Class 10: Classification 2

MGSC 310

Prof. Jonathan Hersh

Class 10: Announcements

1. Quiz 3 posted tonight, due Thursday @ midnight
2. Problem Set 3 Will post later today, Due Oct 13
    - Late problem sets docked 10% per day unless extenuating circumstances
3. Data Analytics Week Next Week!

Data Analytics Industry Week

Register on Handshake to get access to the following virtual events!

Data Analytics Accelerator Program Info Session
Monday, October 5 | 11 a.m. PST
Interested in pursuing a career in the growing field of data analytics? The Argyros School of Business is proud to present the new career skills-focused Analytics Accelerator Program. Learn more about what hard skills are needed to land a successful career in data analytics. Hear from Professor Toplansky and Dr. Hersh about how you can propel your success and prepare for 21st Century jobs that pay a premium.

Careers in Data Analytics
Tuesday, October 6 | 12 p.m. PST
Hear from the renowned authors of Build a Career in Data Science, Jacqueline Nolis and Emily Robinson about careers in data analytics.

Data Analytics Industry Panel
Thursday, October 8 | 4:30 p.m. PST
This data analytics panel will feature industry experts in analytics from entertainment, healthcare, technology, and more.

Entertainment Analytics: Turning Data Into Insights
Friday, October 9| 12 p.m. PST
Come see a live demo and learn about turning data into actionable insights in Entertainment Analytics with Andre Vargas Head of the data department at leading entertainment and sports agency, Creative Artists Agency (CAA).

CHAPMAN UNIVERSITY | Argyros School of Business and Economics

Class 10: Outline

1. Logistic Function
2. Log Odds Ratio
3. Estimating Logistic Regressions
4. Classification Lab 1
5. False/True Positives False/True Negatives
6. Confusion Matrices
7. ROC Curves and AUC
8. Classification Lab 2

Using the Logistic/Sigmoid Function to Generate Probabilities

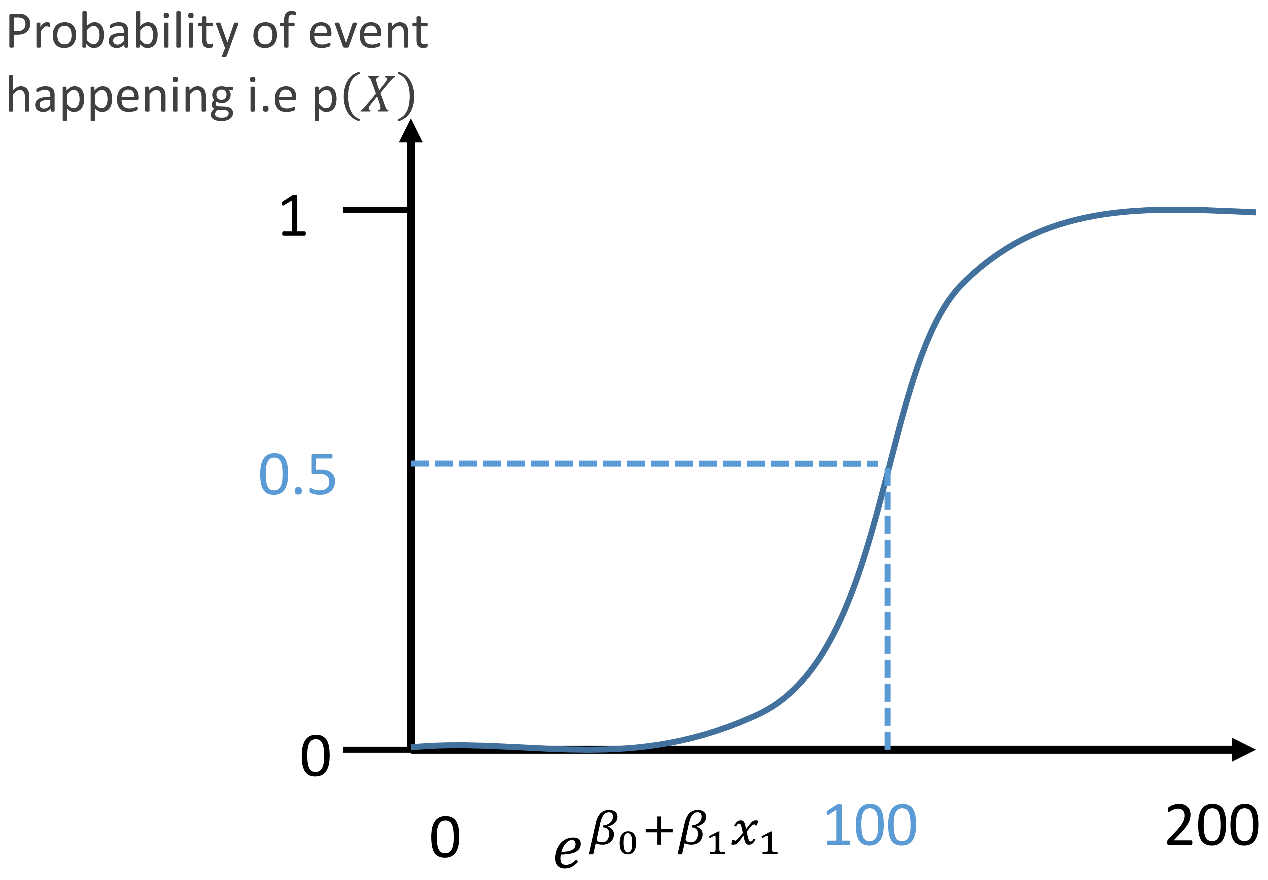
- How do we generate probabilities from this function?

- We let X = $\beta_0 + \beta_1 \cdot x_1 + \cdots + \beta_k \cdot x_k$ and plug this into the logistic function

$$\sigma(X) = \frac{1}{1 + e^{-X}} = \frac{e^X}{e^X + 1}$$

$$Pr(Y = 1|X) = \frac{e^{\beta_0 + \beta_1 \cdot x_1}}{e^{\beta_0 + \beta_1 \cdot X} + 1}$$

$$Pr(Y = 1|X) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1)}}$$



This is equivalent mathematically! I promise. Work it out on pen and paper if you don't believe me

Probability Note on The Complement

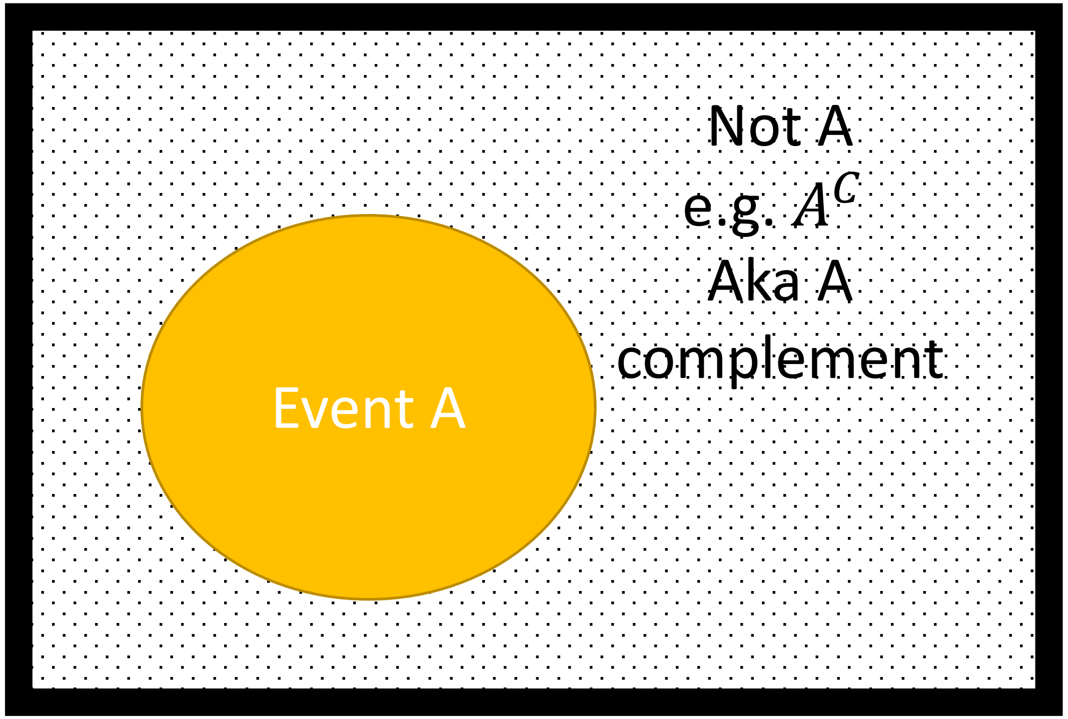
- Q: if $\Pr(A) = 30\%$
- What is the probability of A not happening (the complement) or $\Pr(A^C)$?
- Because events A and not A fully partition the sample space

$$\Pr(A^C) = 1 - P(A)$$

- Fully partition the sample space (i.e. two events are all that can happen):

$$A \cup A^C = \Omega = 1$$

Sample Space (All possible outcomes)



One Weird Trick to Find P(Y=0)

$$Pr(Y = 1|X) = \frac{e^{\beta_0+\beta_1 X}}{1 + e^{\beta_0+\beta_1 X}}$$

$$Pr(Y = 0|X) = 1 - \Pr(Y = 1|X)$$

$$Pr(Y = 0|X) = 1 - \frac{e^{\beta_0+\beta_1 X}}{1 + e^{\beta_0+\beta_1 X}}$$

$$Pr(Y = 0|X) = \frac{1 + e^{\beta_0+\beta_1 X}}{1 + e^{\beta_0+\beta_1 X}} - \frac{e^{\beta_0+\beta_1 X}}{1 + e^{\beta_0+\beta_1 X}}$$

$$Pr(Y = 0|X) = \frac{1 + e^{\beta_0+\beta_1 X} - e^{\beta_0+\beta_1 X}}{1 + e^{\beta_0+\beta_1 X}} = \frac{1}{1 + e^{\beta_0+\beta_1 X}}$$

Expressing Ratio of Probabilities: Odds Ratio

- The **odds ratio** is the ratio of the probability event occurs P(Y=1) and the prob it does not occur P(Y=0)

$$\frac{prob\ event\ occurs}{prob\ event\ does\ not\ occur} = \frac{p(Y = 1|X)}{p(Y = 0|\ X)} =$$

- Since we know the mathematical expression for P(Y=1) and P(Y=0) using the logistic function, we can calculate the odds ratio

$$= \frac{e^{\beta_0+\beta_1 X}}{1 + e^{\beta_0+\beta_1 X}} / \frac{1}{1 + e^{\beta_0+\beta_1 X}}$$

$$= \frac{e^{\beta_0+\beta_1 X}}{1 + e^{\beta_0+\beta_1 X}} \cdot \frac{1 + e^{\beta_0+\beta_1 X}}{1}$$

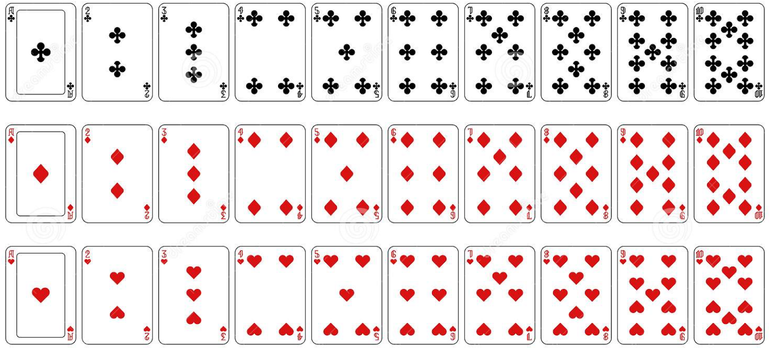
- After some algebra we see the odds ratio is an exponentiated linear model

$$= \frac{prob\ event\ occurs}{prob\ event\ does\ not\ occur} = e^{\beta_0+\beta_1 X}$$

- In fact the log odds ratio is linear!

$$= \ln\left(\frac{prob\ event\ occurs}{prob\ event\ does\ not\ occur}\right) = \ln(e^{\beta_0+\beta_1 X}) = \beta_0 + \beta_1 X$$

Intuition For The Odds Ratio



- The outcome variable in a logit regression is the “odds ratio” (OR)
- In a deck of 52 cards there are 13 spades
- The probability of randomly drawing a spade is 13/52 = 25%
- The probability of not drawing a spade is 39/52 = 75%
- Therefore the ratio of odds of drawing a spade vs not drawing a spade is

$$\frac{ratio\ of\ drawing\ a\ spade}{ratio\ of\ \text{not}\ drawing\ a\ spade} = \frac{{}^{13}/_{52}}{39/52} = \frac{13}{39} = 1:3 = 0.333$$

- Log odds ratio is just log(0.333) = -0.4771…

Logit Models Model the Outcome As a Log Odds Ratio

$$\frac{p(Y=1|X)}{p(Y=0|\ X)} = e^{\beta_0+\beta_1 X}$$

$$log\left(\frac{p(Y=1|X)}{p(Y=0|\ X)}\right) = \log\left(e^{\beta_0+\beta_1 X}\right)$$

$$log\left(\frac{p(Y=1|X)}{p(Y=0|\ X)}\right) = \beta_0 + \beta_1 X$$

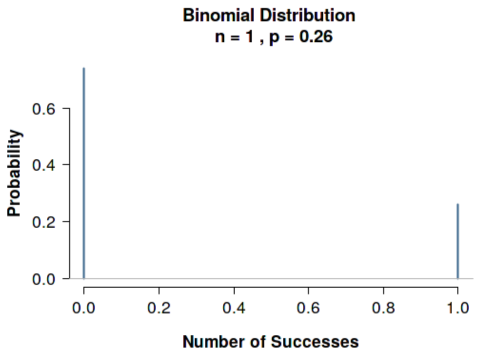
The outcome variable (Y) for a logistic regression is the log odds ratio

Log odds ratio is a linear expression of constants and coefficients of a nonlinear process!

All logistic coefficients can be interpreted as impact on log odds ratio

Estimating Logit Models Using glm()

```r
#------------------------------------------------------------
# Estimating Logistic Regression in R
#------------------------------------------------------------
library('ISLR')
# load data which has credit card default behavior
data(Default)
head(Default)

# make sure to use glm() function!
# set family = binomial to set logistic function
logit_fit1 <- glm(default ~ student,
                  family = binomial,
                  data = Default)
```



- Estimate logistic regression using the function glm() in R
- We still specify a formula in the usual manner
- glm() estimate a variety of "generalized linear models"
- To specific logit we must use the option "family = binomial"
- Binomial is a binary distribution aka the "link" function

If curious see more here: https://shiny.rit.albany.edu/stat/binomial/

Estimating Impact of Being a Student on Default Probability using glm()

$$log\left(\frac{p(Y = default|X)}{p(Y = not\ default|\ X)}\right) = \beta_0 + \beta_1 \cdot student_i + \epsilon_i$$

$$\exp\left(log\left(\frac{p(Y = default|X)}{p(Y = not\ default|\ X)}\right)\right) = \exp(\beta_0 + \beta_1 \cdot student_i + \epsilon_i)$$

$$\frac{p(Y = default|X)}{p(Y = not\ default|\ X)} = \exp(\beta_0 + \beta_1 \cdot student_i + \epsilon_i)$$

```
glm(formula = default ~ student, family = binomial, data = Default)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-0.2970  -0.2970  -0.2434  -0.2434   2.6585

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -3.50413    0.07071  -49.55  < 2e-16 ***
studentYes   0.40489    0.11502    3.52 0.000431 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

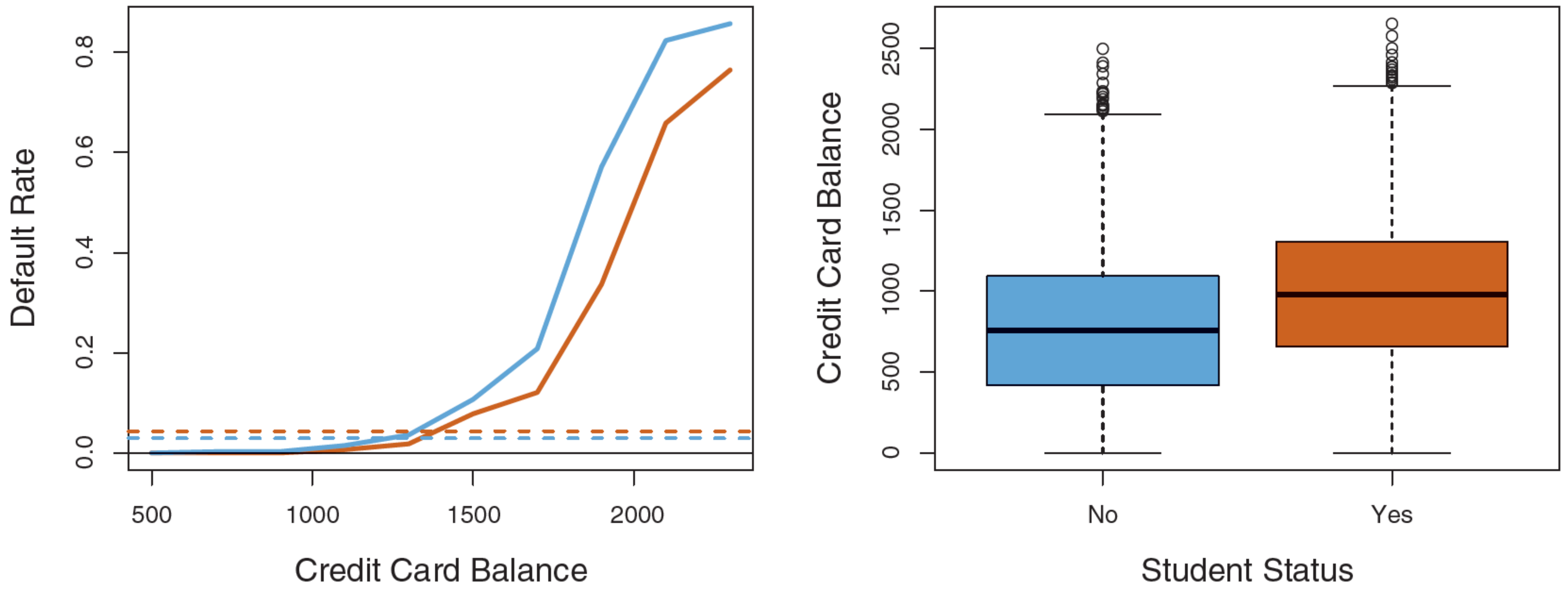
```
> exp(logit_fit1$coefficients)
(Intercept)  studentYes
 0.03007299  1.49913321
```

- Remember the outcome variable in a logistic regression is the log odds ratio
- If we exponentiate the coefficients this tells us the impact of the variable on the unlogged odds ratio
- If we take our estimated logistic model we see $\beta_1 = 0.40489$
- This means students have a 0.40489 higher log odd of defaulting
- Exponentiating the coefficients returns the impact of the X-variable on the odds ratio directly.
- Therefore the ratio of odds of default for student vs non-student is 1.49, or students have a 49% higher probability of default

Lab Time!

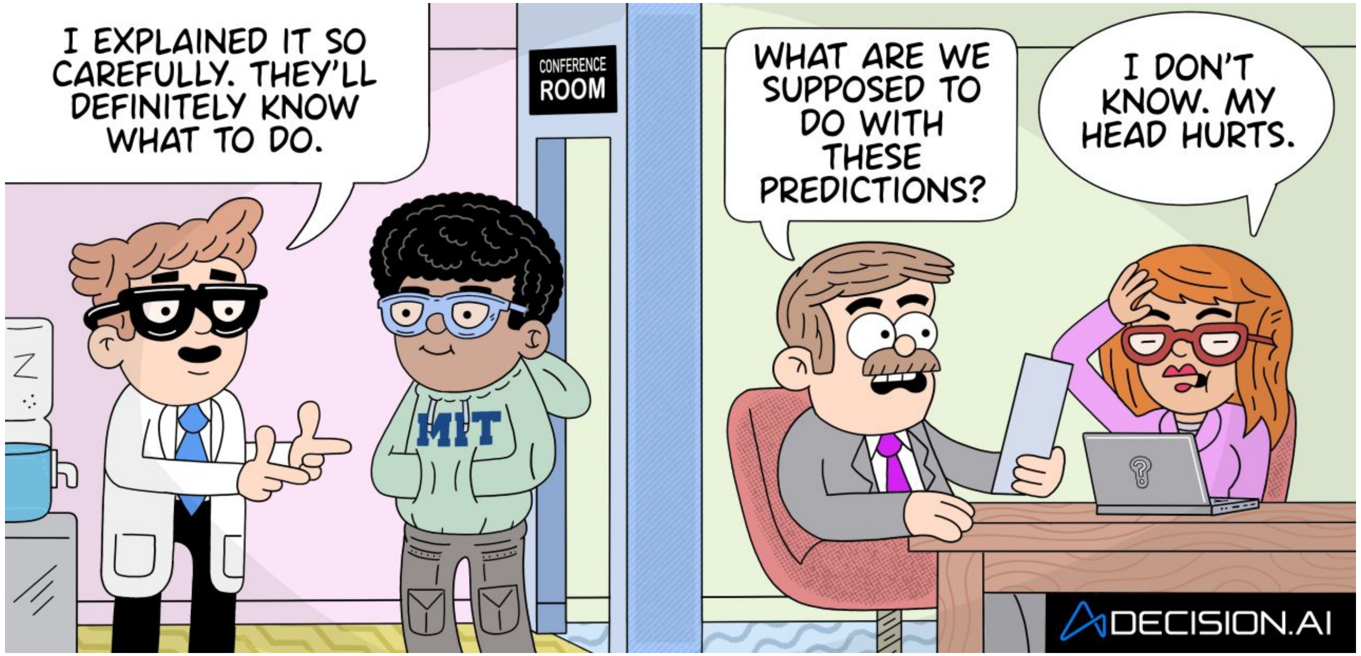
```
#------------------------------------------------------------
# Lab 1
#------------------------------------------------------------
# 1. Estimate a logistic regression model predicting
#    default as a function of student, balance, and income
#    and store this as 'logit_mod2'
# 2. Exponentiate the coefficient vector of logit_mod2
# 3. Interpret the impact of being a student on the probabiliy of default
# 4. Do students face a higher or lower risk of credit card default?
```

Student as "confounder"



Class 10: Outline

1. Logistic Function
2. Log Odds Ratio
3. Estimating Logistic Regressions
4. Classification Lab 1
5. False/True Positives False/True Negatives
6. Confusion Matrices
7. ROC Curves and AUC
8. Classification Lab 2



Generating Predicted Probabilities from a Logit Model

- To generate predictions, we use the estimated coefficients in the logit equation

$$\hat{p}(X = 1000) = \frac{e^{\widehat{\beta}_0 + \widehat{\beta}_1 x_1}}{1 + e^{\widehat{\beta}_0 + \widehat{\beta}_1 x_1}} =$$

- The estimated probability of default with a balance of $1,000 is given by
- The estimated probability of default with a balance of $2,000 is given by
- To generate predicted probability for all observations in a dataset we use the predict function, **but note type = "response"!**

- This is also called "scoring" a dataset

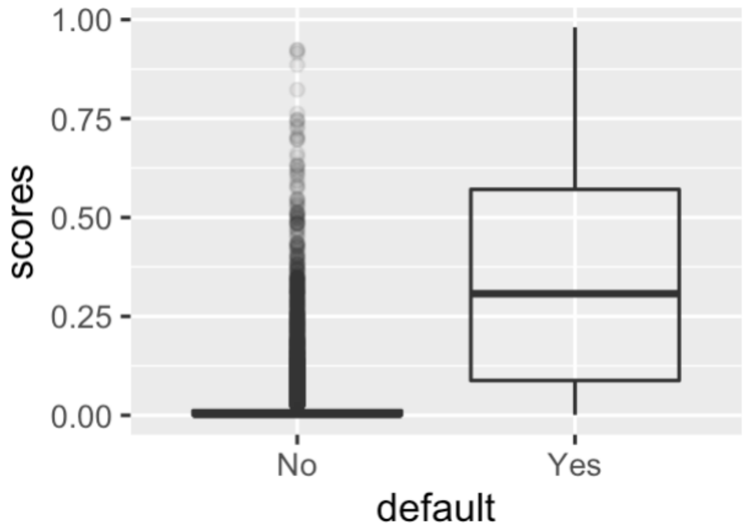
```
glm(formula = default ~ balance, family = binomial, data = Default)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.2697  -0.1465  -0.0589  -0.0221   3.7589

Coefficients:
              Estimate  Std. Error z value Pr(>|z|)
(Intercept) -10.6513306  0.3611574  -29.49   <2e-16 ***
balance       0.0054989  0.0002204   24.95   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

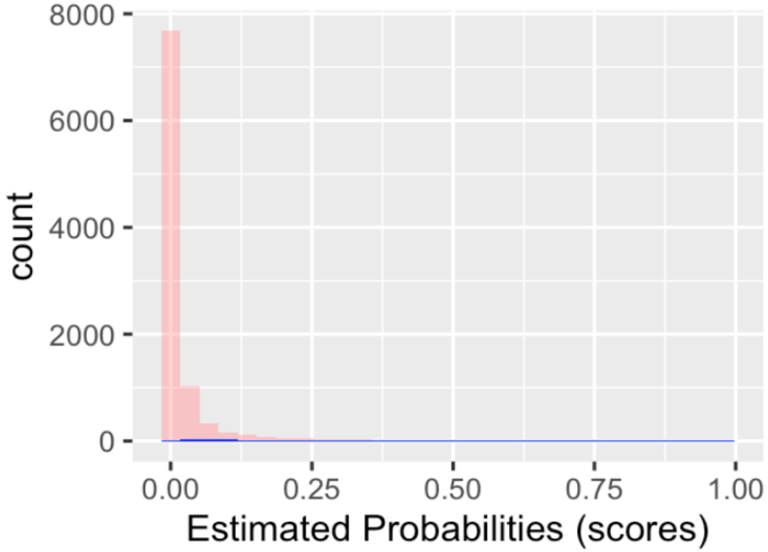
$$\hat{p}(X = 1000) = \frac{e^{-10.6513 + 0.0055 \cdot \mathbf{1000}}}{1 + e^{-10.6513 + 0.0055 \cdot \mathbf{1000}}} = 0.00575$$

$$\hat{p}(X = 2000) = \frac{e^{-10.6513 + 0.0055 \cdot \mathbf{2000}}}{1 + e^{-10.6513 + 0.0055 \cdot \mathbf{2000}}} = 0.55857$$

```
scores <- predict(logit_fit3,
                  type = "response")
```

What Do We Do With Scores or Estimated Probabilities?



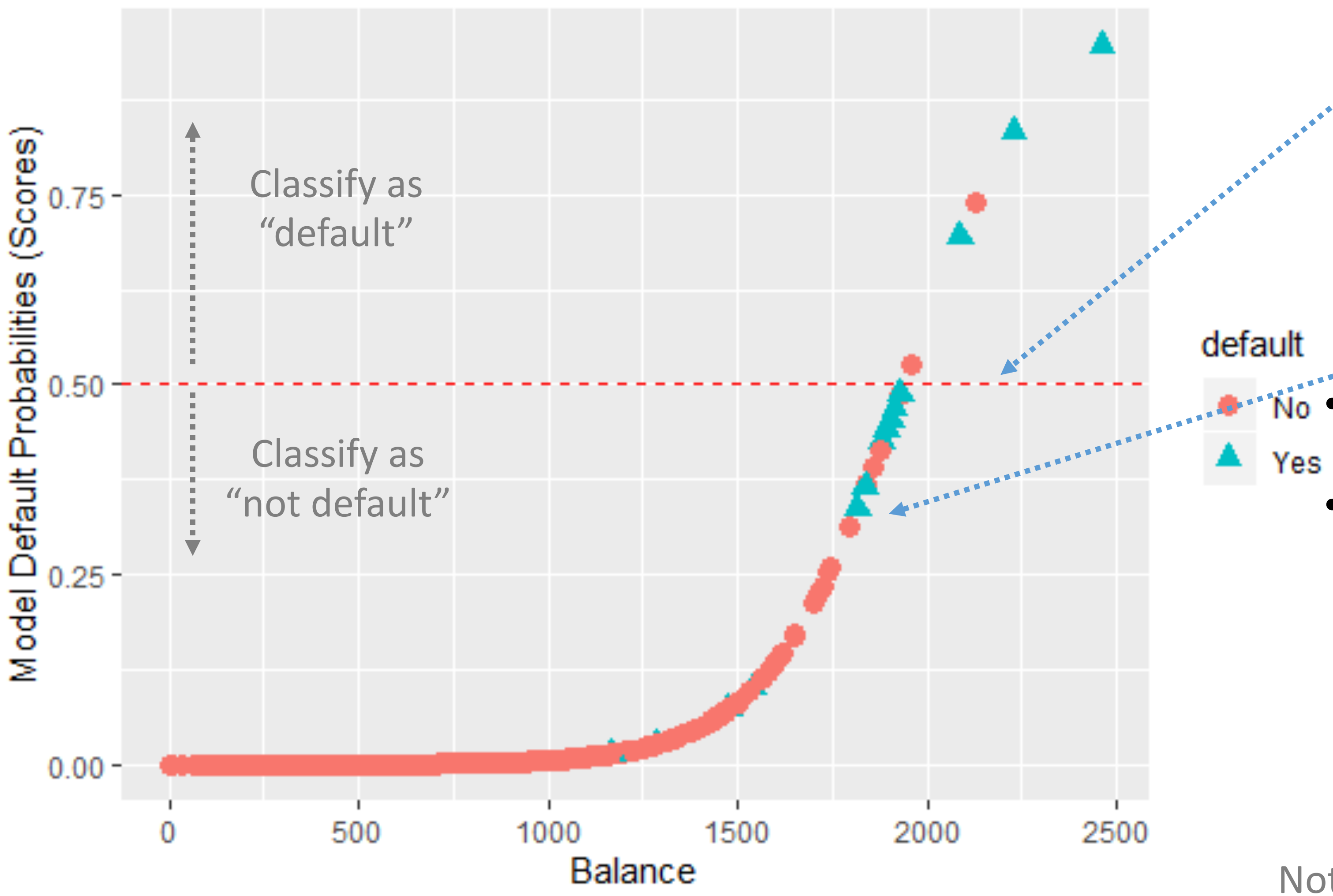


- Okay, so we have probabilities, what then?
- Note there is almost always some overlap between the probabilities of the classes
- We can't choose a probability such that above this all actual defaulters are correctly identified, and below this all non-defaulter are identified
- So we will always have some **false positives** and **false negatives**

Confusion Matrix: Table of False/True Positives and False/True Negatives

		True default status	
		No	**Yes**
Predicted default status	**No**	True negative (TN)	False Negative (FN)
	Yes	False Positive (FP)	True Positive (TP)

Assigning Class=Default to $\hat{p} > 0.5$

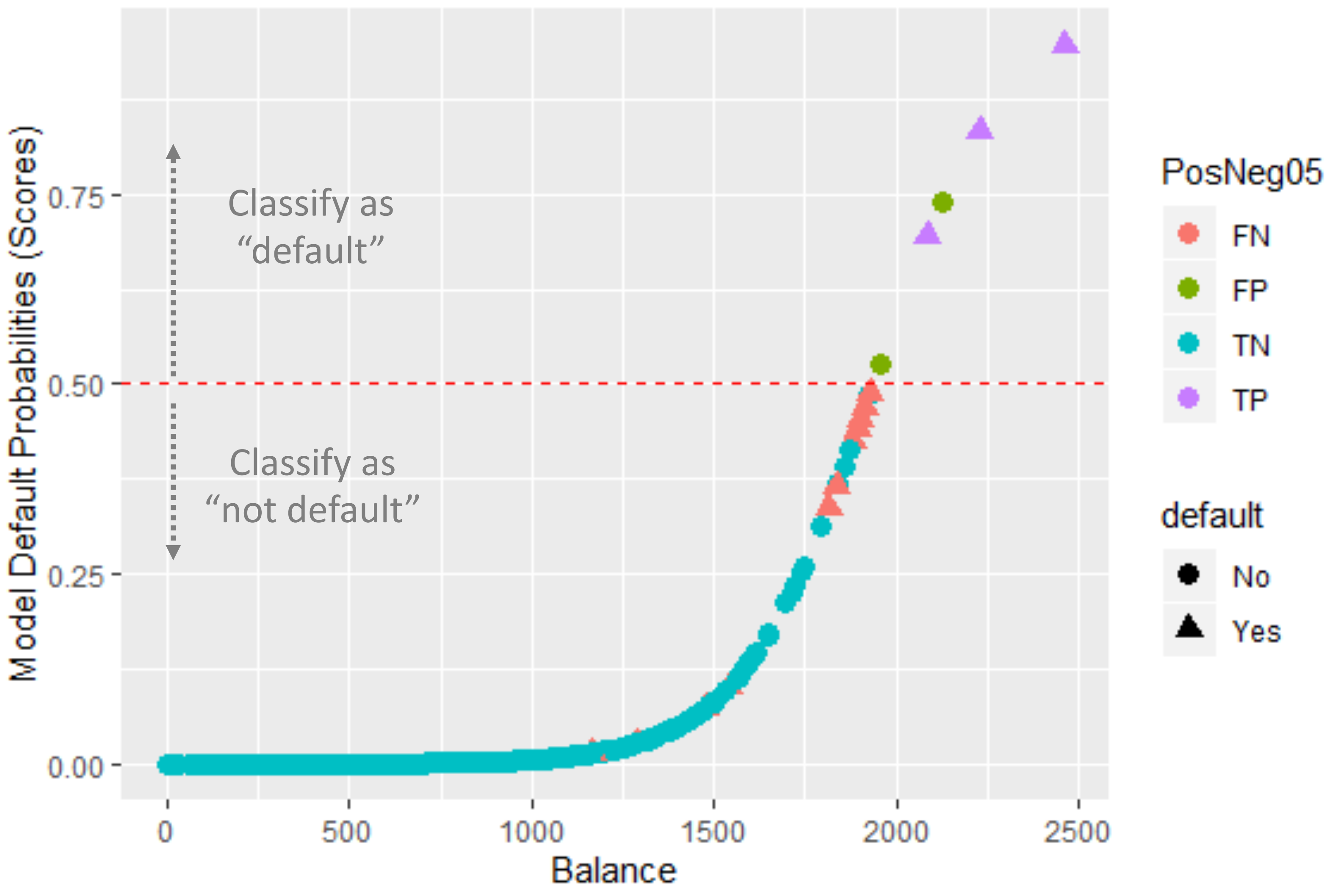


- Above this line, observations are classified as defaulting
- Below this line, observations classified as **not** defaulting
- Actual default = teal triangles
- Actual not default = circle
- If we choose a probability cutoff of 0.5, then we see we have 2 false positives and 11 FN

```
> table(preds_sample$PosNeg05)

FN  FP  TN  TP
11   2 484   3
```

Note I'm working with a 5% sample of the dataset to make the numbers easier

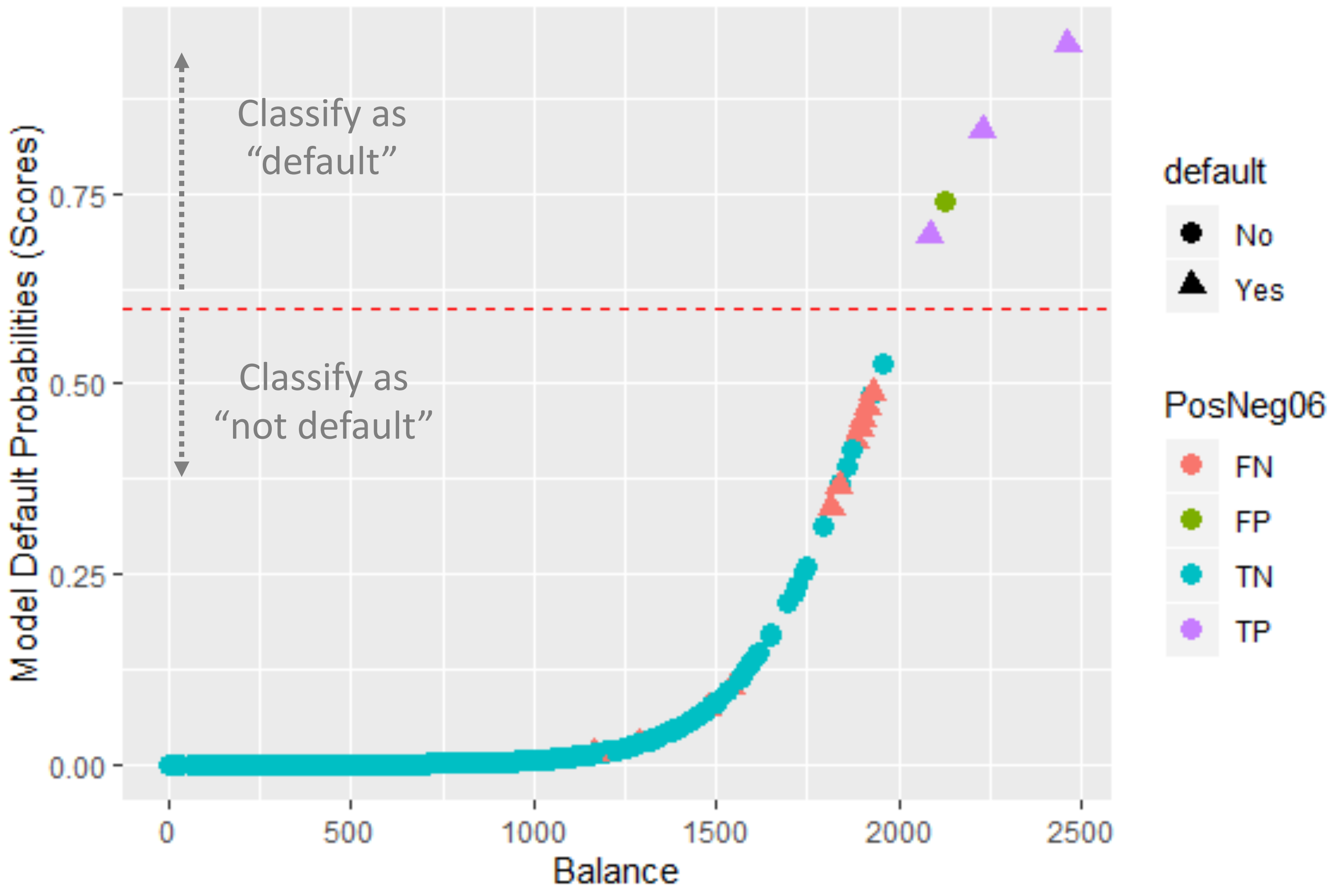
Assigning Class=Default to $\hat{p} > 0.5$



- If we choose a probability cutoff of 0.5, then we see we have 2 false positives and 11 FN

```
> table(preds_sample$PosNeg05)

FN  FP  TN  TP
11   2 484   3
```

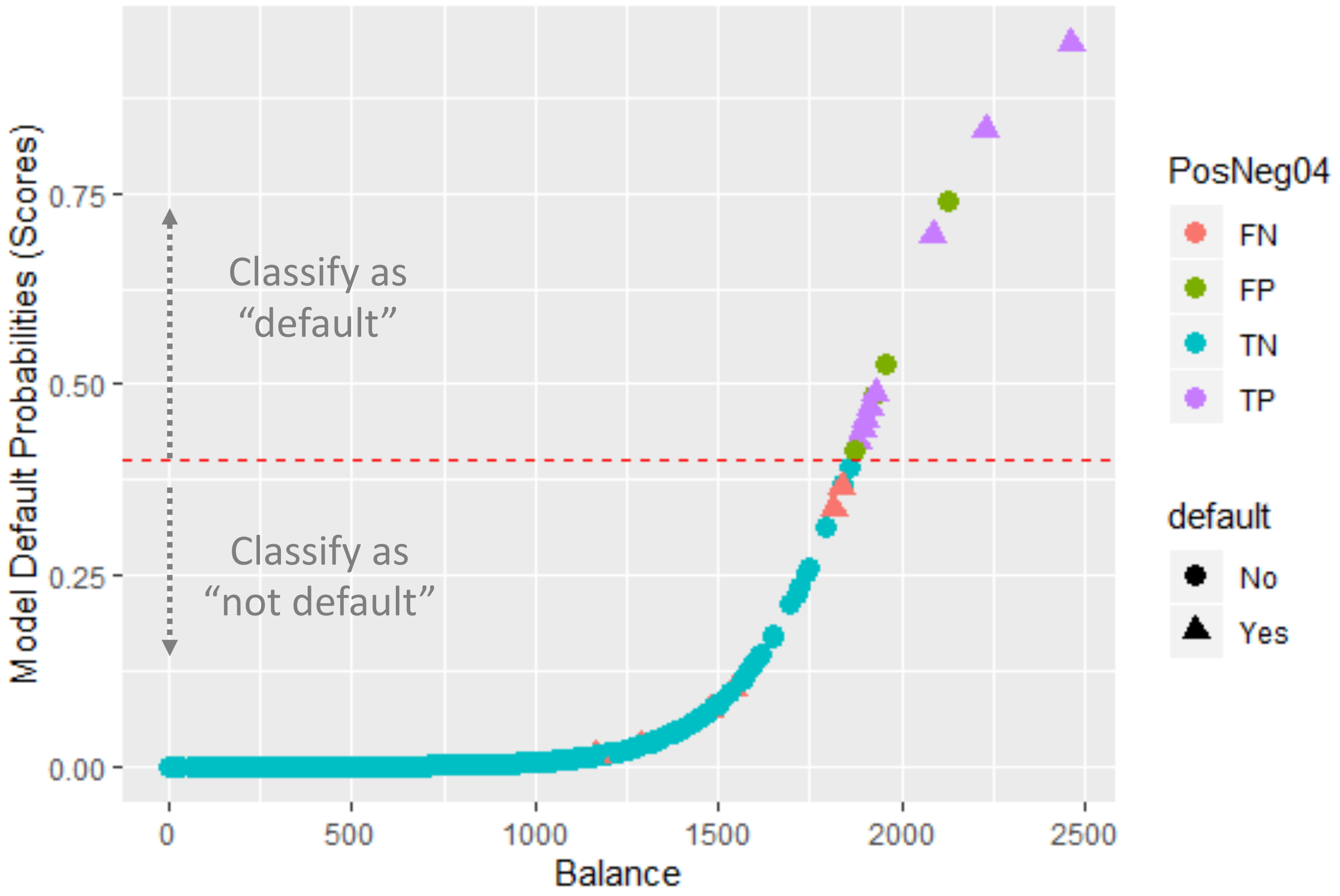
Assigning Class=Default to $\hat{p} > 0.6$



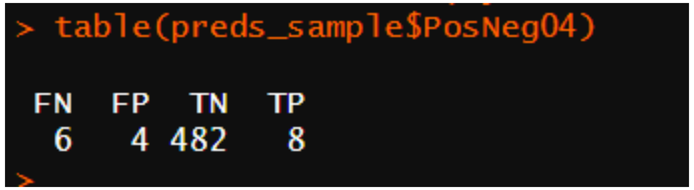
- Raising the cutoff to 0.6, then we see we have 1 false positives and 11 FN

```
> table(preds_sample$PosNeg06)

FN  FP  TN  TP
11   1 485   3
```

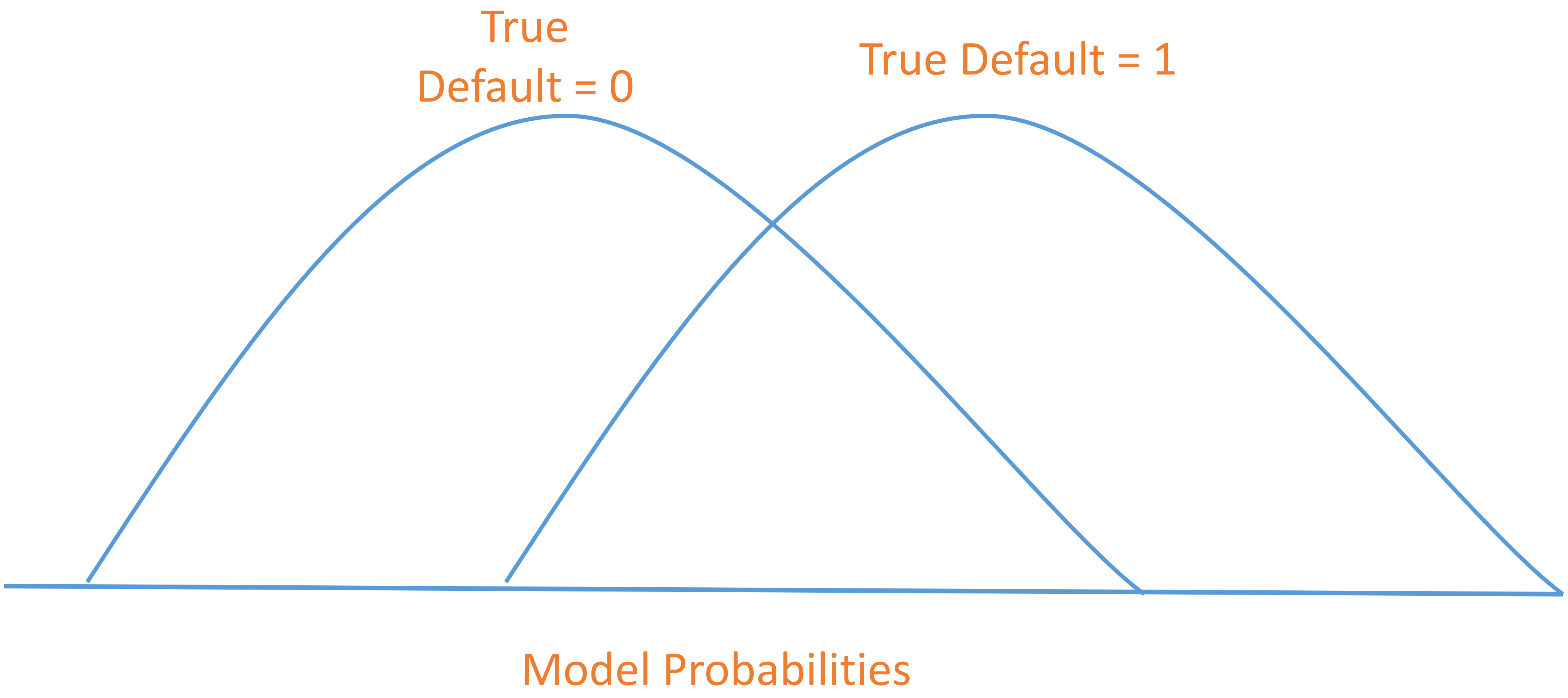
Assigning Class=Default to $\hat{p} > 0.4$



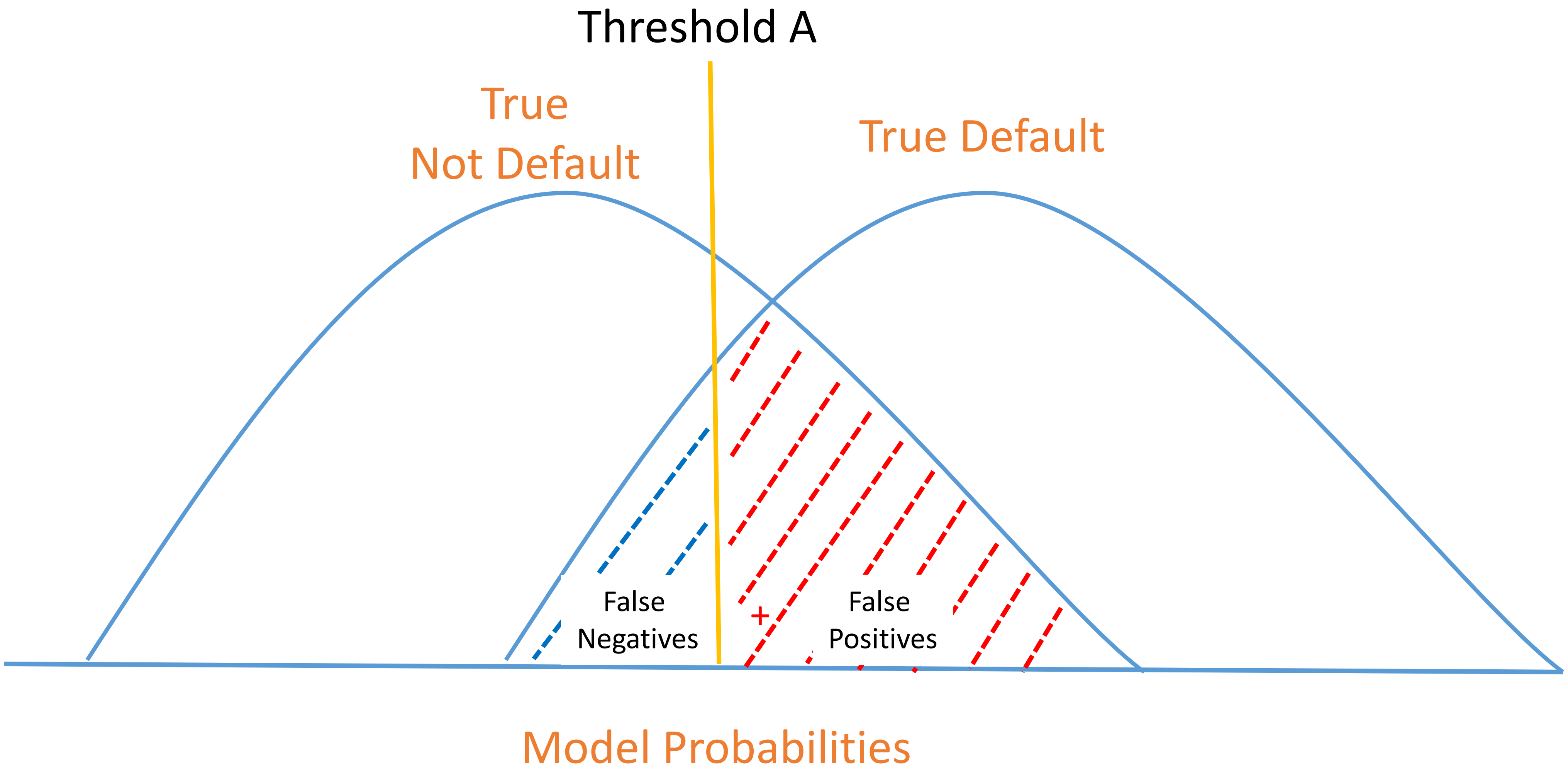
- Lowering the cutoff to 0.4 results in more FPs (4) but fewer FNs (6)



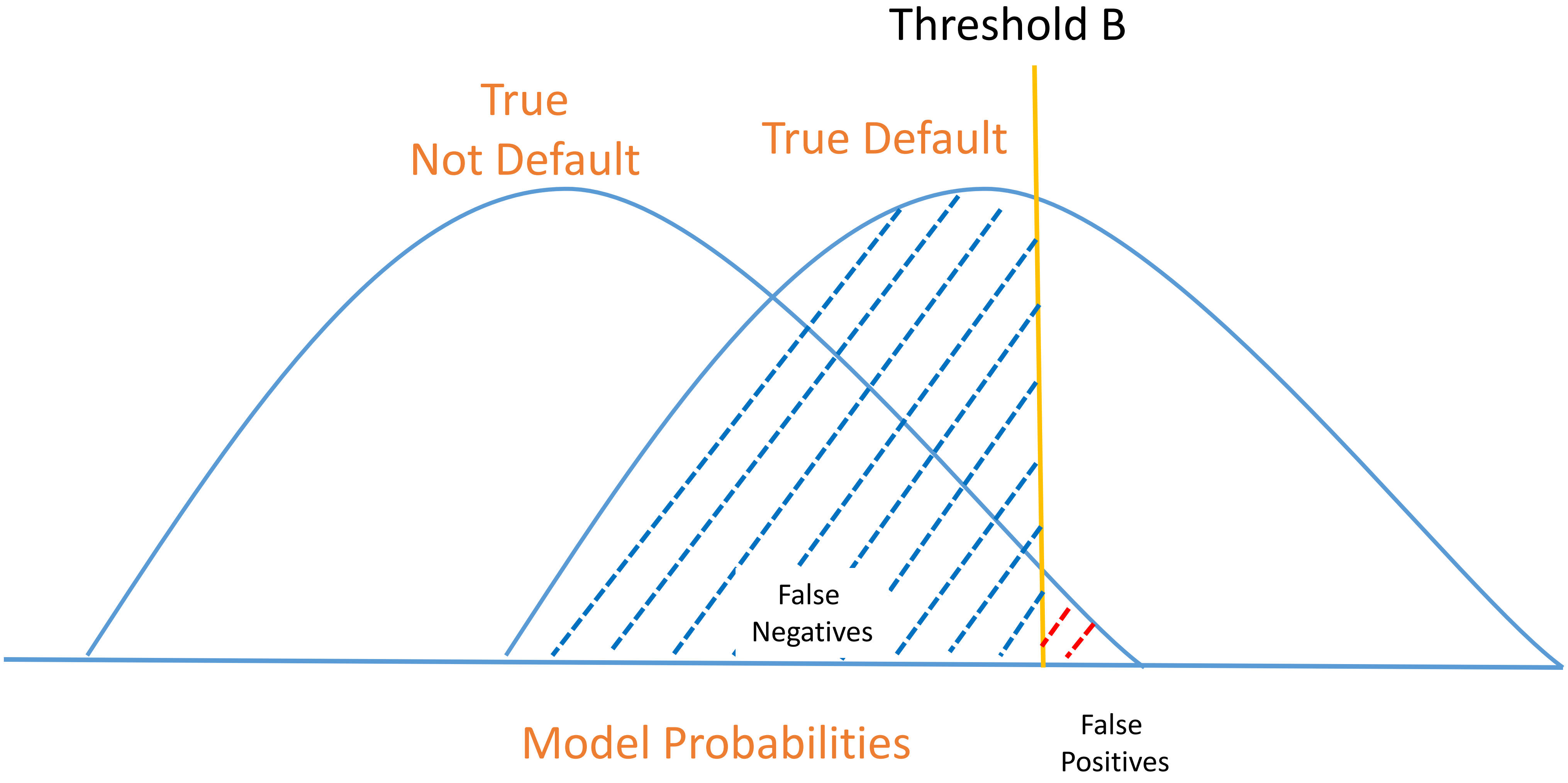
Choosing Probability Cutoff to Assign Class



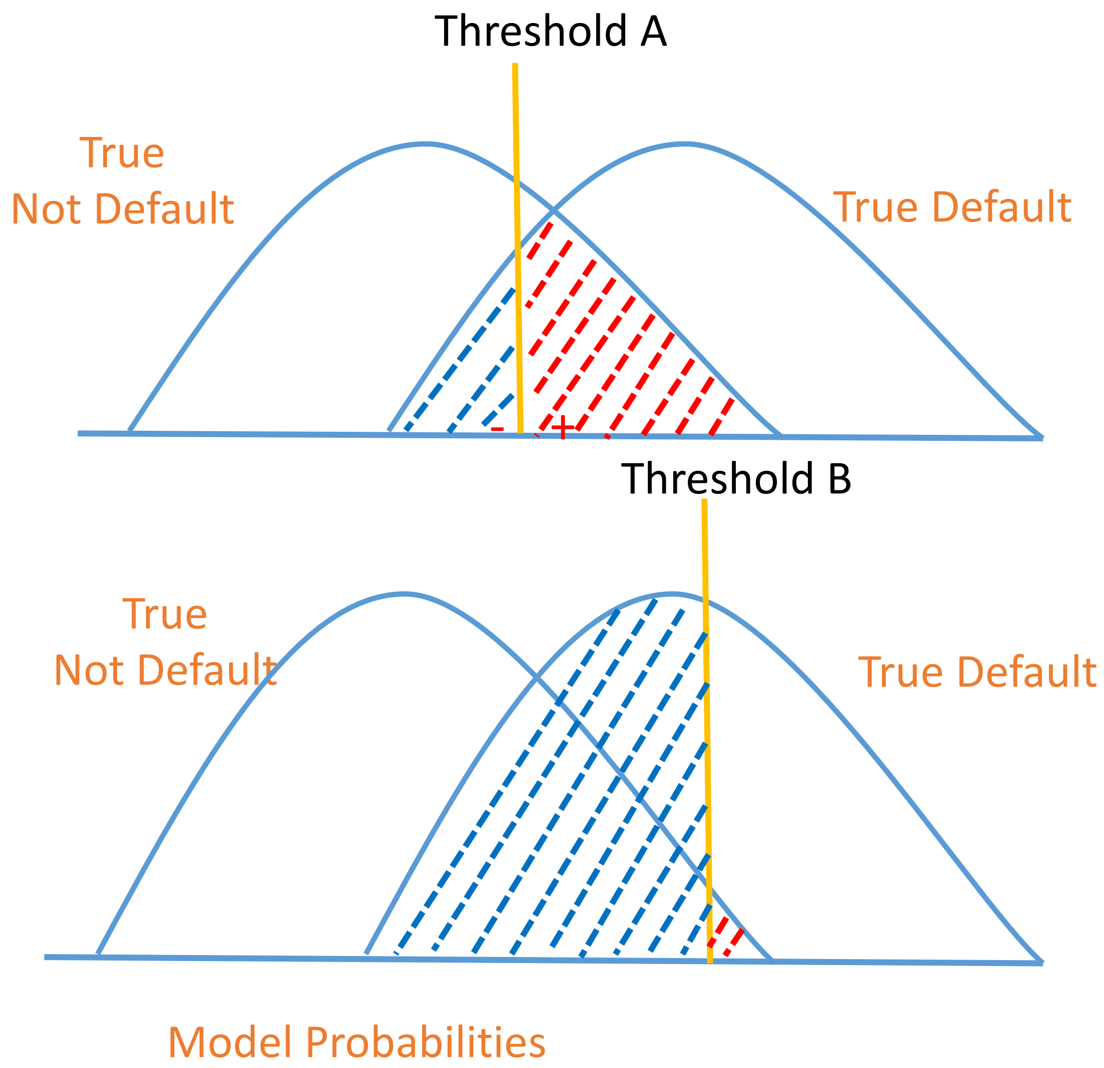
Threshold A: Moderate Threshold



Threshold B: Higher Threshold



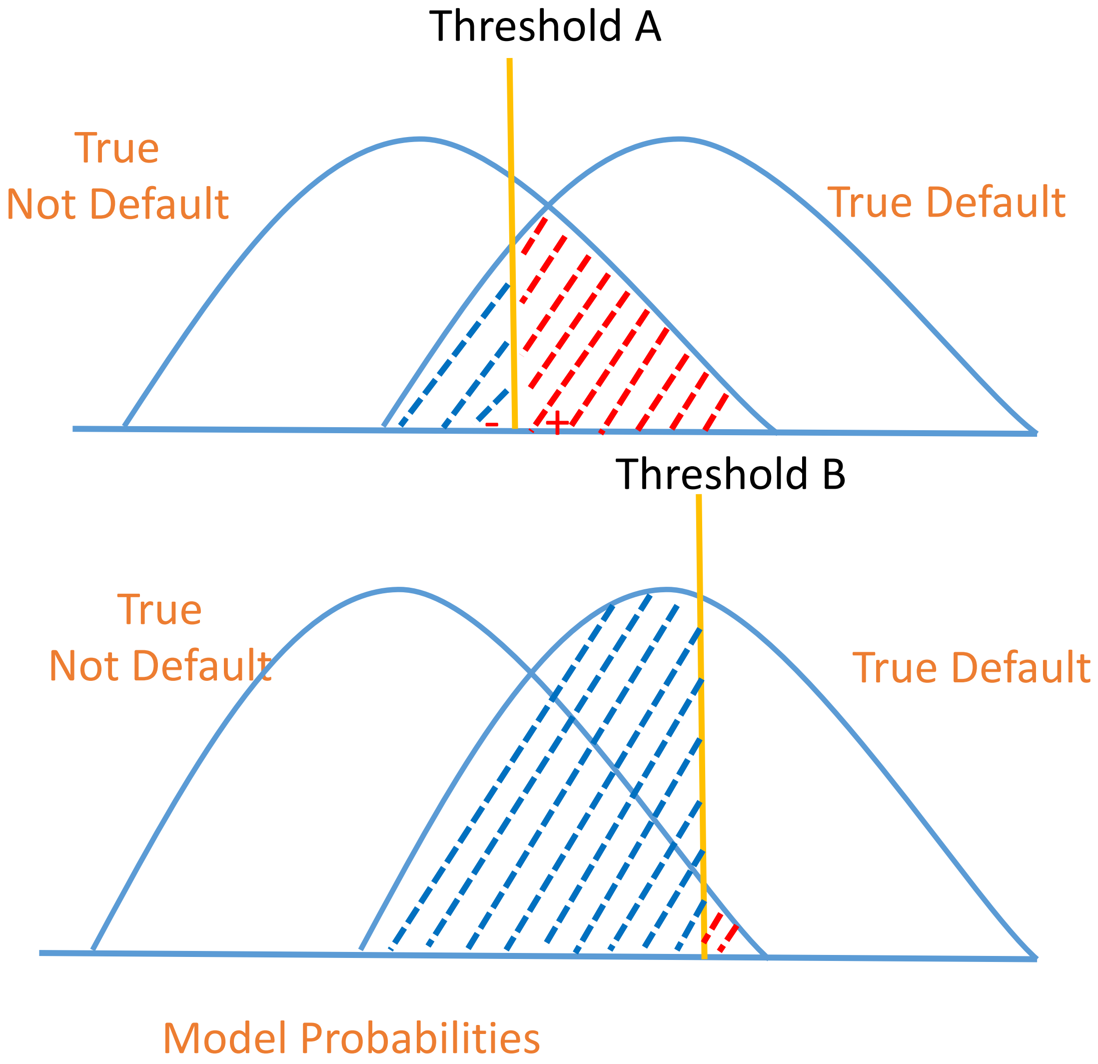
Comparing cutoffs:



Many false positives

Few false positives

Which Probability Cutoff To Use?



- Threshold you choose should depend on relative costs of FPs and FN
  - e.g. screening at airport (cost of false neg high)
  - e.g. direct mail advertisement (cost of false positive low)
- Some common choices
  - **Maximize Accuracy** (equal weighting of FPs and FNs)
  - **Threshold p_hat** > 0.5
  - **Minimize cost**: TC = costFP *FPs + costFN * FNs

Sensitivity and Specificity, Confusion Matrix at P Cutoff > 0.5

		True default status		
		No	Yes	
Predicted default status (cutoff p>0.5)	No	TN = 484	FN = 11	N^* = 495
	Yes	FP = 2	TP = 3	P^*= 5
		N = 486	P = 14	

- **Sensitivity:** True positive rate (aka 1 – power or recall)
  - TP/P = 3 / 14 = 21.4%
- **Specificity:** True negative rate
  - TN/N = 484 / 486= 99.5%
- **False positive rate** (aka Type I error, 1 - Specificity)
  - FP/N = 2/486= 0.004%

Generating Confusion Matrices in R

- To produce a confusion matrix in R we will use the yardstick package
- The function conf_mat() produces confusion matrices but we must format our data correctly
- We need to specify a data frame with
- Actual event (Y = 1) values
- Our estimated probabilities (scores)
- This example data frame shows how we need to structure our results data frame

Usage

```
conf_mat(data, ...)

## S3 method for class 'data.frame'
conf_mat(data, truth, estimate, dnn = c("Prediction", "Truth"), ...)

## S3 method for class 'conf_mat'
tidy(x, ...)

autoplot.conf_mat(object, type = "mosaic", ...)
```

```
> head(two_class_example)
   truth      Class1       Class2 predicted
1 Class2 0.003589243 0.9964107574    Class2
2 Class1 0.678621054 0.3213789460    Class1
3 Class2 0.110893522 0.8891064779    Class2
4 Class1 0.735161703 0.2648382969    Class1
5 Class2 0.016239960 0.9837600397    Class2
6 Class1 0.999275071 0.0007249286    Class1
```

Formatting Results Matrix for Confusion Matrix

- Let's store the model results in a data frame
- We must specify the actual default behavior
- And the probability of class1 (default) as well as probability of class2 (not default)
- We *must* specify a cutoff above which probabilities are classified as "class1" (or having the event) and below which they are not
- The cutoff probability is determined by the relative cost of false positives and false negatives! Do not use rules of thumb!

```r
results_logit <- data.frame(
  `truth` = Default$default,
  `Class1` =  scores,
  `Class2` = 1 - scores,
  `predicted` = as.factor(ifelse(scores > 0.4,
                      "Yes","No"))
)
```

Why Do So Many Practicing Data Scientists Not Understand Logistic Regression?

Posted on June 27, 2020 by W.D.

Logistic Regression is Not Fundamentally a Classification Algorithm

Classification is when you make a concrete determination of what category something is a part of. Binary classification involves two categories, and by the law of the excluded middle, that means binary classification is for determining whether something "is" or "is not" part of a single category. There either are children playing in the park today (1), or there are not (0).

https://ryxcommar.com/2020/06/27/why-do-so-many-practicing-data-scientists-not-understand-logistic-regression/

Producing Confusion Matrix Using Formatted Results Data

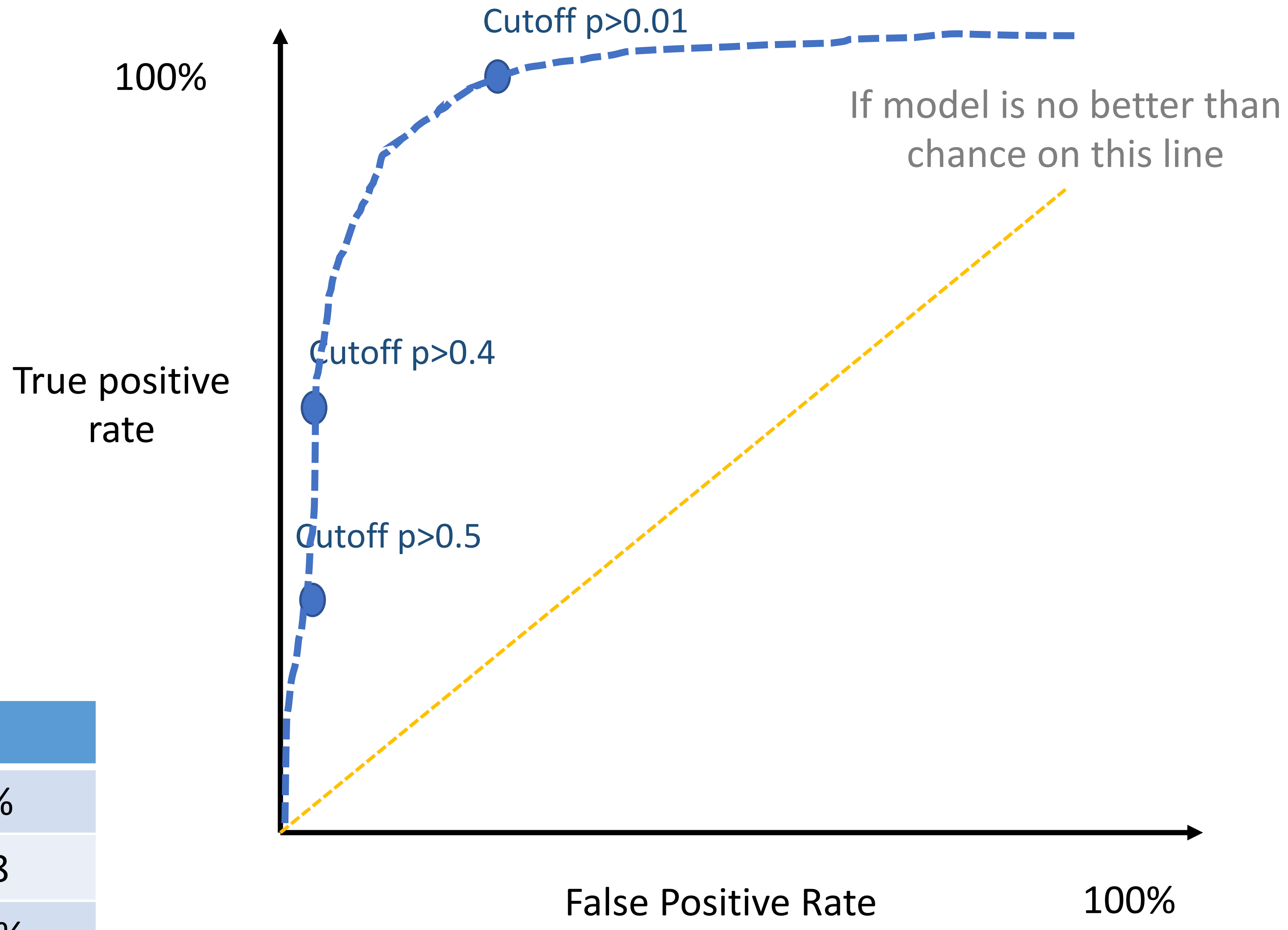
- The conf_mat() function shows the confusion matrix
- If we summarize the conf_mat() object we see more binary metrics of classification (don't need to know all of these)
- **Sensitivity** is the true positive rate (TP/P) and here we identify of the true positives = 131/333 = 39.3%
- We may need to lower our threshold of cutoff probability

```
> cm <- conf_mat(results_logit,
+                truth = truth,
+                estimate = predicted)
> print(cm)
          Truth
Prediction   No  Yes
       No  9594  202
       Yes   73  131
```

Continuous Cutoff: ROC Curve

- Can we show consequences of FPs and FNs as we vary the cutoff probability to assign classes?
- Idea of a ROC (Receiver Operator Curve) plot

Cutoff	TPR	FPR
0.01	100%	22.6%
0.4	57%	0.008
0.5	21.4%	0.004%
0.6	21.4%	0.002%



ROC Curves in R

```r
#-----------------------------------------------------------
# ROC plots
#-----------------------------------------------------------
library('ggplot2')
library('plotROC')

p <- ggplot(results_logit,
        aes(m = Class1, d = truth)) +
  geom_roc(labelsize = 3.5,
           cutoffs.at =
             c(0.99,0.9,0.7,0.5,0.3,0.1,0)) +
  theme_minimal(base_size = 16)
print(p)
```

- At a cutoff of 0.3, we get a true positive fraction of 0.5 and a false positive fraction of a very low number
- Better models lie up and to the left in the ROC plot
- AUC calculates how much total area is under a particular curve
- AUC of 0.947 is pretty good

```
> calc_auc(p)
  PANEL group       AUC
1     1    -1 0.9479842
```

Lab (time permitting)

```
#-----------------------------------------------------------
# Exercises
#-----------------------------------------------------------
# 1. Generate predictions using your logit_mod2 model
#    that predicts default as a function of
#    student, balance, and income
# 2. Generate predicted probabilities (score the model)
# 3. Create a results data frame and print a confusion
#    matrix using the results data
# 4. Plot a ROC curve using the results data
# 5. How well does the model perform?
```

Class 10 Summary

- Logit functions compress predictions to lie between 0 and 1, which are valid probabilities
- The logistic model models the outcome (Y) as the log odds ratio!
- Confusion matrices show the true/false positives/negatives.
- ROC plots measure the consequence on true positive fraction and false positive fraction for different cutoff probabilities
- Higher AUC scores mean a better ROC plot indicating a better model