Concepts in Statistical Modeling

Ryan Miller

Big Picture

Supervised Learning:

- Outcome variable - goal is to model or predict the outcome variable
  - *regression problems* where the outcome is quantitative (numeric)
  - *classification problems* where the outcome is categorical (factor)

Unsupervised Learning:

- No outcome variable - goal is to identify hidden patterns, trends, or groups in the data

Philosophy

- Any model is a simplification of reality
  - Work from simple to complex
- Understand the ideas behind various techniques
  - Knowing how various methods work helps you know when to apply them
- Understand how well a method is working
  - Whether to settle on a method, scrap it for a new one, or collect new data

What is a Statistical Model?

In general, statistical models can be expressed:

$$Y = f(\mathbf{X}) + \epsilon$$

- Y is an outcome (response) variable
- ϵ is a random error term
  - ϵ describes uncertainty in Y that cannot be explained by $f(\mathbf{X})$
- f is a fixed but unknown function of $\mathbf{X} = \{X_1, X_2, \ldots, X_p\}$
  - f is represents the *systematic* information $\mathbf{X}$ provides about Y
  - Estimating f is the primary focus of statistical modeling

Prediction and Inference

There are two main reasons why we might estimate f, *prediction* and *inference*

- For prediction, we can treat f as a black box
  - Details of $\hat{f}$ are less important than our predictions, $\hat{Y} = f(\mathbf{X})$, being close to Y
- One way to quantify if "$\hat{Y}$ is close to Y" is *mean squared prediction error*:

$$\text{MSPE} = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{f}(x_i)\right)^2$$

Prediction

Mathematically, it's easily shown that:

$$\text{MSPE} = (f(\mathbf{X}) - \hat{f}(\mathbf{X}))^2 + Var(\epsilon)$$

- The first term is sometimes called *reducible error*, while the second is *irreducible error*
- All models focus on minimizing reducible error
- It is important to know that irreducible error will always limit the accuracy of our predictions

Inference

Another goal of modeling is *inference*, or understanding *how* Y is influenced by $\mathbf{X}$

1. Determining which components of $\mathbf{X}$ are important and which are irrelevant
2. Understanding how each component affects Y

- Consider the simple linear regression model: $Y = \beta_0 + \beta_1 X_1 + \epsilon$
  - Here $f(\mathbf{X}) = \beta_0 + \beta_1 X_1$
- Inference is straightforward using this model:
  - ANOVA testing can identify important variables
  - 1 unit increases in X_1 lead to expected β_1 unit increases in Y
  - Estimating β_1 is all that is necessary to describe the relationship between X_1 and Y

Statistical Significance

Two important inferential tools are *hypothesis testing* and *confidence intervals*.

- In the simple linear model, we might test the **null hypothesis** that H_0: $\beta_1 = 0$
  - A small p-value indicates it would be unlikely to see data leading to an estimate of $\hat{\beta}_1$ if the null hypothesis is true
- Relatedly, a confidence interval can describe values of β_1 that are considered plausible in light of the data we observed
- These tools are only *loosely related* to prediction
  - Variables with coefficients that aren't statistically significant can still improve the model's predictive ability

Determining the Best f?

Whether our goal is prediction or inference will influence how we determine f, specifically whether a **parametric** or **non-parametric** model is used.

- *parametric models* assume a functional form of f. For example, linear regression assumes: $f(\mathbf{X}) = \beta_0 + \beta_1 X_1 + \ldots + \beta_p X_p$
  - After specifying a functional form, we use a procedure to *fit* or *train* the model. For linear regression "training" is estimating β using least squares

Determining the Best f?

Whether our goal is prediction or inference will influence how we determine f, specifically whether a **parametric** or **non-parametric** model is used.

- *parametric models* assume a functional form of f. For example, linear regression assumes: $f(\mathbf{X}) = \beta_0 + \beta_1 X_1 + \ldots + \beta_p X_p$
  - After specifying a functional form, we use a procedure to *fit* or *train* the model. For linear regression "training" is estimating β using least squares
- *non-parametric models* don't make any explicit assumptions about the functional form of f
  - Instead they rely on algorithmic approaches to finding an $\hat{f}$ that is as close to the data points as possible (with "closeness" being constrained by the nature and parameters of the algorithm)

Determining the Best f?

Whether our goal is prediction or inference will influence how we determine f, specifically whether a **parametric** or **non-parametric** model is used.

- *parametric models* assume a functional form of f. For example, linear regression assumes: $f(\mathbf{X}) = \beta_0 + \beta_1 X_1 + \ldots + \beta_p X_p$
  - After specifying a functional form, we use a procedure to *fit* or *train* the model. For linear regression "training" is estimating β using least squares
- *non-parametric models* don't make any explicit assumptions about the functional form of f
  - Instead they rely on algorithmic approaches to finding an $\hat{f}$ that is as close to the data points as possible (with "closeness" being constrained by the nature and parameters of the algorithm)

Parametric models tend to well-suited for inference, while non-parametric models may offer better prediction

Assessing Model Accuracy

Previously we mentioned MSPE as an accuracy measure:

$$\text{MSPE} = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{f}(x_i)\right)^2$$

- It can be shown that MSPE will always *overestimate* the true predictive accuracy of the model
- We are really interested in the expected prediction error $E(\text{PE})$:

$$E(\text{PE}) = \frac{1}{m}\sum_{i=1}^{m}\left(y_i^{\text{new}} - \hat{f}(x_i)\right)^2$$

- For $E(\text{PE})$, we consider the same predictor variables (X's) and systematic component ($\hat{f}$), but new outcomes values that were not used in the fitting of $\hat{f}$

Cross-Validation

Cross-validation is one approach to estimating the expected prediction error (and thereby maximizing predictive ability)

- Cross-validation uses separate subsets of data to fit the model (ie: estimate $\hat{f}$) and evaluate its predictions
- We will focus on **k-fold cross-validation**, which uses the following algorithm:

1. Randomly divide the original data into k equally sized, non-overlapping subsets
2. Fit the candidate model using data from $k-1$ folds, then find the model's prediction error on the k^{th} fold (the "left out" fold)
3. Repeat step two a total of k times until each fold has been left out

Cross-Validation

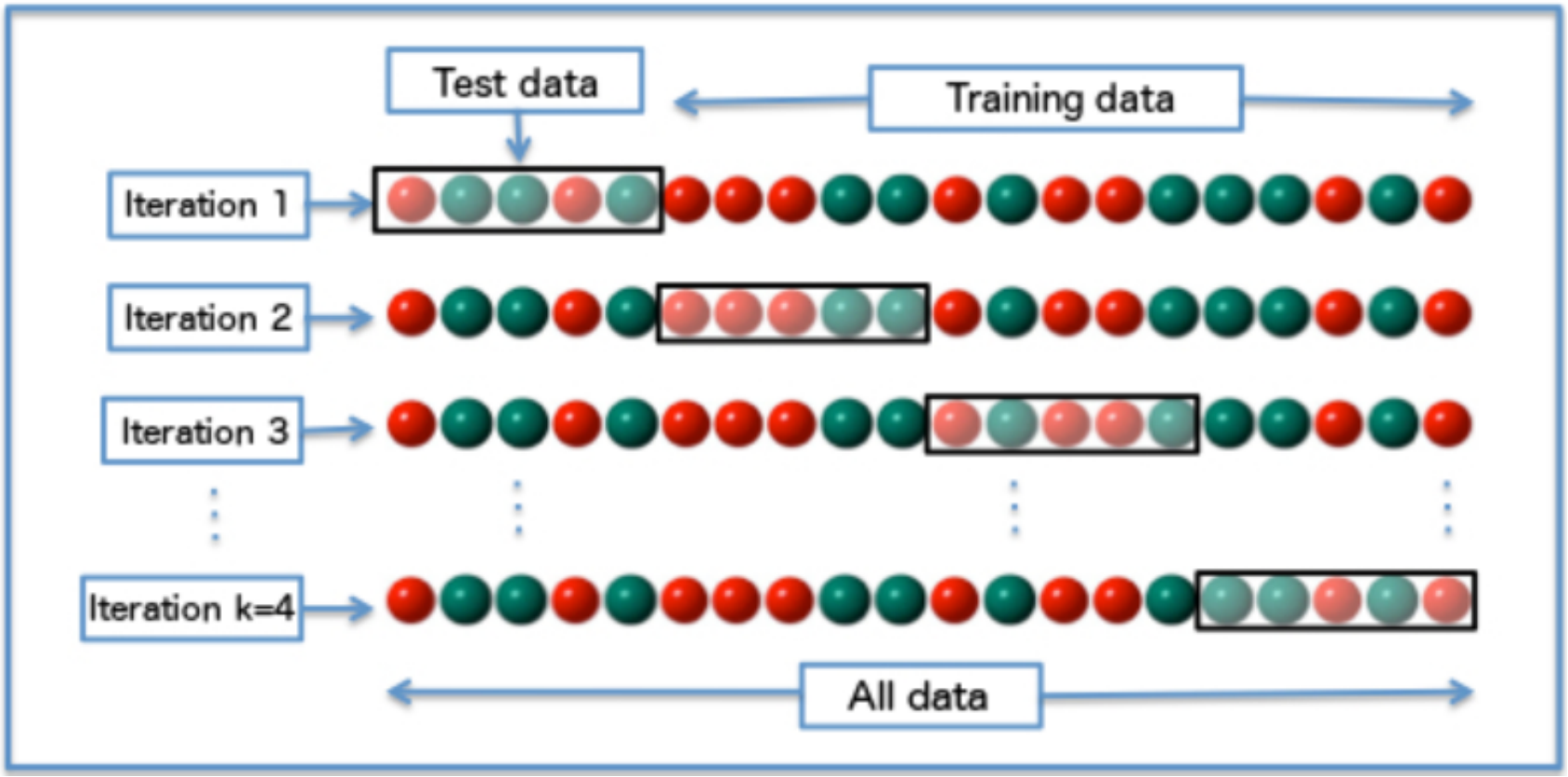


Image from https://en.wikipedia.org/wiki/Cross-validation_(statistics)

Bias vs Variance Tradeoff

- As the complexity of f increases, the model's *bias* decreases
  - $\hat{f}$ can better mimic trends in the data

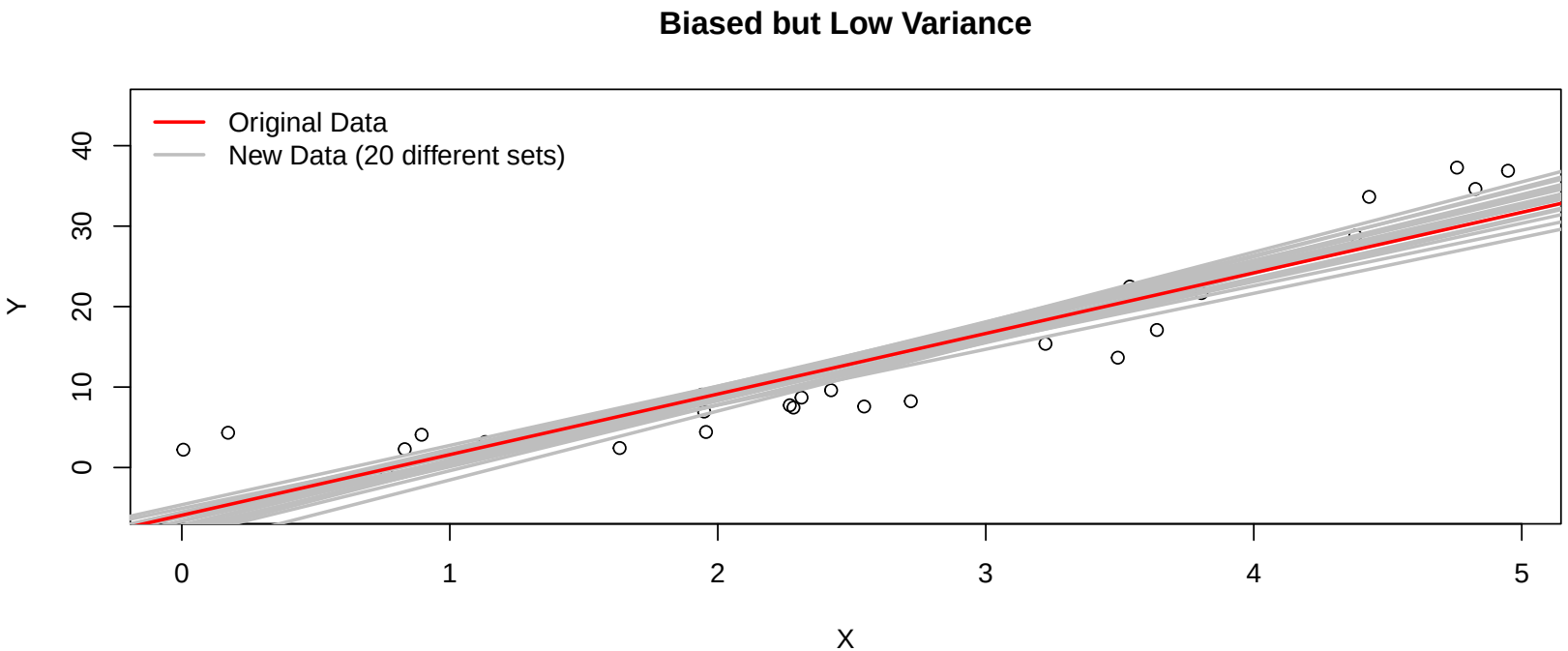
Bias vs Variance Tradeoff

- As the complexity of f increases, the model's *bias* decreases
  - $\hat{f}$ can better mimic trends in the data
- However, as complexity increases, the model's *variance* increases
  - $\hat{f}$ is too specific to the data it was fit on

Bias vs Variance Tradeoff

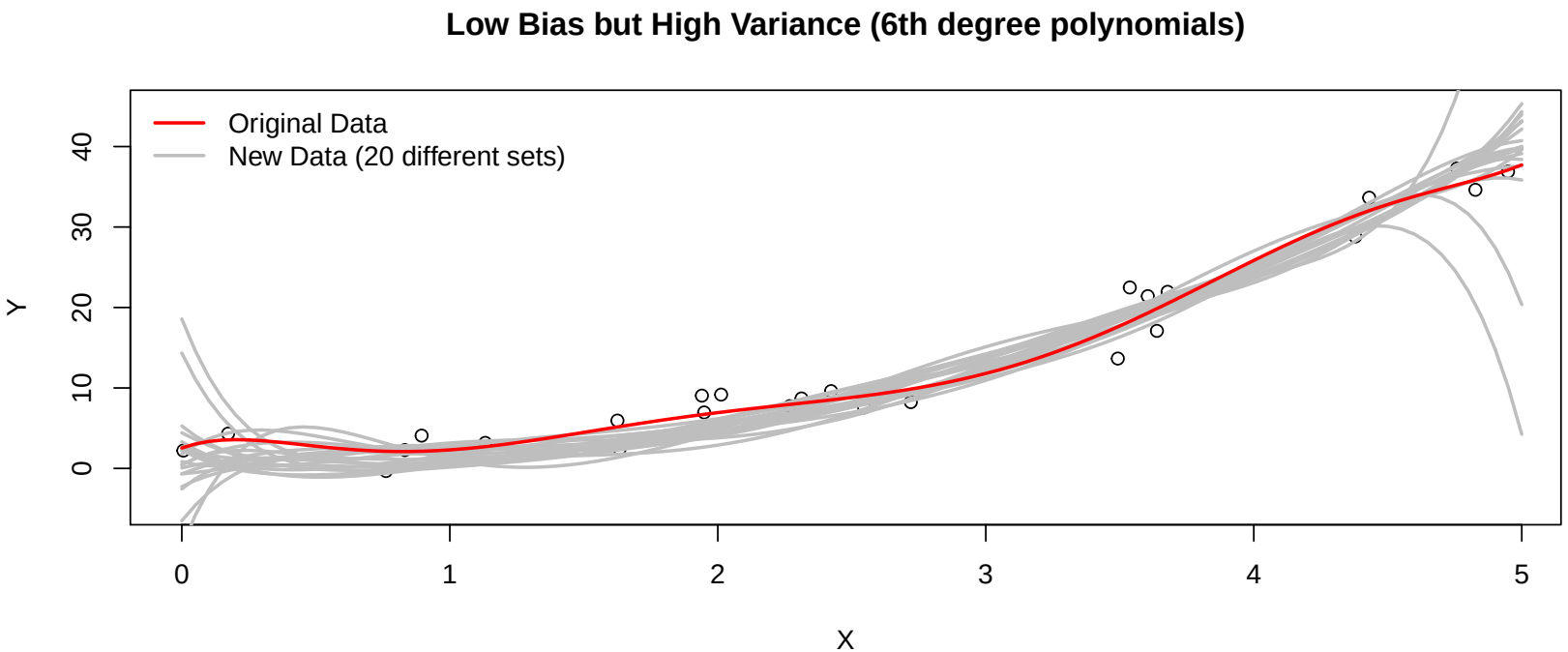
- As the complexity of f increases, the model's *bias* decreases
  - $\hat{f}$ can better mimic trends in the data
- However, as complexity increases, the model's *variance* increases
  - $\hat{f}$ is too specific to the data it was fit on
- Cross-validation is a tool for finding a balance between bias and variance

Bias vs Variance Tradeoff (example)



- Here simple linear regression is biased because it doesn't account for the curvature in the true relationship between X and Y
- However, it is has low variance, fitting it to a different sample doesn't change much

Bias vs Variance Tradeoff (example)



- This model is very capable of capturing the curvature in the true relationship between X and Y
- However, it contains too many parameters, it changes dramatically depending on the specific sample that it is fit to

Bias vs Variance and Cross Validation

Which model do you think has lower cross validation error (CVE), the linear model or the 6th degree polynomial model?

Bias vs Variance and Cross Validation

Which model do you think has lower cross validation error (CVE), the linear model or the 6th degree polynomial model?

- The linear model has CVE of 27, while the 6th degree polynomial model had CVE of 25.4 (you'll see how to apply cross validation yourselves in the next lab)
- Both models have their problems, in this example, bias ended up being more problematic than overfitting
  - It is worth noting that the irreducible error here was 5.81; so even the perfect model can't do better than this benchmark

Classification

- So far, we've focused on scenarios with numeric outcomes, we typically call these *regression problems*
- Situations involving categorical outcomes are called *classification problems*
  - In classification problems, we usually build models that estimate class probabilities, so these situations aren't really that different from regression problems
  - That said, we'll need different ways of assessing model accuracy since $E(\text{PE}) = \frac{1}{m}\sum_{i=1}^{m}\left(y_i^{\text{new}} - \hat{f}(x_i)\right)^2$ is no longer well defined

Misclassification Error

- The simplest approach is classify observations using class probability estimates, then evaluate accuracy using misclassification error:

$$\text{Misclassification Error} = \frac{\text{N Incorrect}}{\text{N Correct} + \text{N Incorrect}}$$

- Misclassifications are summarized via the **confusion matrix**:

		Actual class		
		Cat	**Dog**	**Rabbit**
Predicted class	**Cat**	**5**	2	0
	Dog	3	**3**	2
	Rabbit	0	1	**11**

Image from https://en.wikipedia.org/wiki/Confusion_matrix

Downsides of Misclassification Error

Relying on Misclassification Error (or its reciprocal, *accuracy*) is intuitive and widely used; however, treating each misclassified observation equally can be problematic: 1. $Pr(A) = .51$ vs $Pr(B) = .49$ is no worse than $Pr(A) = .99$ vs $Pr(B) = .01$ 2. A healthy person screening positive for cancer is no worse than a person with cancer screening negative 3. In *imbalanced class* problems, the majority class dominates

Example

Suppose we fit a train classification model to predict fraudulent transactions:

ID	Actual	Predicted Probability (of fraud)	Classification
1	Legit	0.41	Legit
2	Legit	0.10	Legit
3	Fraud	0.67	Fraud
4	Legit	0.59	Fraud
5	Legit	0.11	Legit
...	...	...	...

Further, suppose our data contain 100,000 transactions and only 100 are fraudulent

Binary Outcomes

When the outcome variable is binary, there are following four possibilities for each observation:

	Actually positive (Fraud)	Actually negative (Legit)
Classified positive (Fraud)	True Positive	False Positive
Classified negative (Legit)	False Negative	True Negative

- In some applications, certain misclassifications can be costlier than others
- The fraudulent transaction example, false negatives might be more costly than false positives
- A human can later correct a false positive with relatively low costs, but a false negative will go undetected

Sensitivity/Recall, Specificity, and Precision

In addition to accuracy, we also might consider:

- **Sensitivity**, also known as the *true positive rate* or **Recall**:

$$\text{Sensitivity} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

- **Specificity**, also known as the *true negative rate*:

$$\text{Specificity} = \frac{\text{True Negatives}}{\text{True Negatives} + \text{False Positives}}$$

- **Precision**, also known as *positive predictive value* (PPV):

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

,

Example

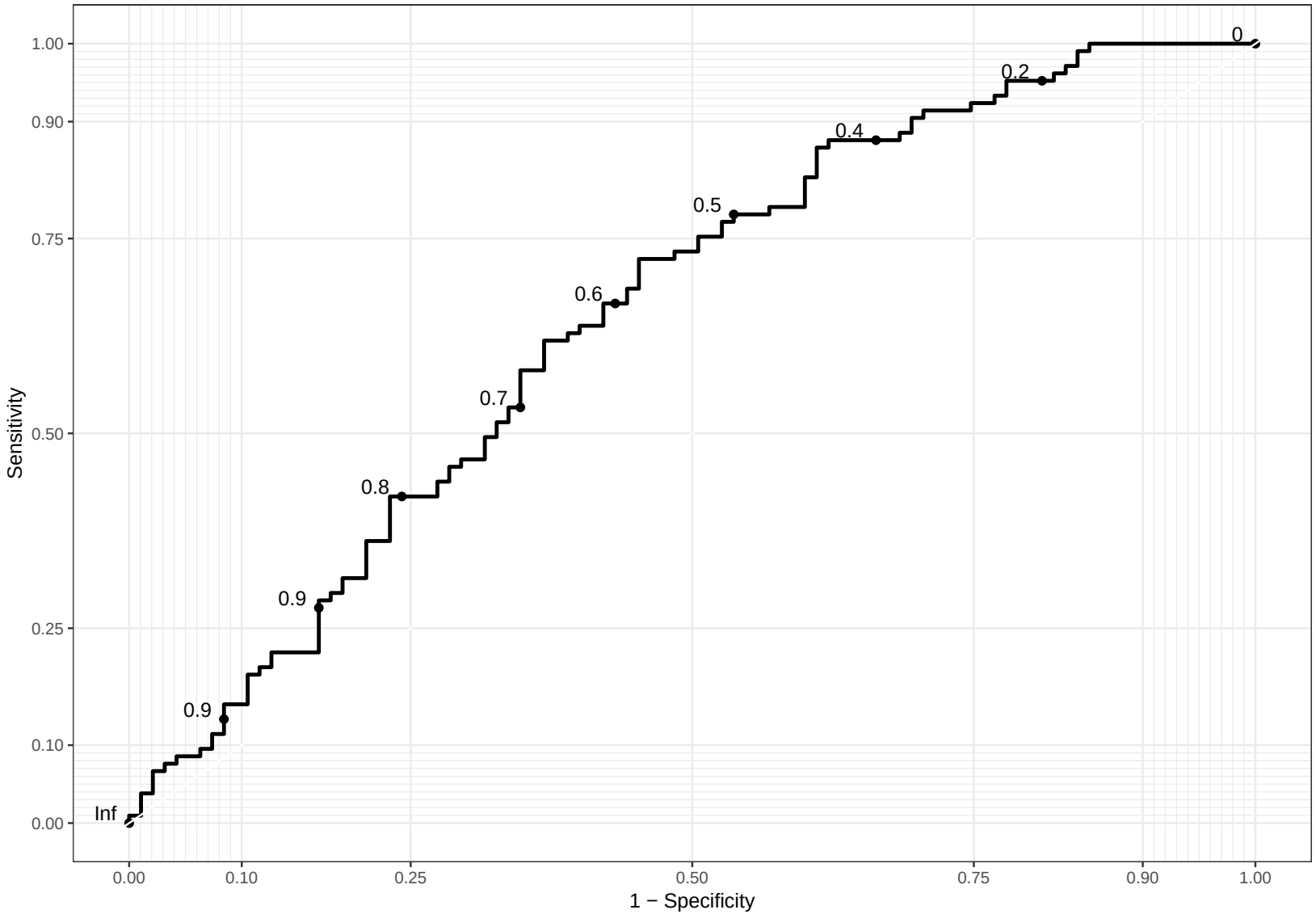
In the fraud detection example:

- Sensitivity/Recall describe the fraction of fraudulent transactions that are classified as fraud
- Specificity describes the fraction of legitimate transactions that are classified as legitimate
- Precision describes the fraction of actual fraudulent transactions among the transactions classified as fraud

ROC Curves

- It is easy to maximize either sensitivity or specificity if the other is ignored
  - Classifying everything as fraud maximizes sensitivity (but specificity is 0)
  - Classifying everything as legit maximizes specificity (but sensitivity is 0)
- This tradeoff is summarized using a Receiver Operating Characteristic Curve (ROC Curve)
  - The area under the ROC Curve (AUC) is used to summarize the curve
  - An AUC of 1 indicates perfect classification
  - An AUC of 0.5 is akin to random guesses

ROC Curves


1.00
0.90
0.75
0.50
0.25
0.10
0.00
Sensitivity
0.00
0.10
0.25
0.50
0.75
0.90
1.00
1 − Specificity
0
0.2
0.4
0.5
0.6
0.7
0.8
0.9
0.9
Inf

Cross Validation and Classification

- Regardless of the metric chosen, cross-validation allows us to estimate the *expected* out-of-sample value
- In practice, all of these different metrics tend to suggest similar models, so they are frequently used in tandem
  - For example, we might consider all models that achieve a certain accuracy standard, then choose the one with the highest sensitivity

Cross Validation and Classification

- Regardless of the metric chosen, cross-validation allows us to estimate the *expected* out-of-sample value
- In practice, all of these different metrics tend to suggest similar models, so they are frequently used in tandem
  - For example, we might consider all models that achieve a certain accuracy standard, then choose the one with the highest sensitivity
- Imbalanced classes might be a concern
  - *Stratified* or *balanced* (synonyms) cross-validation approaches are easy to implement using software
  - Measures such as balanced accuracy have been proposed, using AUC is good idea in these scenarios

Putting it all Together

- **Supervised learning** uses a model to describe an outcome variable Y using a set of features $\mathbf{X}$
  - We might use that model for **prediction** or for **inference**

Putting it all Together

- **Supervised learning** uses a model to describe an outcome variable Y using a set of features $\mathbf{X}$
  - We might use that model for **prediction** or for **inference**
- A critical concern in modeling is balancing the trade-off between **bias** and **variance**
  - We want our model to capture important trends in the data without being overfit

Putting it all Together

- **Supervised learning** uses a model to describe an outcome variable Y using a set of features $\mathbf{X}$
  - We might use that model for **prediction** or for **inference**
- A critical concern in modeling is balancing the trade-off between **bias** and **variance**
  - We want our model to capture important trends in the data without being overfit
  - Cross validation is an excellent tool for evaluating a model's performance in light of the bias-variance trade-off

Putting it all Together