GENERAL ASSEMBLY

EVALUATING SUPERVISED MODELS

Mason Gallo, Data Scientist

AGENDA

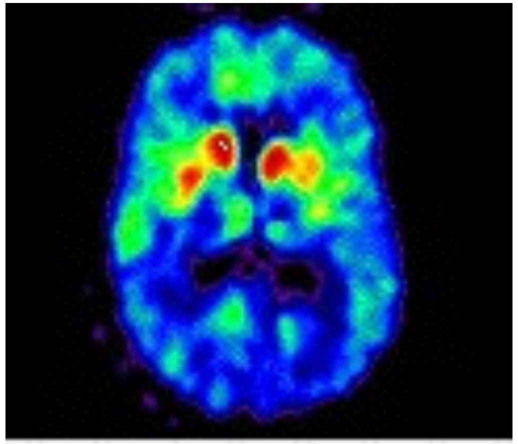
- Formalize Bias/Variance Tradeoff
- Complexity vs Interpretability
- Underfitting vs Overfitting
- Cross Validation
- Curse of Dimensionality

OBJECTIVES

- Bias/Variance and complexity tradeoff big picture
- Follow the thought process for evaluating a model used by top ML researchers
- Evaluate the performance of a model in Python

MOTIVATING EXAMPLES: DETECTING PARKINSON’S

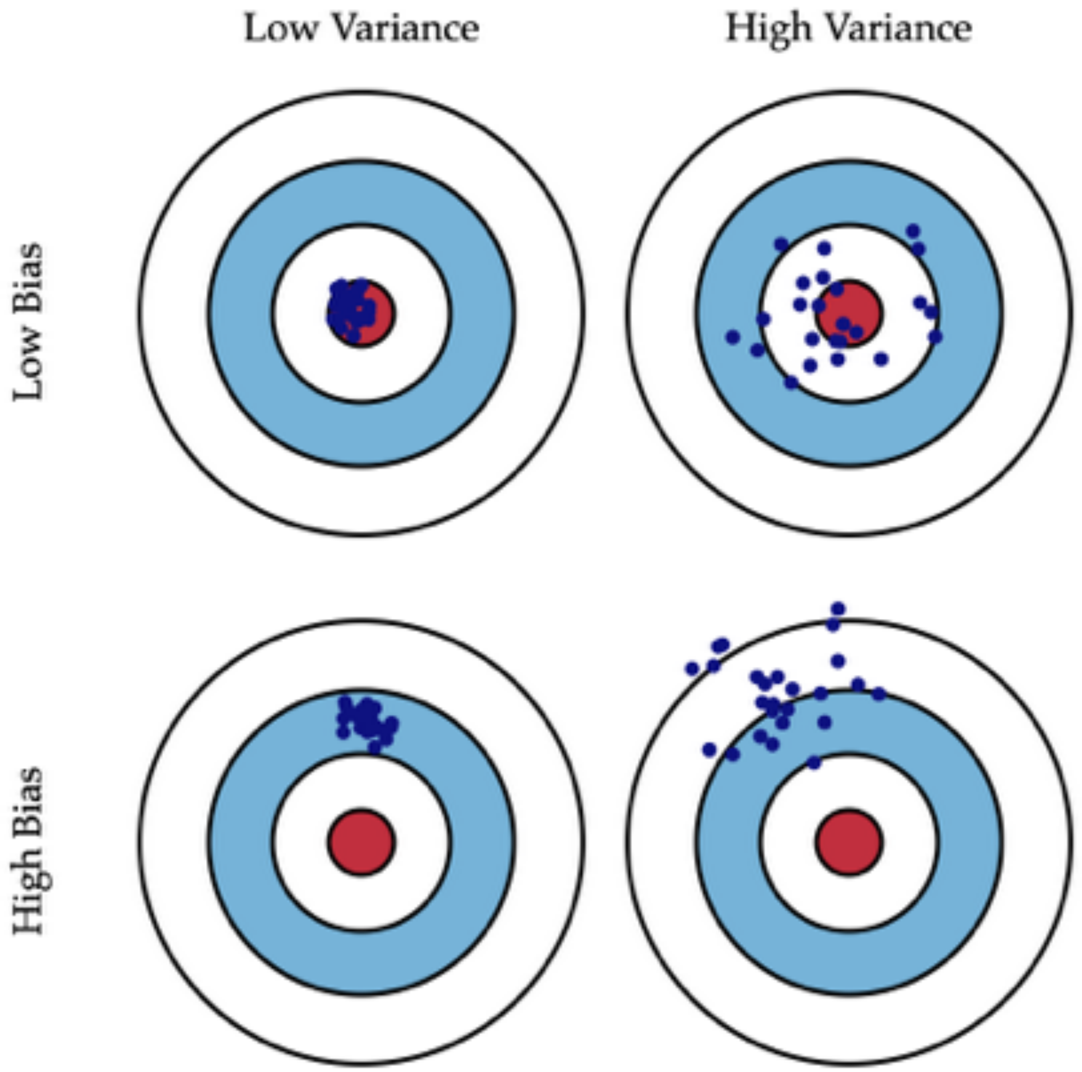
PARKINSON’S DISEASE



YOU WILL BUILD A LEARNER THAT PREDICTS THE DISEASE BASED ON VOCAL CHORD MEASUREMENTS

HOW DO WE KNOW HOW GOOD OUR MODELS ARE?

THINK ABOUT ERROR IN TERMS OF BIAS AND VARIANCE



FORMAL DEFINITIONS

The error due to BIAS is the difference between our prediction and the actual

Think: how close are our predictions to what we want over time?

The error due to VARIANCE is the difference between realizations of the same model

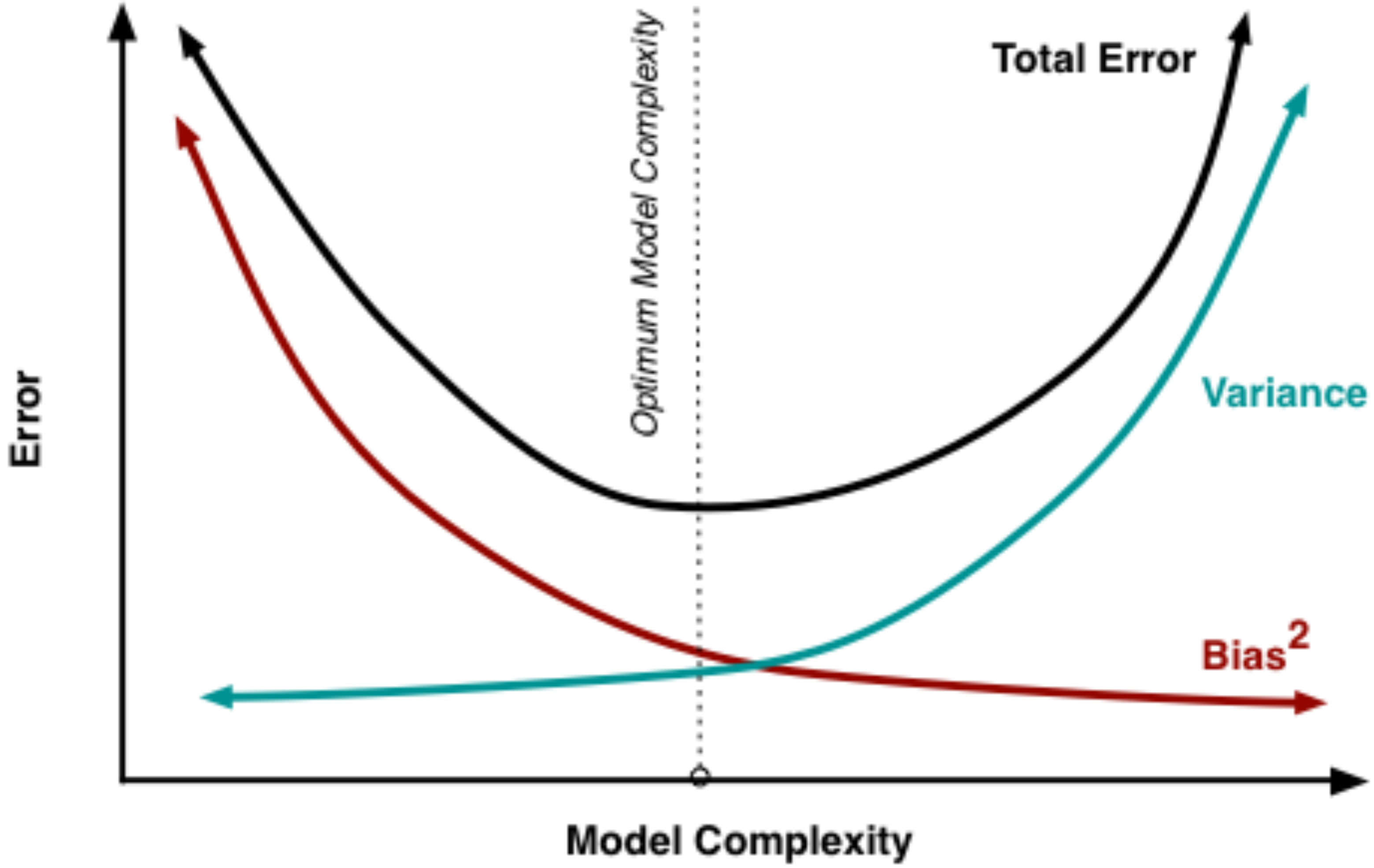
Think: how spread out are our predictions over time?

FORMAL DEFINITIONS

The error due to BIAS is the difference between our prediction and the actual

The error due to VARIANCE is the difference between realizations of the same model

RELATIONSHIP BETWEEN BIAS AND VARIANCE



HOW I THINK ABOUT BIAS AND VARIANCE

High bias means we practically ignore the training data

Intuition: we're good at improvising with little to go on

HOW I THINK ABOUT BIAS AND VARIANCE

High bias means we practically ignore the training data

Intuition: we’re good at improvising with little to go on

High variance means we practically memorize the training data

Intuition: we fail at anything we haven’t seen before

HOW I THINK ABOUT BIAS AND VARIANCE

What’s better for low amount of training data - high bias or high variance?

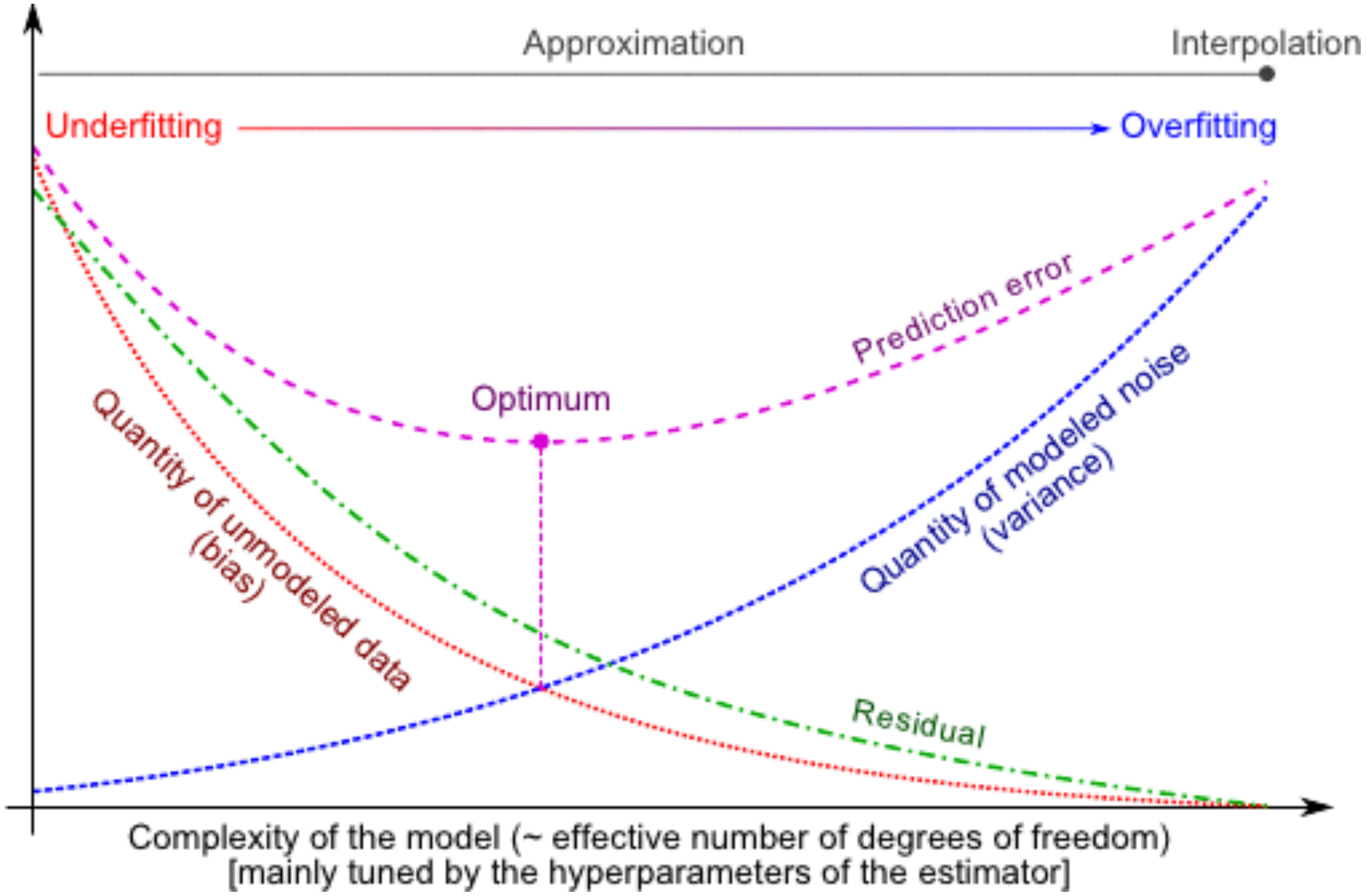
HOW I THINK ABOUT BIAS AND VARIANCE

What’s better for low amount of training data - high bias or high variance?

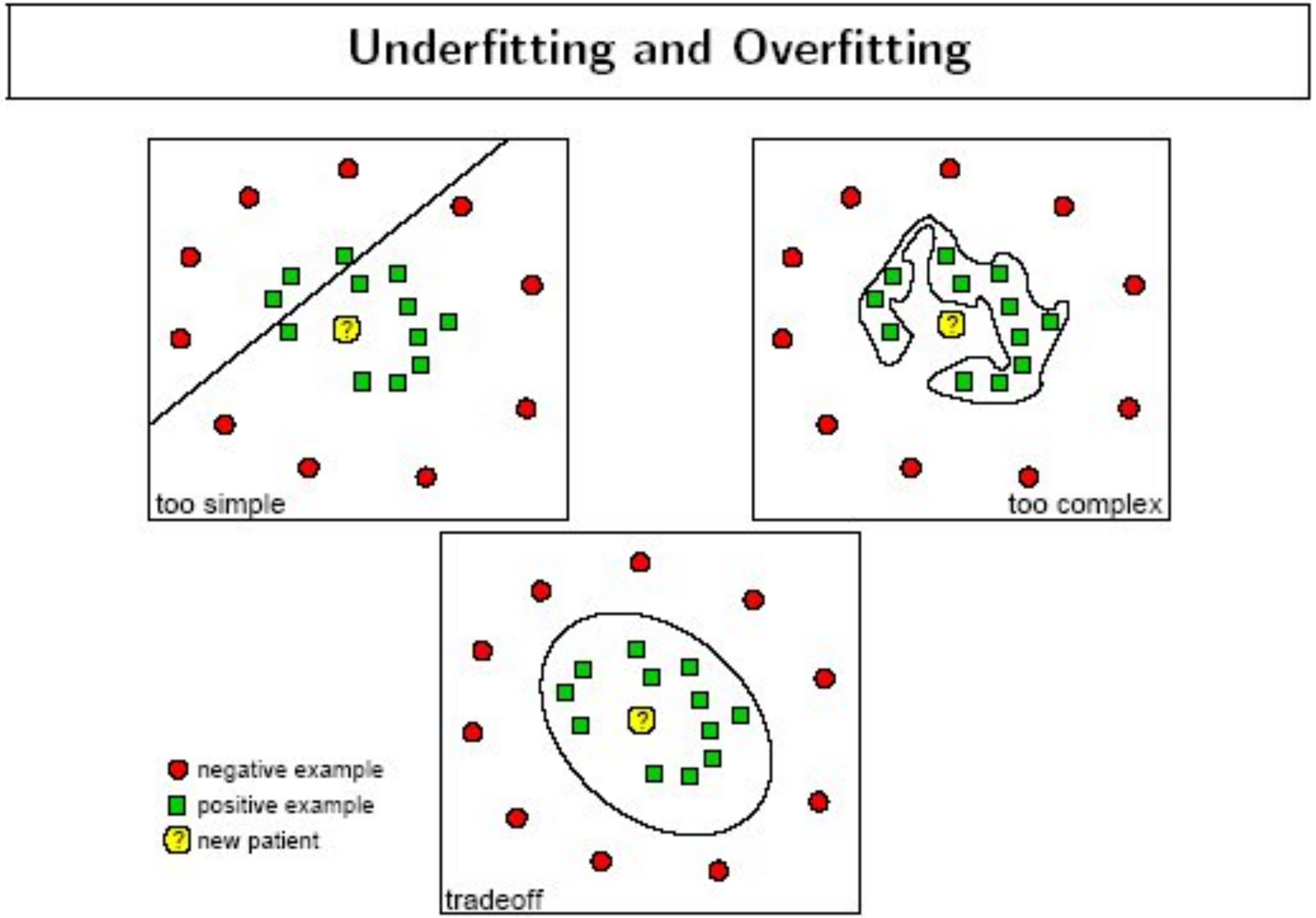
High bias is better because we are more comfortable with “unseen” data

Intuition:
If we have a low amount of training data, it’s more likely we’ll have “unseen” data

RELATIONSHIP BETWEEN BIAS AND VARIANCE



OVERFITTING - EXAMPLE



source: http://www.dtreg.com

OVERFITTING AND UNDERFITTING

If we have small number rows and many columns we risk OVERFITTING

We risk fitting a model that's overly complex with the small amount of data

OVERFITTING AND UNDERFITTING

If we have small number rows and many columns we risk OVERFITTING

We risk fitting a model that's overly complex with the small amount of data

If we have high number rows and few columns we risk UNDERFITTING

We risk fitting a model that's too simple

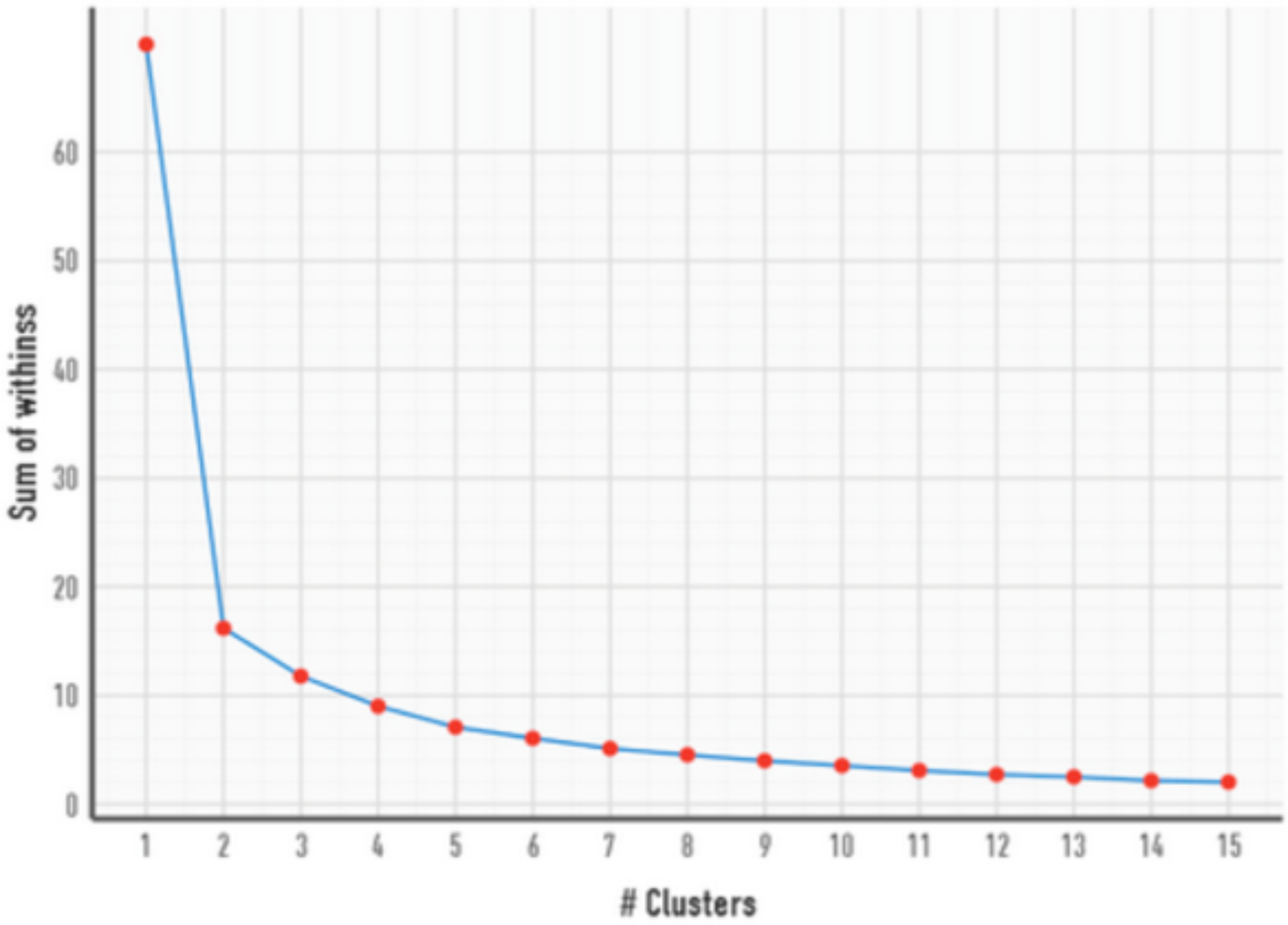
A NOTE ABOUT COMPLEXITY VS INTERPRETABILITY

Consider your audience:

Is it worth getting that last .00001% if it makes the model harder to explain?

Is it worth adding more predictors if it makes the story less clear?

A NOTE ABOUT COMPLEXITY VS INTERPRETABILITY



Where you stop depends on your audience

HOW SHOULD WE CONSTRUCT THE EXPERIMENT?

RECALL OUR TEST/TRAIN DISCUSSION FROM LAST CLASS

What if we change the portion that we allocate as a test set?

Will our conclusions change?

GENERALIZATION ERROR

Suppose we do the train/test split.

Q: How well does generalization error predict OOS accuracy?

Thought experiment:

Suppose we had done a different train/test split.

Q: Would the generalization error remain the same?

A: Of course not!

A: On its own, not very well.

NOTE

The generalization error gives a *high-variance estimate* of OOS accuracy.

Something is still missing!

Something is still missing!

Q: How can we do better?

GENERALIZATION ERROR

Something is still missing!

Q: How can we do better?

Thought experiment:

Different train/test splits will give us different generalization errors.

Something is still missing!

Q: How can we do better?

Thought experiment:

Different train/test splits will give us different generalization errors.

Q: What if we did a bunch of these and took the average?

GENERALIZATION ERROR

Something is still missing!

Q: How can we do better?

Thought experiment:

Different train/test splits will give us different generalization errors.

Q: What if we did a bunch of these and took the average?

A: Now you're talking!

Something is still missing!

Q: How can we do better?

Thought experiment:

Different train/test splits will give us different generalization errors.

Q: What if we did a bunch of these and took the average?

A: Now you're talking!

A: Cross-validation.

Steps for n-fold cross-validation:

Steps for n-fold cross-validation:

1) *Randomly split the dataset into n equal partitions.*

Steps for n-fold cross-validation:

1) *Randomly split the dataset into n equal partitions.*
2) *Use partition 1 as test set & union of other partitions as training set.*

Steps for n-fold cross-validation:

1) *Randomly split the dataset into n equal partitions.*
2) *Use partition 1 as test set & union of other partitions as training set.*
3) *Find generalization error.*

Steps for n-fold cross-validation:

1) *Randomly split the dataset into n equal partitions.*
2) *Use partition 1 as test set & union of other partitions as training set.*
3) *Find generalization error.*
4) *Repeat steps 2-3 using a different partition as the test set at each iteration.*

Steps for n-fold cross-validation:

1) *Randomly split the dataset into n equal partitions.*
2) *Use partition 1 as test set & union of other partitions as training set.*
3) *Find generalization error.*
4) *Repeat steps 2-3 using a different partition as the test set at each iteration.*
5) *Take the average generalization error as the estimate of OOS accuracy.*

Features of n-fold cross-validation:

Features of n-fold cross-validation:

1) *More accurate estimate of OOS prediction error.*

Features of n-fold cross-validation:

1) *More accurate estimate of OOS prediction error.*
2) *More efficient use of data than single train/test split.*
   - *Each record in our dataset is used for both training and testing.*

Features of n-fold cross-validation:

1) More accurate estimate of OOS prediction error.
2) More efficient use of data than single train/test split.
   - Each record in our dataset is used for both training and testing.
3) Presents tradeoff between efficiency and computational expense.
   - 10-fold CV is 10x more expensive than a single train/test split

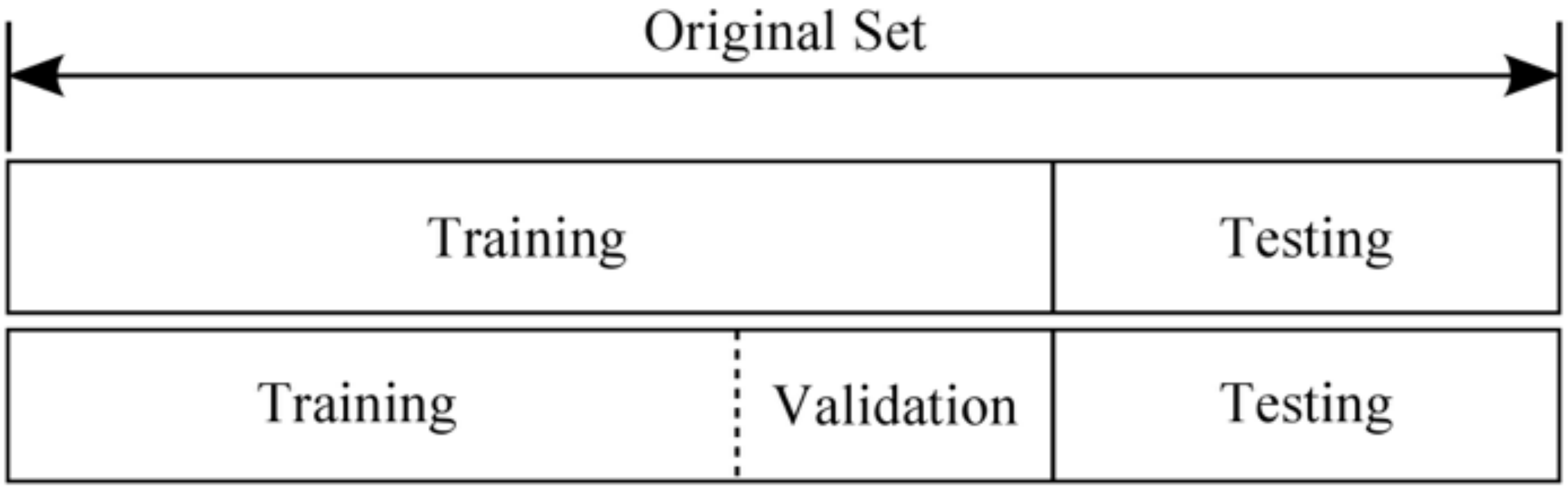
Features of n-fold cross-validation:

1) *More accurate estimate of OOS prediction error.*
2) *More efficient use of data than single train/test split.*
   - *Each record in our dataset is used for both training and testing.*
3) *Presents tradeoff between efficiency and computational expense.*
   - *10-fold CV is 10x more expensive than a single train/test split*
4) *Can be used for model selection.*

ANYTIME YOU RUN A ML EXPERIMENT, YOU NEED:

- A test set that you NEVER touch until the final evaluation

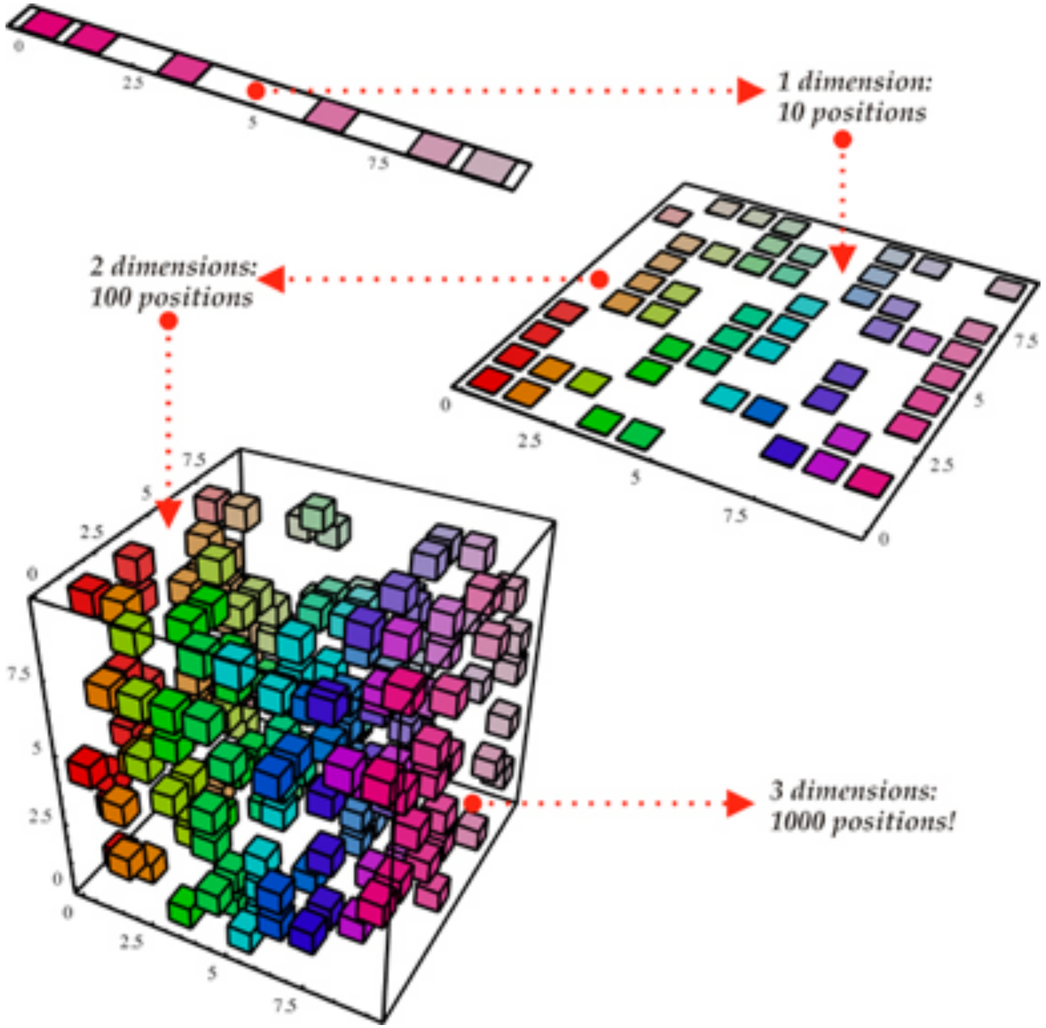
- The remaining data are split into a training set / validation set



THE CURSE

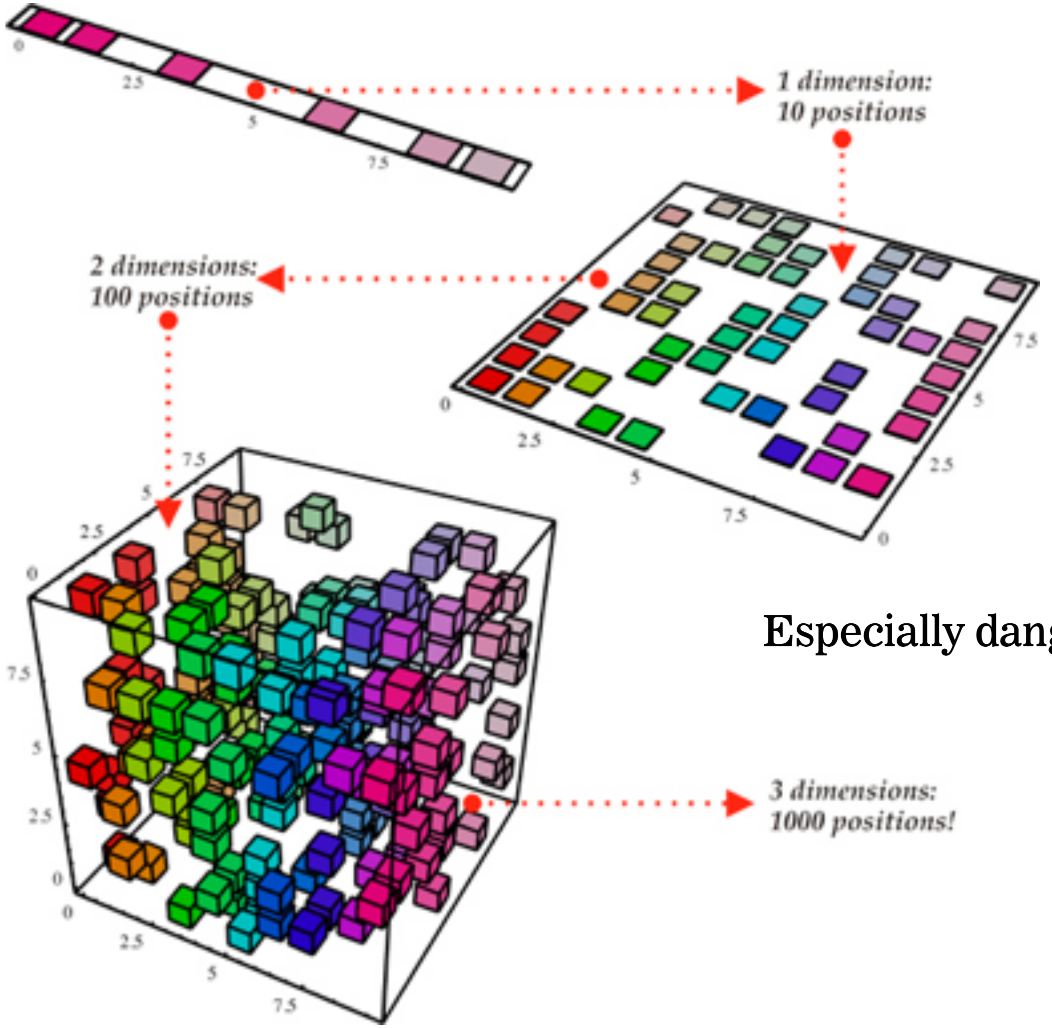
CURSE OF DIMENSIONALITY VISUALLY

As we keep adding features (dimensions), it gets harder to make predictions



CURSE OF DIMENSIONALITY VISUALLY

As we keep adding features (dimensions), it gets harder to make predictions



Especially dangerous for methods that rely on distance!

CURSE OF DIMENSIONALITY IN WORDS

Let's say you have a straight line 100 yards long and you dropped a penny.

It wouldn't be too hard to find. You walk along the line and it takes few minutes.

Now let's say you have a square 100 yards on each side and you dropped a penny.

It would be pretty hard, like searching across two football fields stuck together.

Now a cube 100 yards across.

That's like searching a 30-story building the size of a football stadium. Yikes!

TAKEAWAYS

We need to be careful of overly complex and overly simple models

Do we have enough data?
Are we using irrelevant or overly complex features?
Did we tune our model using cross validation and measure performance with the test set?

This is a constant balance and you should always be paranoid!

ALMOST ANY PROBLEM WILL BE: OVERFITTING OR CURSE OF DIMENSIONALITY

LET'S CODE!