Machine Learning for All

Session 5. Model Selection and Regularization

César E. Montiel-Olea (SPD)

February 22th, 2019

What do we know so far?

- In linear regression, we know that **adding predictors** to the model **always reduces the Sum of Square Residuals** (SSR) on the training data.

- Recall that adding predictors to the model does not necessarily improve the SSR on the testing data (testing error).

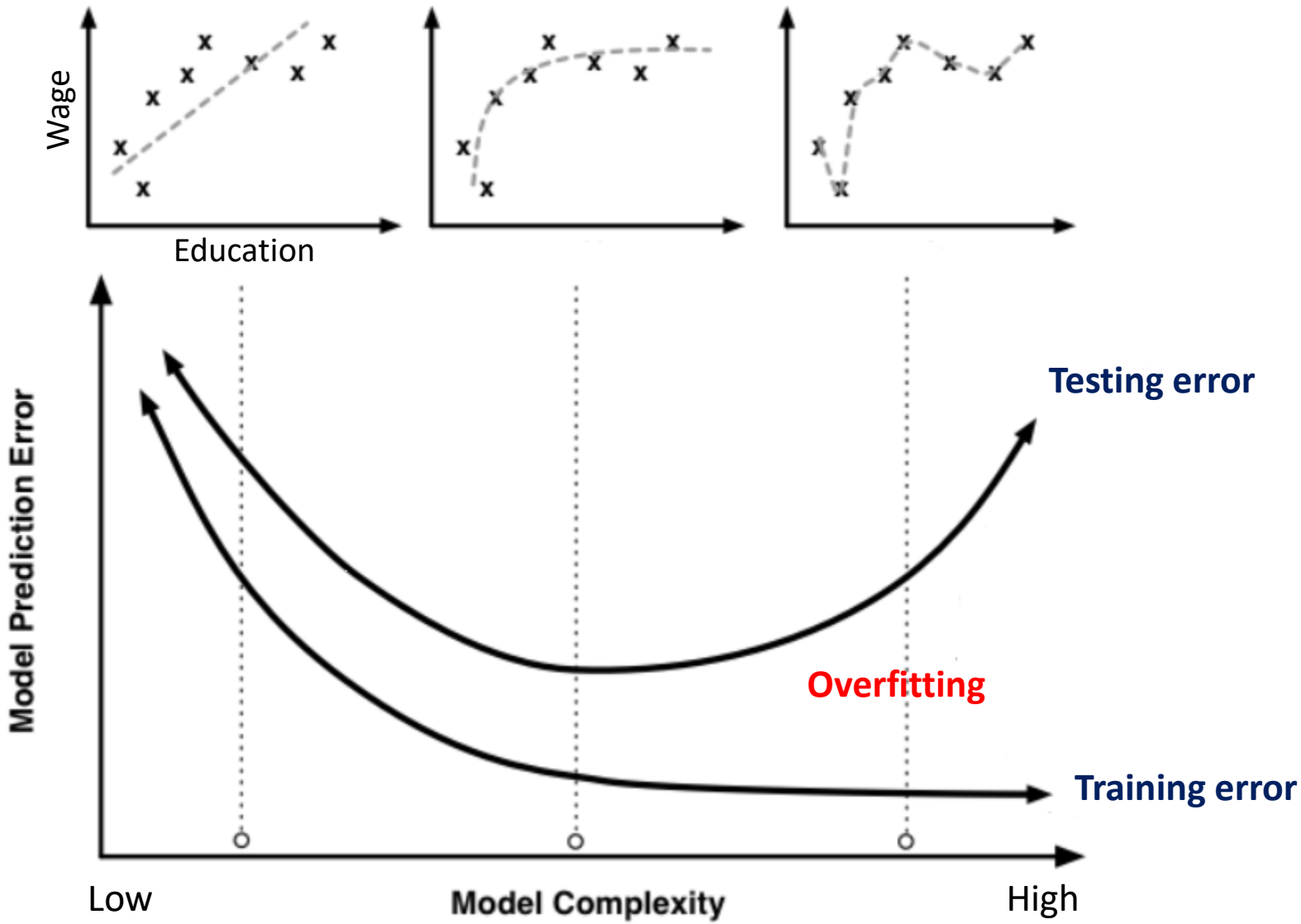
- In general, in supervised machine learning **we want to find a learning method that minimizes the testing error** not the training error.

What do we know so far?

- A **more flexible learning method** might fit the data better and in principle **decrease the testing error**.

- But fitting a **more complex model (more flexible)** requires estimating a greater number of parameters (we have less degrees of freedom).

- Therefore, **increasing flexibility** reduces the testing error but it can then **lead to overfitting** (increasing at some point the testing error again).


Wage
Education
Model Prediction Error
Testing error
Overfitting
Training error
Low
Model Complexity
High

How can we address overfitting?

1. **Subset selection methods**: We identify a subset of the predictors that might be related to the response. We then fit a model using least squares on the reduced set of variables.

2. **Regularization methods**: Using of all our predictors, we use a method that allows us to **regularize or constrain** the coefficient estimates **towards zero** (LASSO) or **close to zero** (Ridge).

Subset Selection

- Best subset selection: We compare all models with different number of predictors

Example:

Y = wage
$x_1 = \text{years of education}$
$x_2 = \text{years of experience}$

We want to predict wage using years of experience and years of education.

How many linear models can we fit?

Use least squares for all possible combinations

Let's see how many models I can fit:

1. {Constant}
2. {Constant + years of education}
3. {Constant + years of experience}
4. {Constant + years of education + years of experience}

We can fit $2^2 = 4$ models

Problem

- How many models can we fit for 35 covariates?

$$\text{We can fit } 2^{35} = 34{,}359{,}738{,}368 \text{ models}$$

- In fact, the best subset selection becomes infeasible when we have more tan 40 covariates.

- The problem is that many data structures have tons of covariates.

So we must find another way to select fewer predictors

Subset Selection

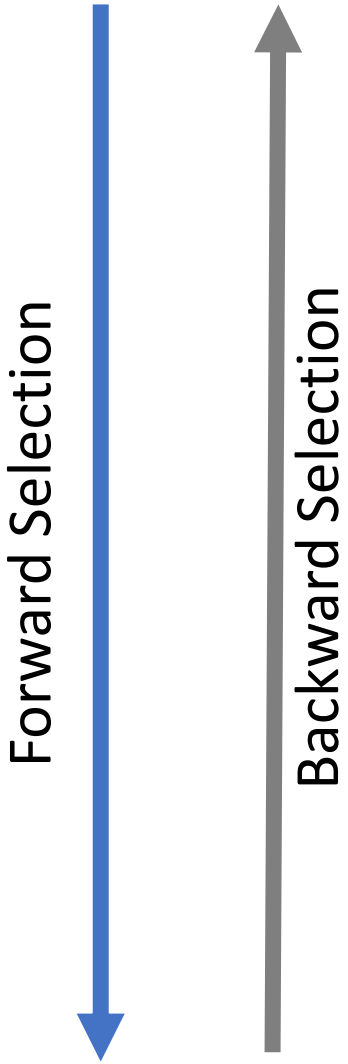
Stepwise Selection:

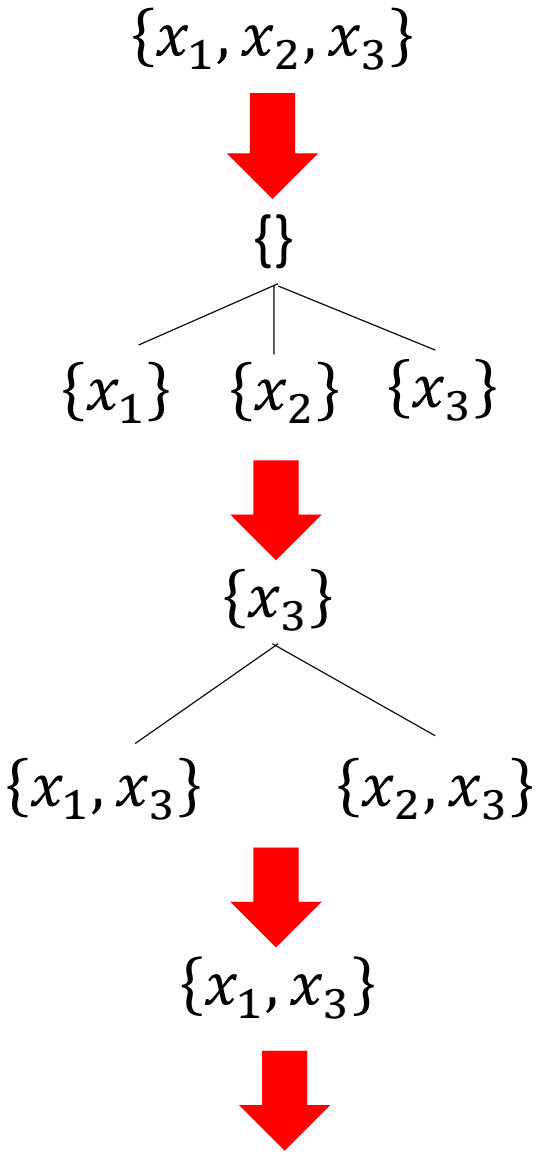
- More efficient alternative to best subset selection
- We identify a subset of the d predictors and then fit a model using least squares on the reduced set of variables.

- The idea is the following:

1. Construct a sequence of *n* models with different number of variables
2. Select the best among them using some metric that we choose

Variants of Stepwise Approach





Original set of variables

First we start with a null model

Then we consider all the possible models with one variable

We choose the best (lowest SSR) one-variable model

Now we add the second predictor from all possible combinations

We choose the best (lowest SSR) two-variable model

We keep doing this until we finish with our predictors

Different methods to choose the best model

- Sum of Squares Residual (SSR)
- Adjusted R-Squared
- Akaike Information Criteria (AIC)
- Bayes Information Criteria (BIC)

Let’s do some code!

Issues with Least Squares

In least squares (OLS):

- When the number of observations is not much larger than the number of predictors ($\boldsymbol{n \approx d}$), we will have **a lot of variability** in the least squares resulting in **overfitting**.

- When the number of observations is less than the number of predictors (that is **when** $\boldsymbol{n < d}$), we **cannot use this method anymore.**

Issues with Least Squares

- That least squares method works only if:

$$\hat{\beta} = \underbrace{(\mathbf{X}^T\mathbf{X})}^{-1}\ \mathbf{X}^T\ \boldsymbol{y} \quad \text{if} \quad n \geq d$$

Not singular

- If $n < d$, then $\mathbf{X}^{\mathrm{T}}\mathbf{X}$ will be singular (matrix which is not invertible) and least squares will yield an infinite number of solutions.

- If we use this model to predict an outcome in the testing data, the variance will be infinite.

Ridge Regression

In Ridge regression we have to minimize:

$$\min_{\beta} \sum_{i=1}^{n} (y_i - f(\boldsymbol{x}_i; \boldsymbol{\beta}))^2 + \lambda \sum_{j=1}^{p} \beta_j^2$$

In blue, we have the regular SSR of the model

In red we have a penalty term: $\boldsymbol{\lambda}$ is a tuning parameter and makes $\mathbf{X}^{\mathrm{T}}\mathbf{X}$ invertible (we regularize that matrix)

Regularization

- The tuning parameter λ shrinks our coefficients to be close to zero

  If $\lambda \downarrow 0 \rightarrow$we obtain the least squares solution
  If $\lambda \uparrow \infty \rightarrow$ coefficients start to shrink towards zero until we just have an intercept

- We will have different coefficient estimates for different values of λ.

- Furthermore, λ helps to control the variance of the prediction error.

- λ controls the amount of regularization $\rightarrow$ we need a discipline way to select λ.

Implementation

1. In practice, what do we do?

   - We **scale each variable** such that it has sample variance 1 before running the regression.

   - Scaling prevents penalizing some coefficients more than others.

2. How do we choose the tuning parameter λ?

   - We find **an estimate $\boldsymbol{\beta_{\lambda}^{ridge}}$ for many values of $\boldsymbol{\lambda}$** and then choose it by cross-validation.

The LASSO

In LASSO regression we minimize:

$$\min_{\beta} \sum_{i=1}^{n} (y_i - f(\boldsymbol{x}_i; \boldsymbol{\beta}))^2 + \lambda \sum_{j=1}^{p} |\beta_j|$$

In blue, we have the regular SSR of the model.

In red we have the penalty term

Features of LASSO

- LASSO: Least Absolute Shrinkage and Selection Operator
- Again, we have a tuning parameter λ which controls which coefficients enter the model.
- Unlike Ridge regression, for a large enough λ the solution will set some coefficients exactly equal to zero (sparse regression).

The LASSO will perform model selection for us!

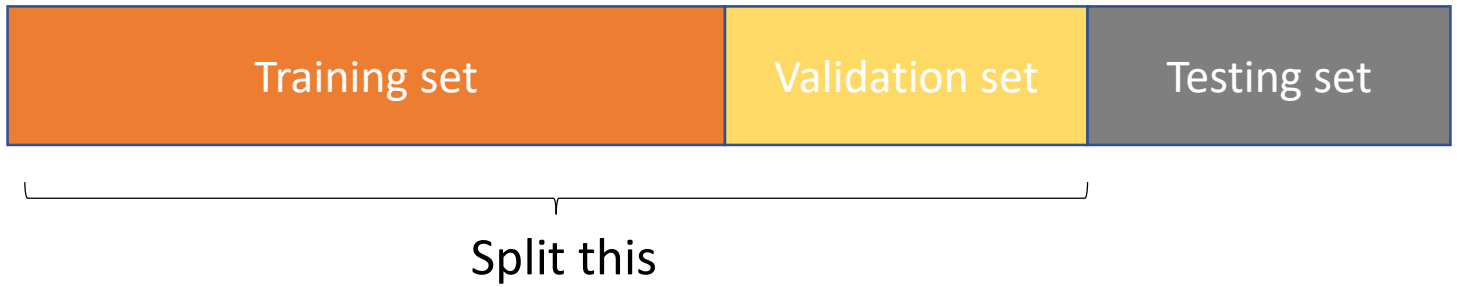
LASSO vs. Ridge

Why choosing LASSO instead of Ridge?

- Ridge regression shrinks all the coefficients to non-zero values.
- Alternative to subset selection or stepwise approach.
- Which one yields the lowest variance and MSE depends on the selection of the tuning parameter λ.

Cross-Validation

- To choose the optimal value of the tuning parameter λ^* we use a cross-validation procedure:



- Goal:
  1. Estimate testing error of a learning method
  2. Select the best model from a given set of models. In this case, "set of models" means models with different parameter values for λ

Model selection by cross-validation

We consider different parameter values of $\lambda_1 \ldots \lambda_n$

Model selection

1) For different values of λ_i, we train our algorithm on the training set
2) Use the validation set to compute SSR
3) Select the value λ^* with lowest SSR
4) Re-train the algorithm with parameter λ^* on data (training + validation data) except the testing set

Model assessment

5) Estimate SSR in testing data using optimal model from step 4)

Let’s do some code!

K-fold Cross-Validation

Strategy:

- Randomly divide the training set into K blocks or folds and use one block as a validation set at a time (e.g. K = 5)

Validation	Train	Train	Train	Train
Train	Validation	Train	Train	Train
⋮	⋮	⋮	⋮	⋮
Train	Train	Train	Train	Validation

We change our validation at each turn until reaching the fifth fold

- Train model on k - 1 folds (with a different value of the parameter) and then compute prediction error on fold k.
- We average the results over all K combinations, and choose best λ^*

Thank you!