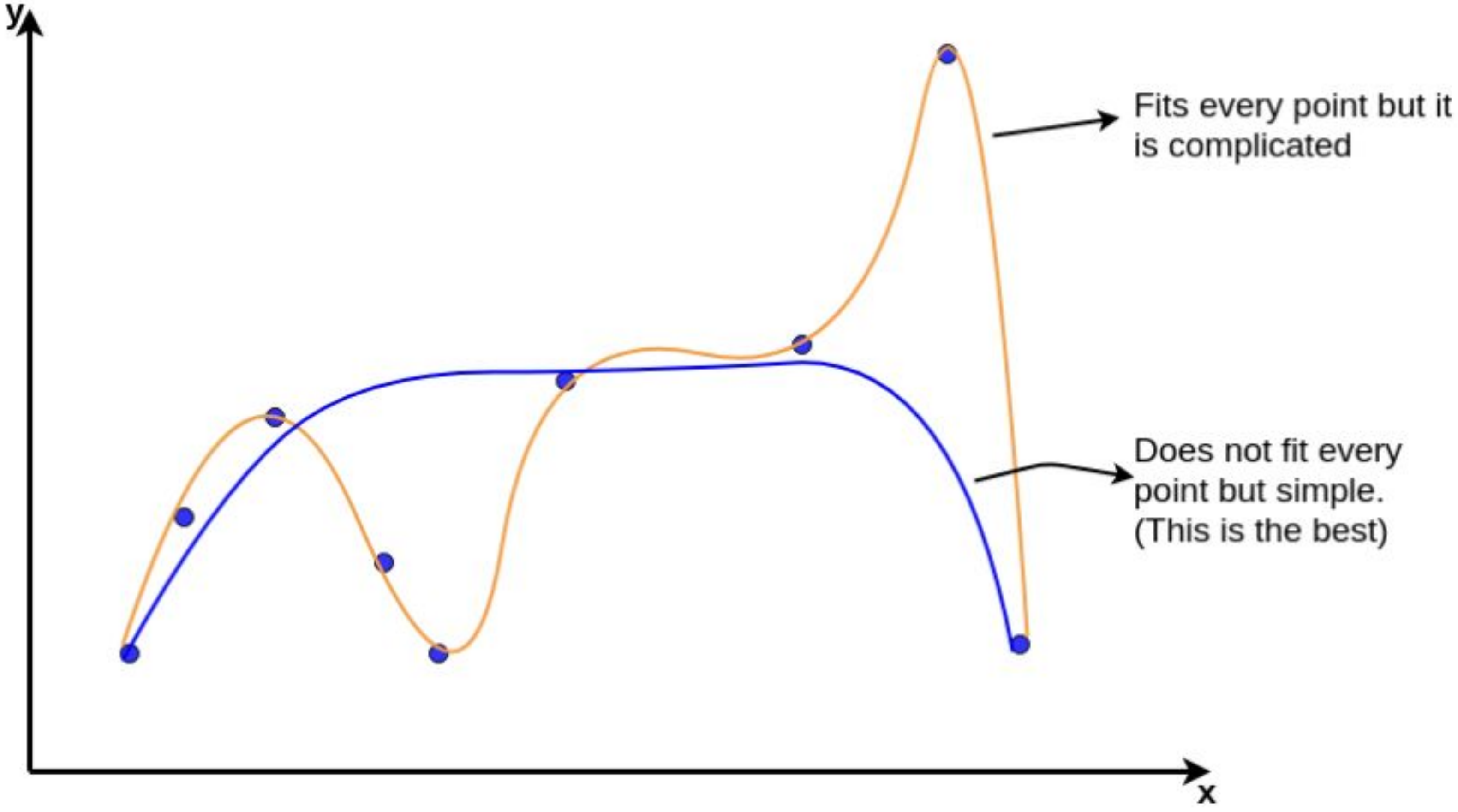
- λ controls the amount of regularization.

Generalization capacity and Regularization

- One way of regularization is to put a bias on the model forcing the learning to prefer certain types of weights over others.
- What makes for a "simpler" model for a linear model? Two ideas.
  1. If weights are large, a small change in a feature can result in a large change in the prediction.
  2. Might also prefer weights of 0 for features that aren't useful

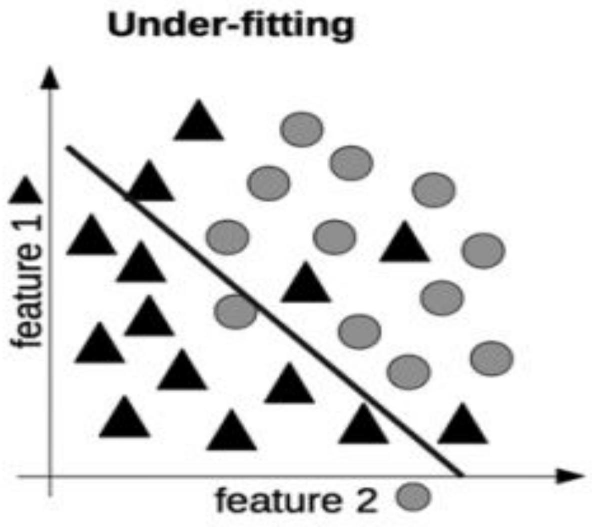
$$\mathrm{f} = \arg\min_{f \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} l\big(Y_i, f(X_i)\big) + \lambda\, \Omega(\theta)$$

Generalizing Capacity



The blue curve has better generalization capacity. The orange curve overfits the data

Generalizing Capacity (contd..)



What is a good Regularizer?

- Case 1: If weights are large, a small change in a feature can result in a large change in the prediction, Also gives too much weight to any one feature.

$$L2 \text{ Regularizer} = \sqrt{\sum_{\theta_j} {\theta_j}^2}$$

- Case 2: Might also prefer weights of 0 for features that aren't useful.

$$L1 \text{ Regularizer} = \sum_{\theta_j} |\theta_j|$$

Squared weights penalizes large values more
Sum of weights will penalize small values more

On Regularization

Claim: Small weights $\theta = (\theta_1, \dots, \theta_d)$ ensure function $y = f(x) = \theta^\top x$ is *smooth* *(why smoothness?)*

Justification:

- Let $x_n, x_m \in \mathbb{R}^d$ such that

$$x_{n_j} = x_{m_j},\ j = 1,2,\dots,d-1, \text{ but } |x_{n_d} - x_{m_d}| = \epsilon$$

- Then $|y_n - y_m| = \epsilon w_d$
- If w_d is large, the difference would be large.

=> $f(x)$ is not smooth.

Predicting say Student grades to admissions (0/1).
- 2 students, all but 1 is same.

Linear Regression with Regularizers (Ridge Regression)

Ridge Regression

- Modified Objective: Given $\{(x_n, y_n)\}_{n=1}^{N}$, find w such that

$$L_{emp}(f) = \frac{1}{N}\sum_{n=1}^{N}(y_n - \theta^T x_n)^2 + \lambda||\theta||^2$$

  - $||\theta||^2 = \theta^T\theta$
  - λ is a hyperparameter, controls the amount of regularization.
- Solution:

$$\frac{\partial L(\theta)}{\partial \theta} = \sum_{n=1}^{N} 2(y_n - \theta^T x_n)(-x_n) + 2\lambda\theta = 0$$

Ridge Regression

- $$\frac{\partial L(\theta)}{\partial \theta} = \sum_{n=1}^{N} 2(y_n - \theta^T x_n)(-x_n) + 2\lambda\theta = 0$$

$$\Rightarrow \gamma(\theta) = \sum_{n=1}^{N} x_n(y_n - x_n^T\theta)$$

$$\Rightarrow \gamma(\theta) = \sum_{n=1}^{N} x_n y_n - \sum_{n=1}^{N} x_n x_n^T\theta$$

$$\Rightarrow \gamma\theta = X^T Y - X^T X W$$

$$\Rightarrow \gamma\theta + X^T X W = X^T Y$$

$$\Rightarrow \theta = (X^T X + \gamma I)^{-1} X^T Y$$

Ordinary Regression

- $$\frac{\partial L(\theta)}{\partial \theta} = \sum_{n=1}^{N} 2(y_n - \theta^T x_n)\frac{\partial L(\theta)}{\partial \theta} = 0$$

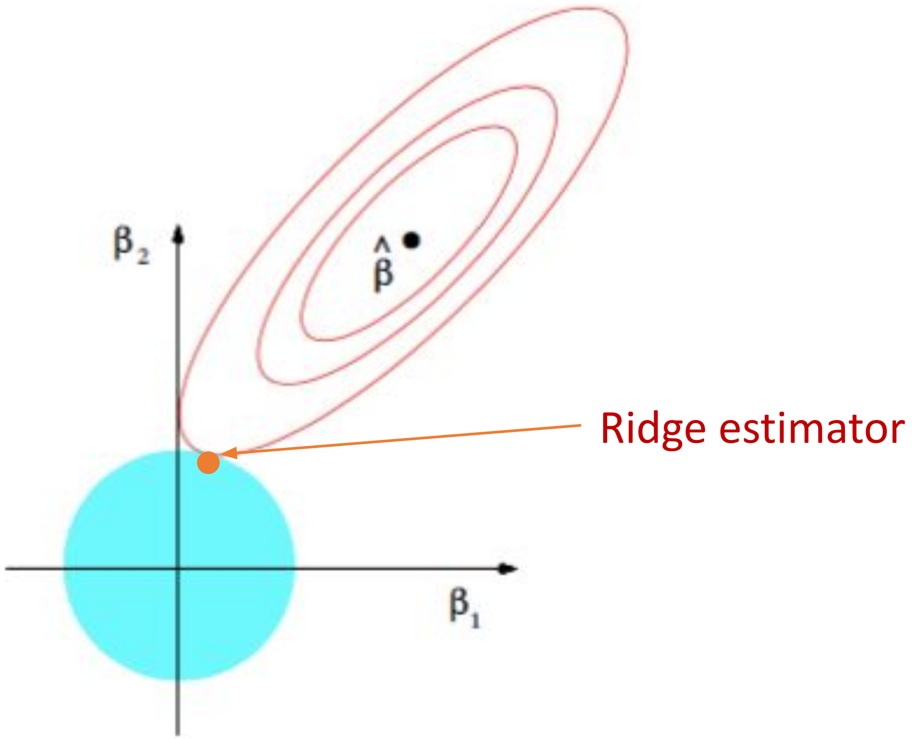
$$\Rightarrow \sum_{n=1}^{N} 2(y_n - \theta^T x_n)(-x_n) = 0$$

$$\Rightarrow \sum_{n=1}^{N} x_n y_n - \sum_{n=1}^{N} x_n x_n^\top\theta = 0$$

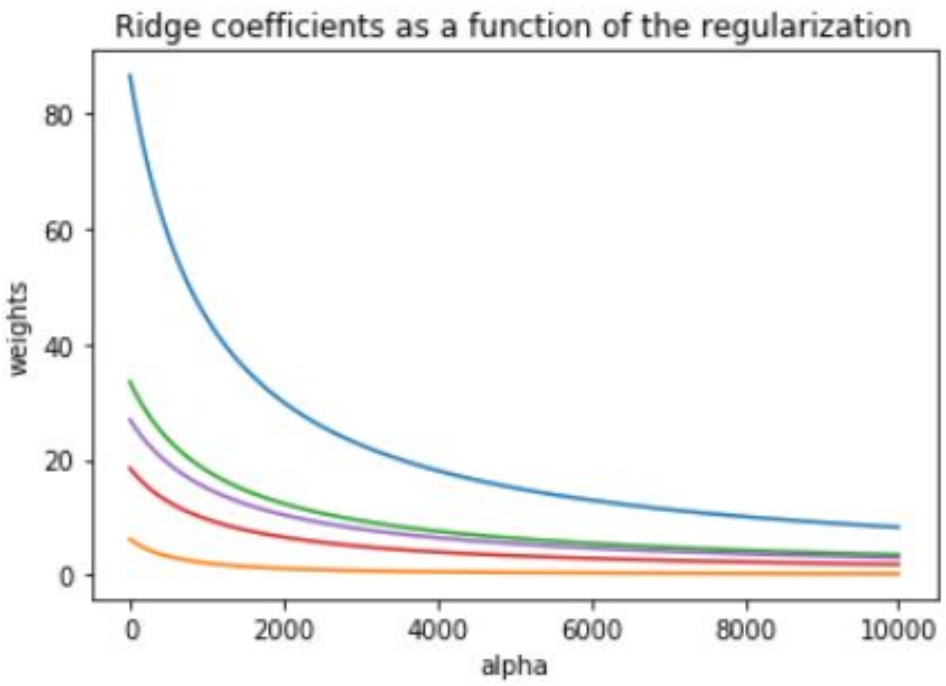
$$\Rightarrow \sum_{n=1}^{N} x_n x_n^\top\theta = \sum_{n=1}^{N} x_n y_n$$

$$\Rightarrow \theta = (X^\top X)^{-1} X^\top Y$$

Ridge Regularization visualized



The ridge estimator is where the constraint and the loss intersect.

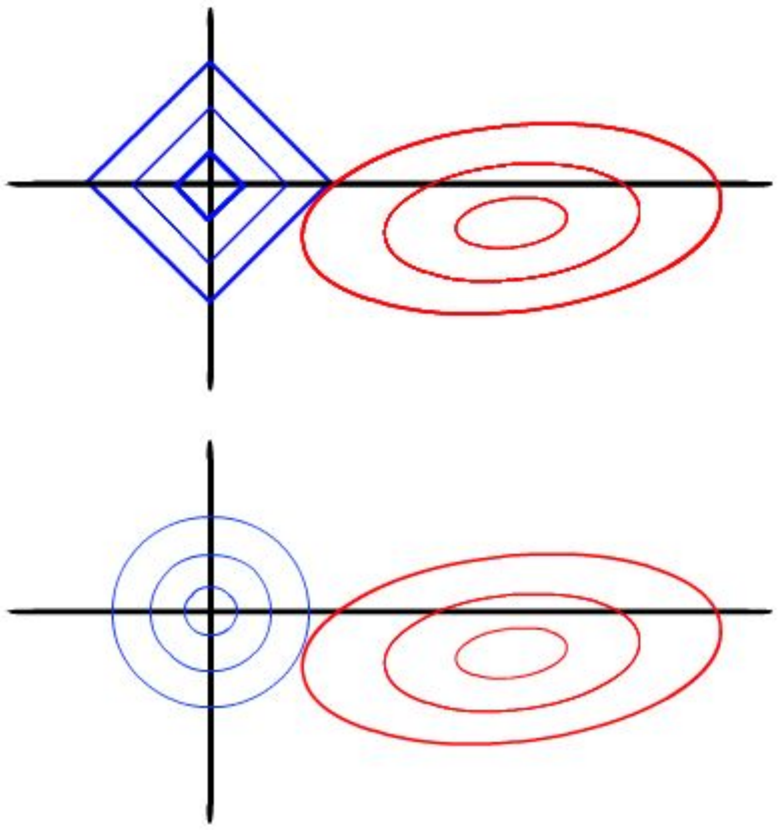


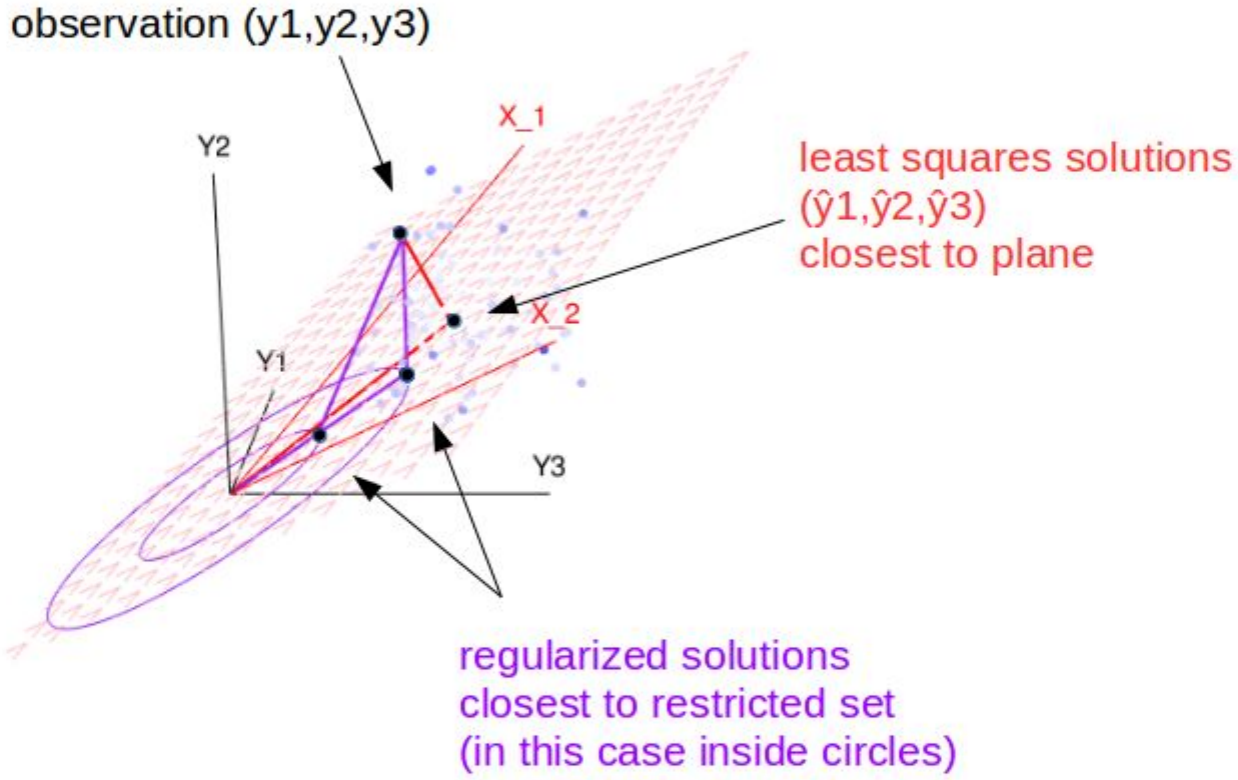
The values of the coefficients decrease as lambda increases, but they are not nullified.

- Ellipses are the contours of $\sum_{i=1}^{n}(y_i - \hat{\beta}_0 - \hat{\beta}_1 x_{i1} - \hat{\beta}_2 x_{i2})^2$, which centered at the OLS estimates $(\hat{\beta}_{1,OLS}, \hat{\beta}_{2,OLS})$.
- (Left) Ellipse intersects the circle of radius t at the Ridge estimate.

Khan Academy Video Link

Geometric Explanation





On Convexity

- The squared loss function in linear regression is convex.
    - L2 regularizer it is strictly convex.

- <u>Convex Functions</u>:
    - For scalar functions : Convex if the second derivative is nonnegative everywhere
    - For vector valued : Convex if Hessian is positive semi definite

Gradient Descent Solution for Least Squares

- Ridge Regression has an analytical solution.

$$\theta^* = (X^T X + \lambda I)^{-1} X^T Y$$

- Involves inverting a $d \times d$ matrix.
- Difficult for large d
- Gradient Descent solution may be used.

L_1 Regularizer LASSO

- l_1 Regularizer $R(\theta) = \left|\left|\theta\right|\right|_1 = \Sigma_{j=1}^{d}|\theta_j|$

- Promotes θ to have very few non zero components.
- Since LASSO regression tend to produce zero estimates for a number of model parameters - we say that LASSO solutions are sparse - we consider LASSO to be a method for variable selection.

- LASSO has no conventional analytical solution, as the L1 norm has no derivative at 0.

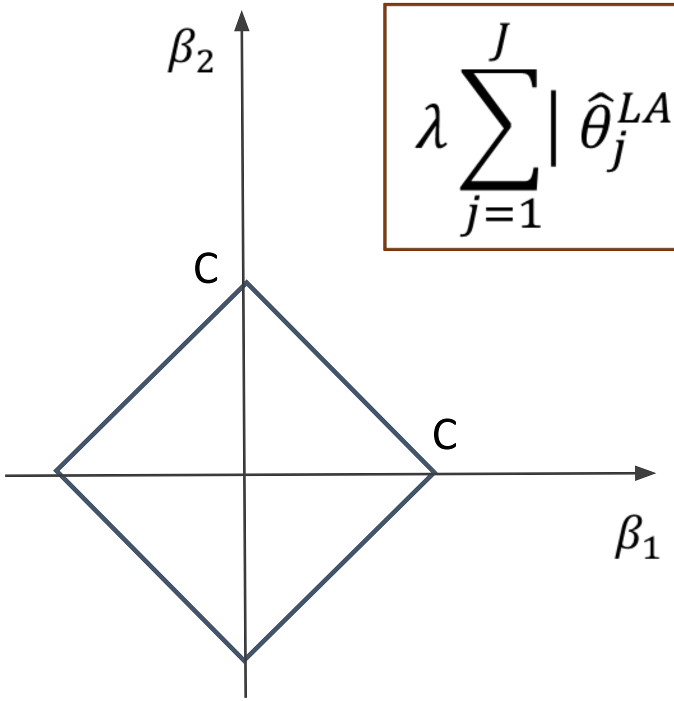
The Geometry of Regularization (LASSO)

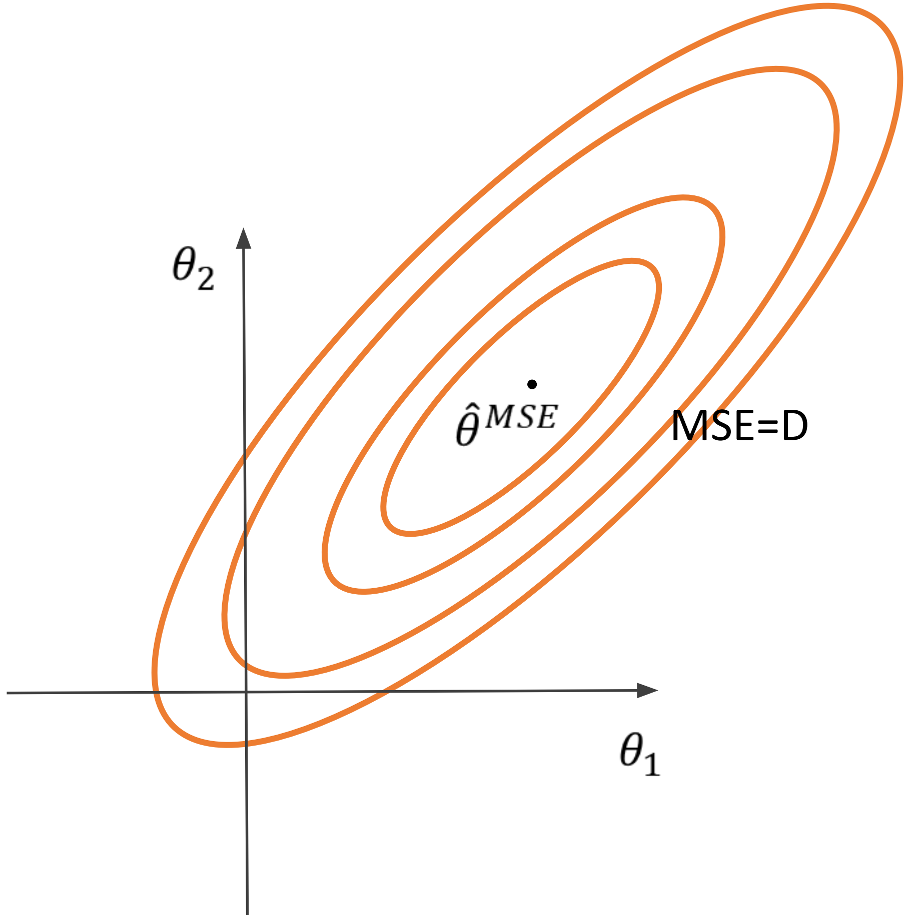
- $L_{LASSO}(\boldsymbol{\theta}) = \frac{1}{n}\sum_{i=1}^{n} |y_i - \boldsymbol{\theta}^T \boldsymbol{x}|^2 + \lambda \sum_{j=1}^{J} |\theta_j|$

$$\widehat{\boldsymbol{\theta}}^{LASSO} = \operatorname{argmin} L_{LASSO}(\boldsymbol{\theta})$$

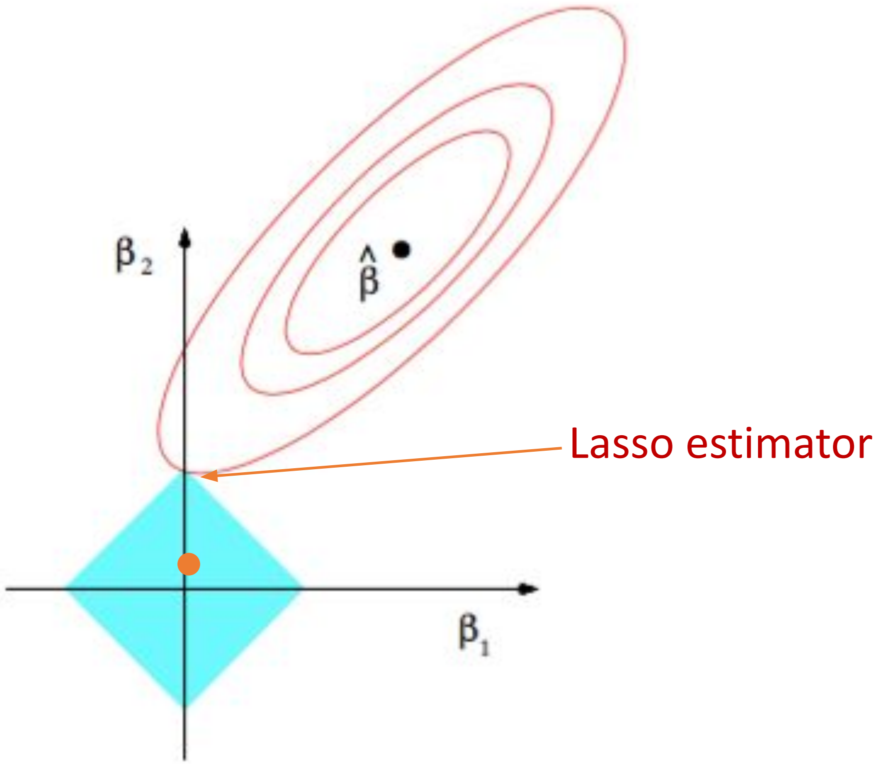
$$\frac{1}{n}\sum_{i=1}^{n} |y_i - \widehat{\boldsymbol{\theta}}^{LASSO^T} \boldsymbol{x}|^2 = \mathrm{D}$$

$$\lambda \sum_{j=1}^{J} |\hat{\theta}_j^{LASSO}| = C$$

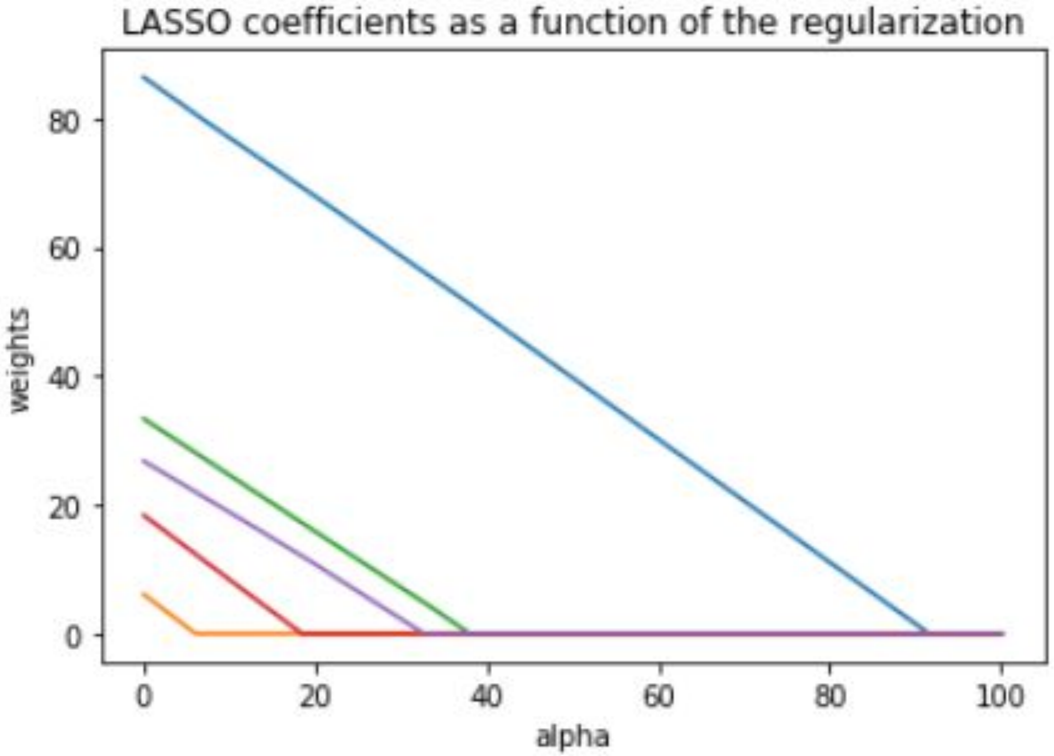




LASSO visualized



The Lasso estimator tends to zero out parameters



The values of the coefficients decrease as lambda increases, and are nullified fast.

- Ellipses are the contours of $\sum_{i=1}^{n}(y_i - \hat{\beta}_0 - \hat{\beta}_1 x_{i1} - \hat{\beta}_2 x_{i2})^2$, which centered at the OLS estimates $(\hat{\beta}_{1,OLS}, \hat{\beta}_{2,OLS})$.
- (Right) Ellipse intersects the square $(|\hat{\beta}_1| + |\hat{\beta}_2| < t)$ at the Lasso estimate

Lasso vs Ridge

Lasso tends to generate sparser solutions than a quadratic regularizer.