Lecture 5: Fundamentals of Machine Learning Pt. 2

INFO 1998: Introduction to Machine Learning

Mid Semester Feedback Form.

(also attendance!)

CDS Education

Lecture 5: Fundamentals of Machine Learning Pt. 2

INFO 1998: Introduction to Machine Learning

Bias vs. Variance & Tuning Models

CDS Education

Announcements

Mid-Semester Check-in

Where you should be right now:

- Have an idea of what your problem statement/hypothesis is
- Have your group chosen
- Have your data set chosen and some progress

Complete **between now and March 24th**

Cornell Drop Deadline: **March 18th**

Announcements

- Web scraping workshop today!
  - 8:20 PM - 9 PM
- A4 due tonight
  - Post on Ed for extensions
- A5 released, due 03/17/2025
-

What We'll Cover

Last Time's Goal: building blocks of ML

This Time's Goal: how to tell if your ML model is *useful (good)*

Agenda

1. Review
    - Types of Machine Learning
2. Measuring Accuracy/Error
3. Model Selection
4. Feature Selection

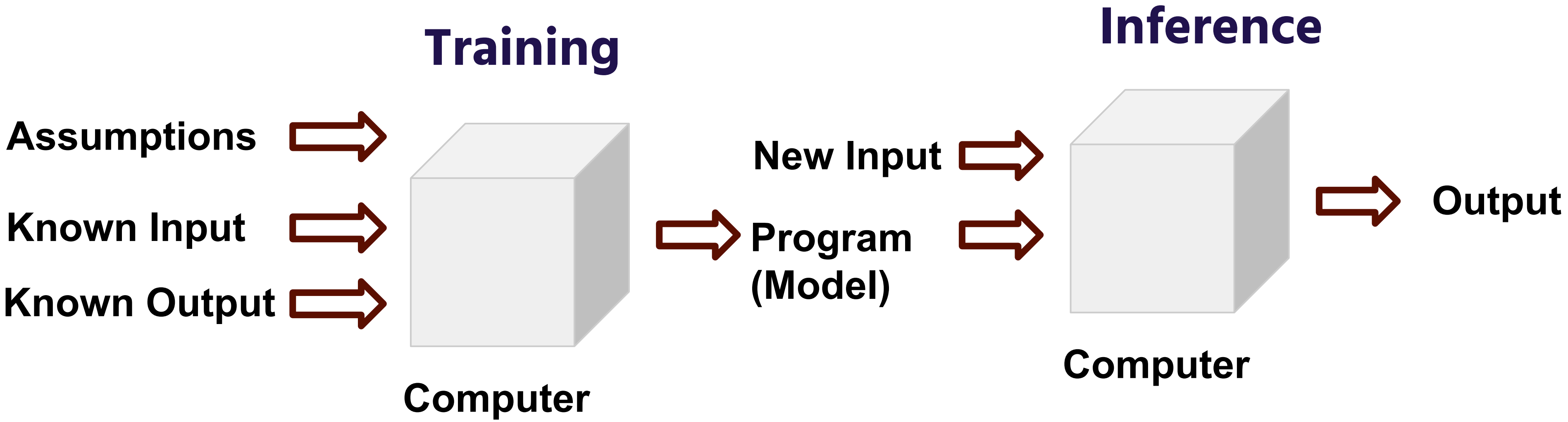
Review: Defining ML

We want to predict the future

- Take some known input and output
- Learn that data's "pattern" to:
  - Given a future input, predict[1] the corresponding output

[1] We model how the output is generated

Review: Machine Learning Pipeline



Review: Model

- “Model training” = learn a relationship
- “Model testing” = check if the learned relationship is generalizes
- “Model validation” = simulates model performance when used in real life

Different Types of ML

(supervised & unsupervised)
(classification & regression)

Supervised vs. Unsupervised

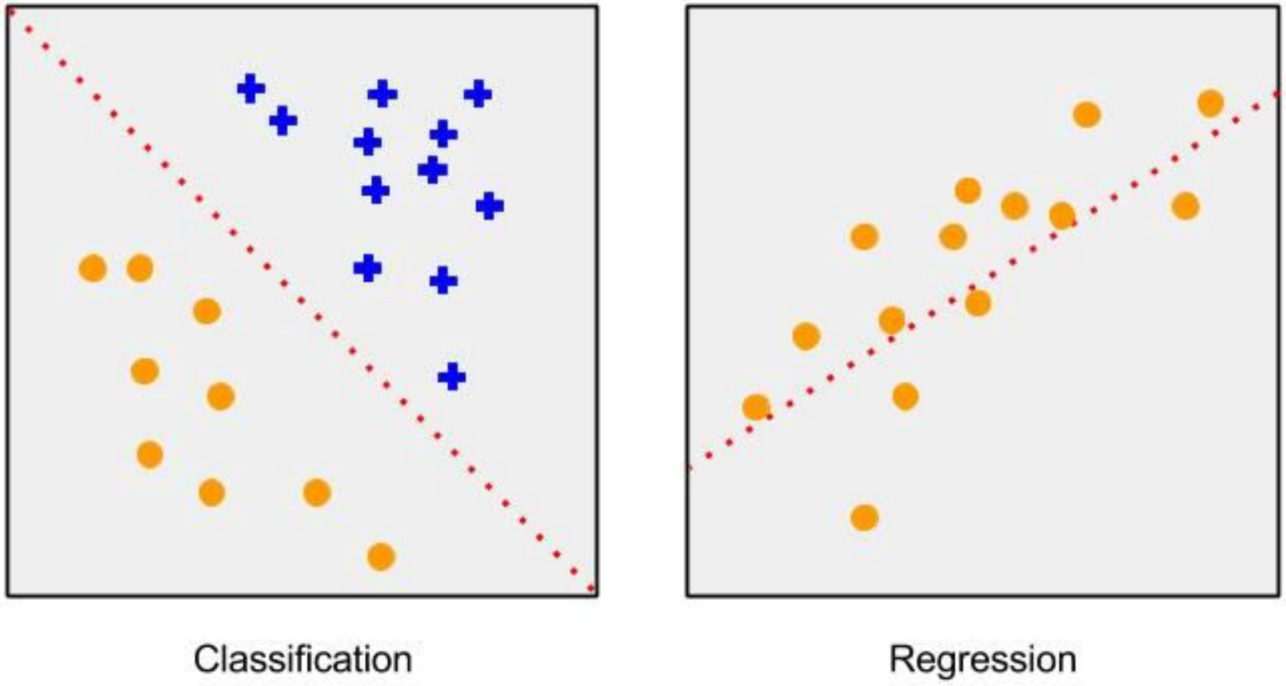
Supervised learning…

- Goal: Predict output
- Needs known output/target

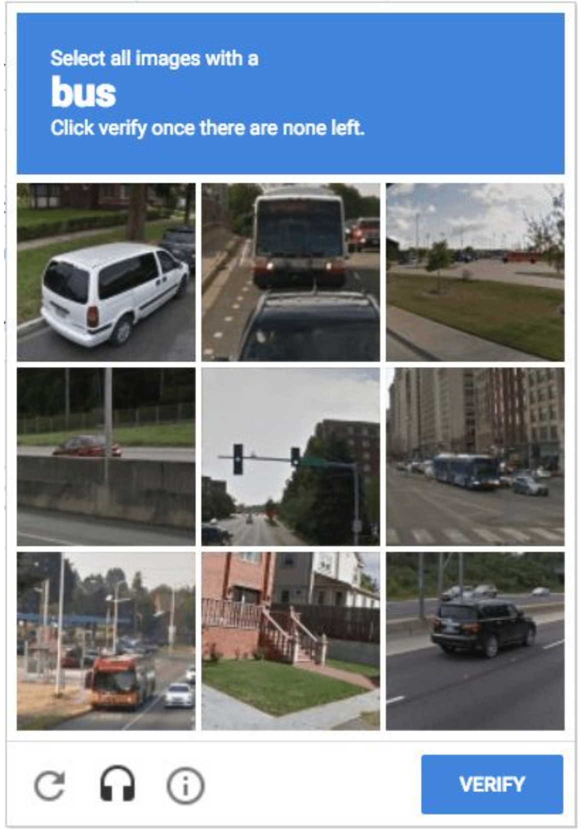
Unsupervised learning…

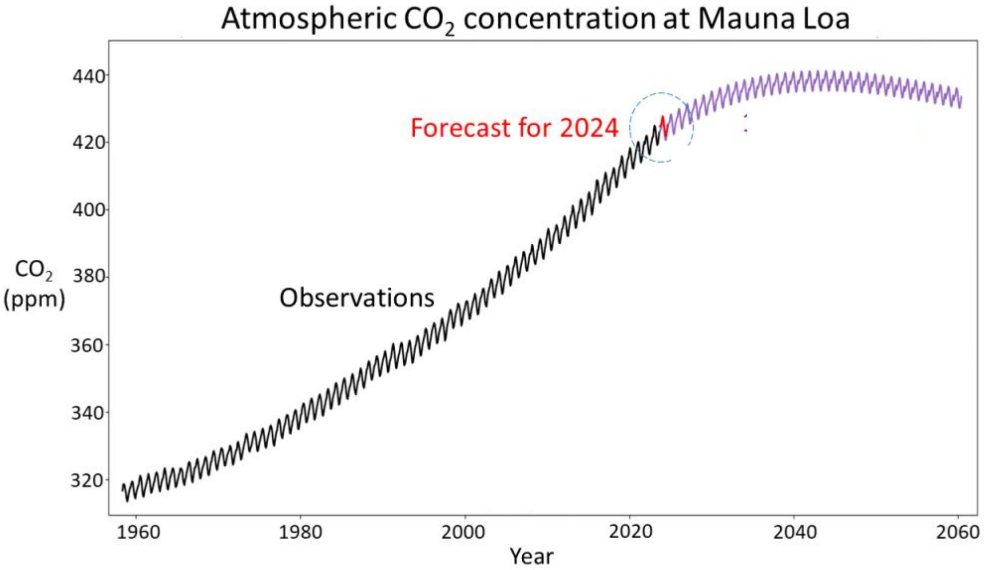
- Goal: learn more about the data (ex. trends)
- Doesn't need known output

Examples of Supervised: Classification and Regression



Classification or Regression?



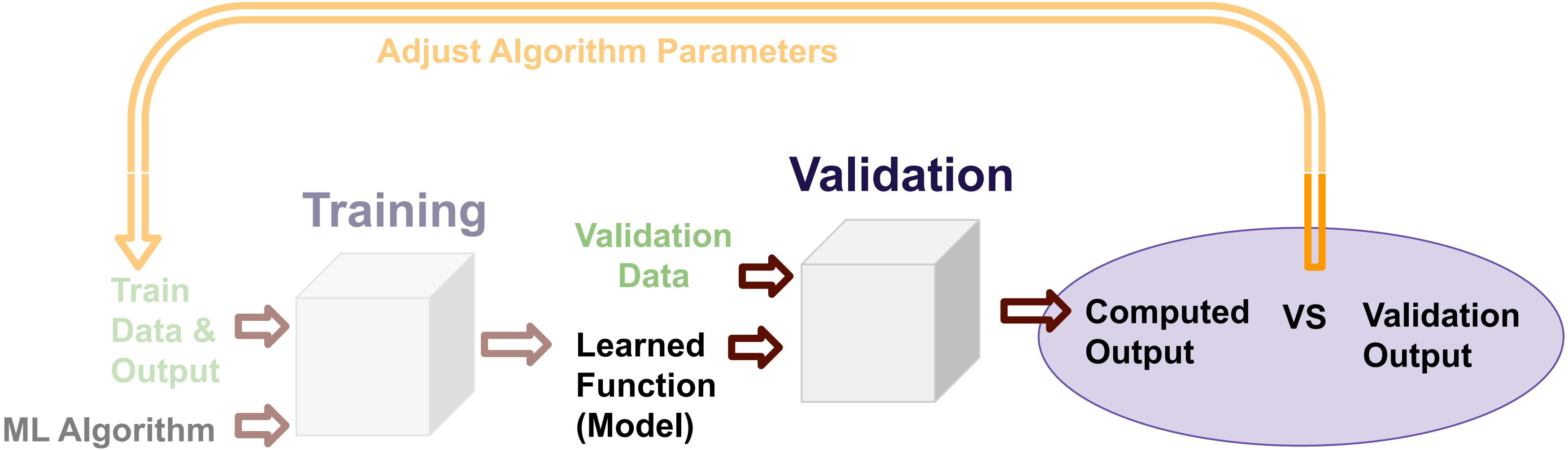


Classification or Regression? Examples from my internship

Detecting fake students
(adults using student discount)

Predicting the value of a customer

Measuring Training Accuracy



1. Split data (lecture 7)
2. Assess model accuracy (today)
3. Adjust Model (a bit today)

Loss, Cost, and Score Functions

- **Loss Function**
  - How far is a prediction from its corresponding answer
  - Used as a penalty for mislabelling in training to help a model learn

- **Cost**
  - Applies loss function to each point, then combines that into a single number

- **Metric (Score Function)**
  - How well the model did across all data points
  - Interpretable, for the model builder

Examples of Loss & Metrics: Multiple Choice Exams

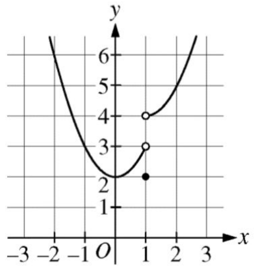
- How would you evaluate these?
  - If the answer is A) but you pick B)

9

Why does Akira say his meeting with Chie is "a matter of urgency" (line 32)?

A) He fears that his own parents will disapprove of Naomi.

B) He worries that Naomi will reject him and marry someone else.



Graph of f

. The graph of the function f is shown in the figure above. The value of $\lim_{x \to 0} f\left(1 - x^2\right)$ is

(A) 1 (B) 2

10

Which choice provides the best evidence for the answer to the previous question?

A) Line 39 ("I don't . . . you")

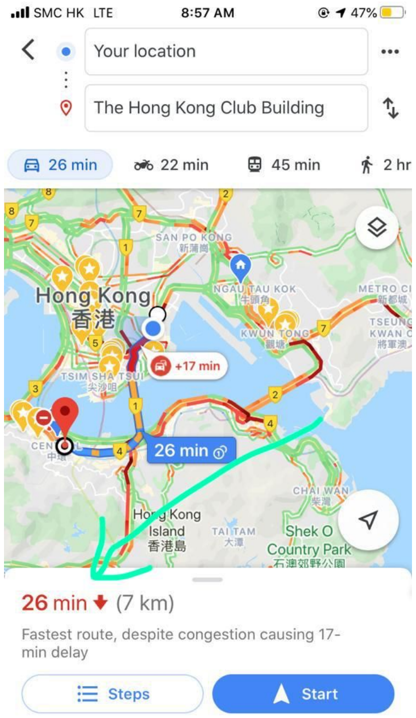
B) Lines 39-42 ("Normally . . . community")

Examples of Loss & Metrics: Multiple Choice Exams

- Zero-one loss:
    - 1 if prediction != answer
    - 0 if prediction == answer

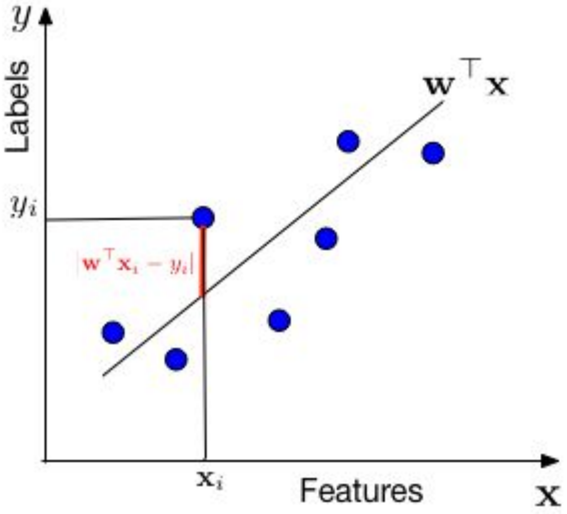
Examples of Loss & Metrics: Google Maps

- How would you evaluate this?
    - If Google Maps says it will take 26 mins but it actually takes x minutes



Linear Regression Loss Formula: Based on Euclidean Distance

$$\text{loss}(x_i, y_i) = (h(x_i) - y_i)^2$$



Two things to note about this loss function:

- Positives and negatives won't cancel
- Large errors are penalized to a power of 2 (more)

In what situations might you want a low penalty loss function as opposed to this high penalty loss function?

Linear Regression Loss Formula: Based on Euclidean Distance

$$\text{loss}(x_i, y_i) = (h(x_i) - y_i)^2$$

What could the **cost function** be?

- MSE = (...)/N
  - Where N is the number of data points

How do you know if something is good?

- “I throw at a speed of 35 ft/sec.”

How do you know if something is good?

- “I throw at a speed of 35 ft/sec. The average for pros is 27 ft/sec.”

Compare to Baseline

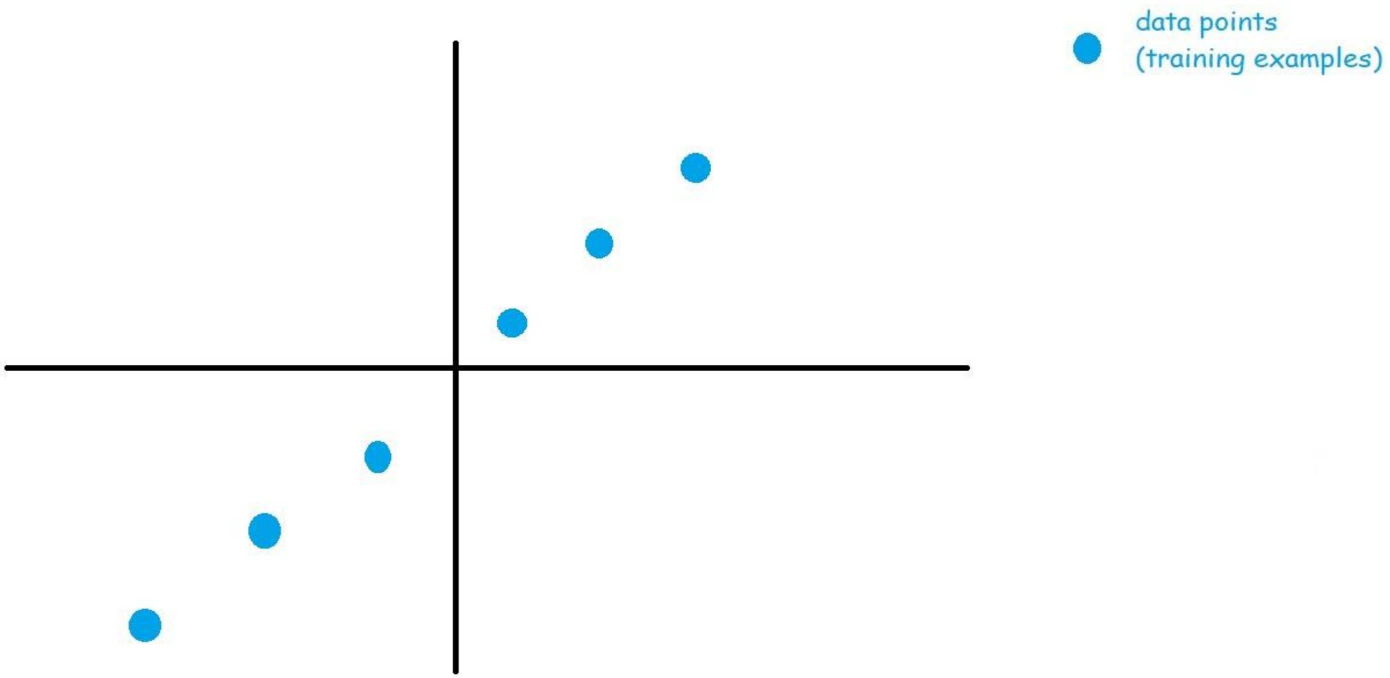
- When evaluating accuracy, compare our model to a **baseline**
    - For regression, one baseline model is the model that predicts the **average** of the target value for every point
    - For our purposes: don’t worry about the baseline *model*

Sk-learn's score function

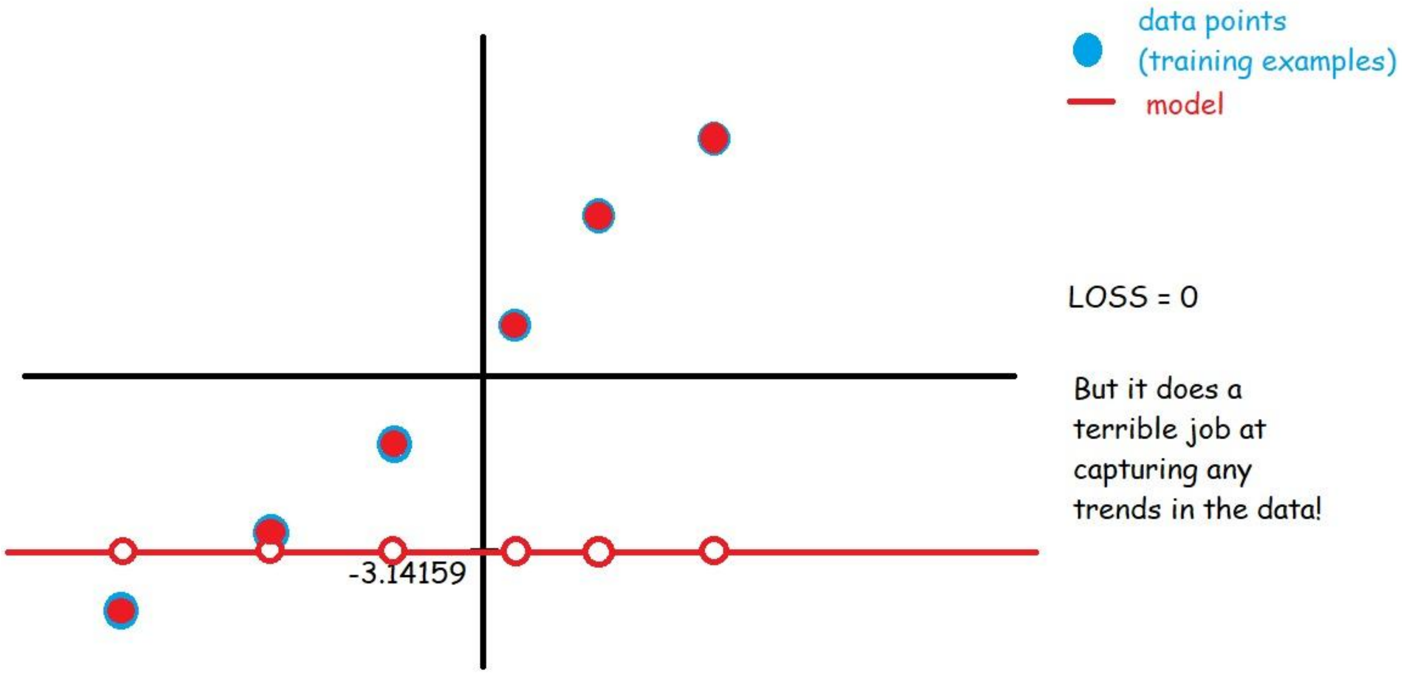
1 - ([Cost of model] / [Cost of baseline])

- **>0** means you beat the baseline
- **0** means you were equal to the baseline
- **<0** means you're worse than the baseline

Training Data



Cost = 0, but model is horrible…



LOSS = 0

But it does a terrible job at capturing any trends in the data!

MORAL: Assumptions are important!

Overfitting and Underfitting

(how generalizable is the performance?)

Model Goals

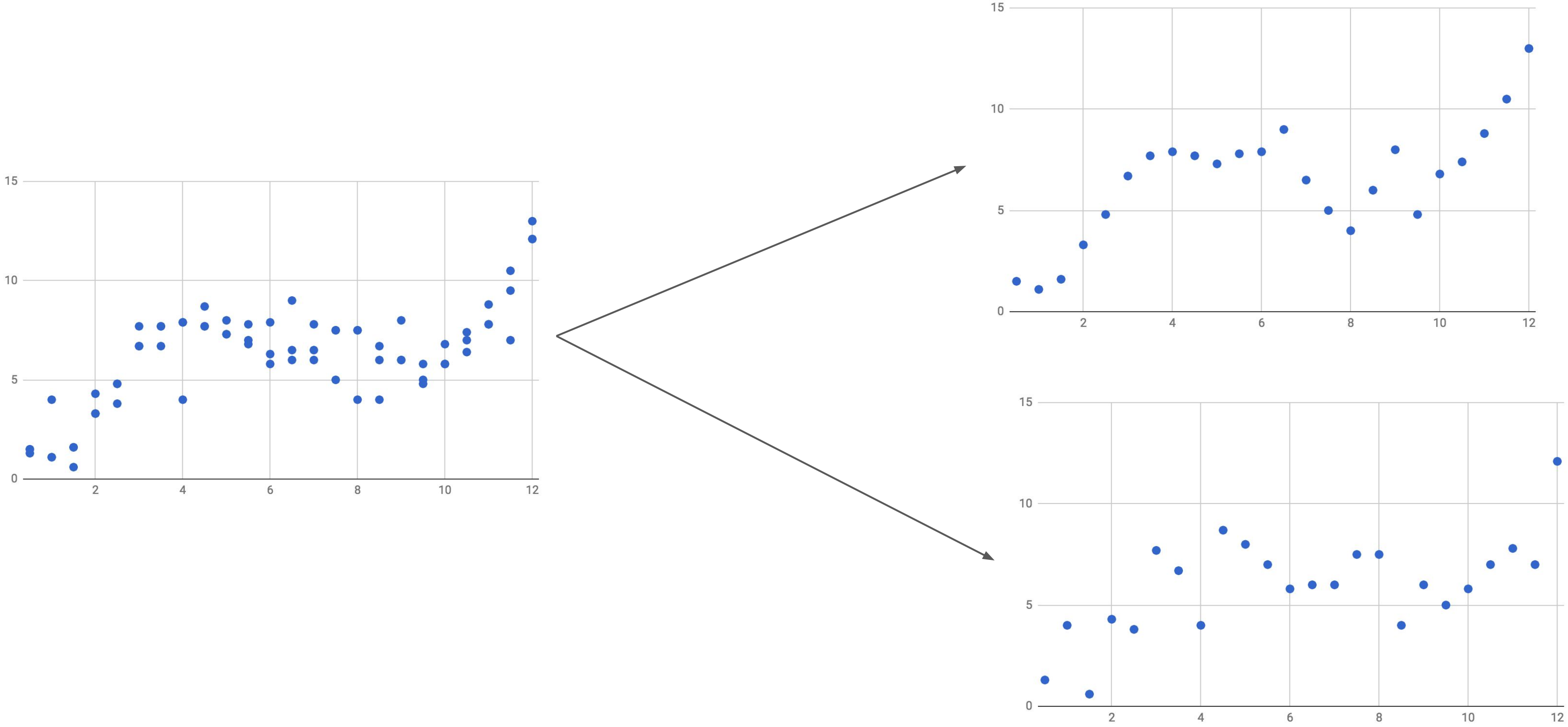
When training a model, we want our model to:

- Capture the trends of the training data sample
- Generalize well to the whole population
- Be moderately interpretable

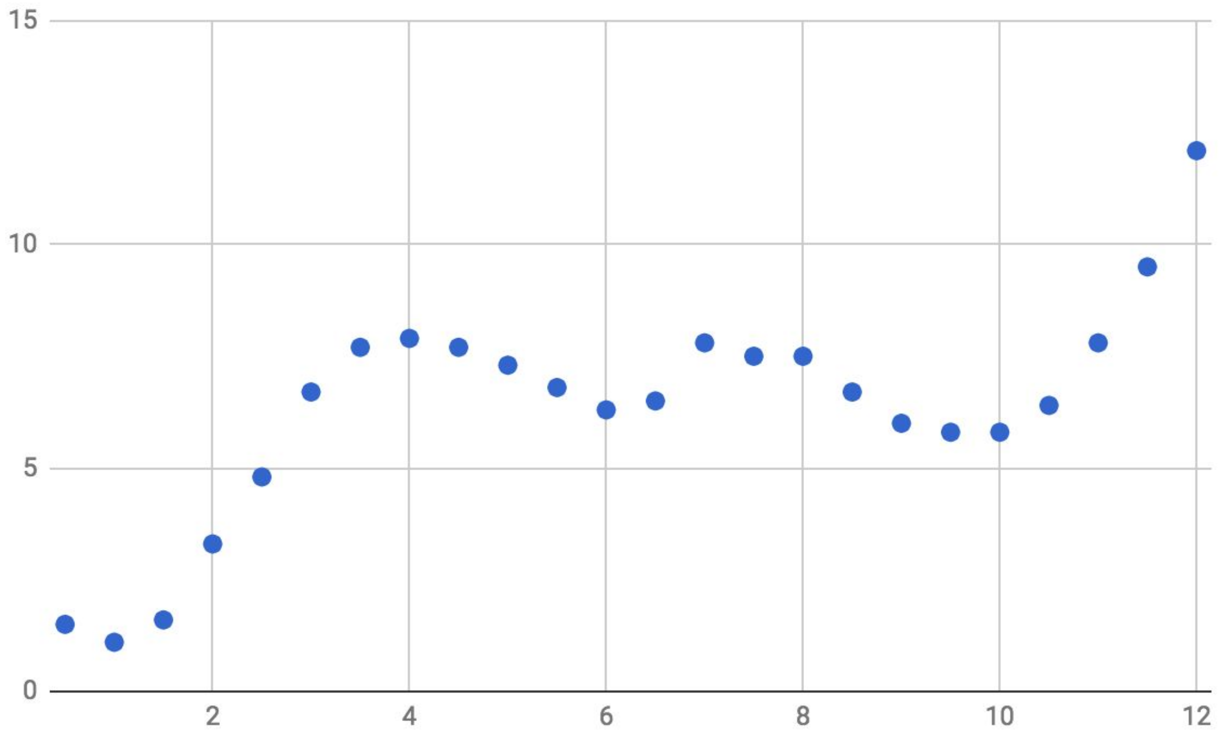
The first two are especially difficult to do simultaneously!

- Want to choose the right amount of complexity

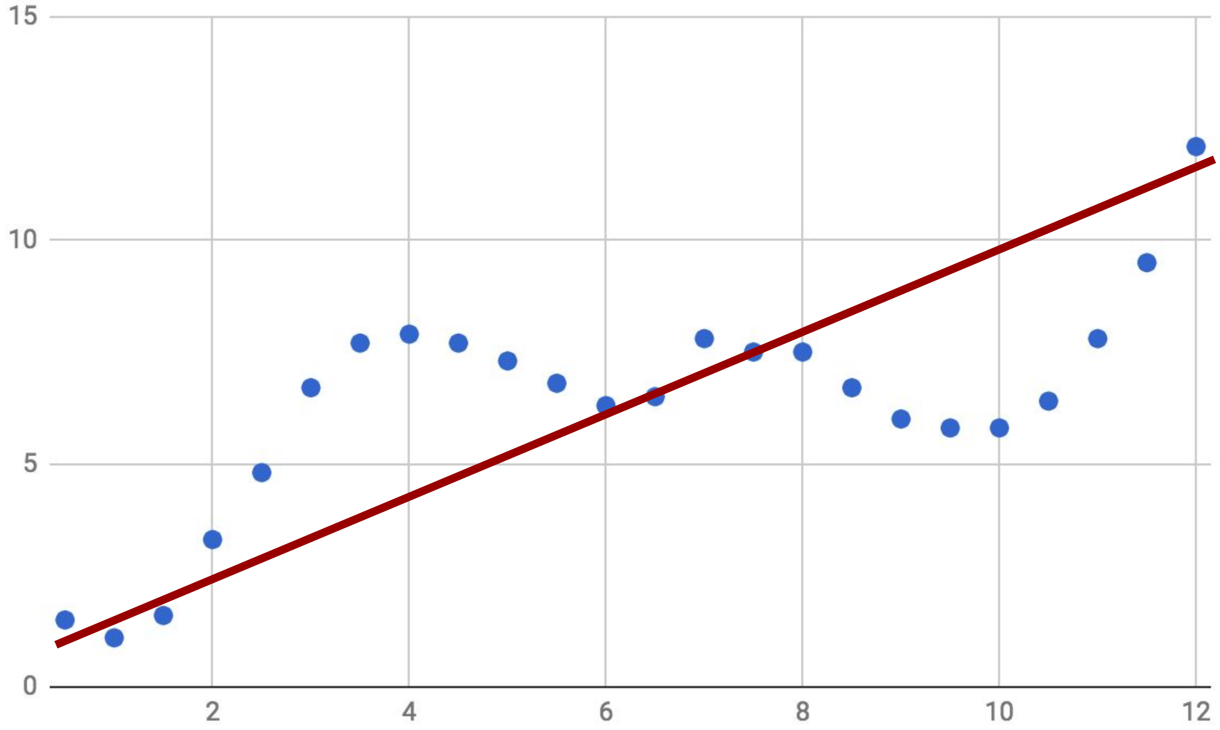
Generate Samples To Illustrate Over/Under fitting



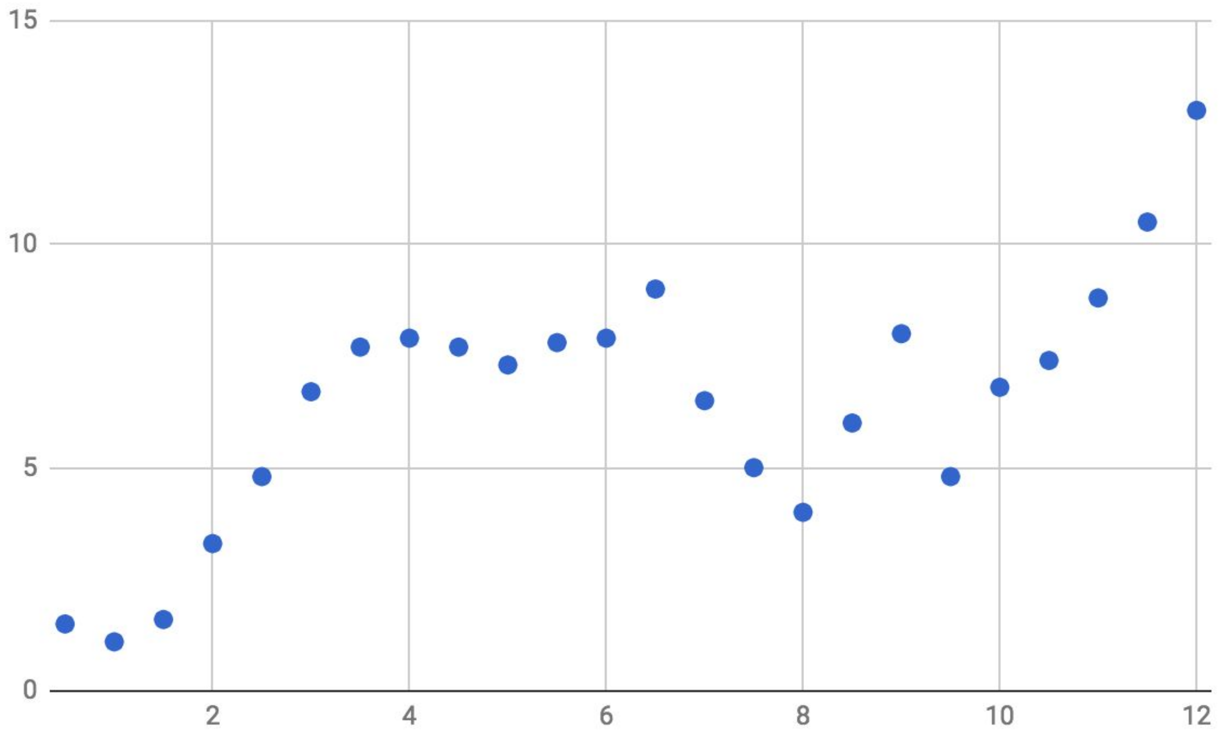
Underfitting



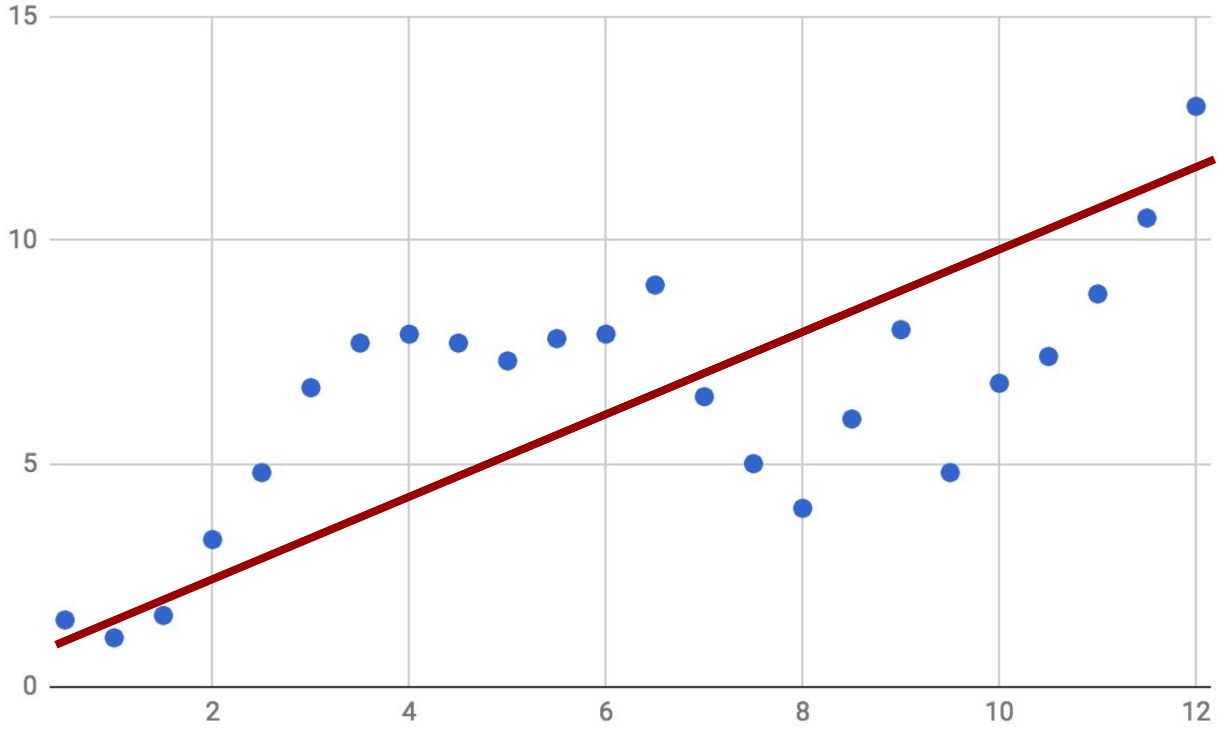
Underfitting: Too simple



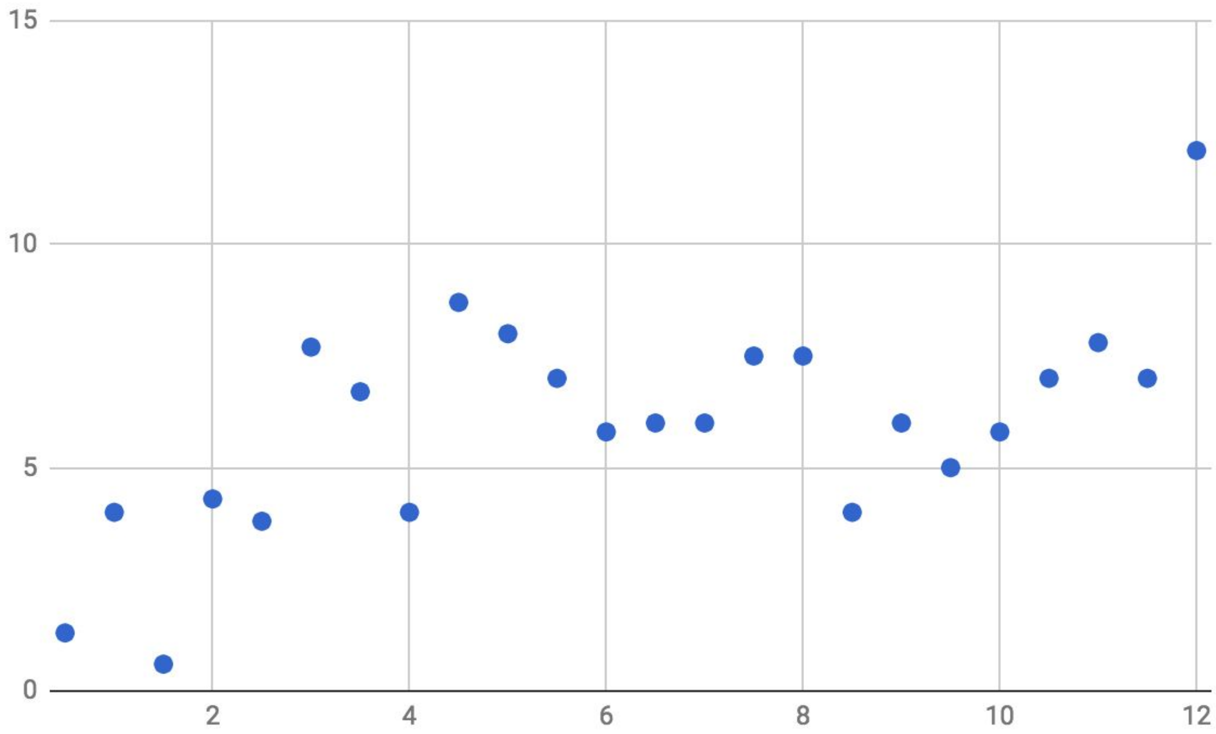
Underfitting



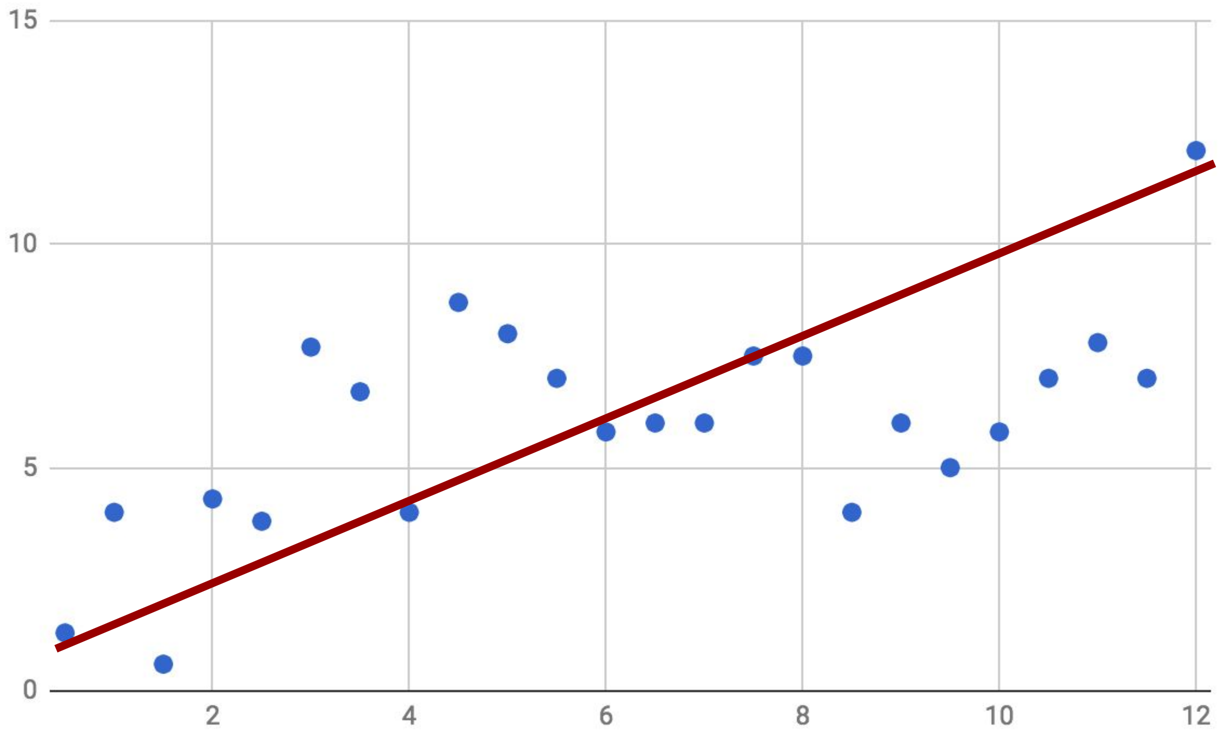
Underfitting: Too simple



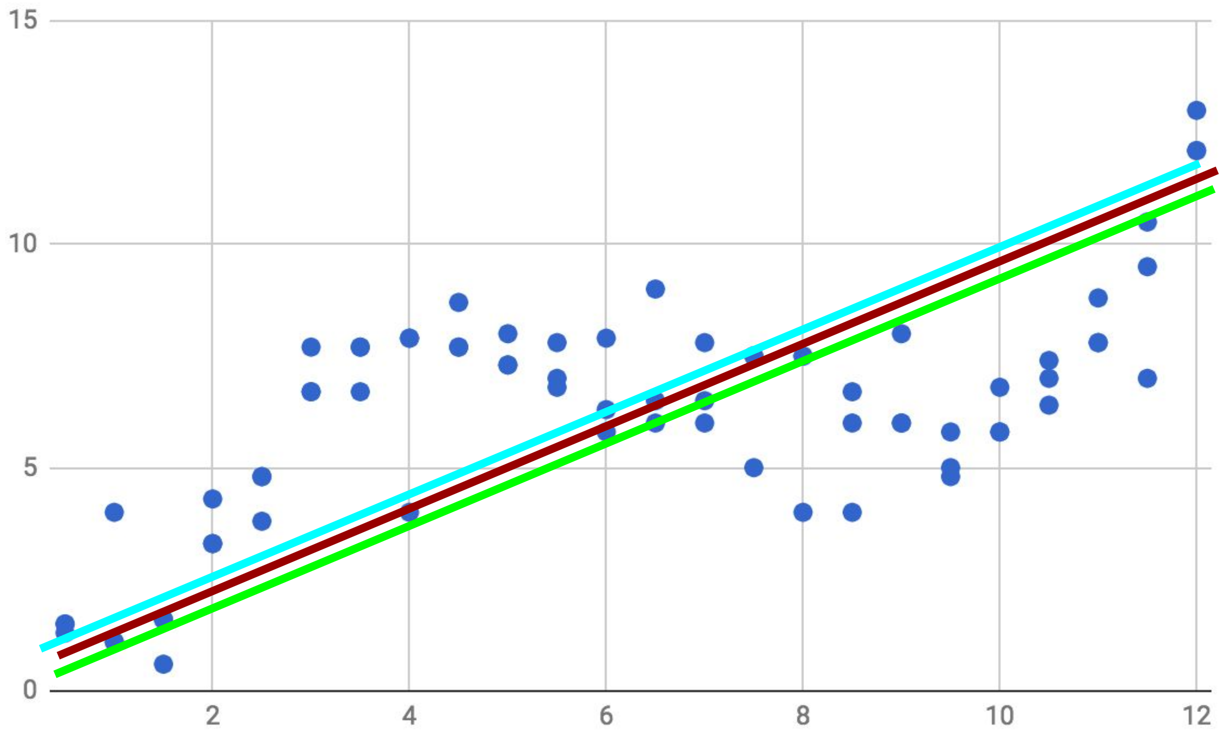
Underfitting



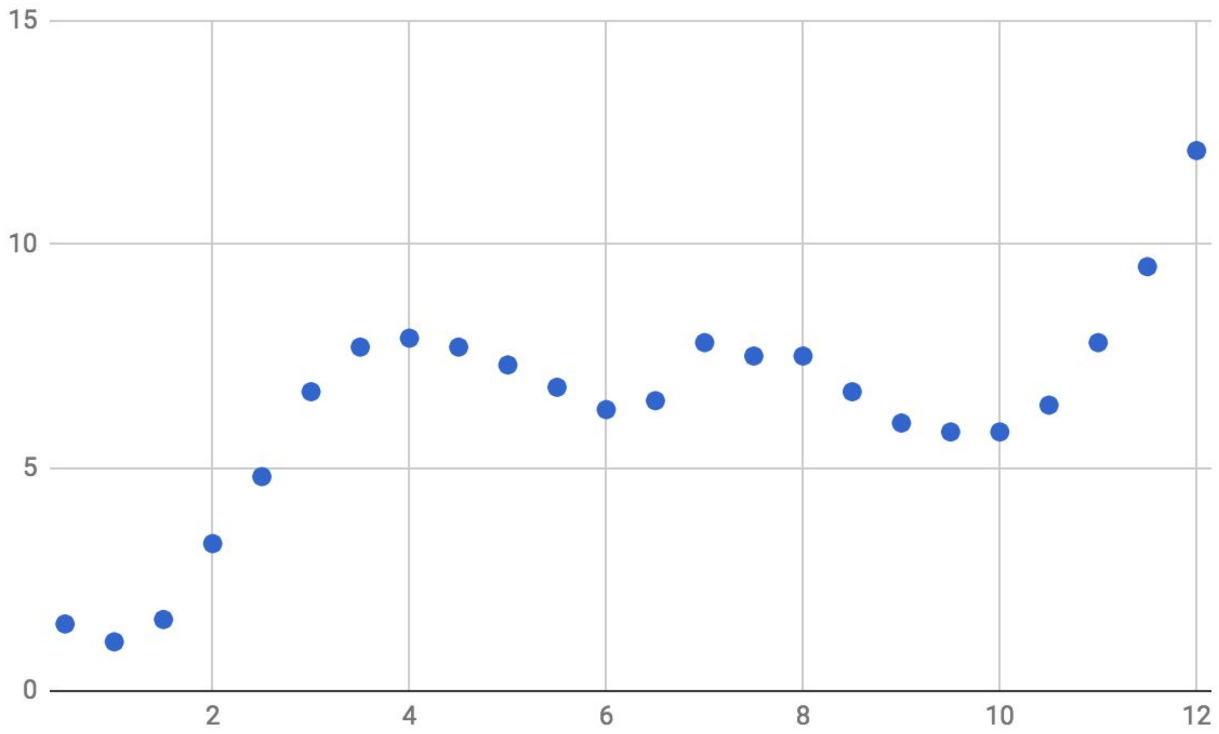
Underfitting: Too simple



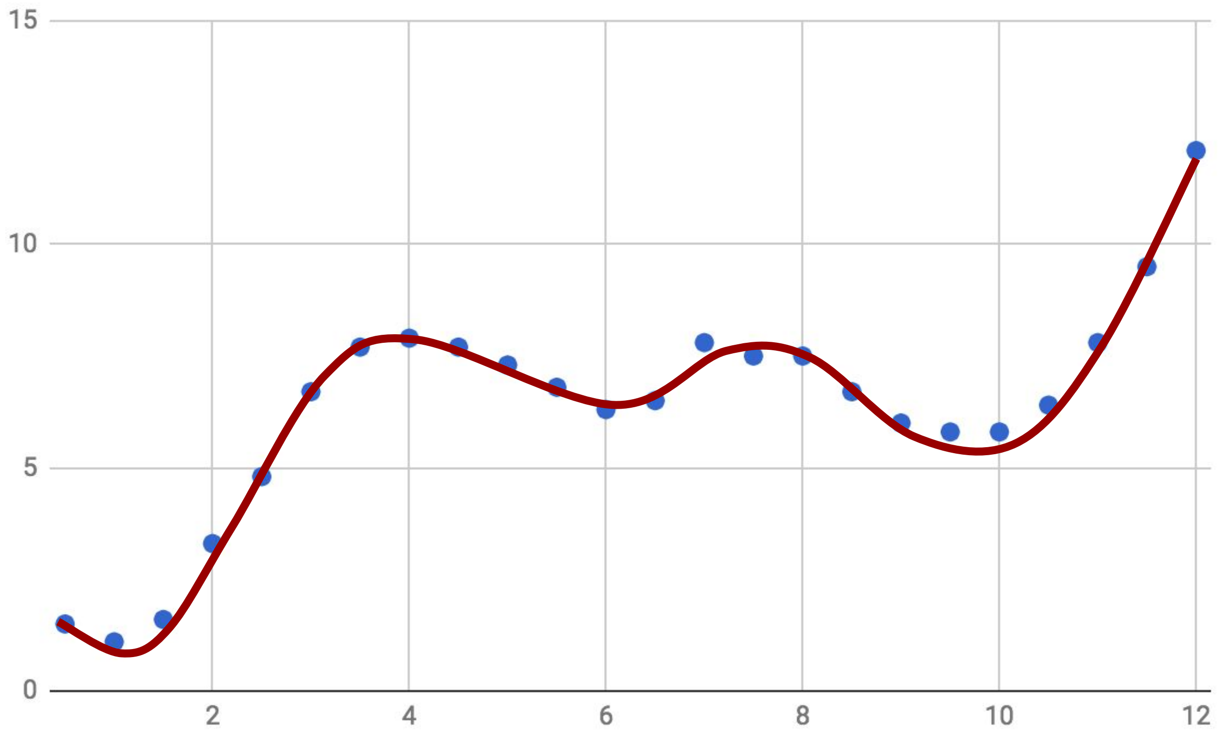
Underfitting: at least the models are consistent...



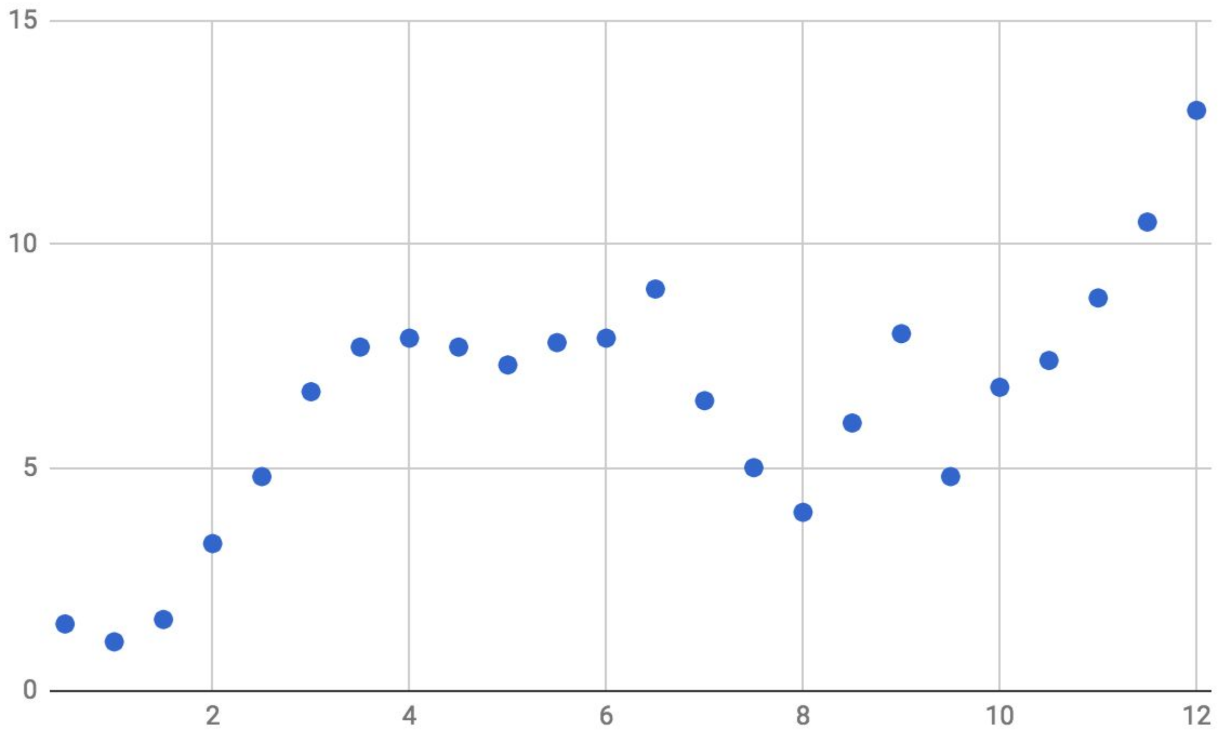
Overfitting



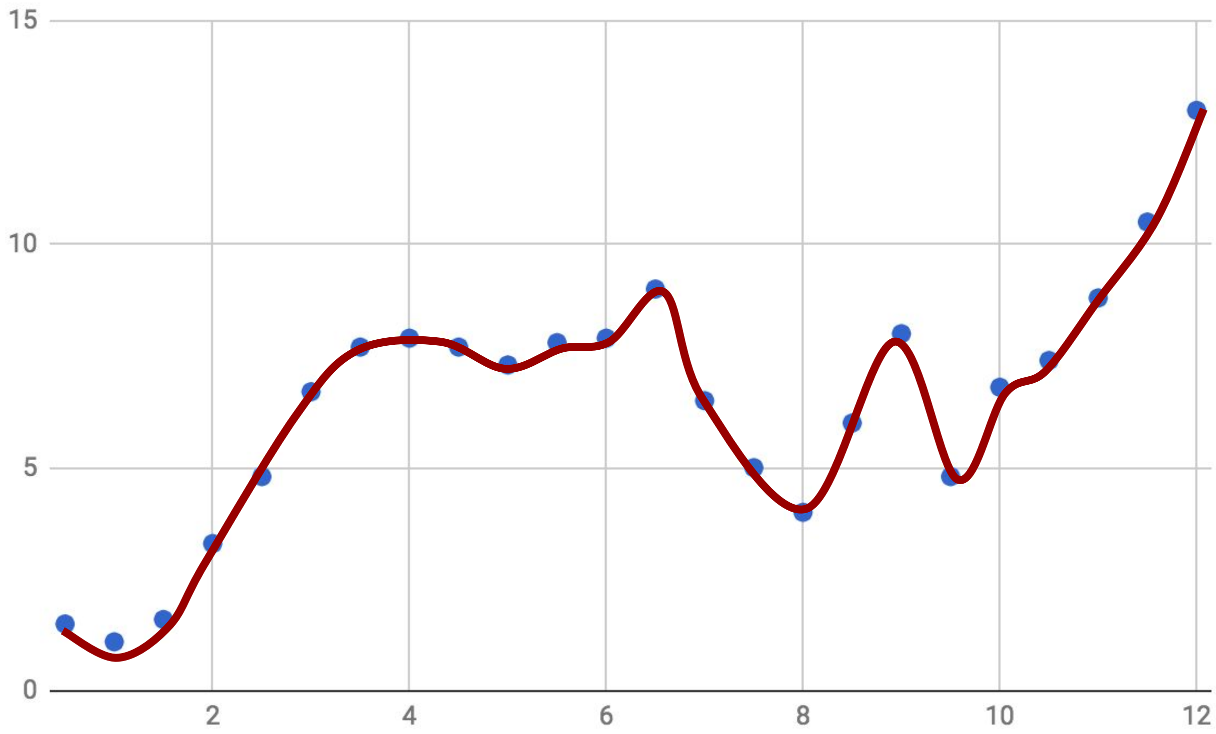
Overfitting



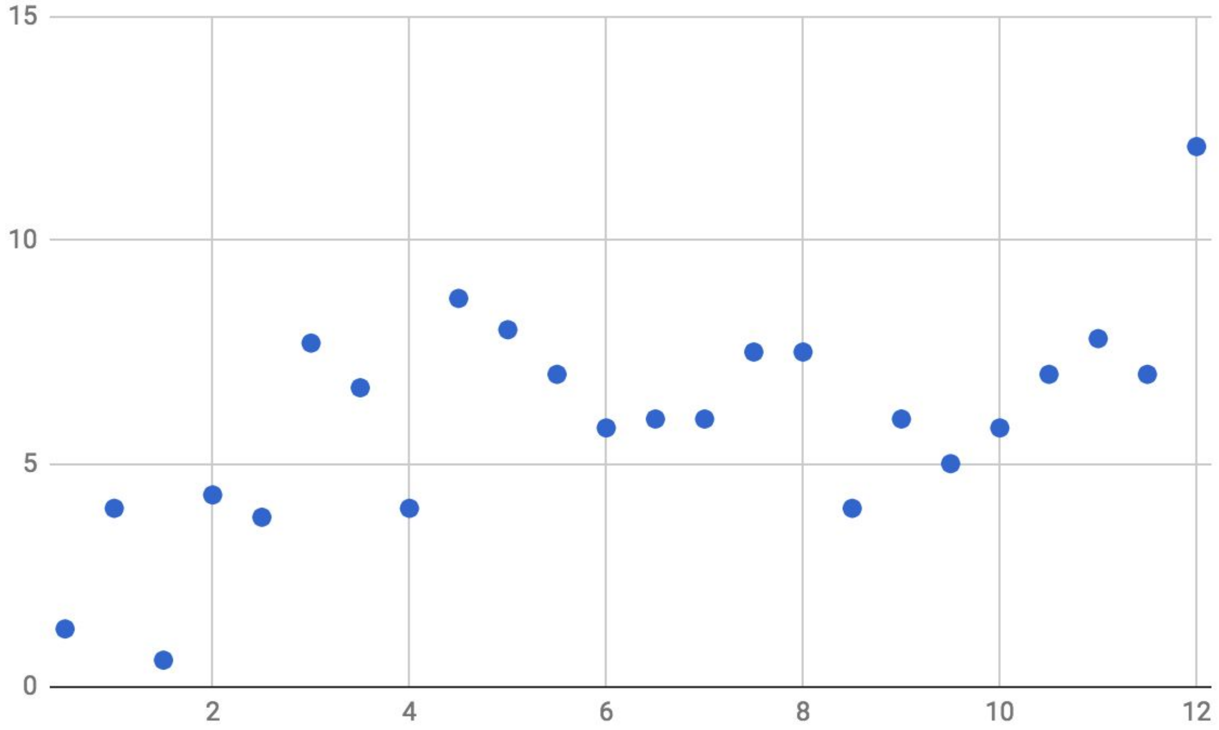
Overfitting



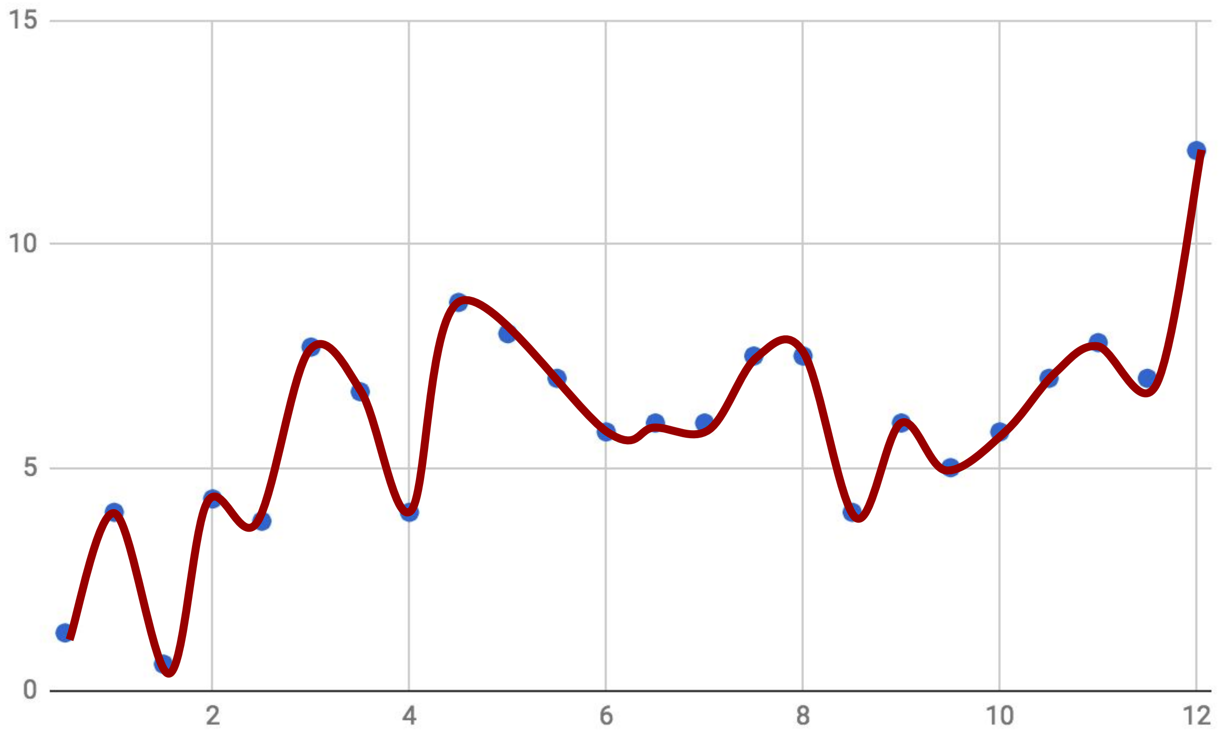
Overfitting



Overfitting

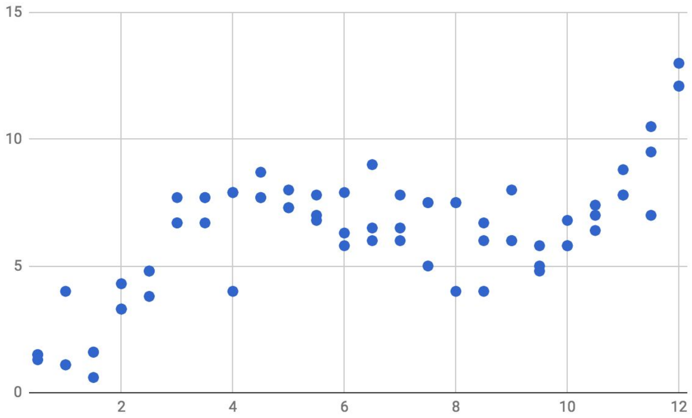


Overfitting

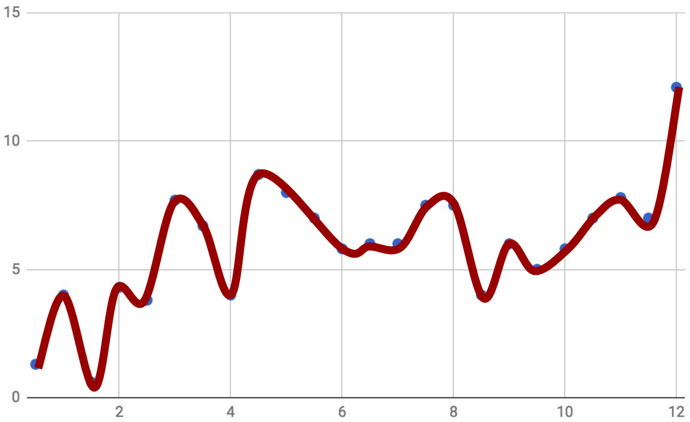


Overfitting: What's the issue?

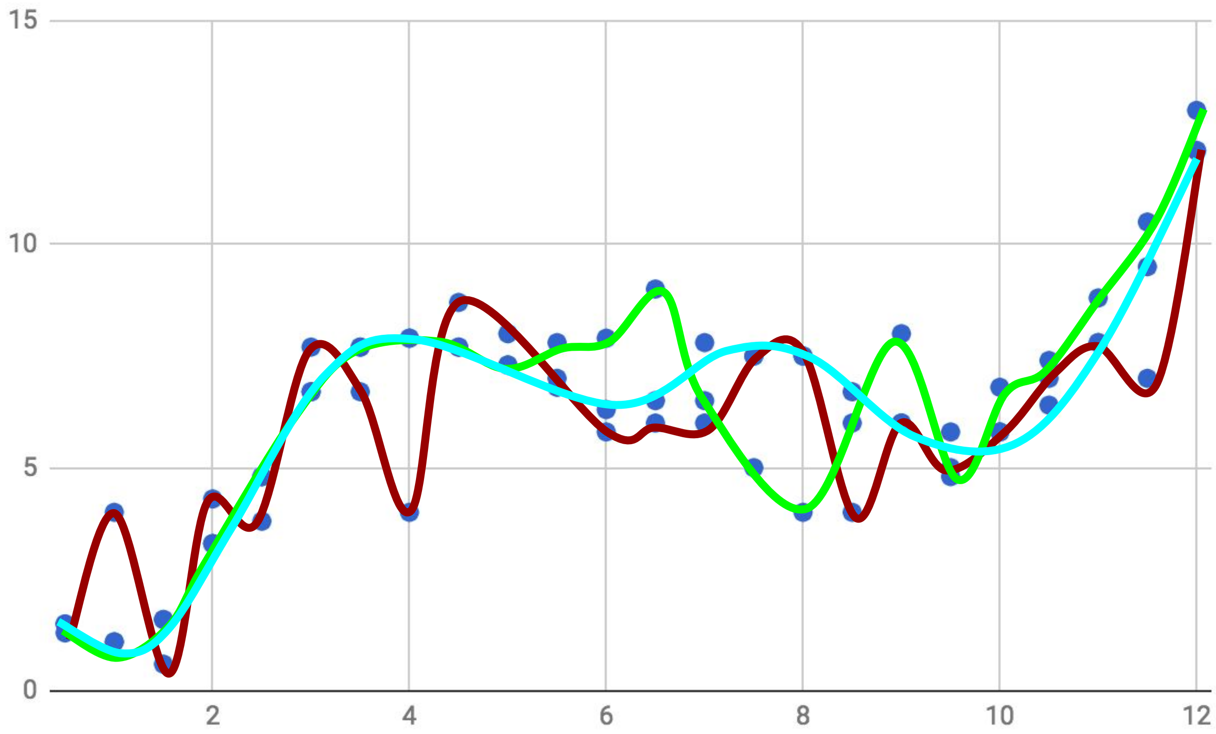
Data before sampling



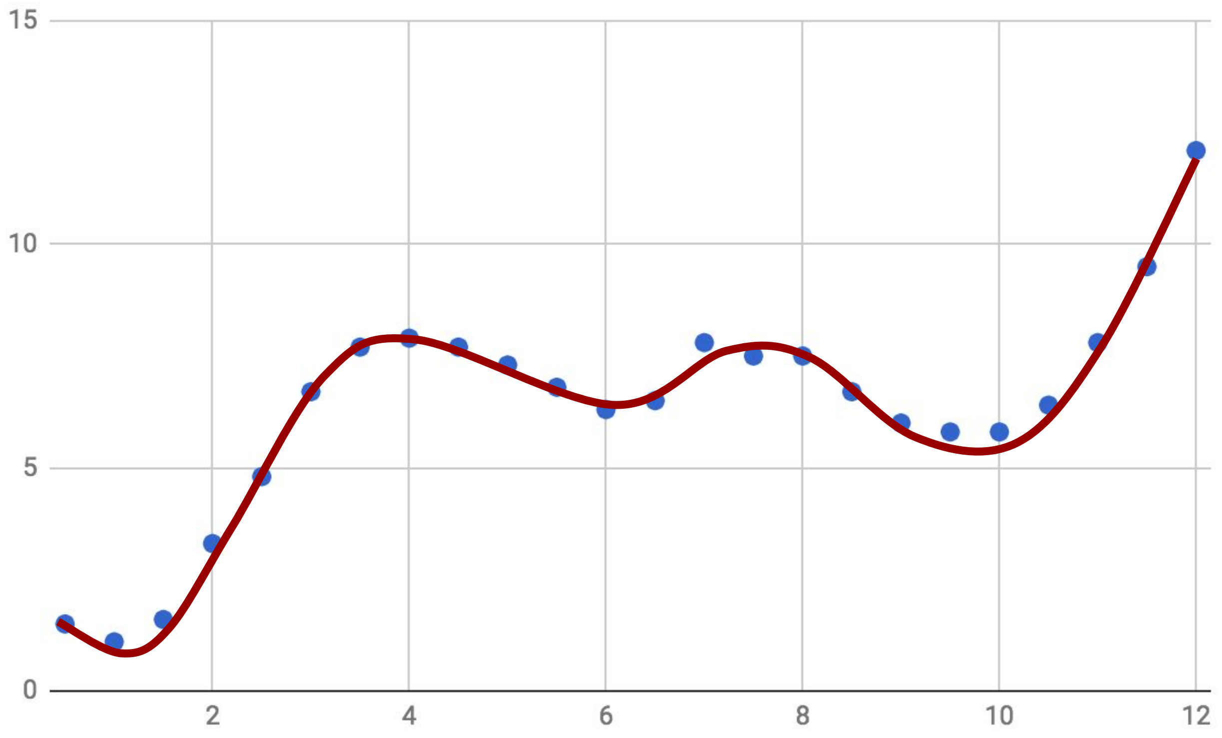
Model trained on sample



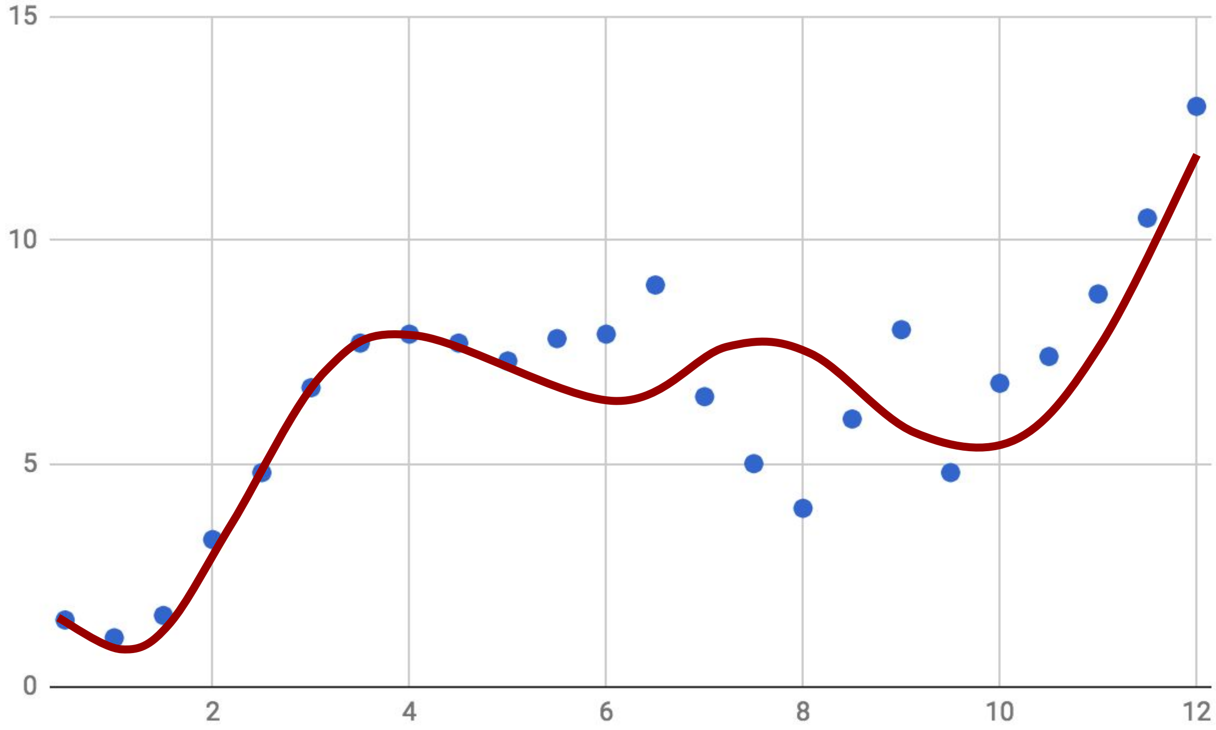
Overfitting: Inconsistent Models!

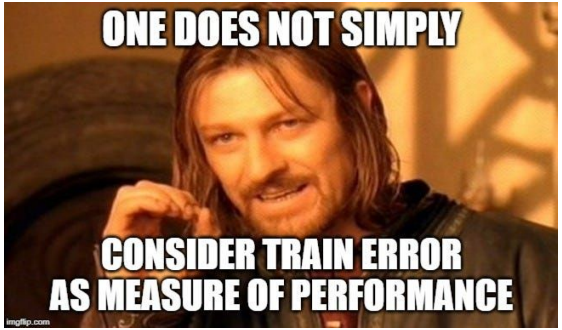


Overfitting: Results from training with high sensitivity



Overfitting: doesn't generalize well!





Understanding Model Error

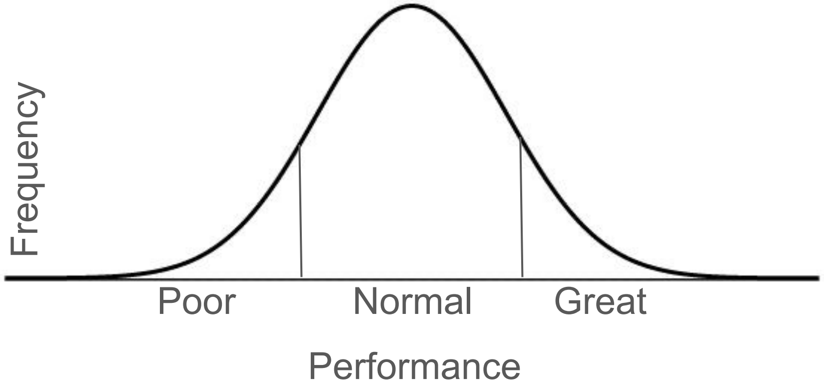
Expected Test Error Decomposition

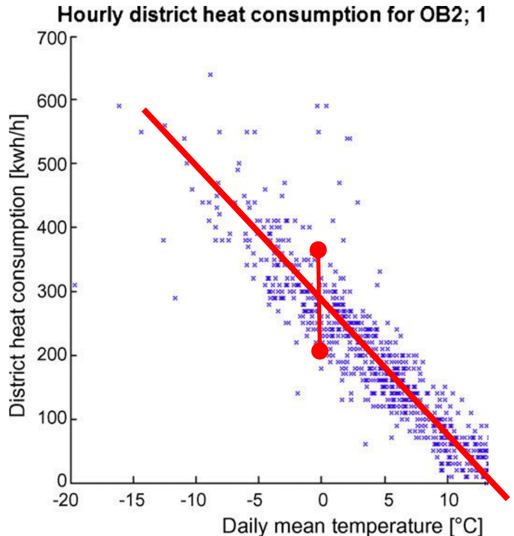
Framework for thinking about data:

- The world has randomness: data is randomly drawn from some distribution

→ general relation but with some variation

- Most things happen once, so we can only observe one of many the possible outcomes





Expected Test Error Decomposition

Bias

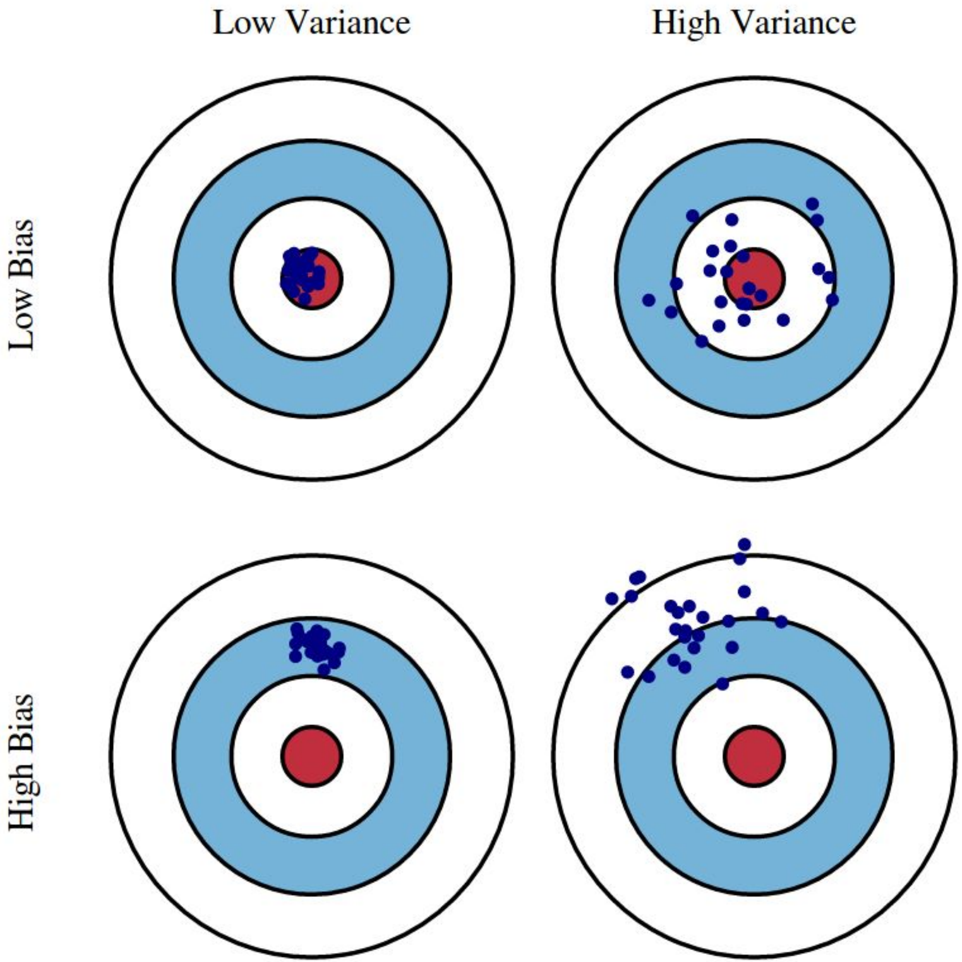
- Error that would still exist if you had an infinite amount of training data
- Inherent to the model
  - ex. We demonstrated high bias by using a linear classifier on non-linear data

Variance

- How would your model change if you had a different training set?
- Measures how specialized your model is to your specific training set

Noise

- Measures inherent ambiguity in the data distribution
- Cannot reduce “noise” by editing algorithm



[Source](#)

What does this mean intuitively?

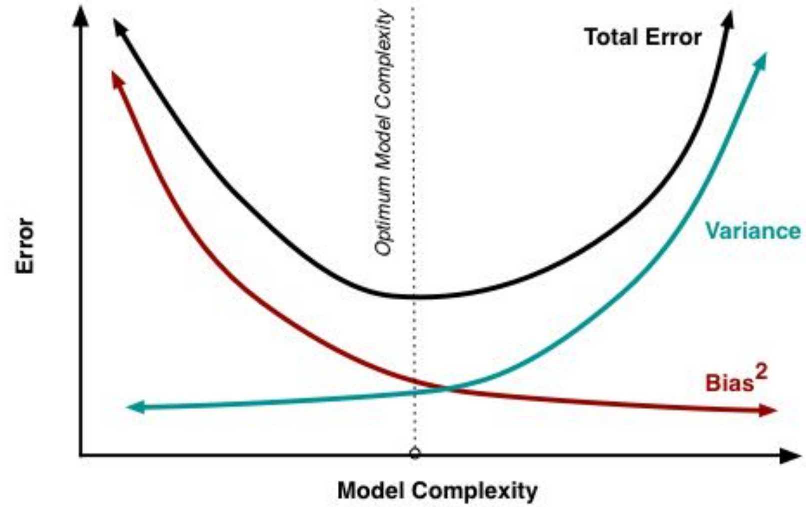
Bias

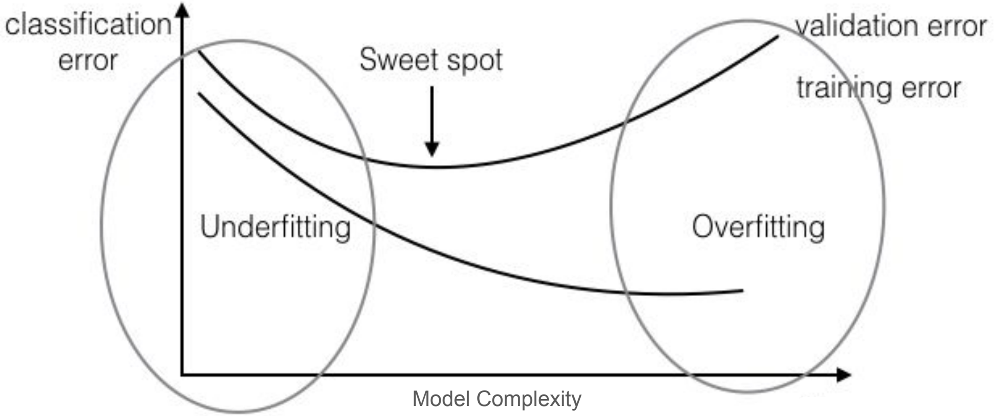
- Bad
- Results from incorrect assumptions in the learning algorithm

Variance

- Bad
- Results from sensitivity to fluctuations in the data

Balancing Bias and Variance

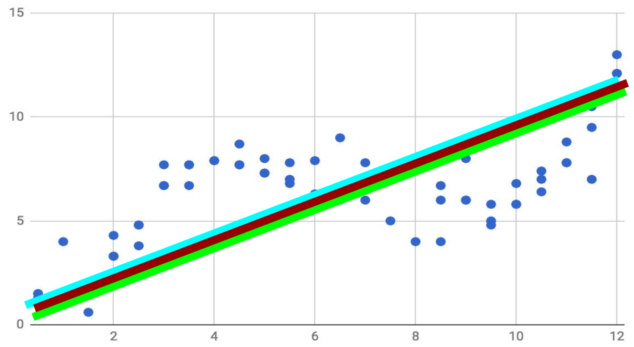




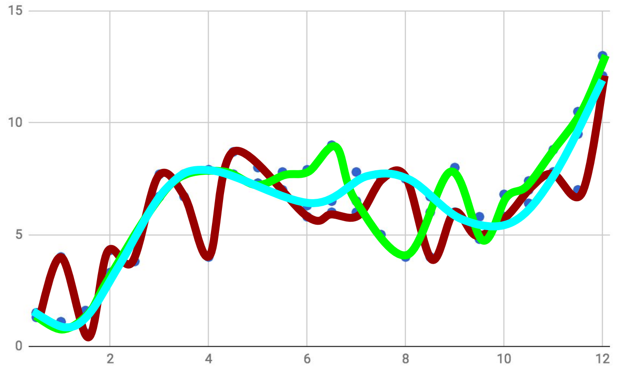
[Source]

Detecting and Resolving Bias and Variance

- If: High train error
  - Increase model complexity
  - Add more information (features)
  - Boost (later lecture)
  - Change model assumptions



- If: Train error << test error (and test error still too high)
  - Reduce model complexity
  - Add more training data
  - Bag (later lecture)



Bias Variance Trade Off

$$\mathrm{E}\left[\left(y-\hat{f}(x)\right)^2\right] = \mathrm{Bias}\left[\hat{f}(x)\right]^2 + \mathrm{Var}\left[\hat{f}(x)\right] + \sigma^2$$

$$\mathrm{Bias}\left[\hat{f}(x)\right] = \mathrm{E}\left[\hat{f}(x) - f(x)\right]$$

$$\mathrm{Var}\left[\hat{f}(x)\right] = \mathrm{E}[\hat{f}(x)^2] - \mathrm{E}[\hat{f}(x)]^2$$

Error = (expected loss of accuracy)^2 + inconsistency of model + irreducible error

Different Topic Ahead

Any questions before we continue

Feature Selection

(adjusting models)

Methods

- **Goal:** Find subset of features that gives a good enough model, in a reasonable amount of time.
- Why:
  - More interpretable
  - More stable results
  - Less redundant/potentially misleading data
  - Faster

Methods

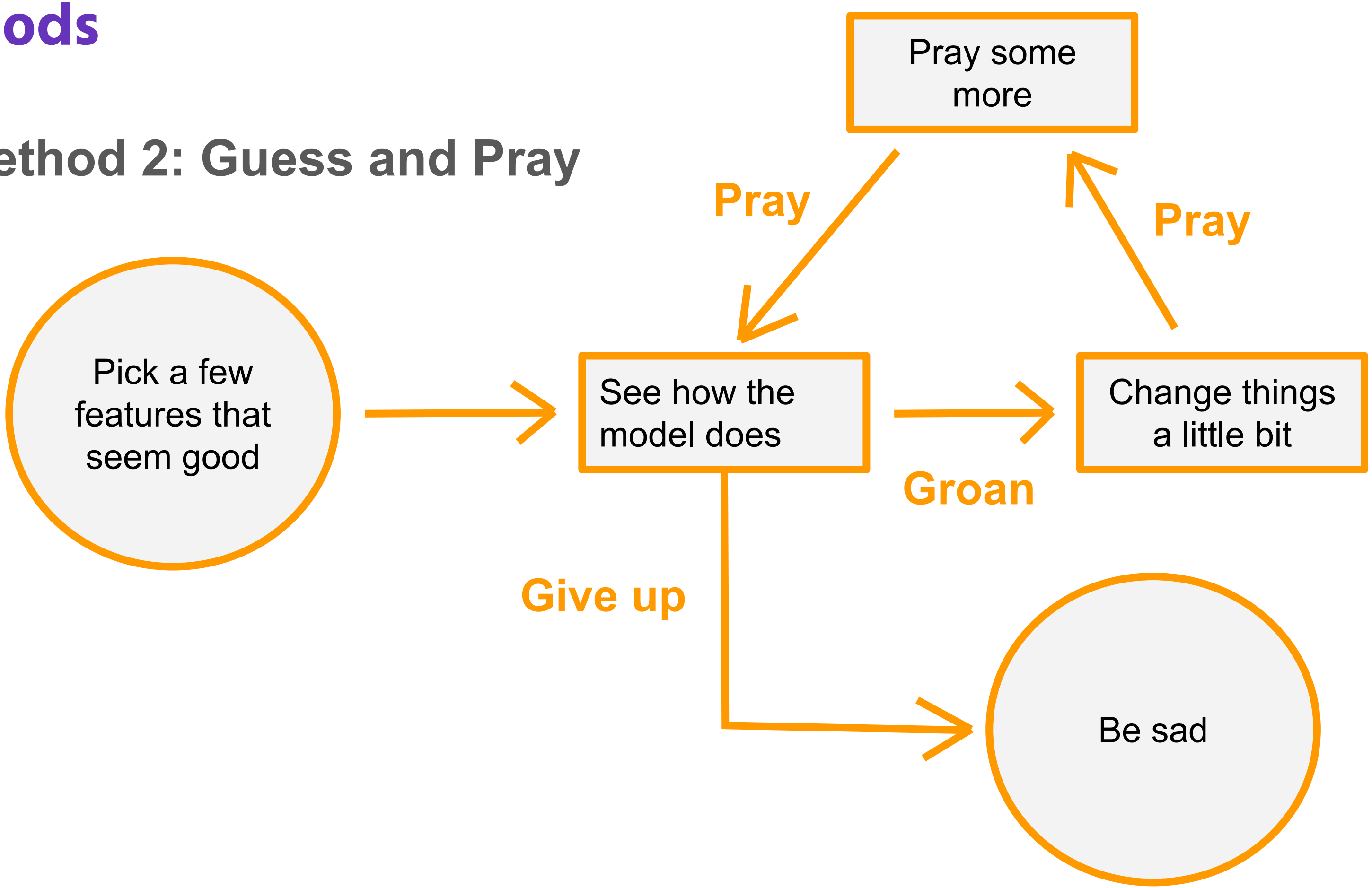
- **Goal:** Find subset of features that gives a good enough model, in a reasonable amount of time.

Methods

- **Goal:** Find subset of features that gives a good enough model, in a reasonable amount of time.
- **Method 1: Best Subset**
    - Test **all** subsets for best one
    - Benefits:
        - **Best** subset out of current features
    - Drawbacks:
        - Slow
        - Even slower with feature engineering

Methods

- **Method 2: Guess and Pray**



Methods

- **Goal:** Find subset of features that gives a good enough model, in a reasonable amount of time.
- **Method 2: Guess and Pray**
  - Guess
  - Benefits:
    - ??
  - Drawbacks:
    - Time consuming for data scientist
    - Unreliable

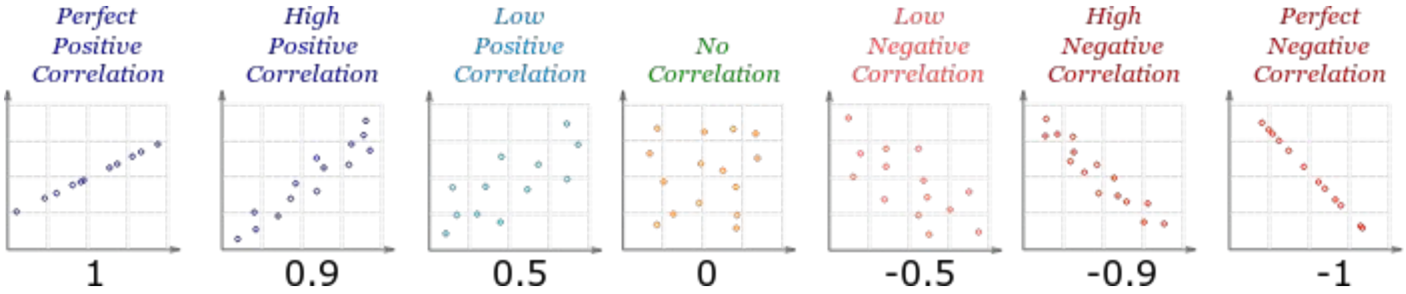
Methods

- **Goal:** Find subset of features that gives a good enough model, in a reasonable amount of time.
- **Method 3: Stepwise**
  - Pick a few features, then programmatically add/remove features using statistics
  - Benefits:
    - Complexity and runtime are adjustable
  - Drawbacks:
    - Can do very badly if you’re not careful
    - Requires more thinking

Correlation, r

The correlation between two variables describes to what extent changing one would change the other.

- Real-valued in [-1,1]
- A variable is always perfectly correlated with itself (correlation=1)



Important Case: Collinearity

Collinear: when two features have a correlation near -1 or 1

- If a feature is collinear with the target, then it's a good choice for linear regression
- If two features are collinear, they're *redundant*
  - Might as well not use one of them
  - Some models *require/assume* no collinear features
  - Takes more time, and doesn't add much information at the cost of *increased variance/sensitivity*

Side Note: Scaling and Normalizing

- Some models require data to be centered

- Some models need features to be on the same scale
  - Can divide by max, minus min divide by max minus min, minus mean divide by standard deviation.

Other Ways to Optimize Model

- Hyper Parameters
- Feature Engineering
- Changing to a different algorithm
  - Q: when should we do this?

Demo

Final Notes

Coming Up

- **Assignment 4:** Due tonight at midnight!
- **Assignment 5:** Due midnight next Friday (10/18)
- **Mid-Semester Check-In:** Now till Wednesday (10/23)
- **Next Lecture**: Intro to Classification

Have a great Fall Break!!