# Prediction and Linear Regression

## Big Data y Machine Learning para Economía Aplicada

Ignacio Sarmiento-Barbieri

Universidad de los Andes

# Agenda

1. Prediction and loss functions
2. Bias/Variance Decomposition
3. Prediction and linear regression
4. GitHub

# Agenda

# Getting serious about prediction

$$y = f(X) + u \tag{1}$$

- Interest on predicting $Y$
- Model? We treat $f()$ as a black box, and any approximation $\hat{f}()$ that yields a good prediction is good enough (*"Whatever works, works..."*).
- How do we measure "what works"?

# Getting serious about prediction

$$y = f(X) + u \tag{1}$$

- Interest on predicting $Y$
- Model? We treat $f()$ as a black box, and any approximation $\hat{f}()$ that yields a good prediction is good enough (*"Whatever works, works..."*).
- How do we measure "what works"?
- Formal statistics can help figure out this: what is a good prediction.

# Minimizing our losses

- A very common loss function in a regression setting is the squared loss $L(d) = d^2$
- Under this loss function the expected prediction loss is the mean squared error (MSE)
- **Result**: The best prediction of $Y$ at any point $X = x$ is the conditional mean, when best is measured using a square error loss

# Minimizing our losses

- Prediction problem solved if we knew $f^* = E[y|X = x]$

# Minimizing our losses

- Prediction problem solved if we knew $f^* = E[y|X = x]$
- But we have to settle for an estimate: $\hat{f}(x)$
- The EMSE of this

$$E(y - \hat{y})^2 = E(f(X) + u - \hat{f}(X))^2 \tag{2}$$

# Reducible and irreducible error

$$E(y-\hat{y})^2 = \underbrace{[f(X)-\hat{f}(X)]^2}_{Reducible} + \underbrace{Var(u)}_{Irreducible} \tag{3}$$

- The focus the is on techniques for estimating $f$ with the aim of minimizing the reducible error
- It is important to keep in mind that the irreducible error will always provide an upper bound on the accuracy of our prediction for $y$
- This bound is almost always unknown in practice

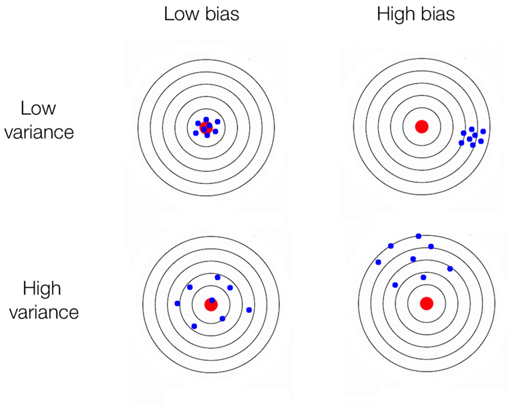
# Agenda

# Bias/Variance Decomposition

Recall that

- $Bias(\hat{f}(X)) = E(\hat{f}(X)) - f = E(\hat{f}(X) - f(X))$
- $Var(\hat{f}(X)) = E(\hat{f}(X) - E(\hat{f}(X)))^2$

# Bias/Variance Decomposition




Source: https://tinyurl.com/y4lvjxpc

# Bias/Variance Decomposition

Recall that

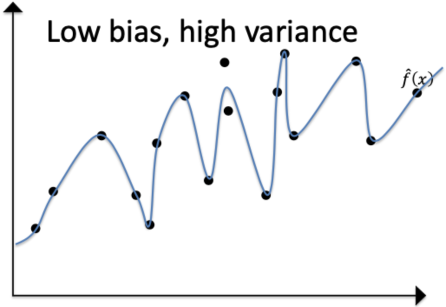
- $Bias(\hat{f}(X)) = E(\hat{f}(X)) - f = E(\hat{f}(X) - f(X))$
- $Var(\hat{f}(X)) = E(\hat{f}(X) - E(\hat{f}(X)))^2$

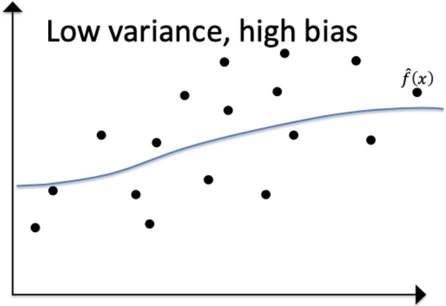
**Result** (very important!)

$$EMSE = Bias^2(\hat{f}(X)) + V(\hat{f}(X)) + \underbrace{Var(u)}_{Irreducible} \tag{4}$$

HW: Proof

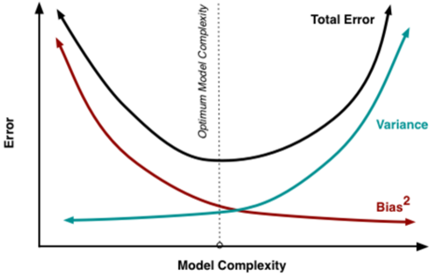
# Bias/Variance Decomposition



# The Bias-Variance Trade-Off

$$EMSE = Bias^2(\hat{f}(X)) + V(\hat{f}(X)) + \underbrace{Var(u)}_{Irreducible} \tag{5}$$




Source: https://tinyurl.com/y4lvjxpc

- The best kept secret: tolerating some bias is possible to reduce $V(\hat{f}(X))$ and lower MSE

# Agenda

# Prediction and linear regression

- The goal is to predict $y$ given another variables $X$.
- We assume that the link between $y$ and $X$ is given by the simple model:

$$y = f(X) + u \tag{6}$$

- we just learned that under a squared loss we need to approximate $E[y|X = x]$

# Prediction and linear regression

- As economists we know that we can approximate $E[y|X = x]$ with a linear regression

$$f(X) = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p \tag{7}$$

- The problem boils down to estimating $\beta$s
- We can estimate these using
  - OLS
  - MLE
  - MM

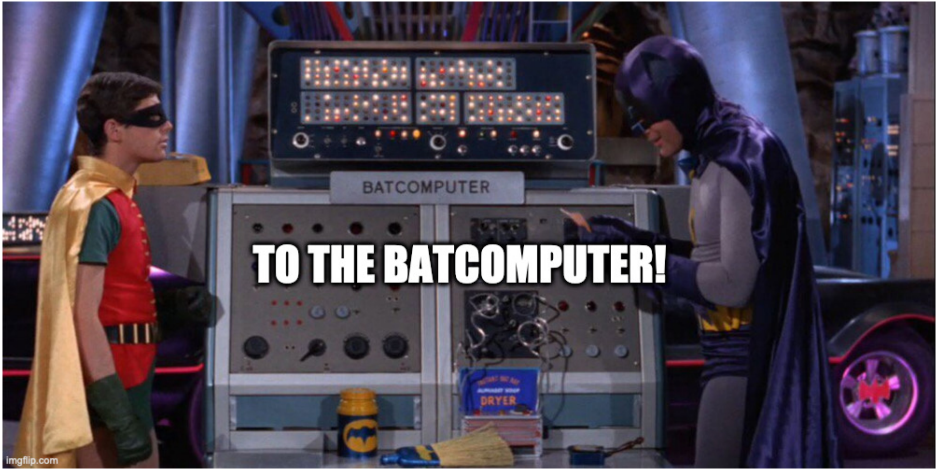



photo from https://www.dailydot.com/parsec/batman-1966-labels-tumblr-twitter-vine/

# Prediction and linear regression

- And the Bias-Variance Trade-Off?

# Prediction and linear regression

- And the Bias-Variance Trade-Off?
- Under the classical assumptions the OLS estimator is unbiased, hence

$$
\begin{aligned}
E(X\hat{\beta}) &= E(\hat{\beta}_1 + \hat{\beta}_2 X_2 + \cdots + \hat{\beta}_p X_p) && (8)\\
&= E(\hat{\beta}_1) + E(\hat{\beta}_2) X_2 + \cdots + E(\hat{\beta}_p) X_p && (9)\\
&= X\beta && (10)
\end{aligned}
$$

- Then,
  - $MSE(\hat{f})$ reduces to just $V(\hat{f})$

# 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

- Classical econometrics, model choice involves deciding between a smaller and a larger linear model.
- Consider the following competing models for $y$:

$$y = \beta_1 X_1 + u_1$$

$$y = \beta_1 X_1 + \beta_2 X_2 + u_2$$

- $\hat{\beta}_1^{(1)}$ the OLS estimator of regressing $y$ on $X_1$
- Prediction is:

$$\hat{y}^{(1)} = \hat{\beta}_1^{(1)} X_1$$

- $\hat{\beta}_1^{(2)}$ and $\hat{\beta}_2^{(2)}$ the OLS estimators of $\beta_1$ and $\beta_2$ of regressing $Y$ on $X_1$ and $X_2$.
- Prediction is:

$$\hat{y}^{(2)} = \hat{\beta}_1^{(2)} X_1 + \hat{\beta}_2^{(2)} X_2$$

# Complexity and the variance/bias trade off

- An important discussion in classical econometrics is that of omission of relevant variables vs. inclusion of irrelevant ones.
  - If model (1) is true then estimating the larger model (2) leads to inefficient though unbiased estimators due to unnecessarily including $X_2$.
  - If model (2) holds, estimating the smaller model (1) leads to a more efficient but biased estimate if $X_1$ is also correlated with the omitted regressor $X_2$.
- This discussion of small vs large is always with respect to a model that is supposed to be true.
- But in practice the true model is unknown!!!

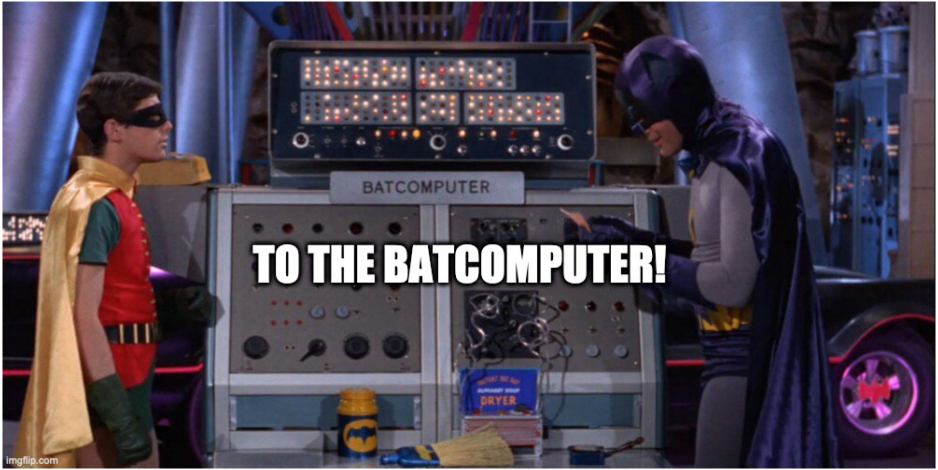



photo from https://www.dailydot.com/parsec/batman-1966-labels-tumblr-twitter-vine/

# Complexity and the variance/bias trade off

- Choosing between models involves a *bias/variance trade off*
- Classical econometrics tends to solve this dilemma abruptly,
  - requiring unbiased estimation, and hence favoring larger models to avoid bias
- In this simple setup, larger models are 'more complex', hence more complex models are less biased but more inefficient.
- Hence, in this very simple framework complexity is measured by the number of explanatory variables.
- A central idea in machine learning is to generalize the idea of complexity,
  - Optimal level of complexity, that is, models whose bias and variance led to minimum MSE.

# Agenda

# Collaboration on GitHub

- This is what I'm going to do (and want you to practice)
  - Partner 1: Invite Partner 2 to join you as a collaborator on your GitHub repo
  - Partner 2: Clone Partner 1's repo to your local machine. Make some edits (e.g. delete lines of text and add your own). Stage, commit and push these changes.
  - Partner 1: Make your own changes to the same file on your local machine. Stage, commit and then try to push them (*after* pulling from the GitHub repo first).

# Collaboration time

... and we are back

- Did Partner 1 encounter a 'merge conflict' error?
- Git is protecting P1 by refusing the merge. It wants to make sure that you don't accidentally overwrite all of your changes by pulling P2's version of the file.

# Collaboration time

```
Some text here.
<<<<<<< HEAD
Text added by Partner 2.
=======
Text added by Partner 1.
>>>>>>> 814e09178910383c128045ce67a58c9c1df3f558.
More text here.
```

# Collaboration time

- Fixing these conflicts is a simple matter of (manually) editing the file.
  - Delete the lines of the text that you don't want.
  - Then, delete the special Git merge conflict symbols.
- Once that's done, you should be able to stage, commit, pull and finally push your changes to the GitHub repo without any errors.
  - P1 gets to decide what to keep because they fixed the merge conflict.
  - The full commit history is preserved, so P2 can always recover their changes if desired.
  - Another solution is using `branches`

# Review

- This Week: The predictive paradigm and linear regression
  - Machine Learning is all about prediction
  - ML targets something different than causal inference, they can complement each other
  - ML best kept secret: tolerating some bias is possible to reduce $V(\hat{f}(X))$ and lower MSE
- Next Week: Out of sample prediction. Overfit, Resampling Methods, Webscrapping