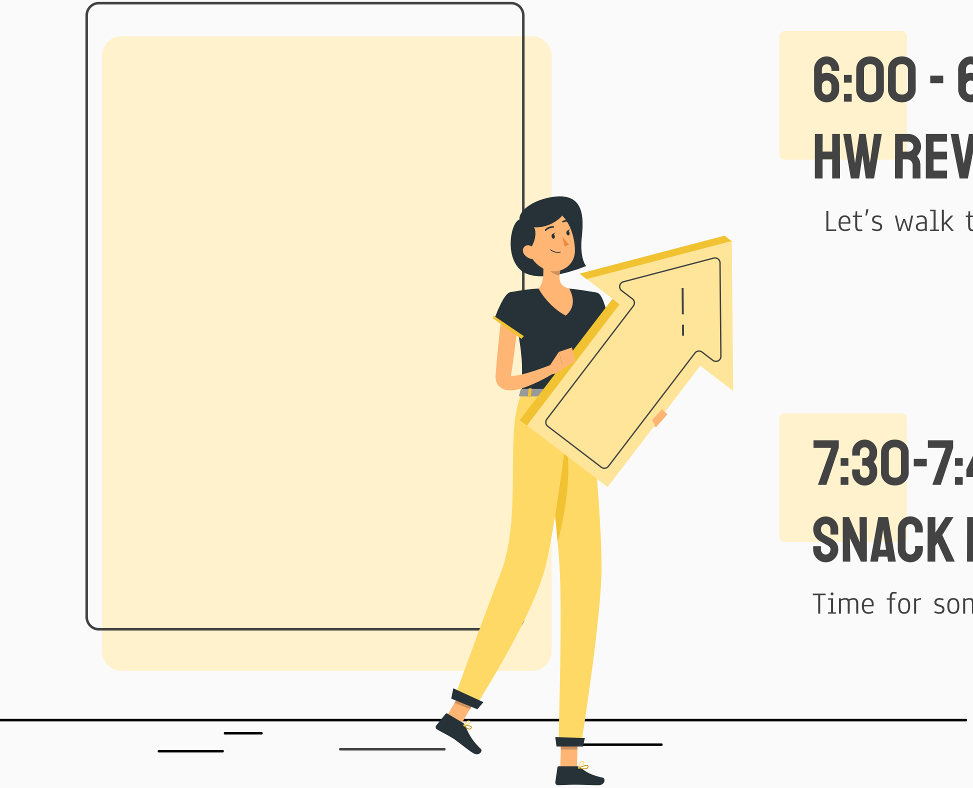
APPLIED DATA SCIENCE II

Week 4: CLASSIFICATION MODELS!

Kyle Scot Shank
WI-22



6:00 - 6:30
HW REVIEW

Let’s walk through it!

6:30-7:30
TOPIC OVERVIEW

Let’s guess if stuff is one thing or another this week with **classification** models!

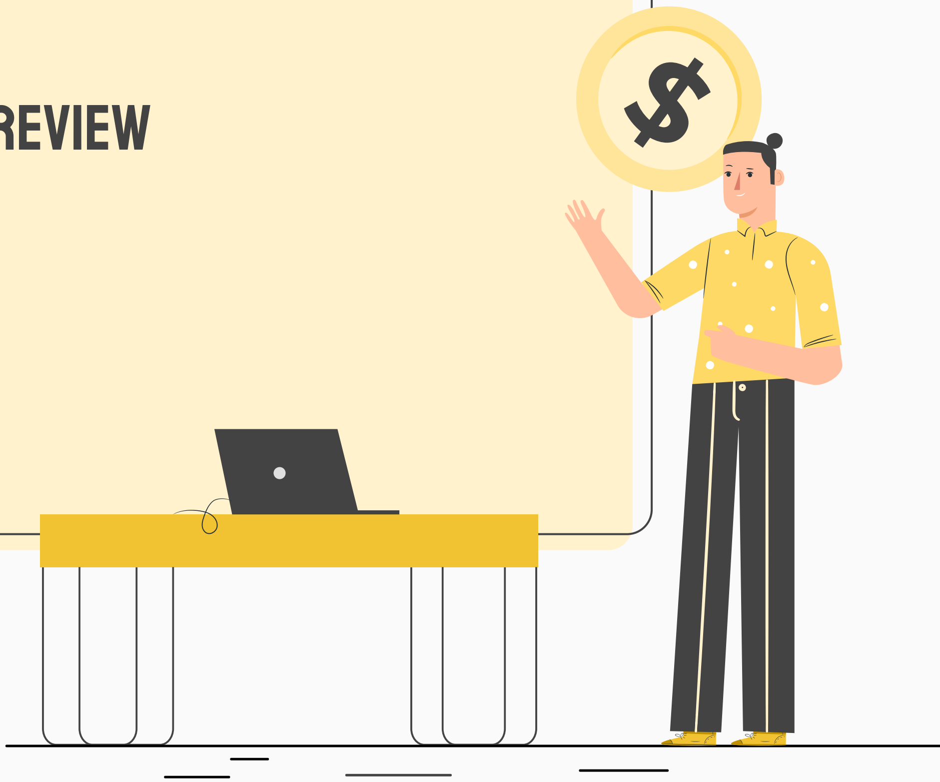
7:30-7:45
SNACK BREAK!

Time for some munchies

7:45 - 9:00
HANDS-ON CODE LAB

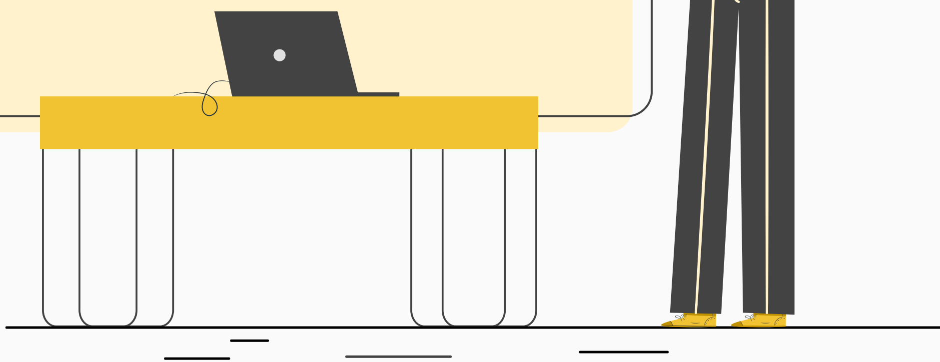
Work through stuff together

HW REVIEW

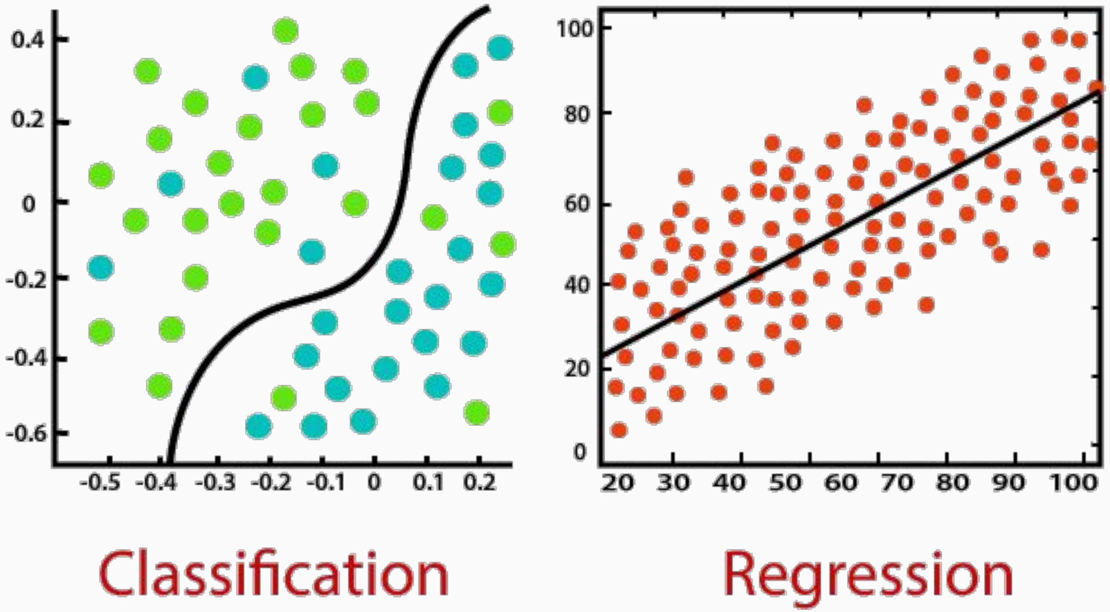


TOPIC OVERVIEW

CLASSIFICATION MODELS!



WHAT ARE CLASSIFICATION MODELS?



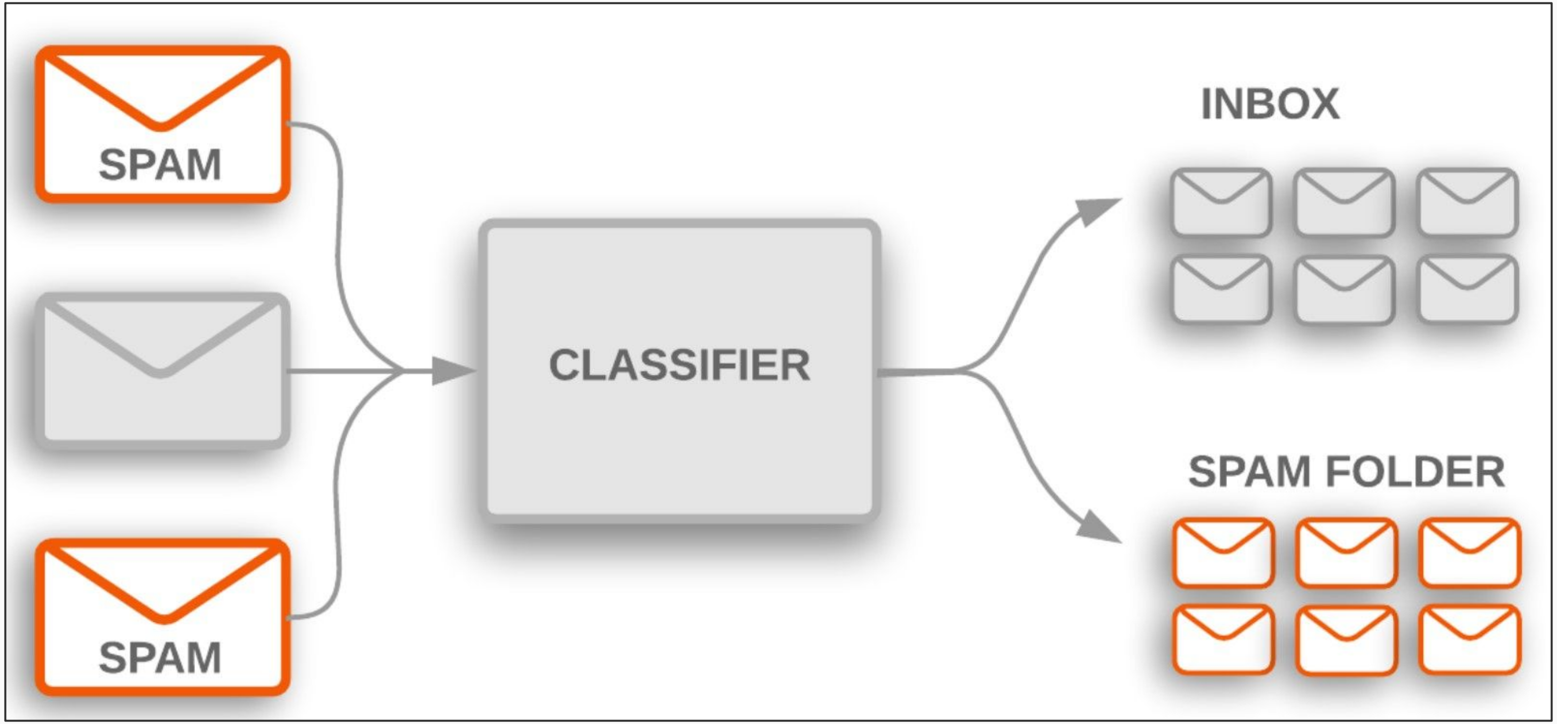
Formal definition:

*Given a feature vector **X** and a qualitative response **Y** taking values in the set **C**, the classification task is to build a function **f(X)** that takes as input the feature vector **X** and predicts its value for **Y** ; i.e. **f(X) ∈ C**.*

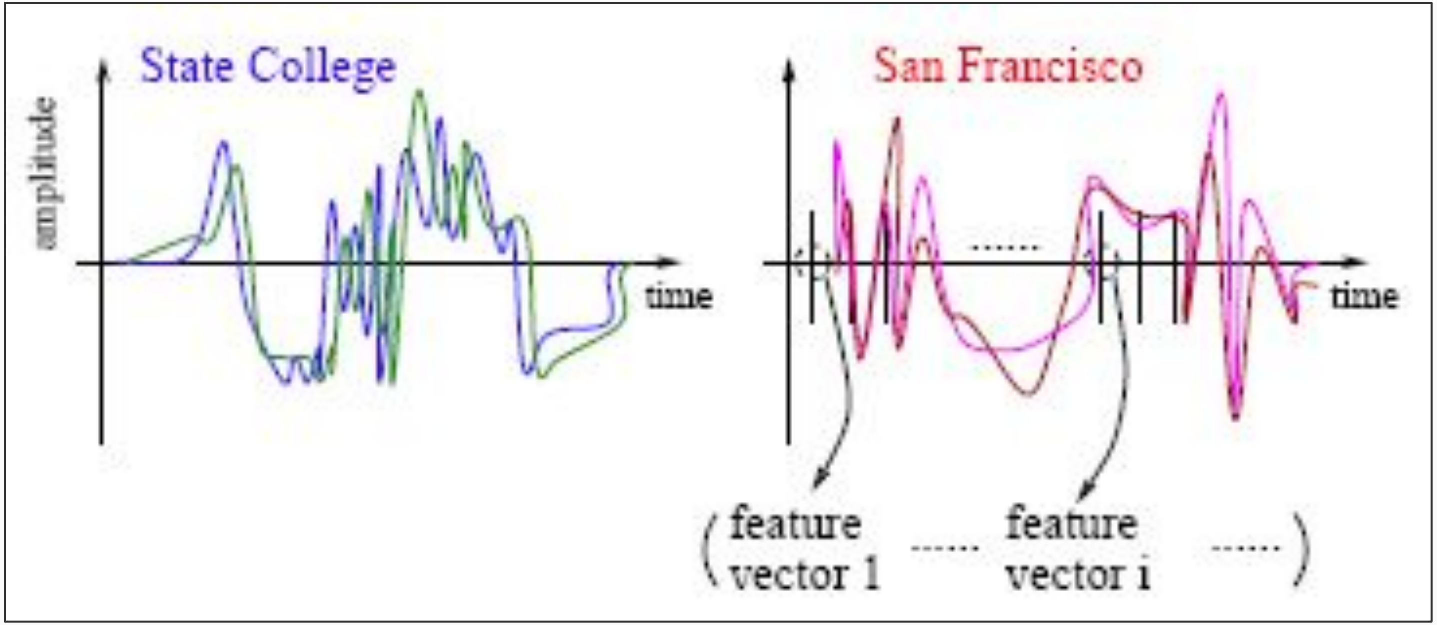
Human language definition:

*Given a set of predictors and a response variable that is qualitative in nature (i.e., some form of a "label") - we build a model that generates **probabilities** of a given row of data belonging to a given label.*

A good example for a binary outcome: spam filters!



A good example for a multinomial outcome: voice recognition!



Can't I just use a linear regression to do this?

Suppose we have a classification task that we think of as this:

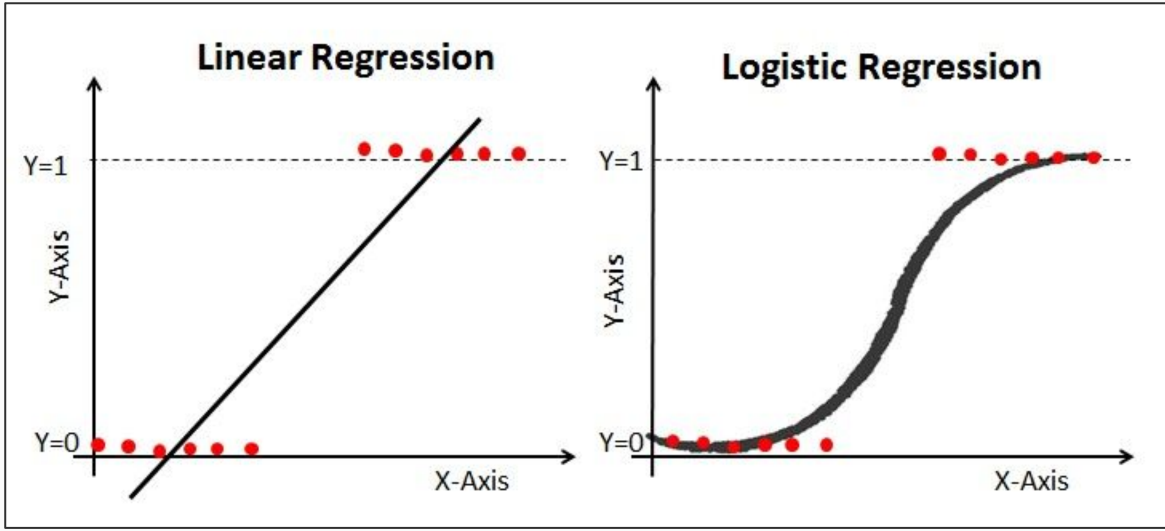
Y = 0 if No
Y = 1 if Yes.

Can we simply perform a linear regression of Y on X and classify as Yes if our predicted value for Y > 0.5?

Sure! Except...linear regression outputs aren't constrained (i.e., your values for your predicted Y might be less than zero or greater than 1) - which means they wont' really make a lot of sense in terms of probabilities. So we have to use other approaches.

THE LOGISTIC REGRESSION

The "workhorse" of old-school classification models is the logistic regression.



THE LOGISTIC REGRESSION

Where p(X) = P(Y = 1 | X)

$$p(X) = \frac{e^{\beta_0+\beta_1 X}}{1 + e^{\beta_0+\beta_1 X}}.$$

This is the only formula today!

...rewritten...

$$\log\left(\frac{p(X)}{1 - p(X)}\right) = \beta_0 + \beta_1 X.$$

This first bit is called the "logit" - or the log odds

THE LOGISTIC REGRESSION

Stuff to remember for a logistic regression model:

- *Make sure you've got enough samples (you generally need more than for a normal linear regression!)*
- *If you're working with multiple classes, make sure you've got good class coverage.*
- *Remember that your coefficient now has a slightly different interpretation (log-odds ratio vs. units!)*

- *Let's do this together!*

Open up R!

FANCIER CLASSIFICATION METHODS

So there's a bunch of other interesting stuff in this chapter that folks tend to only use for very specific tasks.

For example: linear discriminant analysis (LDA) is more efficient in terms of computation than a logistic regression, but that sort of doesn't matter anymore - so folks don't use it.

The only other one we're going to talk through now is KNN, or k-nearest neighbors!

K-NEAREST NEIGHBORS

The best way to describe KNN without math is a simple metaphor:

K-NEAREST NEIGHBORS

How does it work?

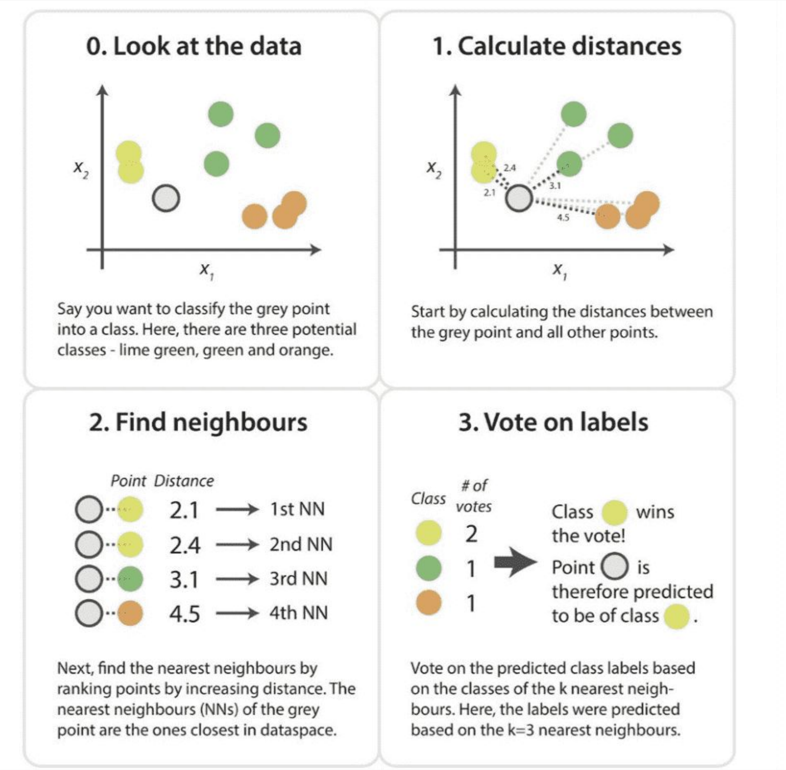
- *In simple words - KNN works by calculating the distance between a given, unknown data point and all the known data points around it, then assigns itself the label most frequently seen.*

- *Unlike the logistic model, the KNN model is truly a "machine learning" approach as it is based on an algorithmic (versus a probabilistic) specification.*

K-NEAREST NEIGHBORS

How does it work (slightly more technical)?

- *Look at all labelled points!*
- *Calculate the distance between them (there are numerous distance metrics, so pick your poison)*
- *Find your nearest neighbors based on the minimized values of your distance metrics*
- *Whichever class/label has the greatest frequency, label yourself that class.*
- *Re-initialize and repeat this entire process again - and keep doing it until you either no longer change classifications for any points and/or hit a stopping rule.*



- ***Let's do this together!***

Open up R!

SNACK BREAK!

COME BACK IN 15!

CODE LAB!

OPEN UP RSTUDIO