DePaul University
College of Computing and Digital Media

Casey Bennett, PhD

Sept.18, 2019

This Week

1) Make sure you have Python and Scikit installed (or Anaconda) – CHECK LIBRARIES
2) Discussion Boards
3) Office Hours
4) Nomenclature

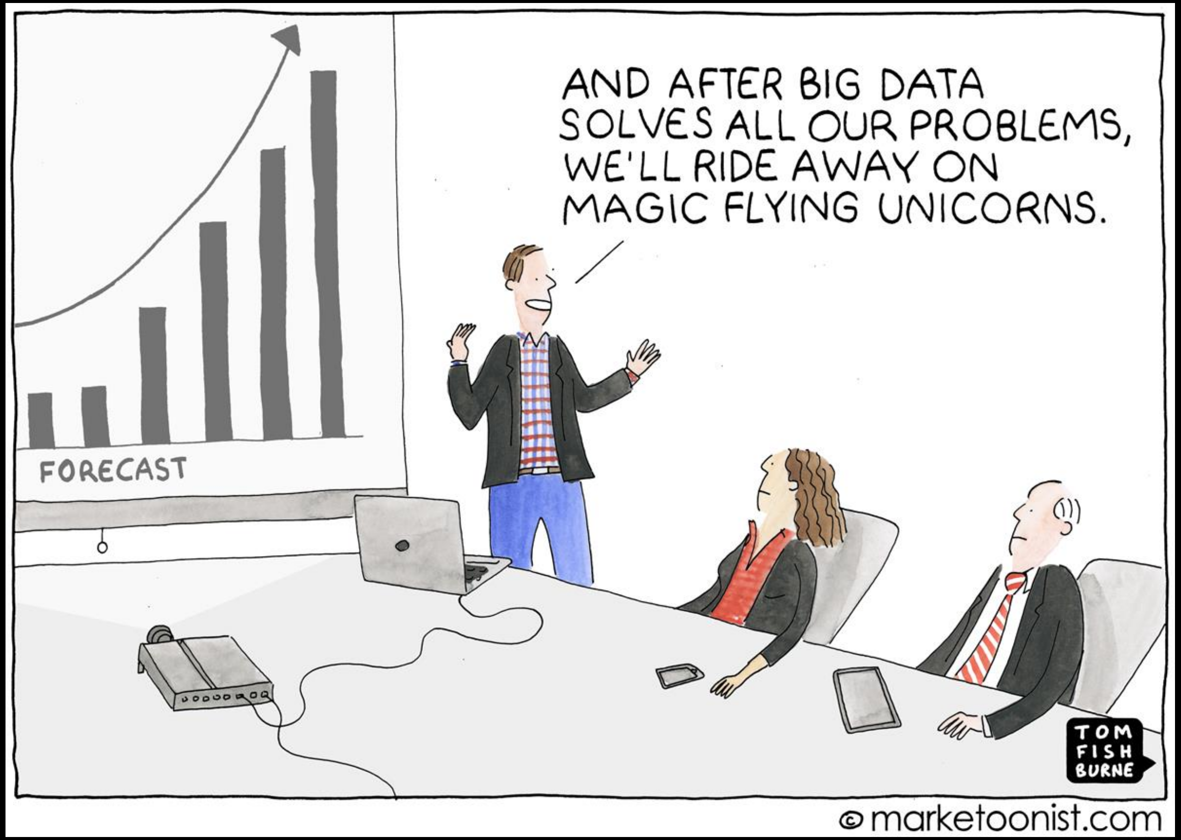
Evaluating Performance

https://pollev.com/caseybennett801

or text “caseybennett801” to 37607

The Papers:

Why did I pick these two? What is the juxtaposition?



So if I gave you a million examples of something (aka "big data") then that would solve the problem?

Generalization Problem

1) Statistical Power

- Sufficient power to detect effects
- Smaller Datasets
- Clinical Trials
- Cohort Analyses

2) Machine Learning

- Cross-validation and overfitting
- Big Data
- Naturalistic Studies
- Public Datasets

ML Stages

1) Load Data
2) Preprocess
3) Feature Selection
4) Train Model
5) Evaluate Performance

Types of ML Models

1) Classification
2) Regression
3) Clustering
4) Time Series/Temporal
5) Search
6) Optimization

Types of Scores

1) Accuracy
2) AUC (ROC Analysis)
3) RMSE
4) Explained Variance
5) AIC/BIC
6) Silhouette Scores
7) etc. etc. etc.

Types of Evaluation

1) Cross-Validation
2) Test/Train split

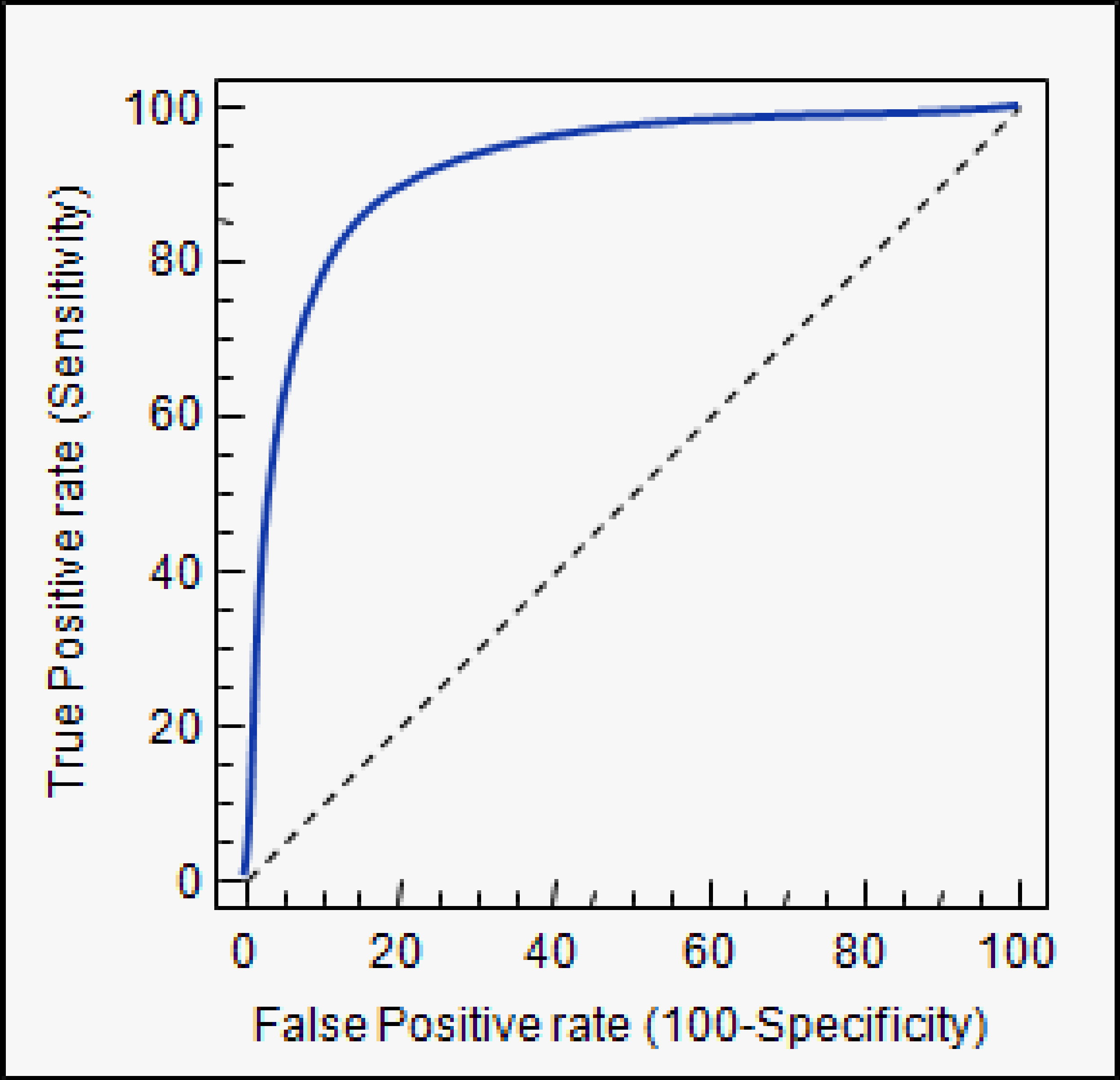
Say we had a breast cancer dataset, where 1 in 100 people develop cancer … is 99% accuracy *good*?

Different Types of Scores

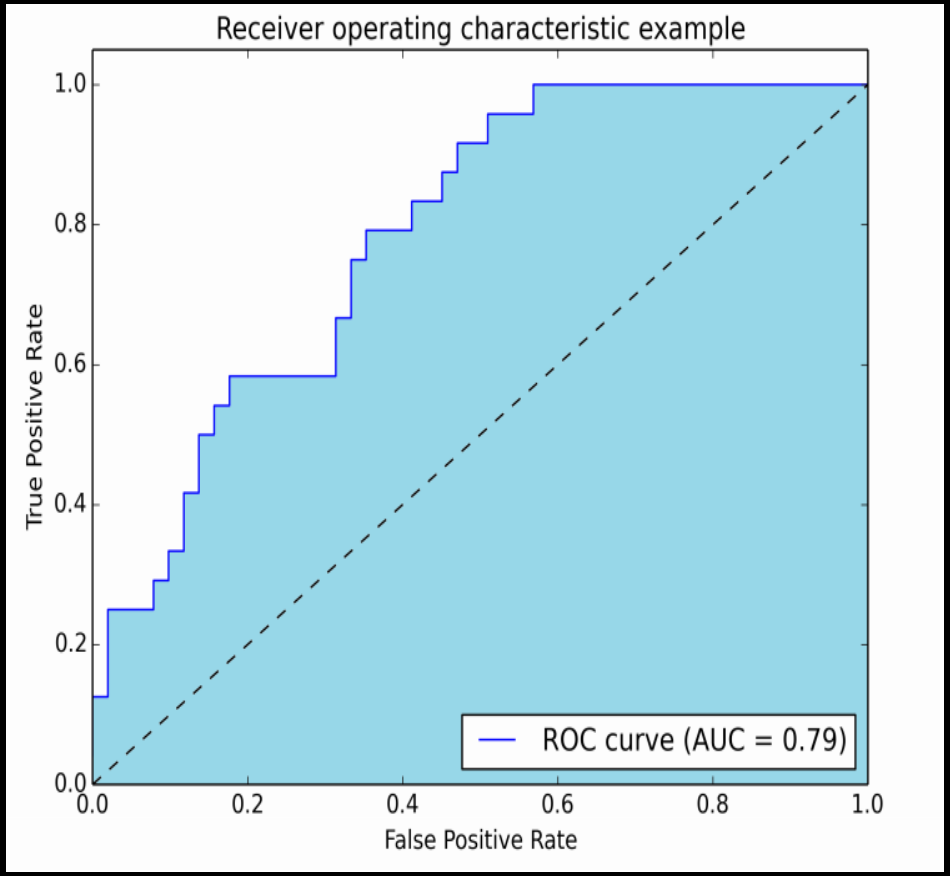
1) Accuracy
2) AUC (ROC Analysis)
3) RMSE
4) Explained Variance
5) AIC/BIC
6) Silhouette Scores
7) etc. etc. etc.

Classification

AUC



True Positive rate (Sensitivity)
100
80
60
40
20
0
0
20
40
60
80
100
False Positive rate (100-Specificity)

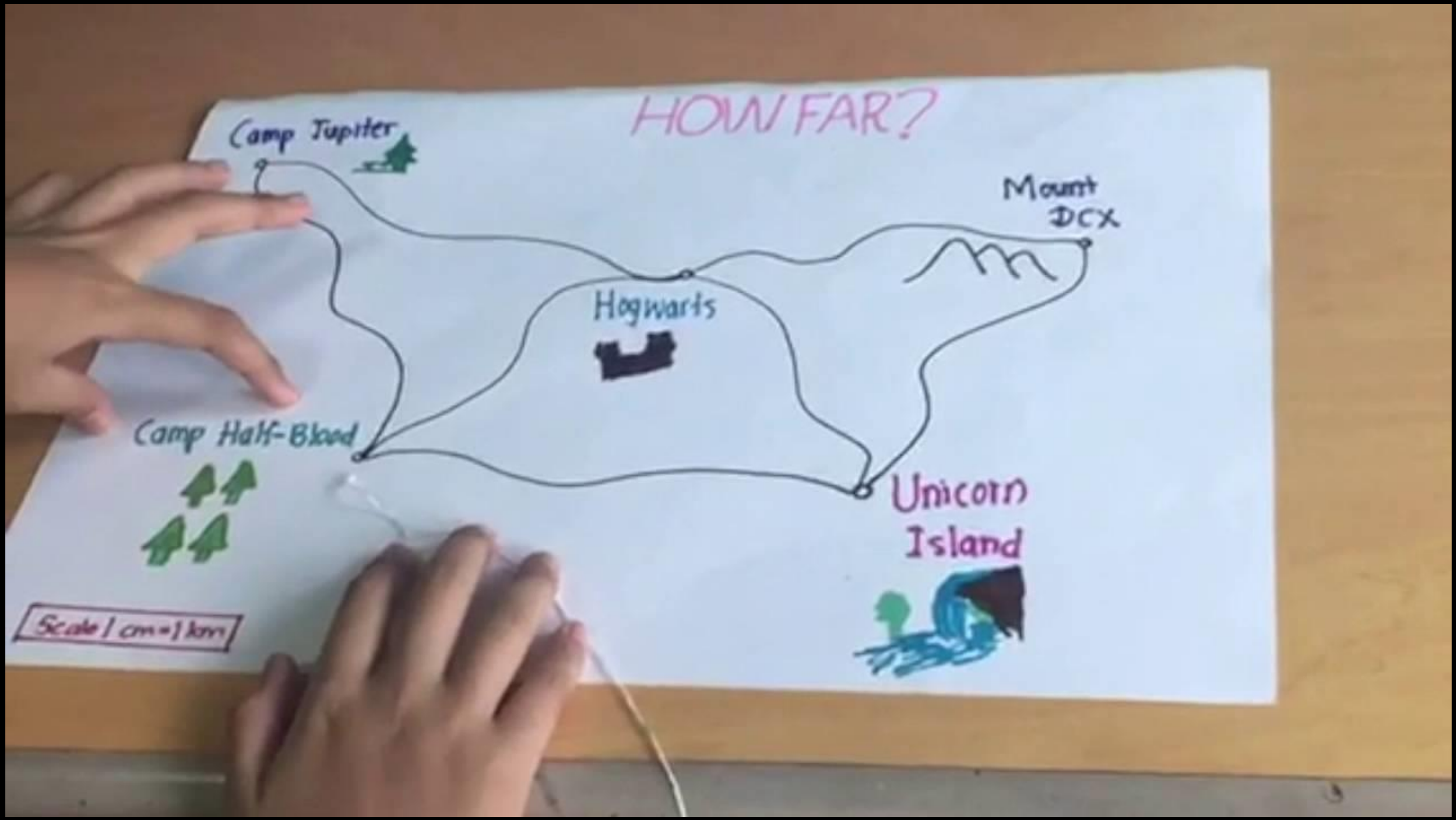


Receiver operating characteristic example
True Positive Rate
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
False Positive Rate
ROC curve (AUC = 0.79)

AUC/ROC *only* works for binary classification

- Yes vs No
- True vs False
- Black vs White
- High vs Low
- Dog vs Not Dog

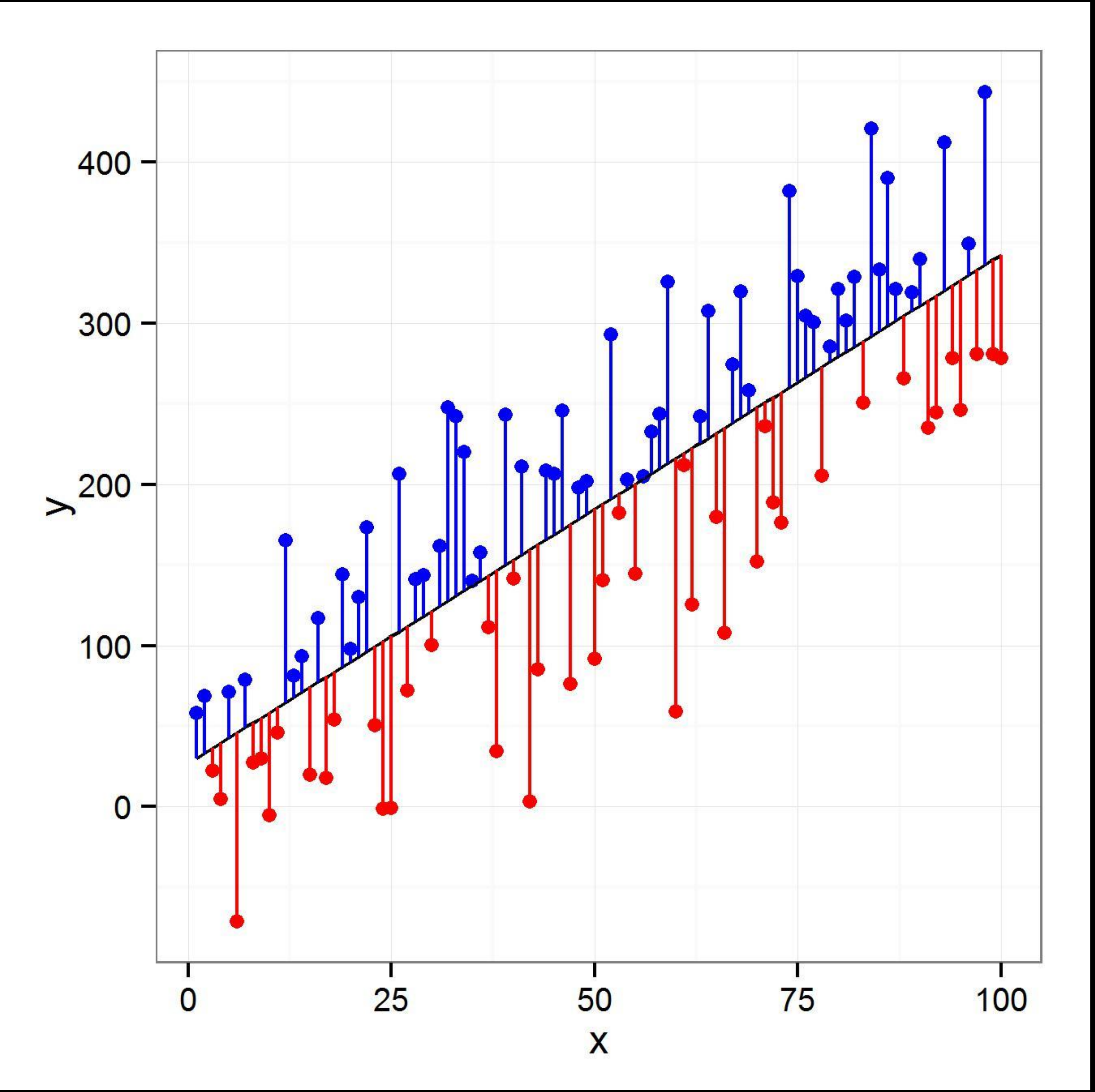
Say I wanted to predict the location of things, how would I know I have a good model?



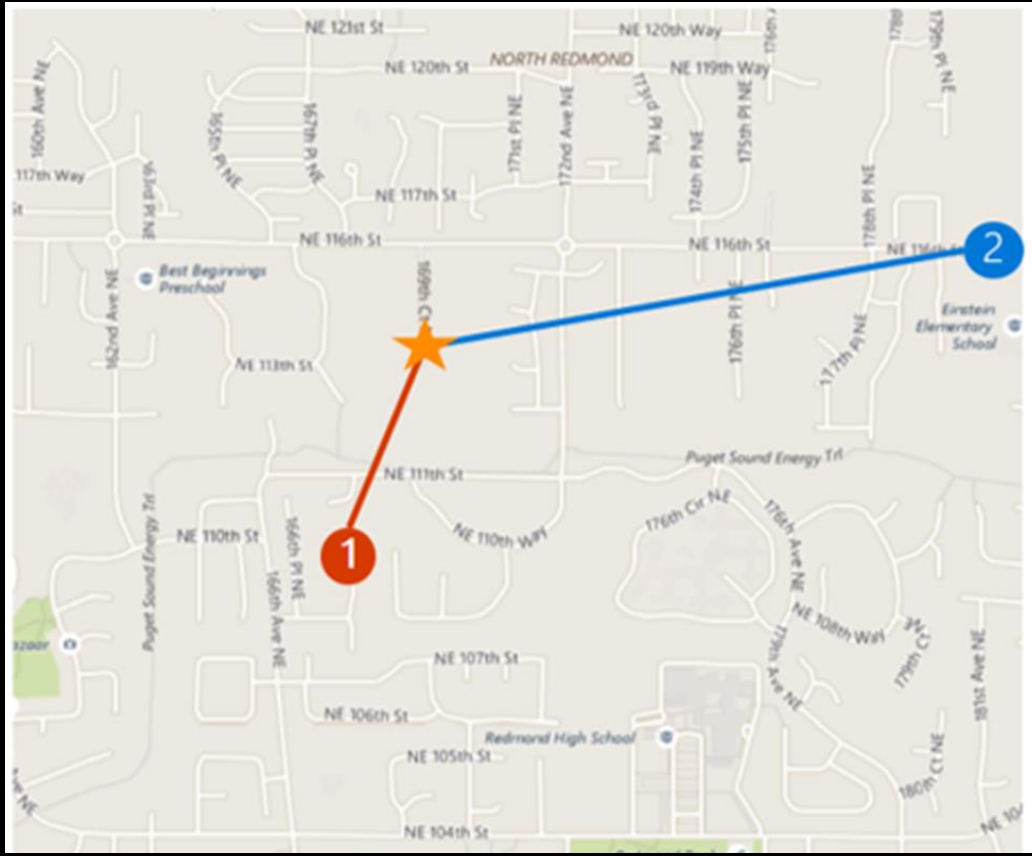
Different Types of Scores

1) Accuracy
2) AUC (ROC Analysis)
3) RMSE
4) Explained Variance
5) AIC/BIC
6) Silhouette Scores
7) etc. etc. etc.

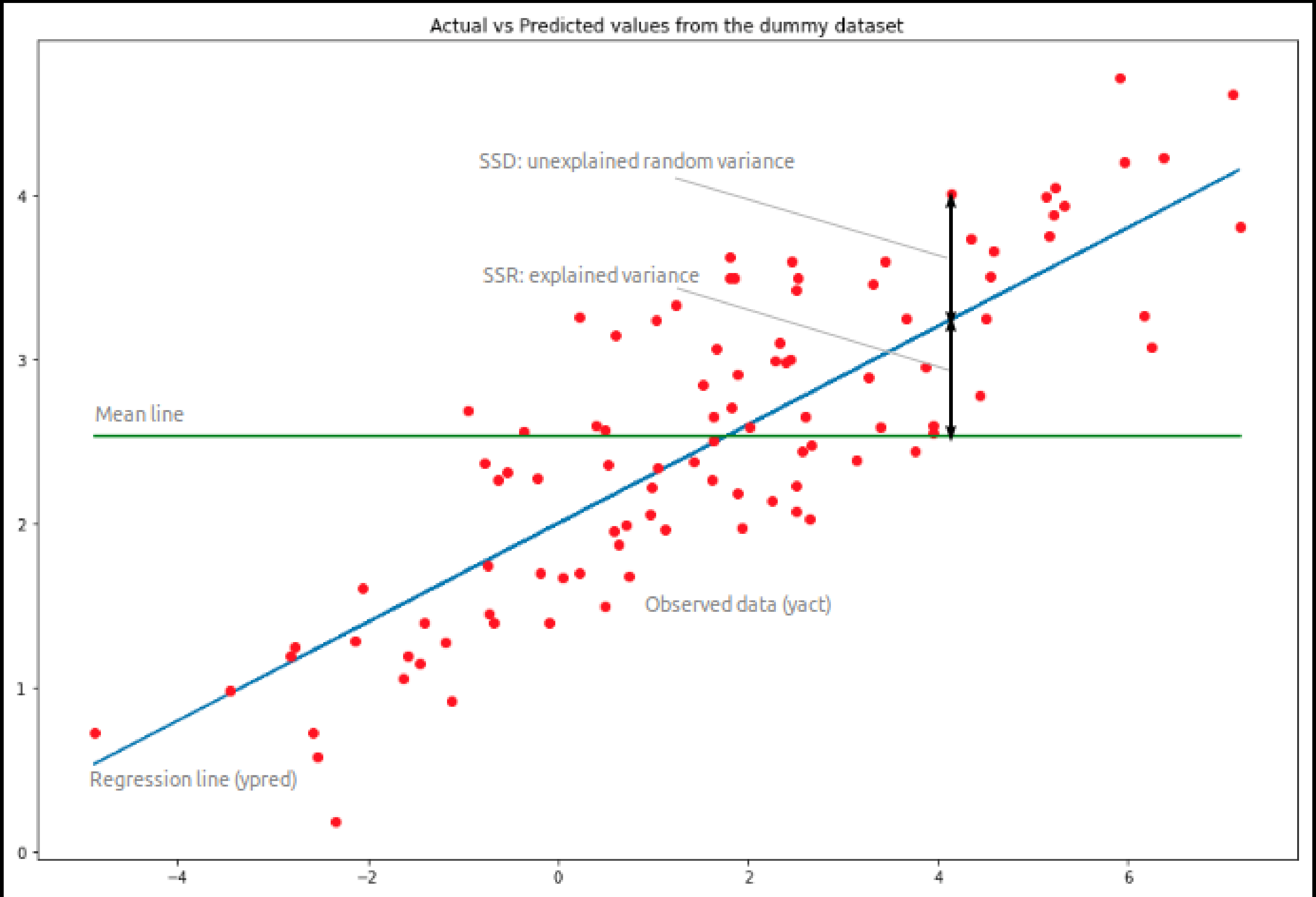
Regression


400
300
200
100
0
y
0
25
50
75
100
x

RMSE


NORTH REDMOND
NE 116th St
NE 111th St
NE 107th St
NE 106th St
Redmond High School
1
2

Explained Variance



Unlike the other metrics, for RMSE *lower* is better

How do I know I have a good model?

It generalizes

We have some sort of *score* for a model, maybe a good score. Does that mean that the model "generalizes" well?

What we are really looking for are models that are *consistently* accurate across different slices of the data

ML Stages

1) Load Data
2) Preprocess
3) Feature Selection
4) Train Model
5) Evaluate Performance

Types of ML Models

1) Classification
2) Regression
3) Clustering
4) Time Series/Temporal
5) Search
6) Optimization

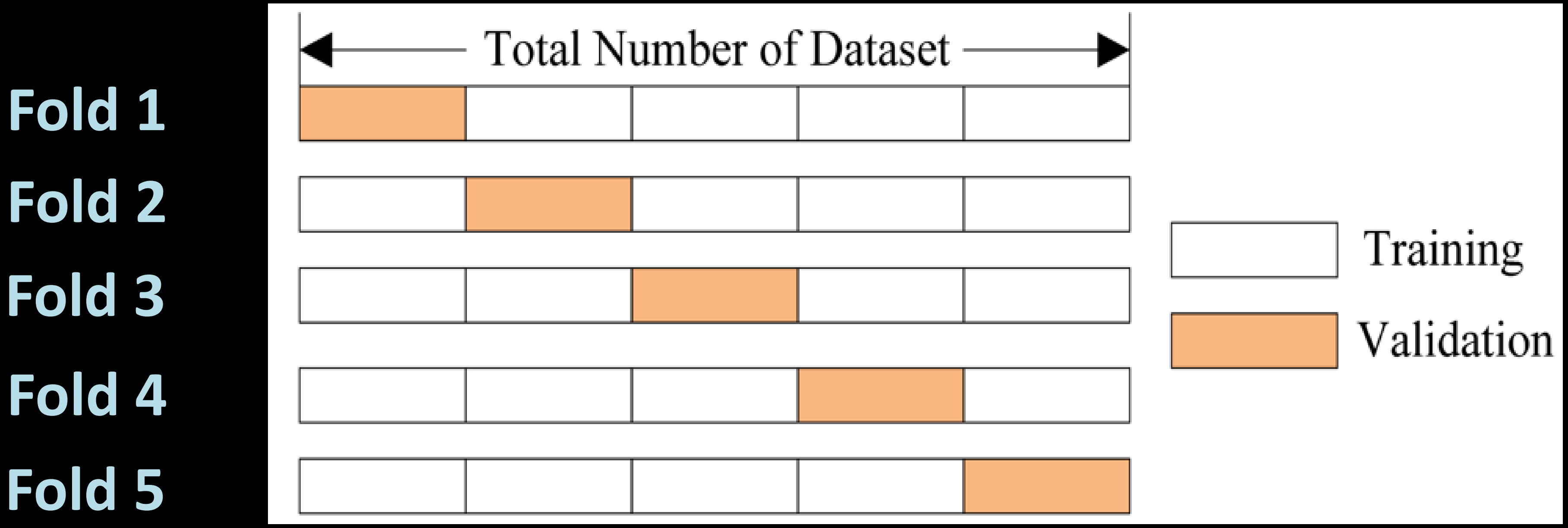
Types of Scores

1) Accuracy
2) AUC (ROC Analysis)
3) RMSE
4) Explained Variance
5) AIC/BIC
6) Silhouette Scores
7) etc. etc. etc.

Types of Evaluation

1) Cross-Validation
2) Test/Train split

K-fold Cross Validation



1) Cross-validation can be any number of folds (k)
2) Test/Train Split is sort of like just doing one fold
3) Also can *stratify* folds, so each fold has representative numbers of each value of target

Be careful about the term “validation”, better to use the term **test set**

Validation set sometimes refers to dataset held out of CV for final testing

ML Stages

Setup Environment,
Import Stuff

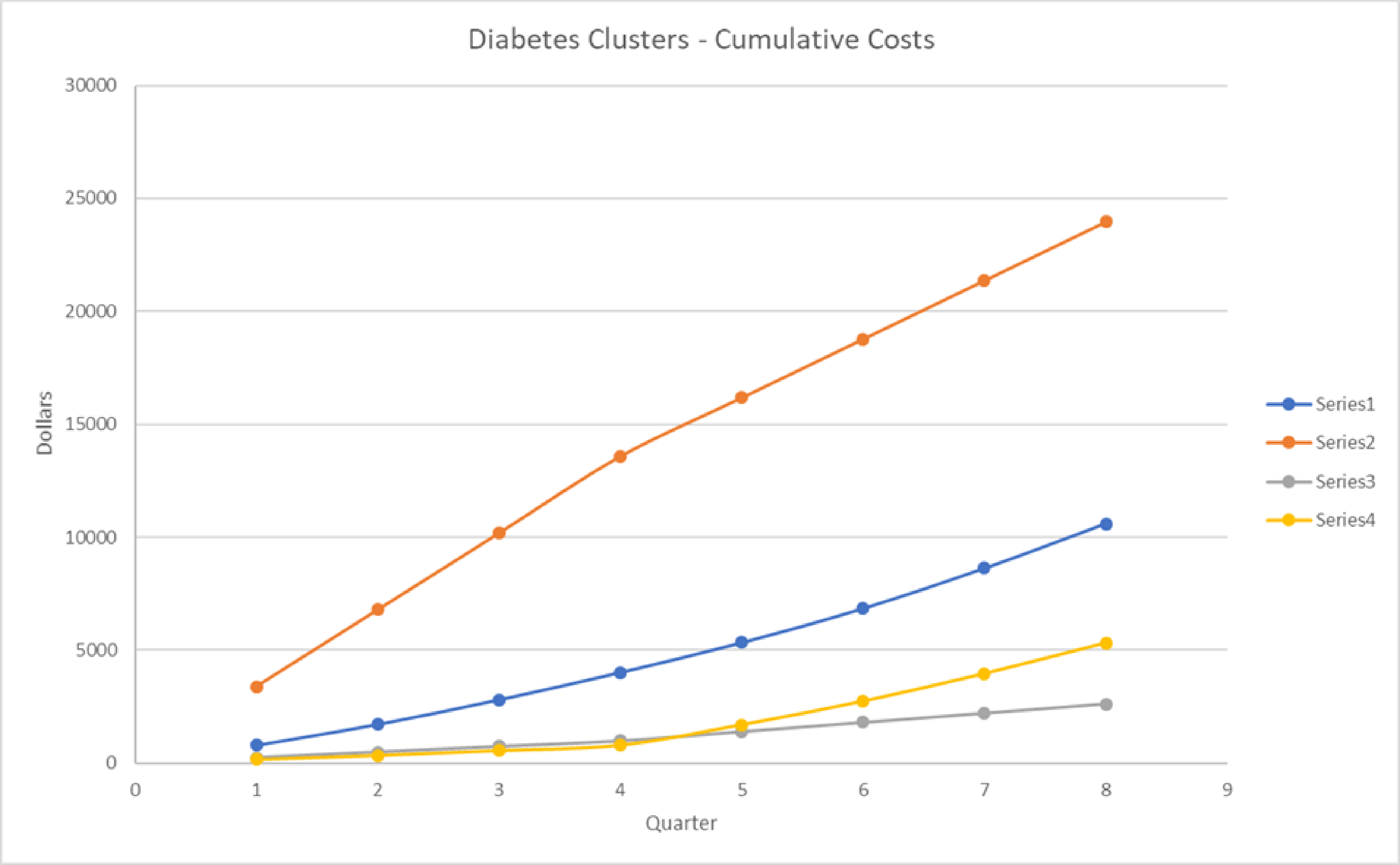
1) **Load Data**
   - *Read File, Parse header and row data*
2) **Preprocess**
   - *Normalize, Discretize, Impute, etc.*
3) **Feature Selection**
   - *Select subset of relevant features*
4) **Train Model**
   - *Fit some model(s) to the dataset*
5) **Evaluate Performance**
   - Did it work?

Real World Example

- Evaluated a large state-wide population in the U.S. of over 300,000 unique patients spanning 3 years from 2014-2016 using random forests
- Payor claims data and social determinants of health data
- Can we detect meaningful clusters of trajectories for *diabetes progression*, in order to create cost-effective screening programs

	Diabetes Progression Models		
Prediction	**Non PredPos %**	**PredPos %**	**Total Acc**
Pre-Diabetes (2014) to Full Diabetes (2015)	30.5%	72.9%	71.6%
Diabetes to Complications (2015)	19.9%	87.0%	83.5%

Real World Example



- Orange Group – High utilizers, high incidence renal complications
- Gray Group – Low Utilizers, with few complications except CV
- Blue Group – Falling in between Orange/Gray
- Yellow Group – “newer” cases with fewer complications, fewer mental health issues, earlier med stage

Orange and Blue groups were TWICE as likely to have mental health comorbidity

The benefit of feature selection is that it helps us overcome the “black box” issue in data science and ML

Feature Selection

1) Filter Methods

- Chi-squared, Gain Ratio, Relief-F, Mutual Information, Low-Variance, Correlation, Regression Based, Symmetrical Uncertainty, etc.

2) Wrapper Methods

- Involves building thousands of models on different sets of features, looking for the optimal one
- Different kinds of search: greedy, random, genetic algorithms

3) Recursive Methods

- Stepwise removal, either forward or backward

Feature Selection (cont.)

1) Filter Methods

- Univariate (chi-sq) vs multivariate (relief-f)
- Target: discrete (gain ratio) vs continuous (mutual info regression)

2) Wrapper Methods

- Feature Importance coming out of tree methods (like Random Forests) can be thought of as a “poor man’s” approach to this
- One can create a full-blown wrapper though, encapsulating any kind of ML algorithm (naïve bayes, neural network, etc.)

3) Recursive Methods

- More traditional statistical approach

Feature Selection – Related Topics

1) **Feature Extraction** (or agglomeration)
   - Dimensionality reduction
   - e.g. PCA, Heirarchical Clustering

2) **Feature Construction** (or engineering)
   - Deep Learning
   - Manual Feature Engineering

Homework

- Homework #1 releases right after class, due next week
- Homework #2 releases after that, 2 weeks to complete
- Check Python installation, and libraries (see '*Python Libraries needed*' file on D2L in Content section under Coding Templates)
- Code will run without any changes, so try running it immediately upon downloading to check Python setup

Project Datasets

- Info is posted in assignment on D2L

Potential Dataset links:

1. Kaggle datasets - https://www.kaggle.com/datasets
2. UCI dataset repo - https://archive.ics.uci.edu/ml/datasets.html
3. Google dataset search - https://toolbox.google.com/datasetsearch

For next week

1) Homework #1
2) Read Papers (posted in Content on D2L)
3) Post on online Discussion Forum (Week 3)
4) First paper review will be due first week of October, so keep that in mind