Lecture 14:
Overfit & Cross Validation

Big Data and Machine Learning for Applied Economics
Econ 4676

Ignacio Sarmiento-Barbieri

Universidad de los Andes

September 24, 2020

Announcements & Recap

- Final Project
  - First deadline. Sept 25. Brief zoom hang out, short presentation (5 slides tops). Present idea and basic plan. soft deadline
  - Second deadline. October 25. Show data. hard deadline
  - Final work, December 17. Bonus for "complete papers" hard deadline
- Details on Spatial Lag Model
  - MLE
  - 2SLS

Agenda

1. Motivation
2. Overfit
   - Overfit and out of Sample Prediction
3. Resampling Methods
   - Validation Set Approach
   - LOOCV
   - K-fold Cross-Validation
4. Further Readings

Motivation: Back to Basics

Complexity and the variance/bias trade off

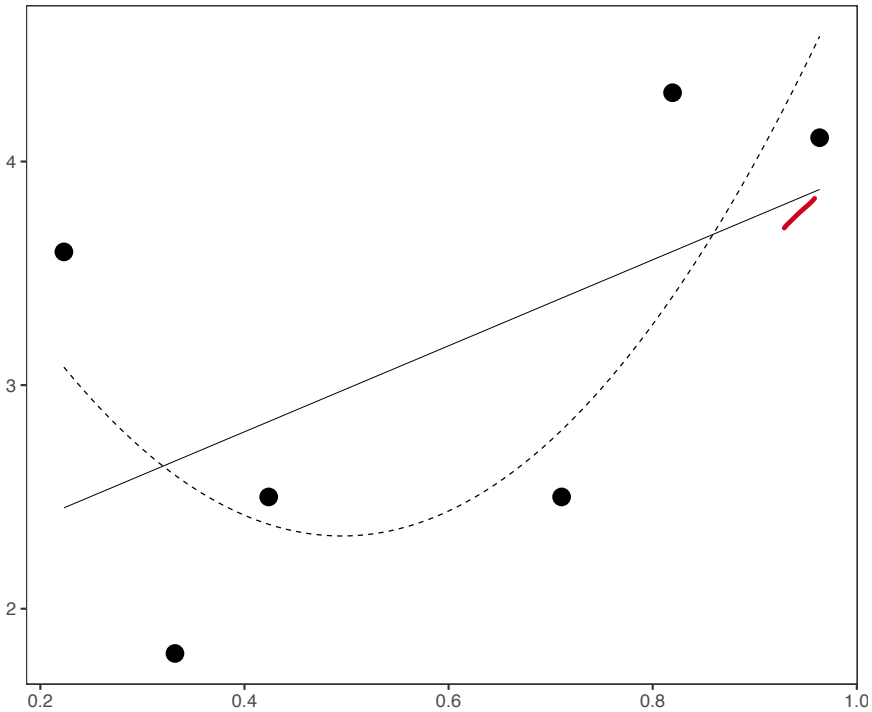
- When the focus switches from estimating f to predicting Y,
- f plays a secondary role, as just a tool to improve the prediction based on X.
- Predicting Y involves *learning* f, that is, f is no longer taken as given, as in the classical view.
- Now it implies an iterative process where initial choices for f are revised in light of potential improvements in predictive performance.
- Model choice or learning involves choosing both f and a strategy to estimate it ($\hat{f}$), guided by predictive performance.

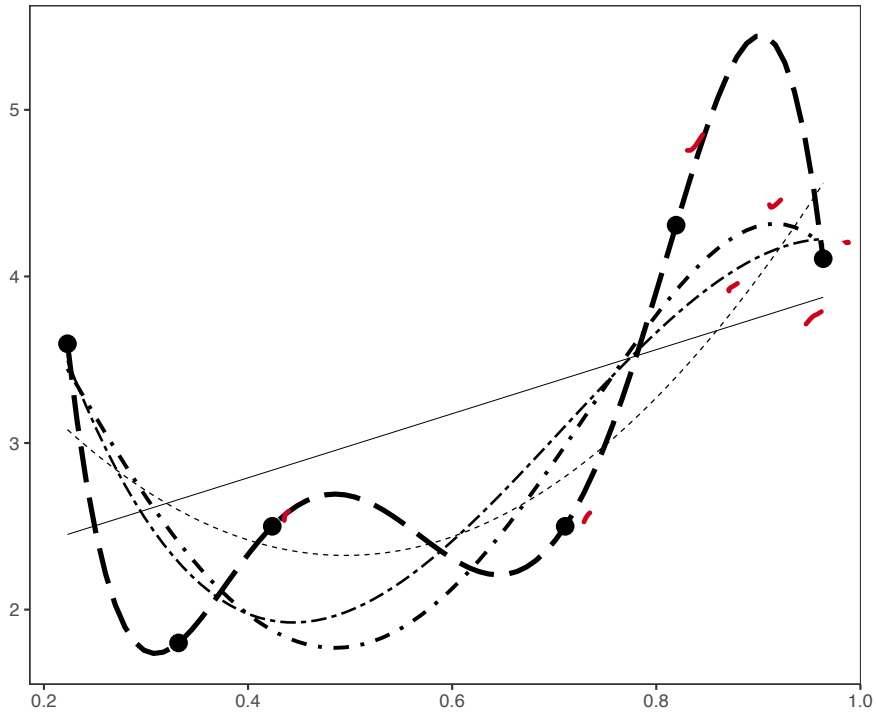
Complexity and the variance/bias trade off

$$Y = \beta_1 X_1 + \beta_2 X_2 + u_2 \tag{1}$$

- Omitting relevant variables, vs including irrelevant variables
- Trade off: more *complex* models tend to have less bias but higher variance
- Complexity: number of variables?
- Classical econometrics: first unbiasedness
- ML: Accept some bias to lower variance

Overfit





- $y = \alpha + \beta x + \varepsilon$
- $y = \alpha + \beta_1 x + \beta_2 x^2 + \varepsilon$

grado 5 $n-1$
hay 6 datos

Overfit

- Suppose that the true model is $y = f(X) + u$ where f is a polynomial of degree p^*, with $E(u) = 0$ and $V(u) = \sigma^2$
- p^* is finite but unknown
- We fit polynomials with increasing degrees $p = 1, 2, ...$
- What happens when we increase the degree of the polynomial?
- The expected prediction error of a regression fit $\hat{f}(X)$ at an input point $X = x_0$, is

$$\begin{aligned} Err(x_0) &= Err(y - \hat{f}(x_0)|X = x_0) \\ &= Bias^2(f, \hat{f}(x_0)) + V(\hat{f}(x_0)) + Irreducible\ Error \end{aligned} \tag{2}$$

- The average expected prediction error

$$\frac{1}{N}\sum_{i=1}^{N} Err(x_i) \tag{3}$$

Overfit

- Bias ?
- Variance:

$$\hat{f}(x_0) = \sum_{s=0}^{p} x_0^s \hat{\beta}_s = x_0' \hat{\beta} \tag{4}$$

where $x_0' = (1, x_0, x_0^2, \ldots, x_0^p)$

$$V(\hat{f}(x_0)) = V(x_0' \hat{\beta}) = x_0' \sigma^2 (X'X)^{-1} x_0 \tag{5}$$

Then

$$\frac{1}{n} \sum_{i=1}^{n} x_i' \sigma^2 (X'X)^{-1} x_i = \sigma^2 \frac{p}{n} \tag{6}$$

After we "hit" p^* increasing complexity does not reduce the bias, but variance increases monotonically for σ^2 and n given

Proof

The fitted model for a polynomial of degree p is :

$$\hat{y}_i = x_i'\hat{\beta} \tag{7}$$

with $x_i' = (1, x_i, , x_i^2, \ldots, x_i^p)$ Then $V(y_i) = V(x_i'\hat{\beta}) = \sigma^2 x_i'(X'X)^{-1}x_i$.
Now:

$$Average\, V(x_i'\hat{\beta}) = \frac{1}{n}\sum_{i=1}^{n} \sigma^2 x_i'(X'X)^{-1}x_i \tag{8}$$

Proof

- Trace.
  - If $A_{m\times m}$ with typical element a_{ij}. The **trace** of A, $tr(A)$ is the sum of the elements of its diagonal: $tr(A) \equiv \sum_{s=1}^{m} a_{ii}$
  - Properties
    - For any square matrices A, B, and C: $tr(A+B) = tr(A) + tr(B)$
    - Cyclic property: $tr(ABC) = tr(BCA) = tr(CAB)$
    - If $m = 1$ tr(A)=A

Now we use traces. Note that $x_i'(X'X)^{-1}x_i$ is a scalar, using the third property of traces

$$Average\, V(x_i'\hat{\beta}) = \frac{1}{n}\sum_{i=1}^{n} \sigma^2 x_i'(X'X)^{-1}x_i \quad (9)$$

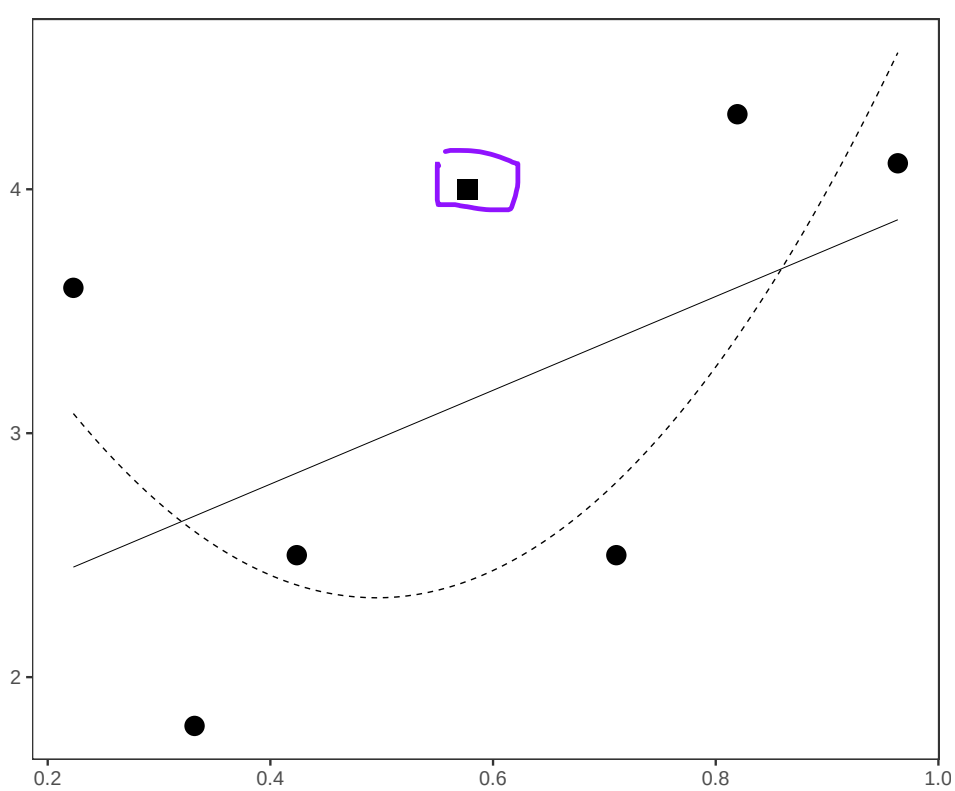
Now, using the cyclic property, $tr(x_i'(X'X)^{-1}x_i) = tr((X'X)^{-1}x_i'x_i)$, and using the first property of traces, we get

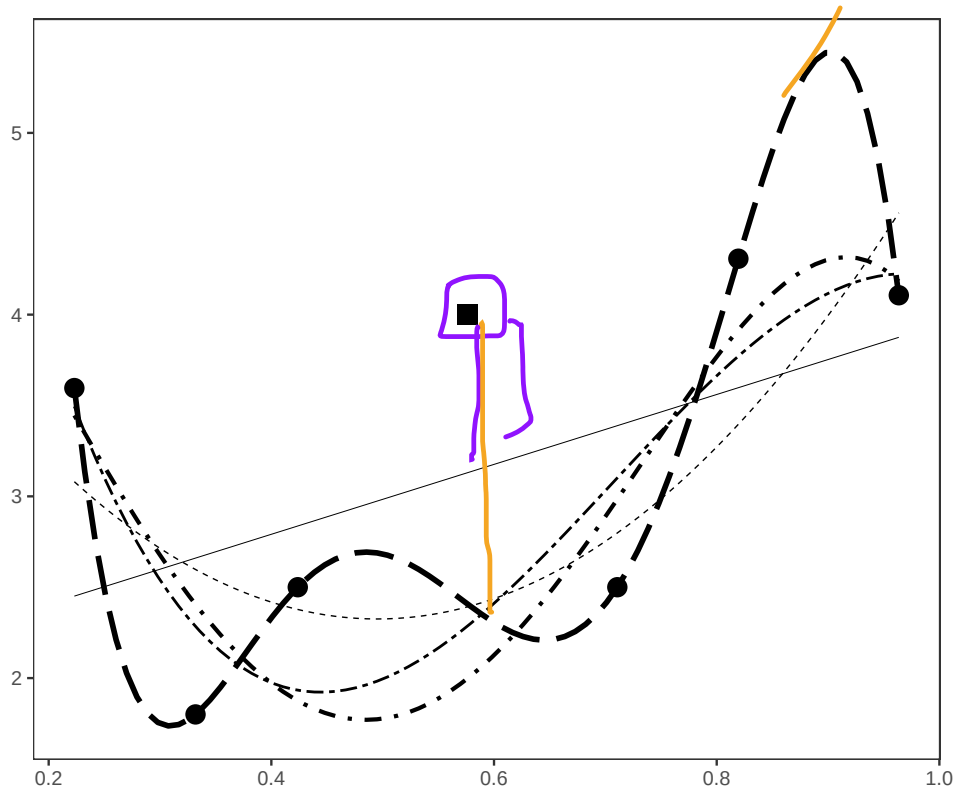
$$\sum_{i=1}^{n} tr((X'X)^{-1}x_i'x_i) = tr(\sum_{i=1}^{n}(X'X)^{-1}x_i'x_i) = tr((X'X)^{-1}(X'X)) = p \quad (10)$$

Overfit and out of Sample Prediction

- ML we care about prediction out of sample
- Overfit: complex models predict very well inside a sample but ”bad” outside
- Choose the right complexity level
- How do we measure the out of sample error?
- R^2 doesn't work: measures prediction in sample, it's non decreasing in complexity (PS1)

Overfit and out of Sample Prediction





Overfit and out of Sample Prediction

- Recall from Lecture 2 we defined loss and risk functions
  - Squared Error Loss $L(y, \hat{y}) = (y - \hat{y})^2$
  - Risk function $E(L(y, \hat{y}))$ that we also call expected prediction error (or sample counterpart average prediction error)
- Now we introduce formally two new/old concepts
  - *Test Error*: is the prediction error in a test sample

$$Err_{\mathcal{T}} = E[L(Y, \hat{Y})|\mathcal{T}] \tag{11}$$

  - *Training error*: is the average loss over the training sample

$$\bar{err} = \frac{1}{N} \sum_{i=1}^{N} L(y_i, \hat{y}_i) \tag{12}$$

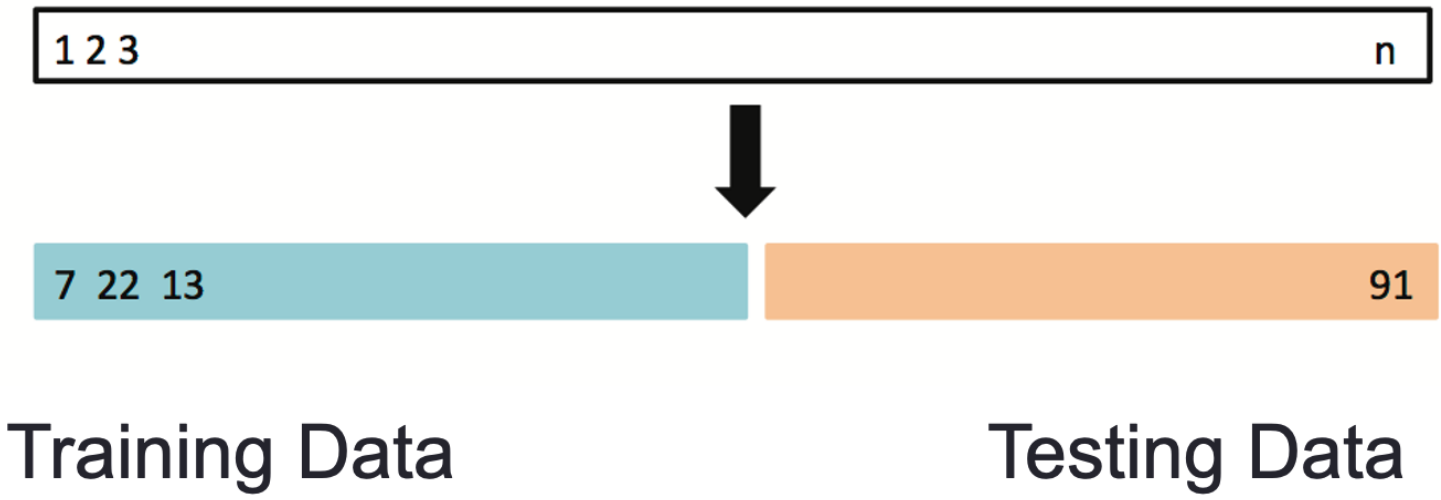
- Then how do we choose $\mathcal{T}$?

What are resampling methods?

- Tools that involves repeatedly drawing samples from a training set and refitting a model of interest on each sample in order to obtain more information about the fitted model
- Model Assessment: estimate test error rates
- Model Selection: select the appropriate level of model flexibility
- They are computationally expensive! But these days we have powerful computers

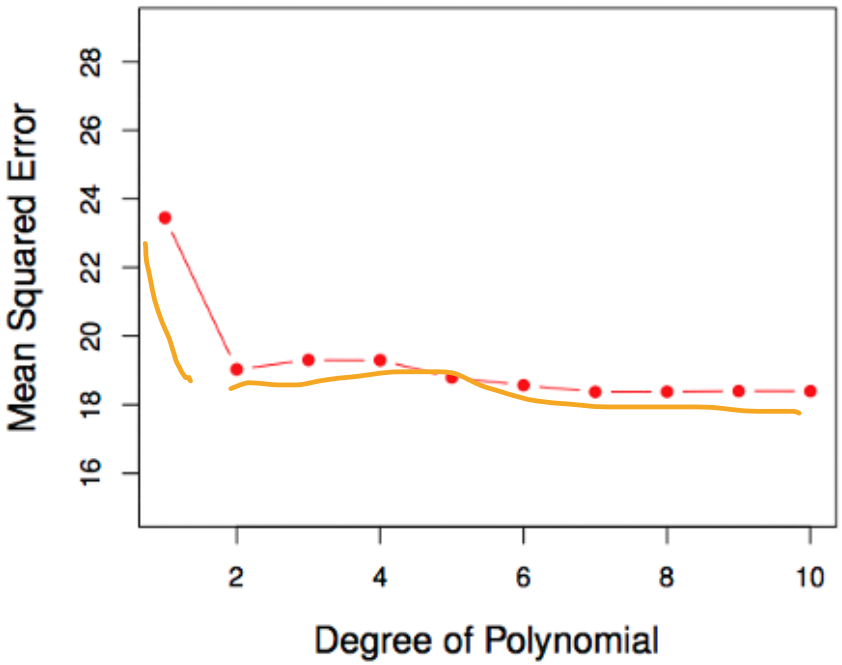
The Validation Set Approach

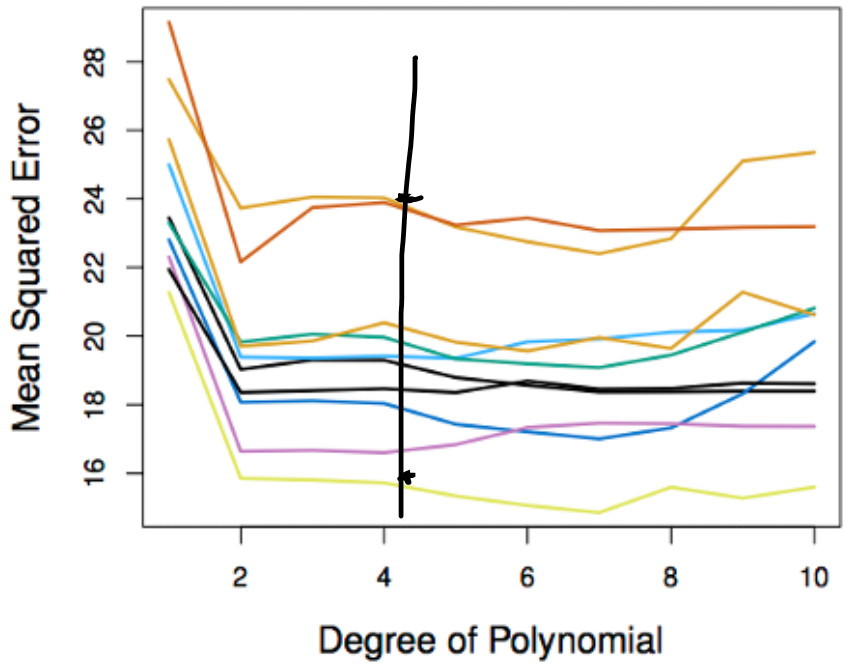
- Suppose that we would like to find a set of variables that give the lowest test (not training) error rate
- If we have a large data set, we can achieve this goal by randomly splitting the data into training and validation(testing) parts
- We would then use the training part to build each possible model (i.e. the different combinations of variables) and choose the model that gave the lowest error rate when applied to the validation data



The Validation Set Approach

- Model $y = f(x) + u$ where f is a polynomial of degree p^*.
- Left: Validation error rate for a single split
- Right: Validation method repeated 10 times, each time the split is done randomly!
- There is a lot of variability among the MSE's... Not good! We need more stable methods!



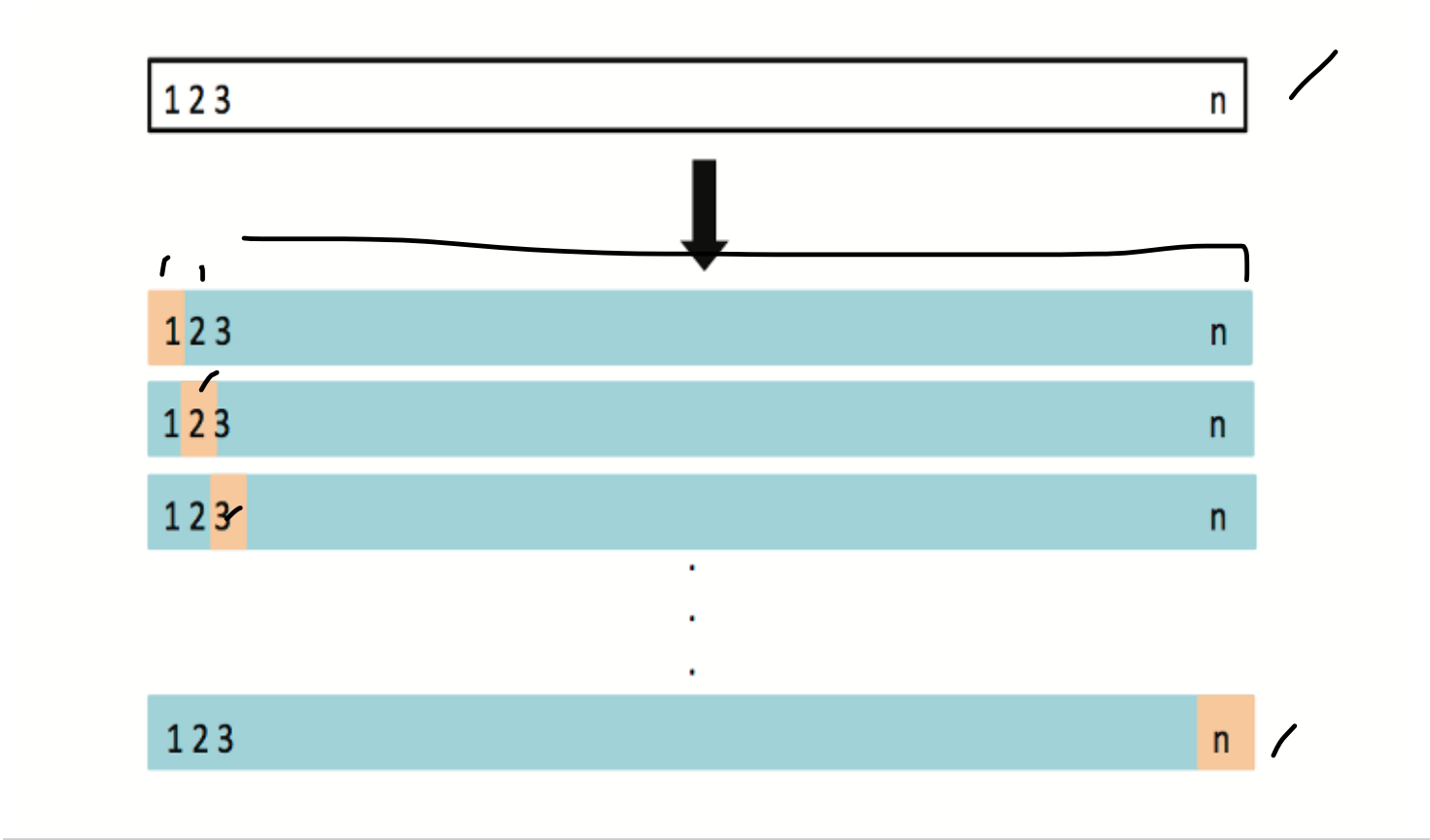


The Validation Set Approach

- Advantages:
  - Simple
  - Easy to implement
- Disadvantages:
  - The validation MSE can be highly variable
  - Only a subset of observations are used to fit the model (training data). Statistical methods tend to perform worse when trained on fewer observations

Leave-One-Out Cross Validation (LOOCV)

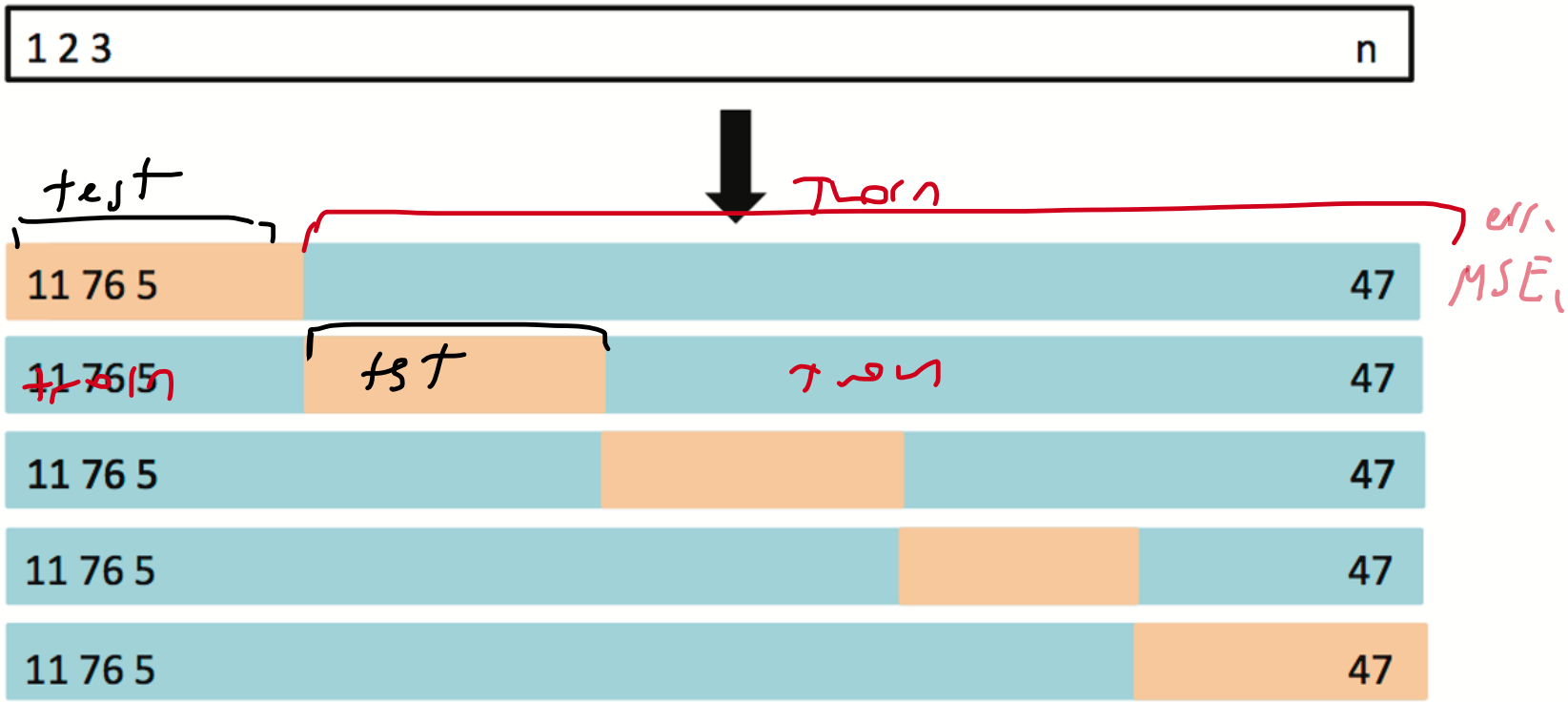
- This method is similar to the Validation Set Approach, but it tries to address the latter's disadvantages



K-fold Cross-Validation

ISLR 5

- LOOCV is computationally intensive, so we can run k-fold Cross Validation instead



K-fold Cross-Validation

- Split the data into K parts ($N = \sum_{j=1}^{K} n_j$)
- Fit the model leaving out one of the folds $\rightarrow f_{-k}(x)$
- Calculate the prediction error in the left out fold

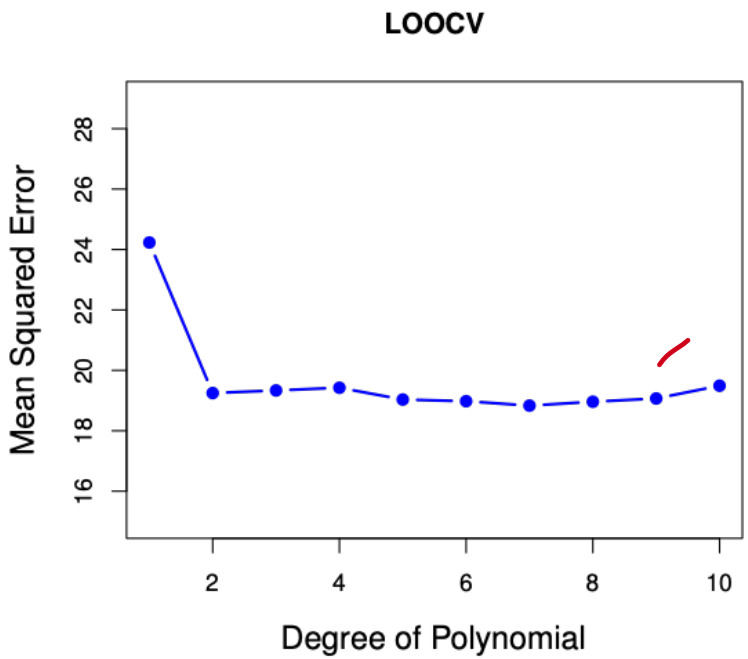
$$\bar{err}_j = MSE_j = \frac{1}{n_j} \sum L(y_j, \hat{y}_{-j}) \tag{13}$$

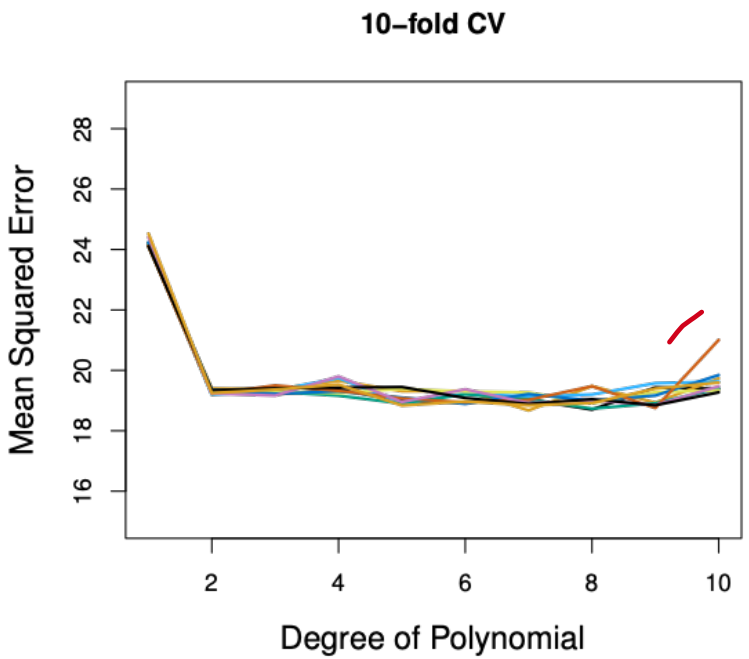
- Average these out

$$CV_{(k)} = \frac{1}{k} \sum_{j=1}^{k} \bar{err}_j = \frac{1}{k} \sum_{j=1}^{k} MSE_j \tag{14}$$

K-fold Cross-Validation

- Left: LOOCV error curve
- Right: 10-fold CV was run many times, and the figure shows the slightly different CV error rates
- LOOCV is a special case of k-fold, where k = n
- They are both stable, but LOOCV (generally) is more computationally intensive!

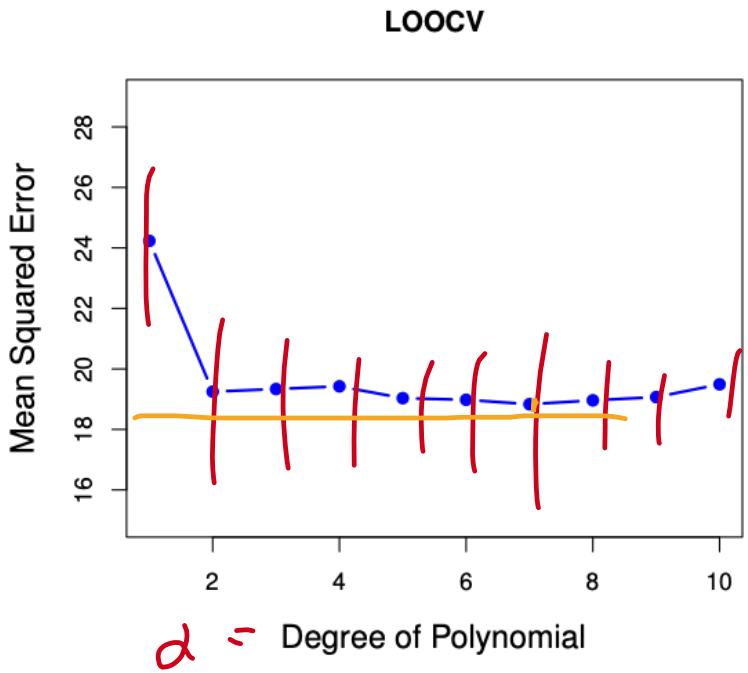


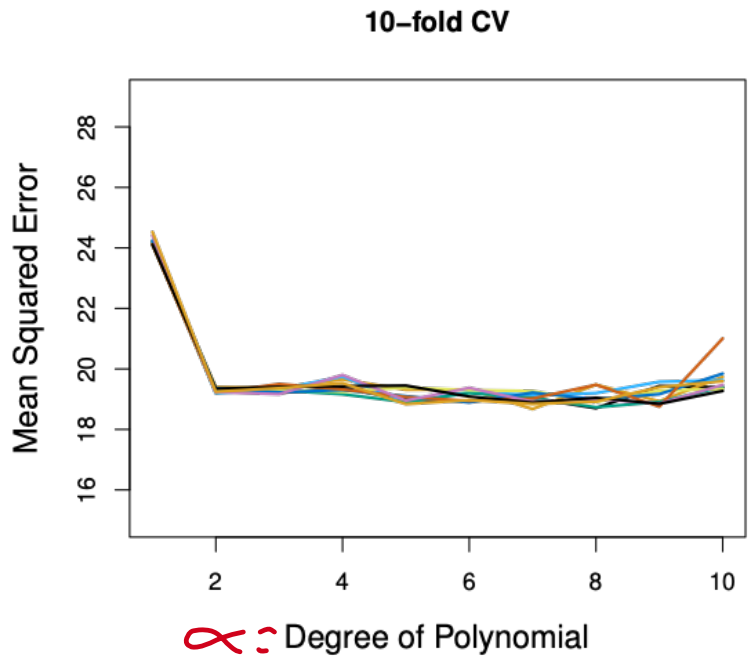


K-fold Cross-Validation for Model Selection

- Let's suppose that α parametrizes the complexity of a model
- In our examples α would be the degree of the polynomial
- First we compute the CV over an grid of α, and then choose the minimum

$$\min_{\alpha} CV_{(k)}(\alpha) \tag{15}$$





Bias- Variance Trade-off for k-fold CV

- Bias:
  - Validation set approach tends to overestimate the test error set (less data, worst fit)
  - LOOCV, adds more data $\rightarrow$ less bias of the test error
  - K-fold an intermediate state
- Variance:
  - LOOCV we average the outputs of n fitted models, each is trained in almost identical set of observations $\rightarrow$ highly (positively) correlated
  - K-fold this correlation is smaller, we are averaging the output of k fitted model that are somewhat less correlated
- Thus, there is a trade-off between what to use
  - We tend to use k-fold CV with (K = 5 and K = 10)
  - It has been empirically shown that they yield test error rate estimates that suffer neither from excessively high bias, nor from very high variance Kohavi (1995)

Spatial K-fold Cross-Validation

- 'First law' of geography states that points close to each other are, generally, more similar than points further away
- Points are not statistically independent because training and test points in conventional CV are often too close to each other
- To alleviate this problem 'spatial partitioning' is used to split the observations into spatially disjointed subsets

Review & Next Steps

- Today:
  - Overfit and out of Sample Prediction
  - Resampling Methods
    - Validation Set Approach
    - LOOCV
    - K-fold Cross-Validation
- Next class: Model selection and Regularization
- Questions? Questions about software?

Further Readings

- Friedman, J., Hastie, T., & Tibshirani, R. (2001). The elements of statistical learning (Vol. 1, No. 10). New York: Springer series in statistics.
- James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An introduction to statistical learning (Vol. 112, p. 18). New York: springer.
- Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. In Ijcai (Vol. 14, No. 2, pp. 1137-1145).
- Lovelace, R., Nowosad, J., & Muenchow, J. (2019). Geocomputation with R. CRC Press. (Chapters 2 & 6)