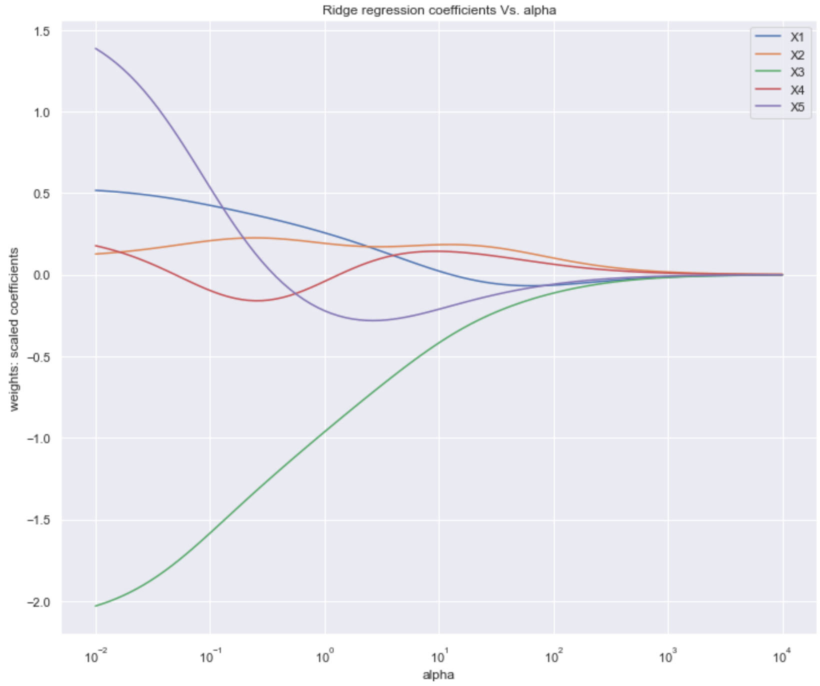
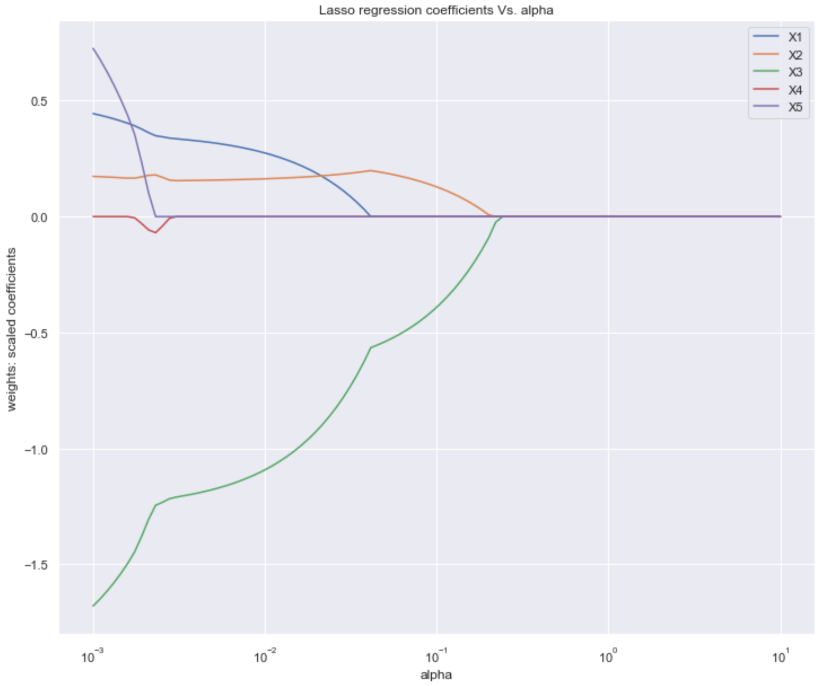
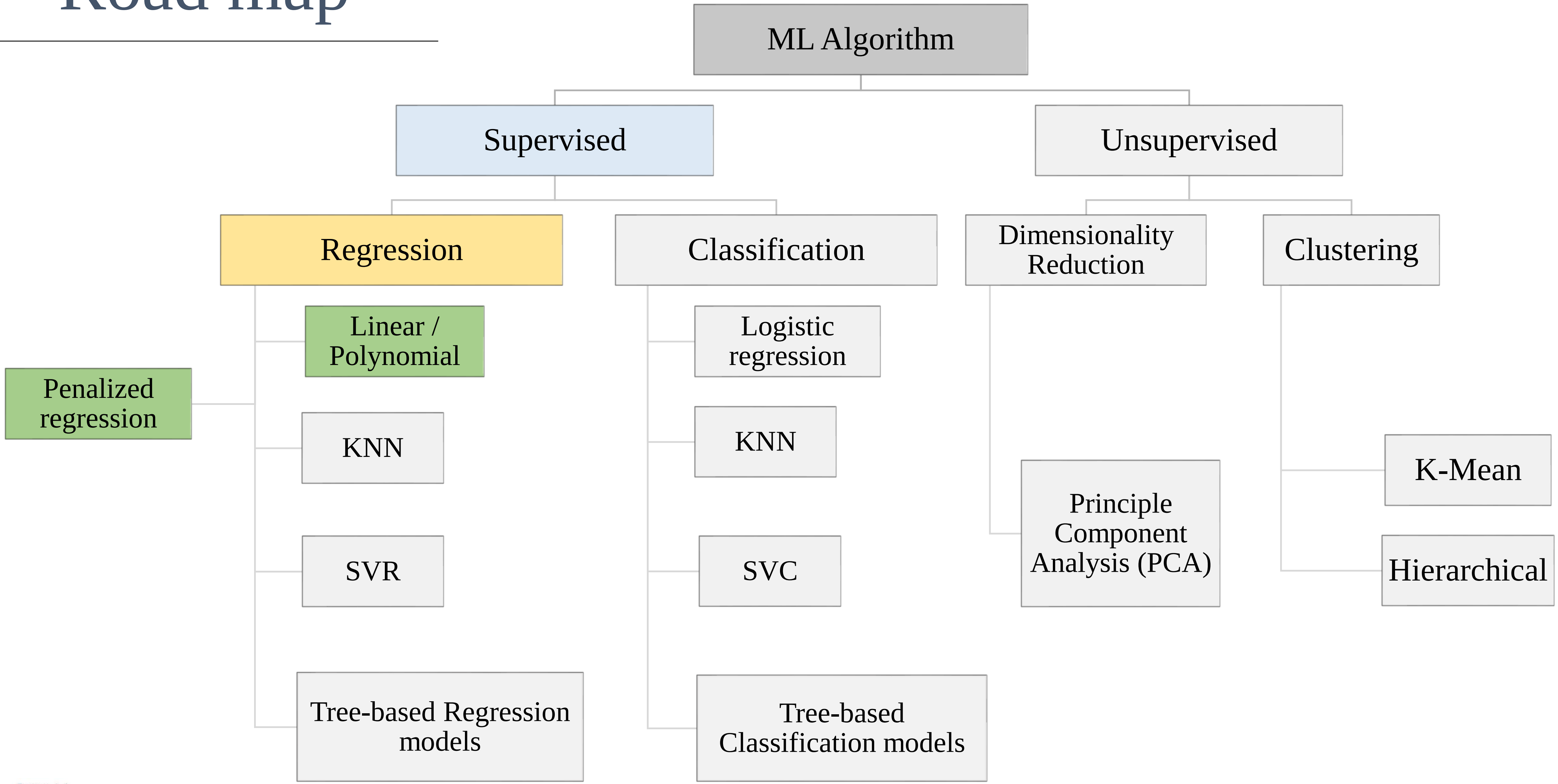
Class 8- Regularization
(Ridge, Lasso and Elastic Net)

Prof. Pedram Jahangiry

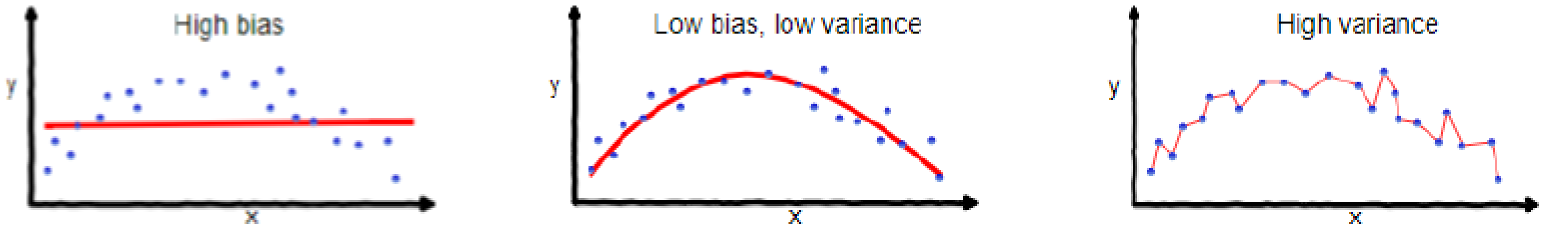




Road map



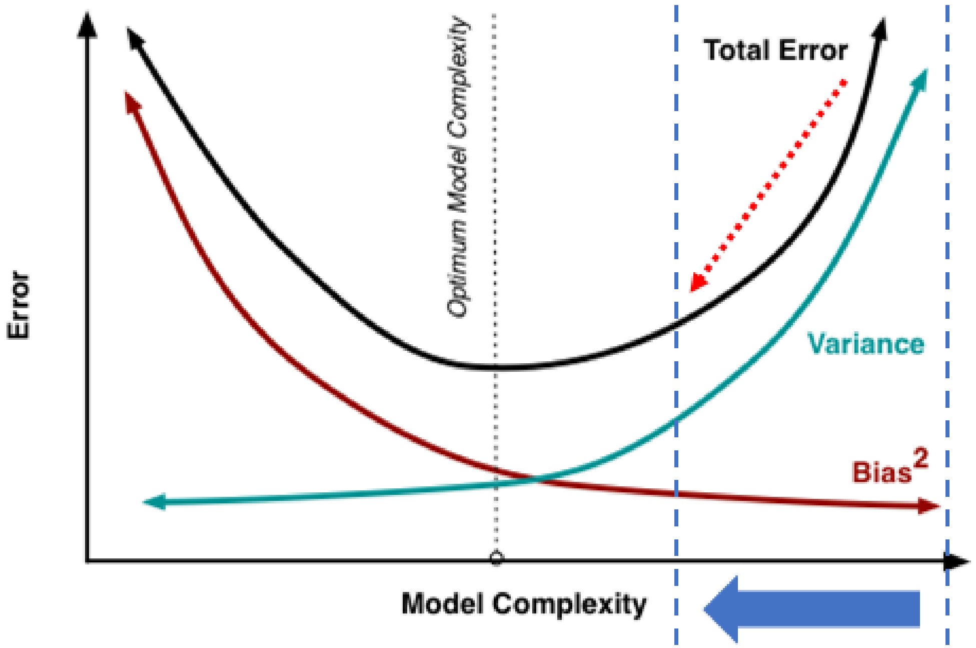
ML Algorithm
Supervised
Unsupervised
Regression
Classification
Dimensionality Reduction
Clustering
Linear / Polynomial
Penalized regression
KNN
SVR
Tree-based Regression models
Logistic regression
KNN
SVC
Tree-based Classification models
Principle Component Analysis (PCA)
K-Mean
Hierarchical



underfitting

Good balance

overfitting



Regularization / Penalized regression

Norms

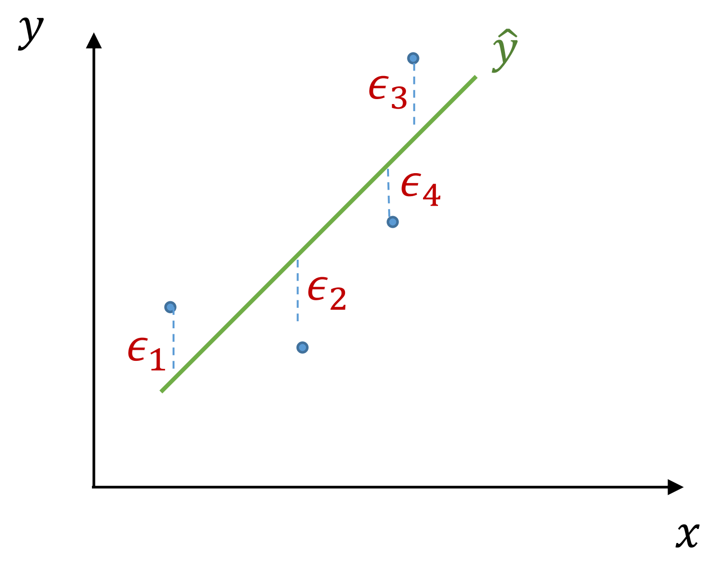
- In mathematics, the norm of a vector is its length.
- In regression analysis, to fit our linear model, we need a measure of mismatch!
- Our vector is error at each training data. We want to measure the length of error!

- L1 norm: Least absolute errors
  Manhattan norm

$$L^1 = \sum_i |\epsilon_i|$$

- L2 norm: Least squares
  Euclidean norm

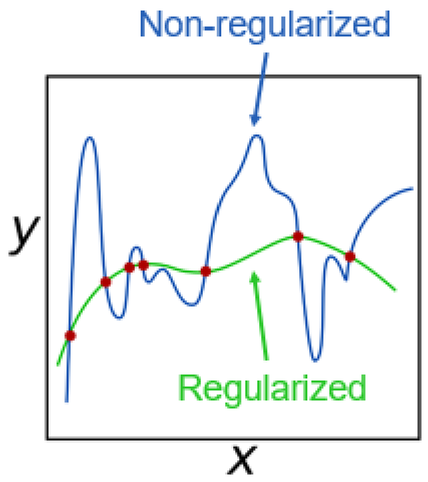
$$L^2 = \sum_i (\epsilon_i)^2$$



Regularization

- In machine learning there are often many features (usually correlated with each other). This can lead to overfitting and models that are unnecessarily complex.
- Regularization force the learning algorithm to build a less complex model. In practice, that often leads to slightly higher bias but significantly reduces the variance.

✓ The two most widely used types of regularization are called L1 and L2 regularization. The idea is quite simple. To create a regularized model, we modify the loss function by adding a penalizing term whose value is higher when the model is more complex.



$$Min_{w,b}\ (MSE + penalty) = Min\left[\frac{1}{N}\sum_{i=1}^{N}\left(y_i - f_{w,b}(X_i)\right)^2 + penalty(w)\right]$$

Penalized regression

$$Min_{w,b}\ (MSE + penalty) = Min\left[\frac{1}{N}\sum_{i=1}^{N}\left(y_i - f_{w,b}(X_i)\right)^2 + penalty(w)\right]$$

- Penalized regression is useful for reducing a large number of features to a manageable set and for making good predictions especially where features are correlated (i.e., when classical linear regression breaks down).
- Penalized regression can be used to avoid overfitting.
- To use the penalized regression, we need to first standardize the features. This will allow us to compare the magnitudes of regression coefficients for the feature variables.

1) Ridge regression
2) LASSO regression
3) Elastic Net regression

The only difference is in the penalty term

Part I

Ridge Regression

1) Ridge regression

$$Min_{w,b}\ (MSE + penalty) = Min\left[\frac{1}{N}\sum_{i=1}^{N}\left(y_i - f_{w,b}(X_i)\right)^2 + penalty(w)\right]$$

$$= Min\left[\frac{1}{N}\sum_{i=1}^{N}\left(y_i - f_{w,b}(X_i)\right)^2 + \lambda\sum_{j=1}^{D} w_j^2\right]$$

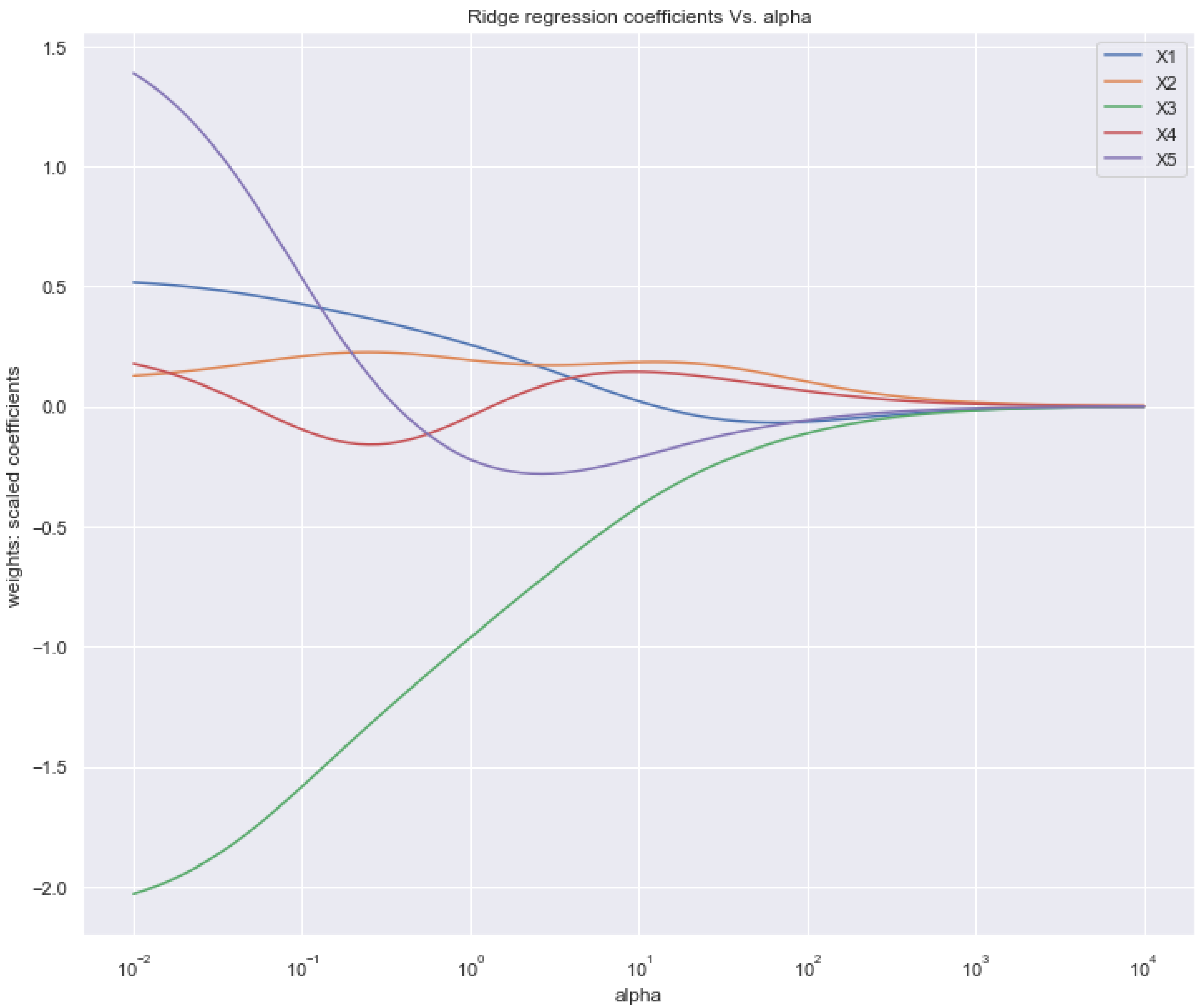
- Ridge regression uses L2 norm.
- The shrinkage penalty has the effect of shrinking the estimates of w_j towards zero.
- The tuning parameter λ serves to control the relative impact of the penalty term on the regression coefficient estimates.
- Selecting a good value for λ is critical; cross-validation is used for this.
- It is best to apply ridge regression after variable standardization.

The true model is:

$$y = f(x) = x + 2x^2 - 3x^3 + \epsilon$$

Imposed functional form:

$$\hat{y} = w_1 x + w_2 x^2 + w_3 x^3 + w_4 x^4 + w_5 x^5$$



Part II
LASSO Regression

2) LASSO regression

$$Min_{w,b}\,(MSE + penalty) = Min\left[\frac{1}{N}\sum_{i=1}^{N}\left(y_i - f_{w,b}(X_i)\right)^2 + penalty(w)\right]$$

$$= Min\left[\frac{1}{N}\sum_{i=1}^{N}\left(y_i - f_{w,b}(X_i)\right)^2 + \lambda\sum_{j=1}^{D}|w_j|\right]$$

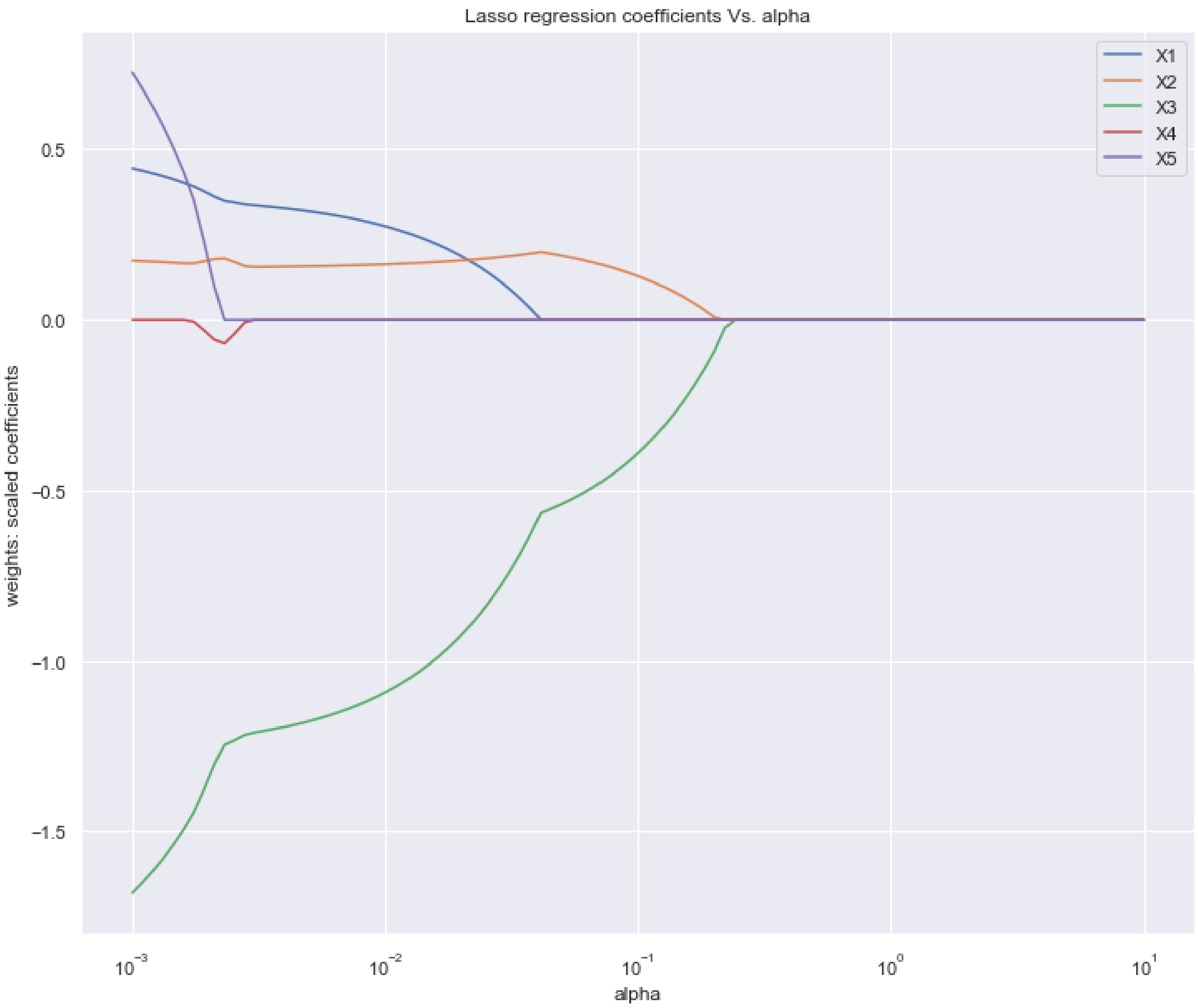
- LASSO stands for "Least Absolute Shrinkage and Selection Operator"
- LASSO regression uses L1 norm.
- LASSO eliminates the least important features from the model, it automatically performs a type of **feature selection**.
- Selecting a good value for λ is critical; cross-validation is used for this.
- It is best to apply LASSO regression after variable standardization.

The true model is:

$$y = f(x) = x + 2x^2 - 3x^3 + \epsilon$$

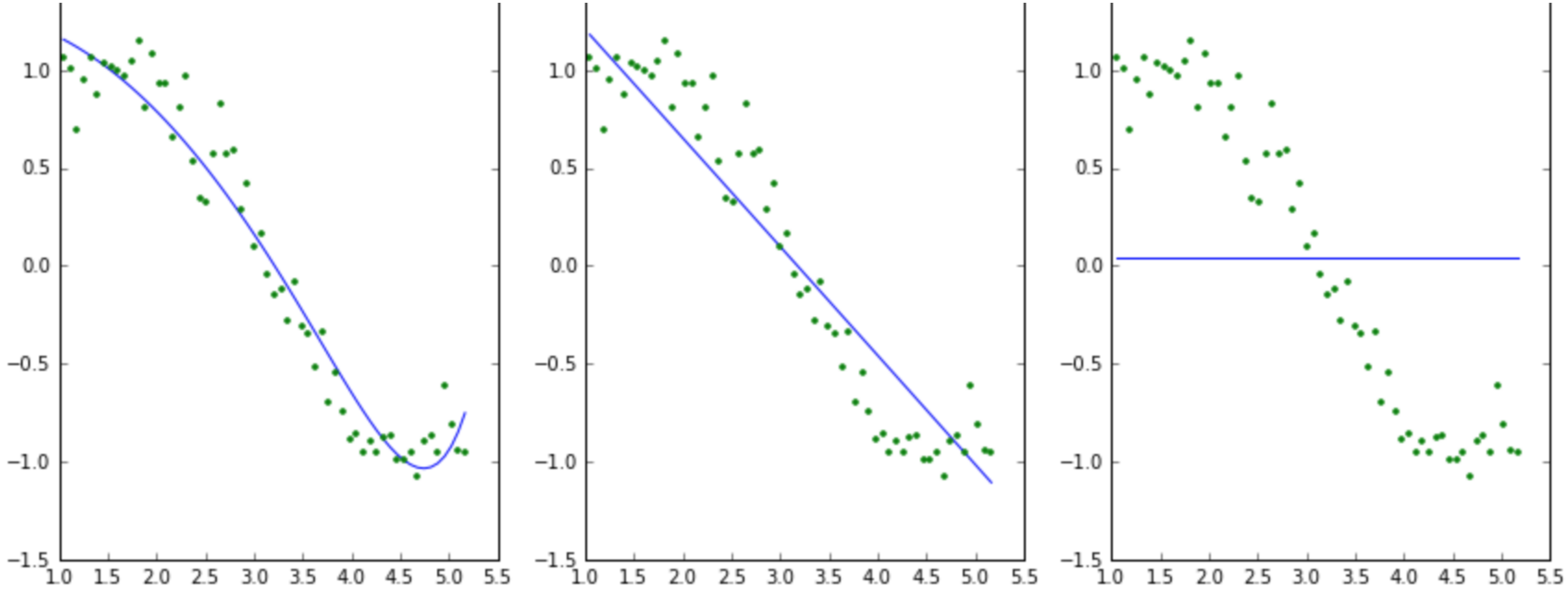
Imposed functional form:

$$\hat{y} = w_1 x + w_2 x^2 + w_3 x^3 + w_4 x^4 + w_5 x^5$$



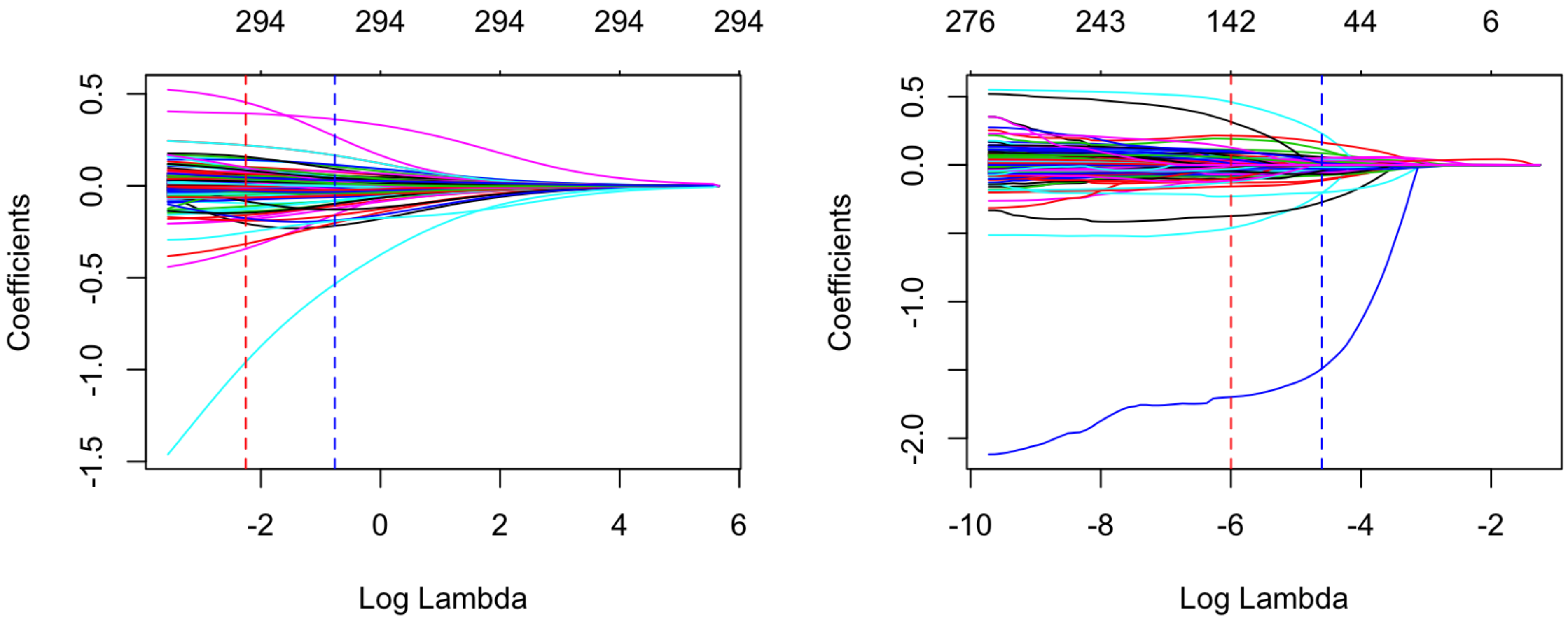
Ridge and LASSO vs Lambda

As λ increases, the model becomes simpler



Question of the day: Ridge vs LASSO?

?



Part III
Elastic Net Regression

3) Elastic Net Regression

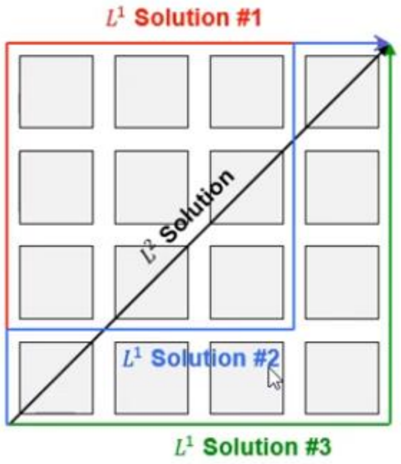
$$Min_{w,b}\,(MSE + penalty) = Min\left[\frac{1}{N}\sum_{i=1}^{N}\left(y_i - f_{w,b}(X_i)\right)^2 + penalty(w)\right]$$

$$= Min\left[\frac{1}{N}\sum_{i=1}^{N}\left(y_i - f_{w,b}(X_i)\right)^2 + \lambda_1\sum_{j=1}^{D}|w_j| + \lambda_2\sum_{j=1}^{D}w_j^2\right]$$

- In LASSO some weights are reduced to zero, but others may be quite large. In Ridge, weights are small in magnitude, but they are not reduced to zero.
- In Elastic Net, we may be able to get the best of both worlds by making some weights zero while reducing the magnitude of the others.

Ridge vs LASSO vs Elastic Net

Property	Ridge	LASSO	Elastic Net
Can shrink the coefficient estimate toward zero?			
Can include all the features in the model even with large λ ?			
Can force some of the coefficient estimates to be exactly = 0? Hence, can be used for feature selection? Or sparse output? More explainable?			
Is robust : resistant to outliers ?			
No Analytical solution i.e., requires gradient descent?			
Always unique solution?			



Appendix

LASSO vs Ridge, behind the scene? (optional)

Why is it that the lasso, unlike ridge regression, results in coefficient estimates that are exactly equal to zero?

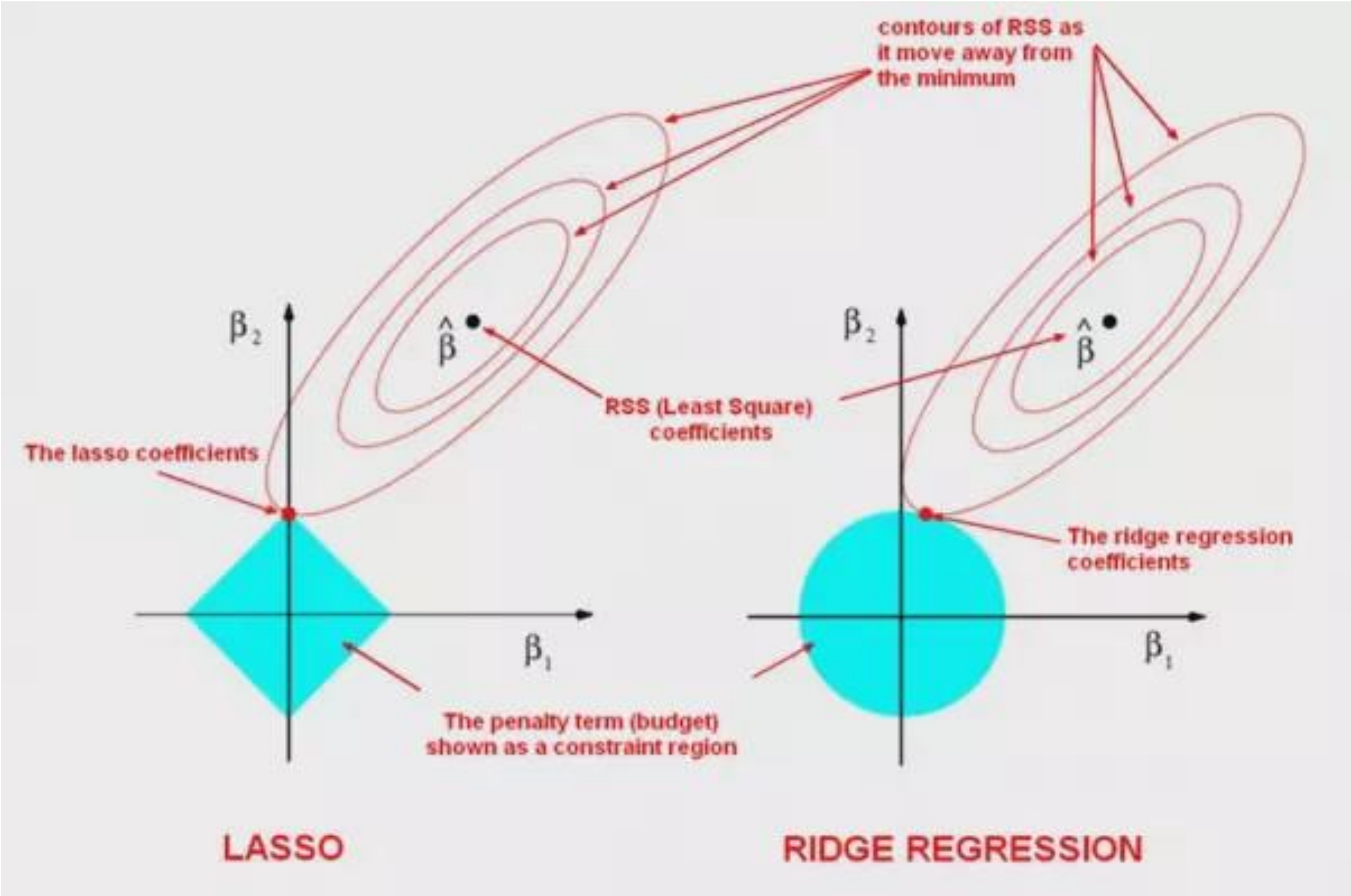
One can show that the lasso and ridge regression coefficient estimates solve the problems

$$\underset{\beta}{\text{minimize}} \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \quad \text{subject to} \quad \sum_{j=1}^{p} |\beta_j| \leq s$$

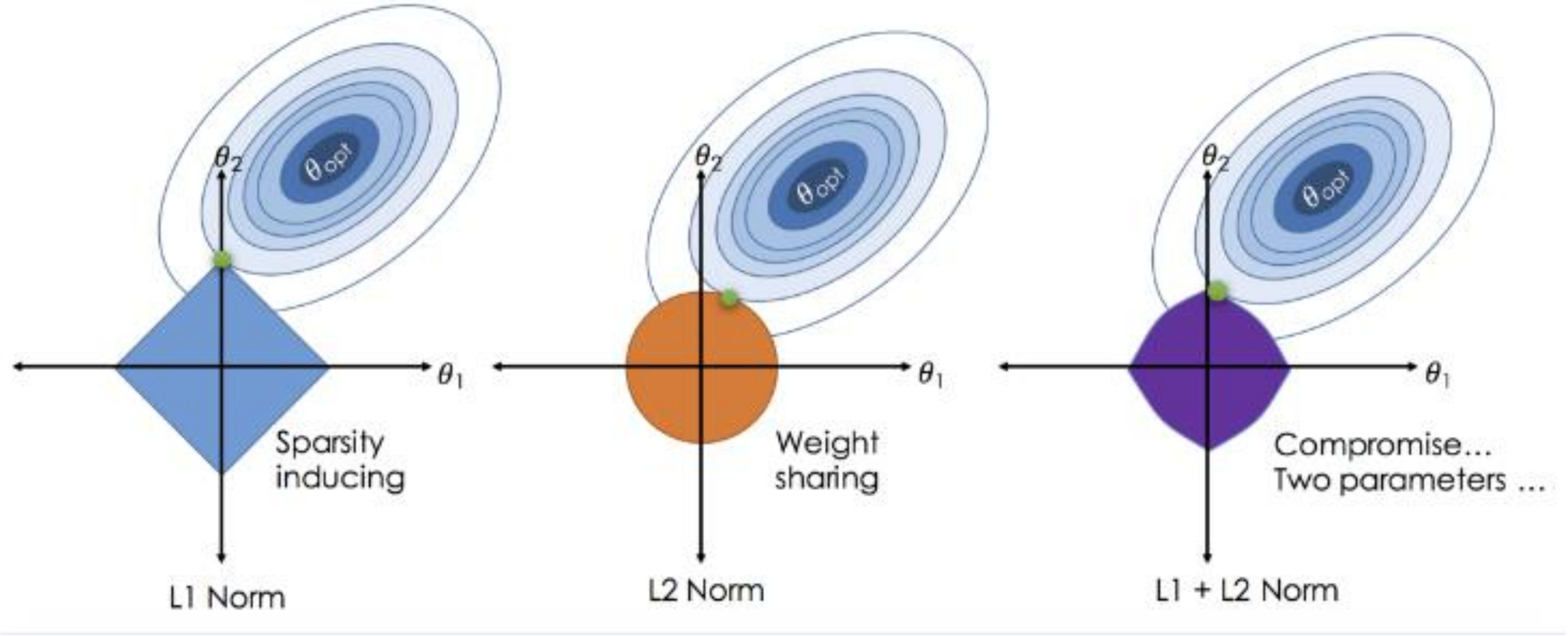
and

$$\underset{\beta}{\text{minimize}} \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \quad \text{subject to} \quad \sum_{j=1}^{p} \beta_j^2 \leq s,$$

LASSO vs Ridge, behind the scene? (optional)



LASSO vs Ridge vs Elastic Net, behind the scene? (optional)




thank you all
CLASS DISMISSED