# Class 5

BUS 696

Prof. Jonathan Hersh

# BUS 696: Class 5 Announcements

1. Pset 3 solutions
2. Problem Set 4 Posted
3. Questions?

# BUS 696: Class 5 Outline

1. AI in the News
2. Classification Intro
3. Why not Linear Models?
4. Sigmoid/Logistic Model
5. Odds Ratio
6. Estimating logit model using glm()
7. Generating predicted probabilities
8. Choosing probability cutoffs
9. Confusion matrices
10. More Diagnostics

# AI in the News

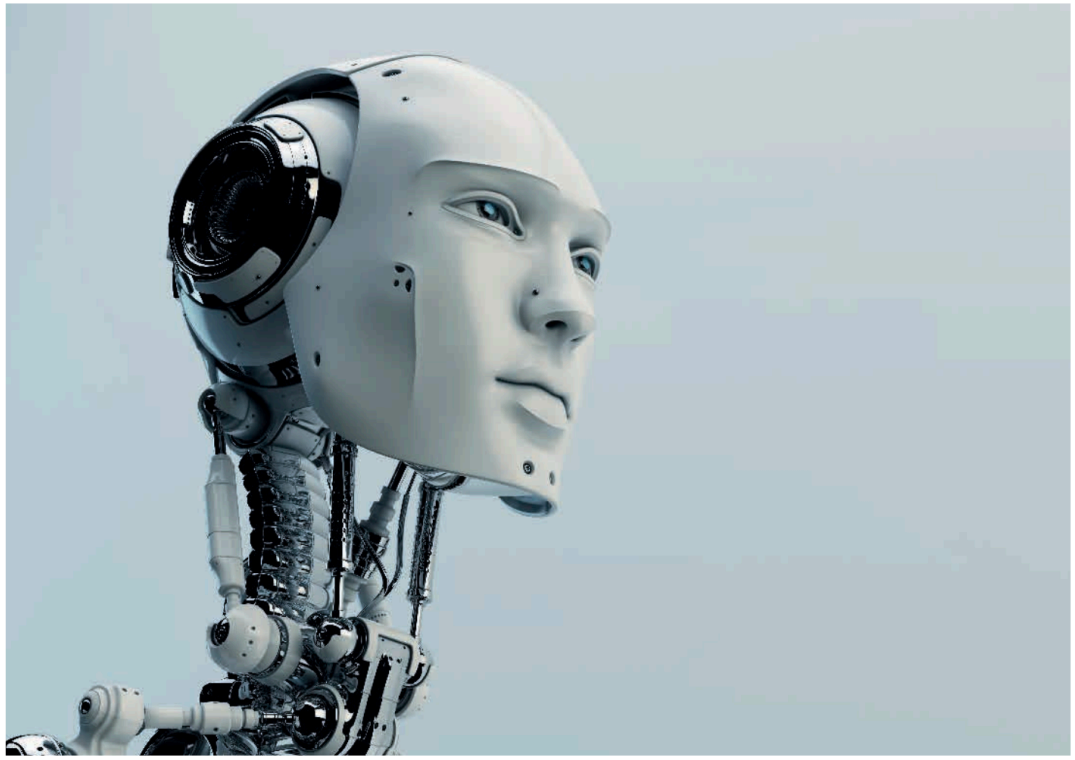
## Vanguard Bets on Robo-Only Adviser

Money manager is aiming to capture younger, tech-savvy investors

Vanguard, founded by John C. Bogle, became a household name through index funds that track markets and don't charge high fees. PHOTO: RYAN COLLERD FOR THE WALL STREET JOURNAL

# Robo-Advisers Background

Deloitte.



## The expansion of Robo-Advisory in Wealth Management

The takeover of the robots in the classic field of Wealth Management is an emerging trend across the industry. Is this

# What is classification?

Regression problem: identifying to which discrete class a particular observation belongs.

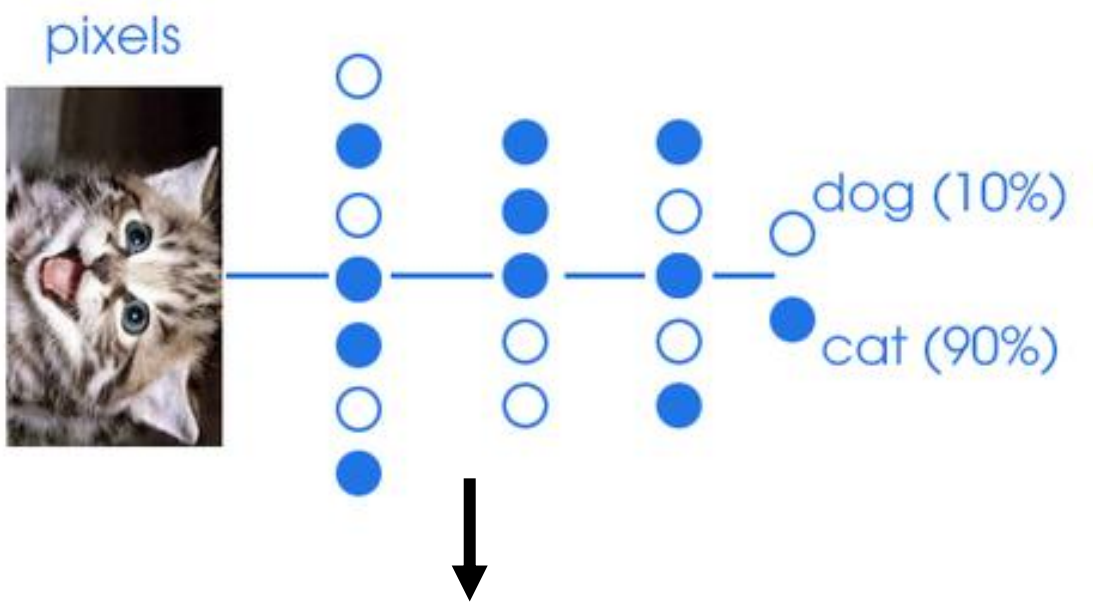
$y_i \in \{0,1\}$

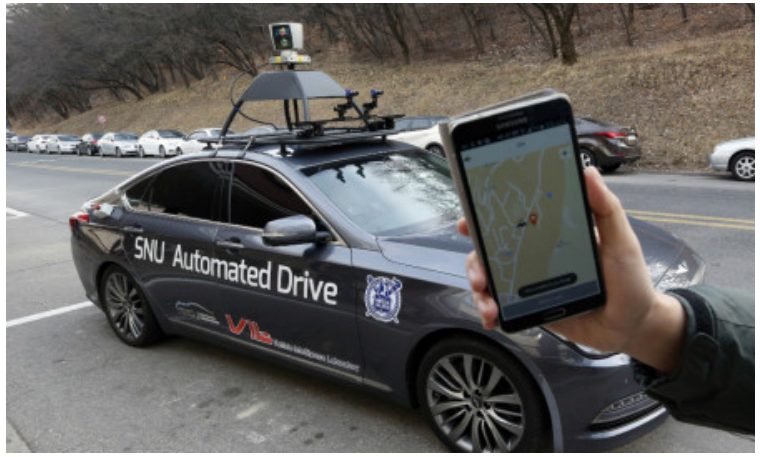
$y_i \in \{red, black, blue, blond, green\}$

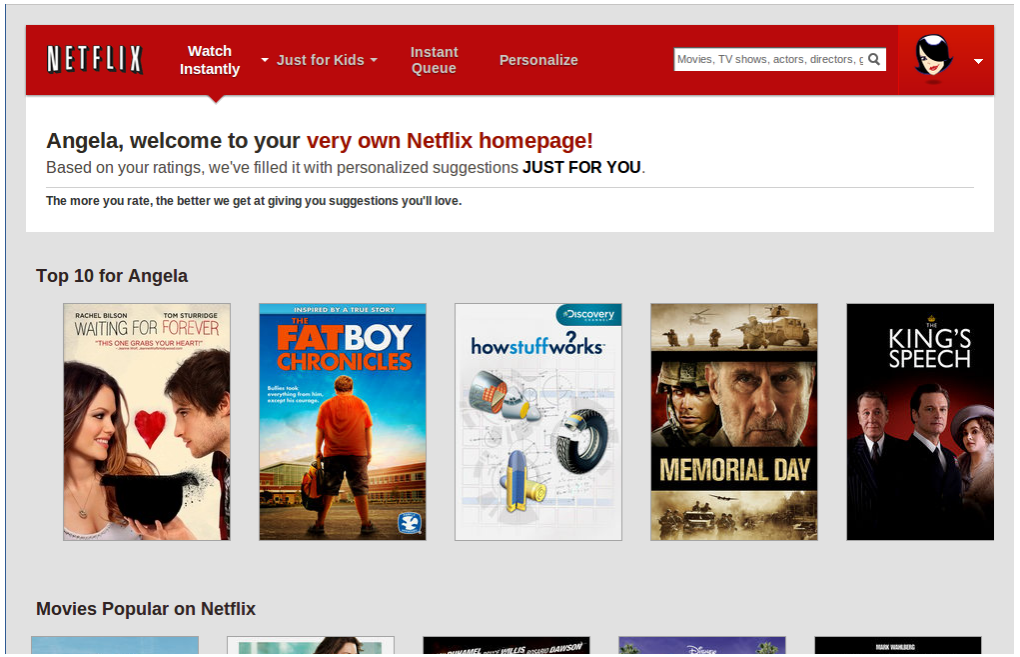
# Classification examples

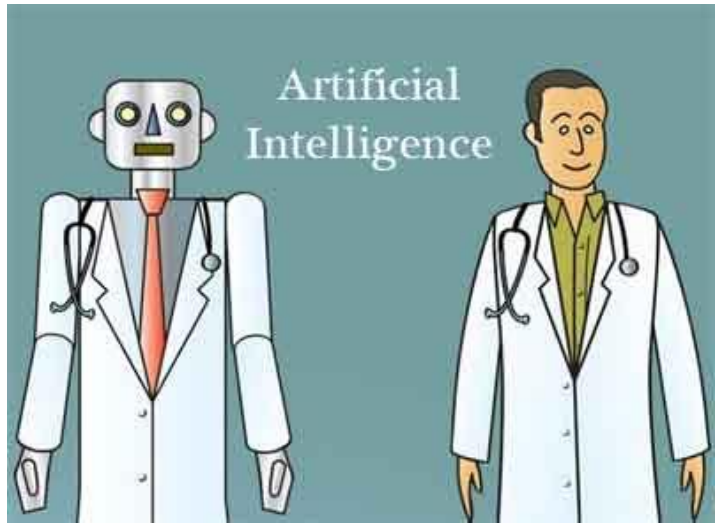


# More classification examples



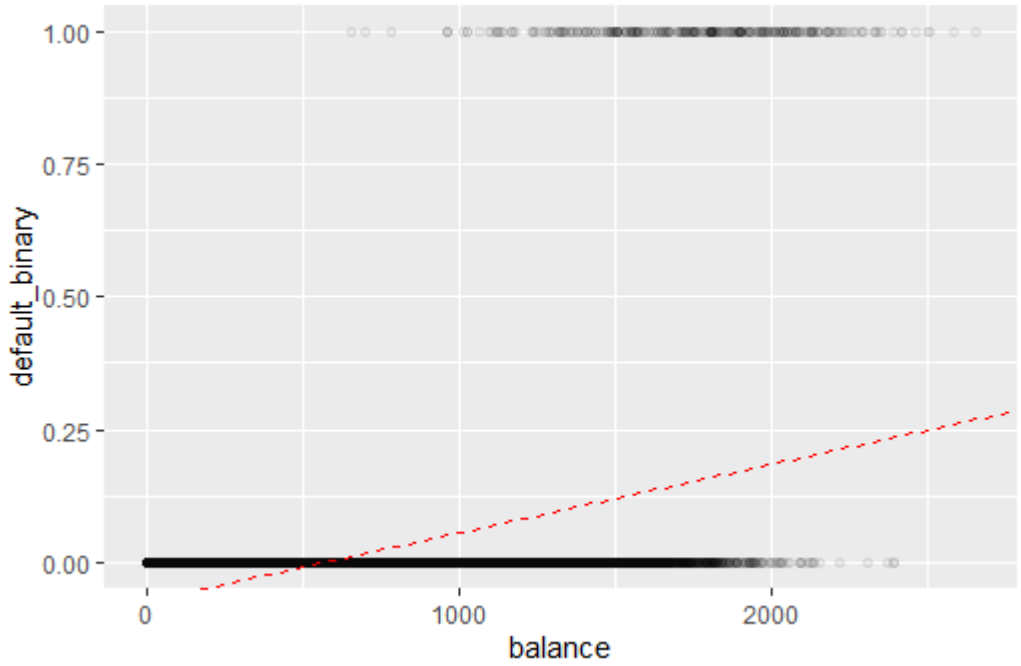
# Why not regression?

```r
library(ISLR)
data(Default)
options(scipen = 3)

library(magrittr)
library(tidyverse)
library(ggExtra)

# create a binary version of default
Default %<>% mutate(default_binary =
                      ifelse(default == "Yes", 1,0))

summary(Default)

# estimate an OLS model using the 0,1
# variable as our dependent variable
mod1 <- lm(default_binary ~ balance,
           data = Default)
summary(mod1)
```

```
> summary(mod1)

Call:
lm(formula = default_binary ~ balance, data = Default)

Residuals:
     Min       1Q   Median       3Q      Max
-0.23533 -0.06939 -0.02628  0.02004  0.99046

Coefficients:
              Estimate   Std. Error t value Pr(>|t|)
(Intercept) -0.075191959  0.003354360  -22.42   <2e-16 ***
balance      0.000129872  0.000003475   37.37   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1681 on 9998 degrees of freedom
Multiple R-squared:  0.1226,    Adjusted R-squared:  0.1225
F-statistic:  1397 on 1 and 9998 DF,  p-value: < 2.2e-16
```

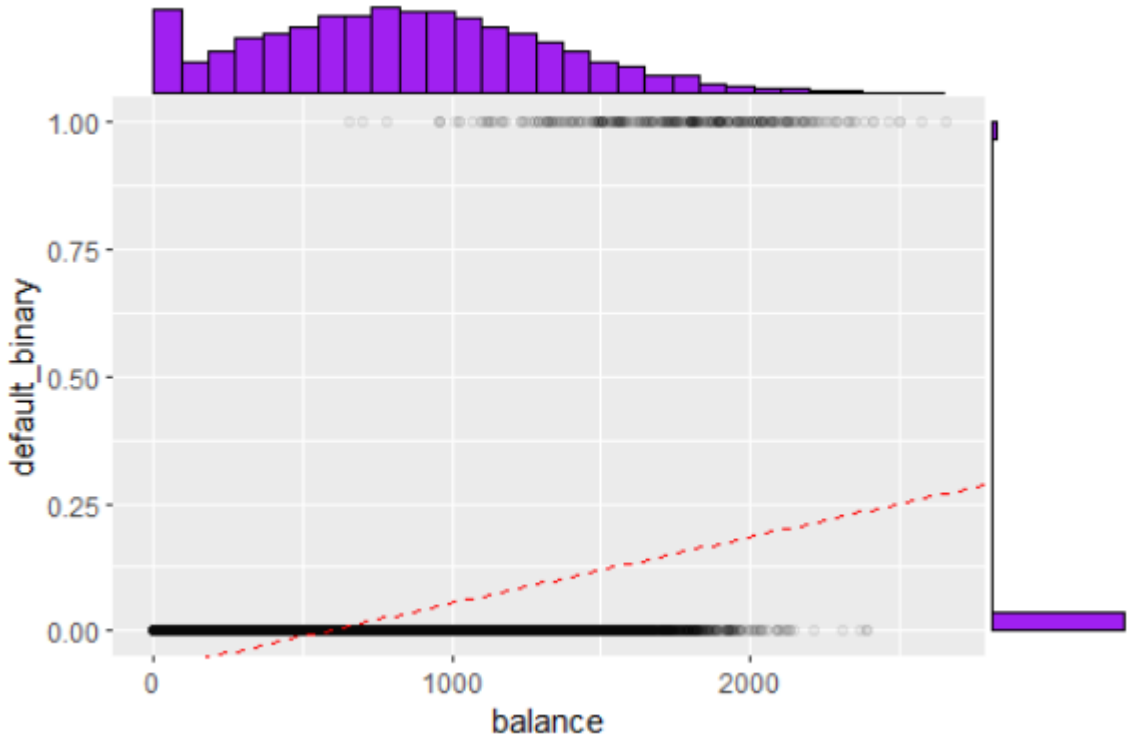
# Why not regression?

```
preds_DF <- data.frame(
  preds = predict(mod1),
  Default
)

# what kind of predictions do we get for this model?
head(preds_DF)

p <- ggplot(preds_DF, aes(x = balance,
                          y = default_binary)) +
  geom_point(alpha = 1/20) +
  geom_abline(intercept = mod1$coefficients[1],
              slope = mod1$coefficients[2],
              color = "red", linetype = "dashed")

plot(p)
```

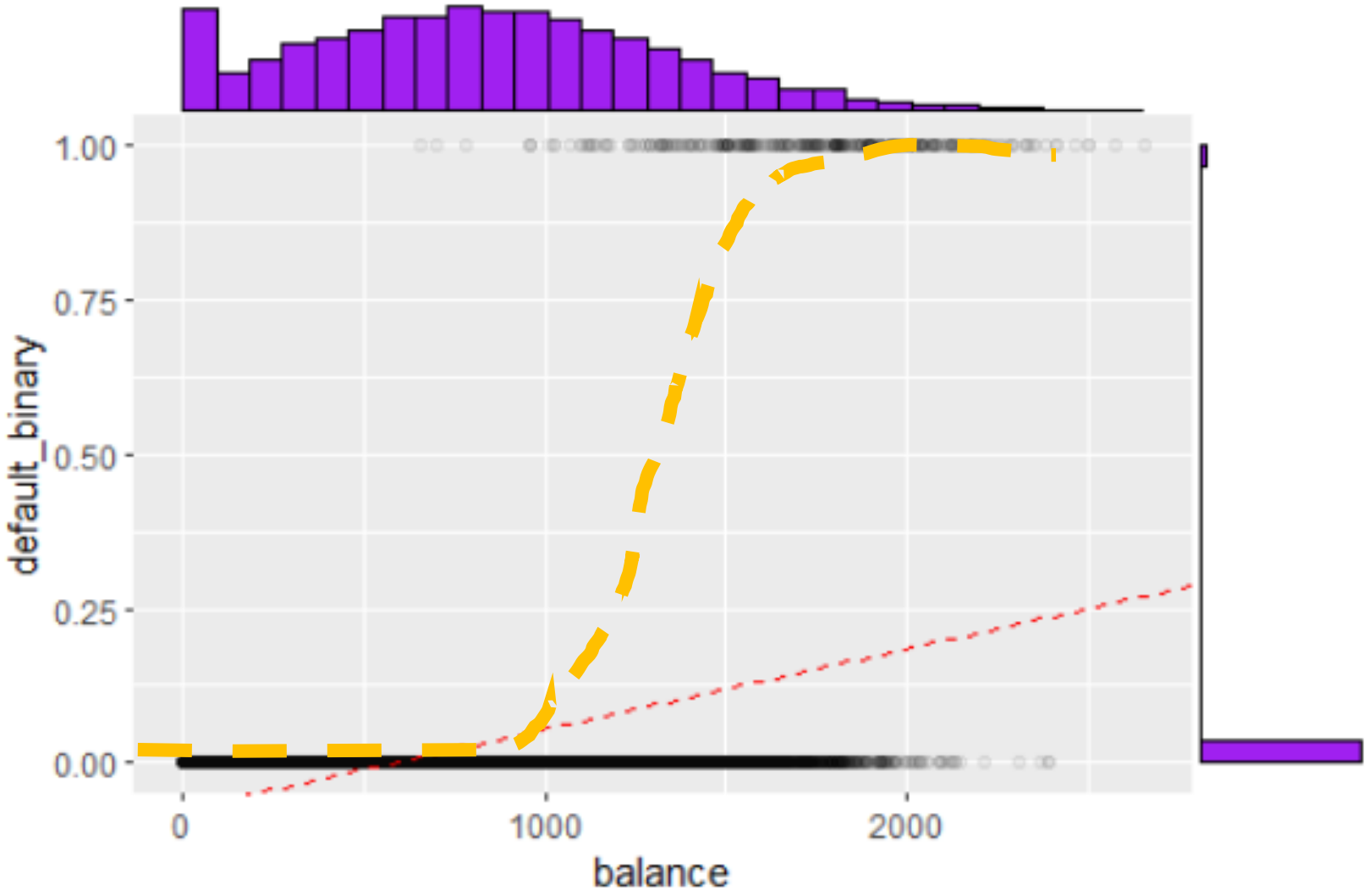


# Why not regression?

```
preds_DF <- data.frame(
  preds = predict(mod1),
  Default
)

# what kind of predictions do we get for this model?
head(preds_DF)

p <- ggplot(preds_DF, aes(x = balance,
                          y = default_binary)) +
  geom_point(alpha = 1/20) +
  geom_abline(intercept = mod1$coefficients[1],
              slope = mod1$coefficients[2],
              color = "red", linetype = "dashed")

plot(p)
p <- ggMarginal(p, type = "histogram", fill="purple", size=6)
plot(p)
```



# What We Want Ideally:




- i.e let $p(X) = Pr(Y = 1|X)$ be the probability the event occurs
- We want our model to output:
  $Pr(Y = 1|X) \in [0,1]$
- Because probabilities are between 0 and 1.

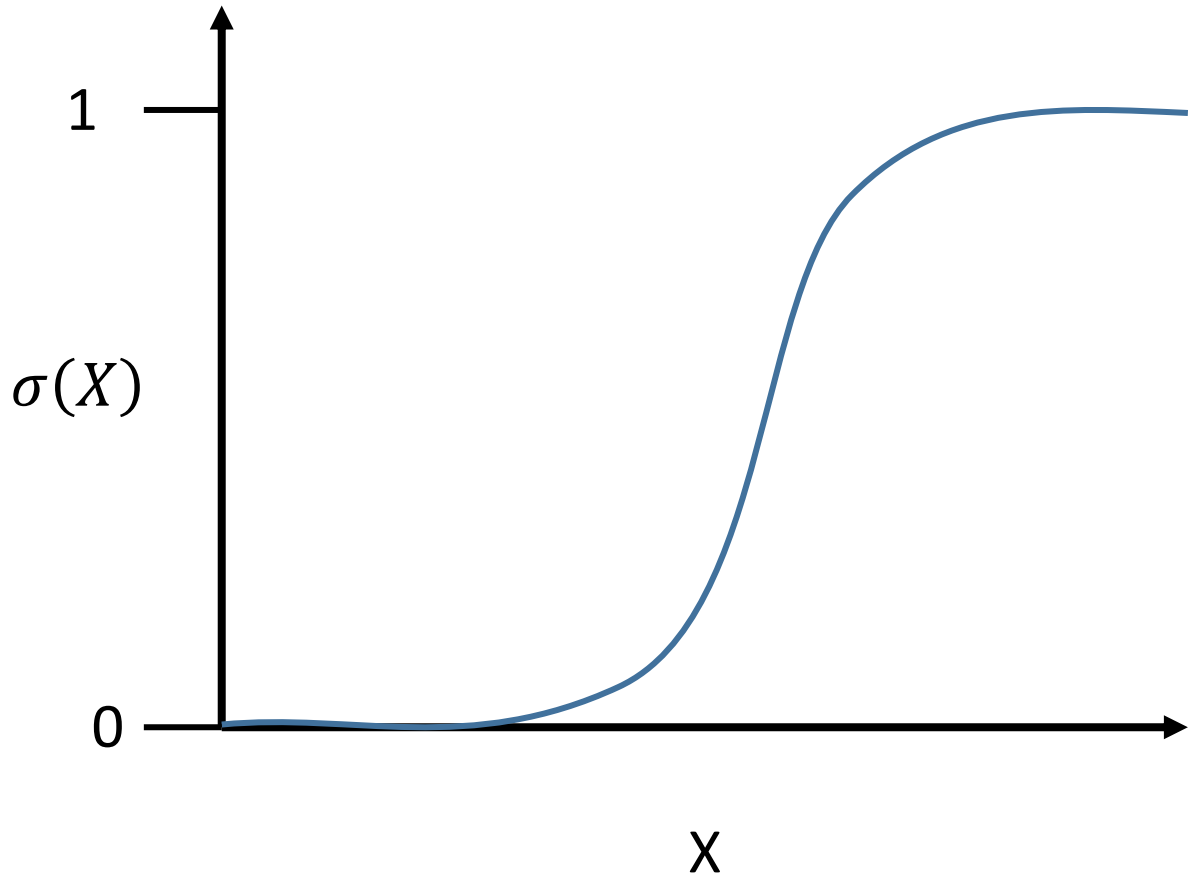
# Logistic/Sigmoid Function

Function that naturally takes

inputs X and transforms

between 0 and 1

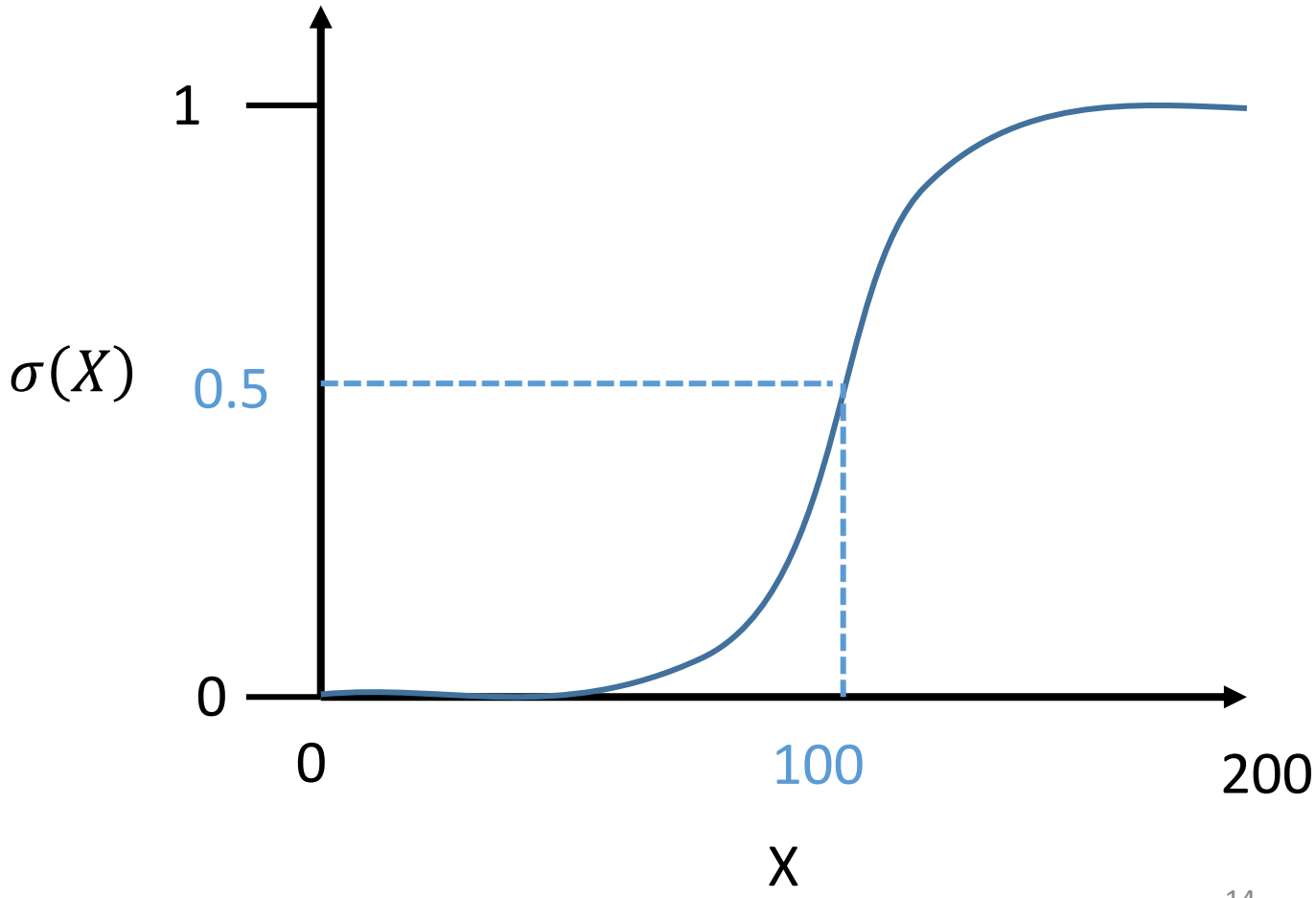
$$\sigma(X) = \frac{1}{1 + e^{-X}} = \frac{e^{X}}{e^{X} + 1}$$



# Logistic/Sigmoid Function

Function that naturally takes inputs X and transforms between 0 and 1

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

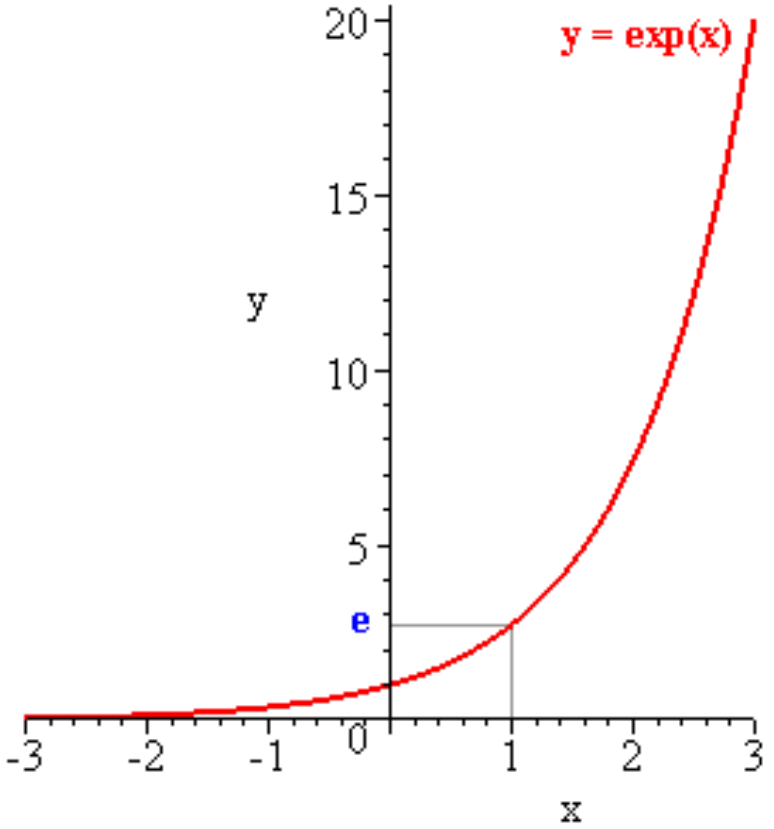


# A note on $e^X = \exp(X)$

- Super spooky mathematical function
- $e = 2.718281828459045 \ldots$
- $\frac{d}{dx} e^{\mathrm{X}} = \mathrm{e}^{\mathrm{X}}$ and $e^0 = 1$
  - e.g. rate of increase in function at

    X is equal to the function at X

Many other ways to characterize function

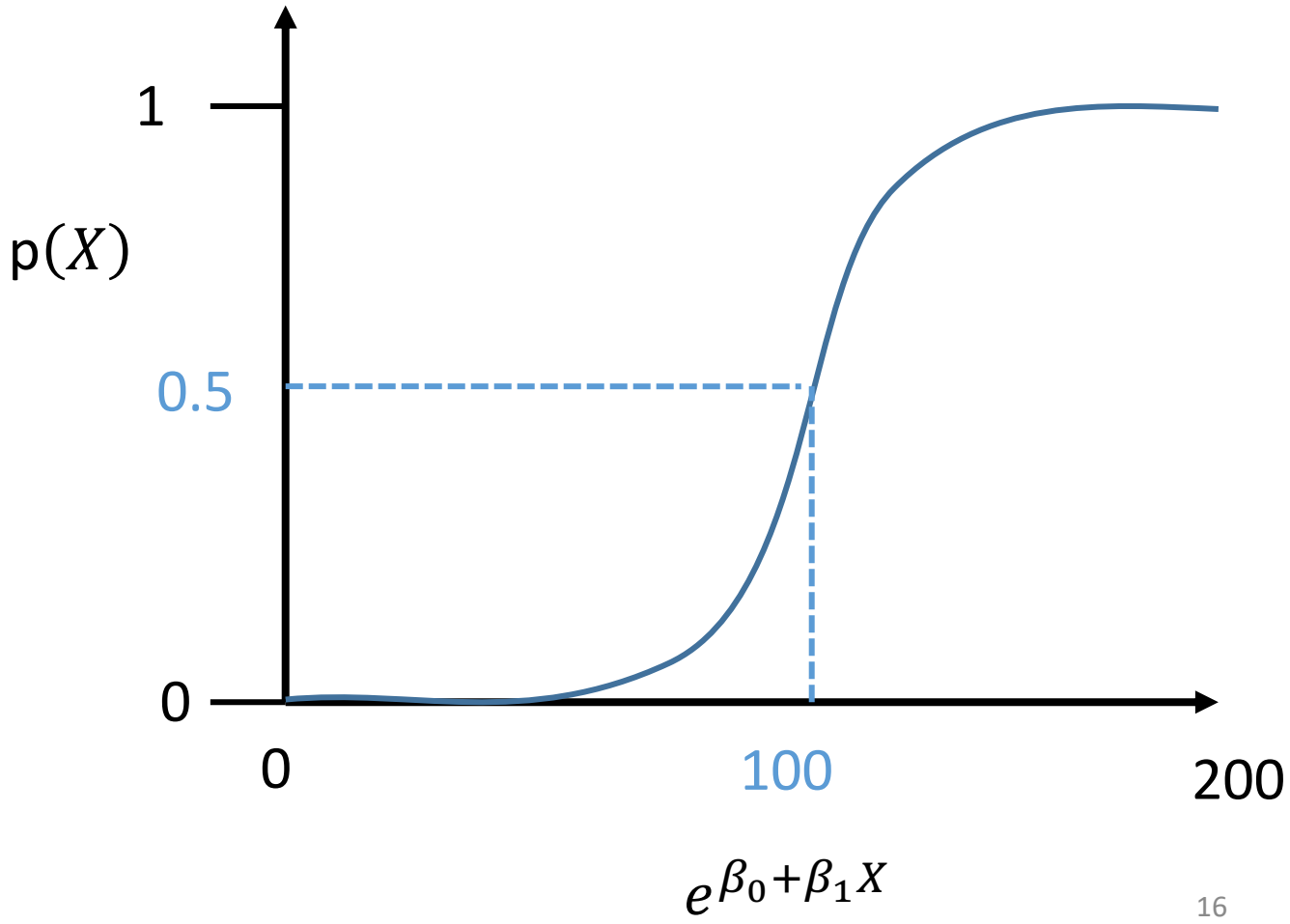


# Logistic/Sigmoid Function to Probabilities

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

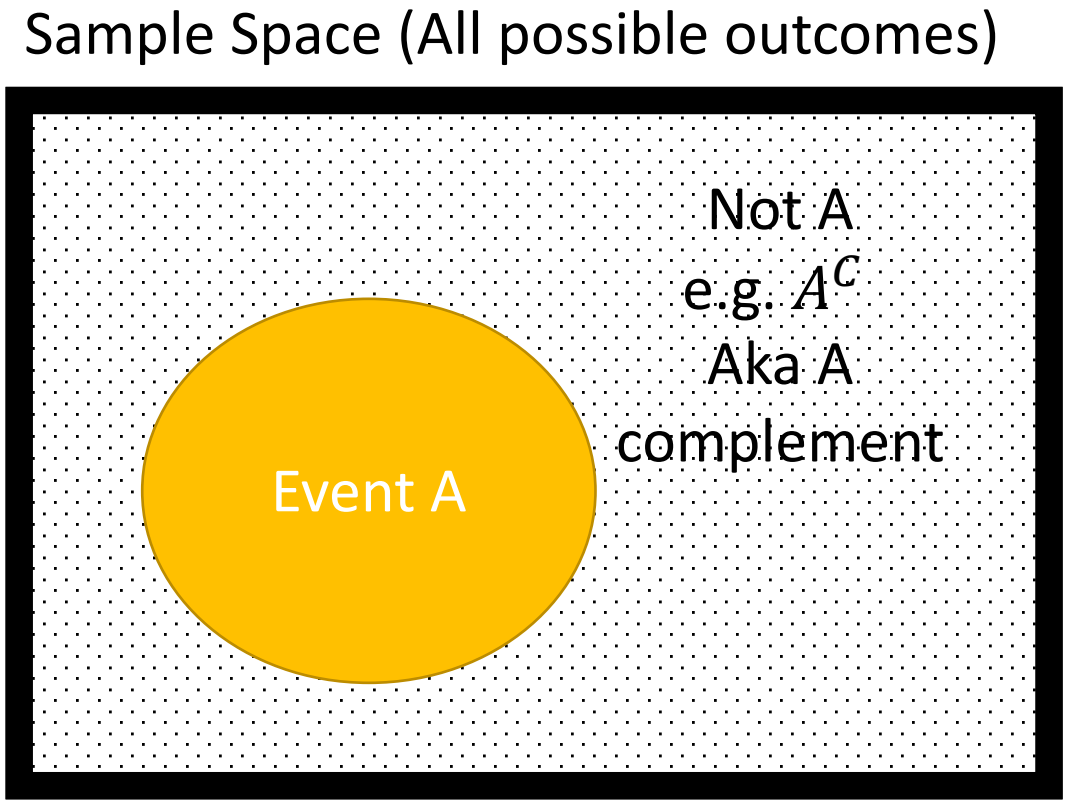
$$Pr(Y = 1|X) = \frac{e^{\beta_0 + \beta_1 \cdot X}}{e^{\beta_0 + \beta_1 \cdot X} + 1}$$

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



# Probability Note on The Complement

- $Q$: if $\Pr(A) = 30\%$
- What is the $\Pr(A^C)$ ?




- Because events A and not A fully partition the sample space

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

- Fully partition the sample space:

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

# One Weird Trick to Find P(Y=0)

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

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

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

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

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

# Credit Card Default Dataset

Default {ISLR} R Documentation

## Credit Card Default Data

### Description

A simulated data set containing information on ten thousand customers. The aim here is to predict which customers will default on their credit card debt.

### Usage

```
Default
```

### Format

A data frame with 10000 observations on the following 4 variables.

`default`

A factor with levels `No` and `Yes` indicating whether the customer defaulted on their debt

`student`

A factor with levels `No` and `Yes` indicating whether the customer is a student

`balance`

The average balance that the customer has remaining on their credit card after making their monthly payment

`income`

Income of customer

### Source

Simulated data

# Estimate Logit Model Using glm()

```
library(ISLR)
data(Default)
options(scipen=9)

glm_fit <- glm(default ~ balance,
               family = binomial,
               data = Default)
```

```
> summary(glm_fit)

Call:
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

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 2920.6  on 9999  degrees of freedom
Residual deviance: 1596.5  on 9998  degrees of freedom
AIC: 1600.5

Number of Fisher Scoring iterations: 8
```

# Making Predictions From Estimated Logit model

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

```
> round(glm_fit$coefficients,4)
(Intercept)     balance
   -10.6513      0.0055
```

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

```
top <- exp(glm_fit$coefficients[1] +
           glm_fit$coefficients[2] * 1000)
p_hat_1000 <- top / (1 + top)
```

# Making Predictions From Estimated Logit Model Balance == $2000

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

```
> round(glm_fit$coefficients,4)
(Intercept)     balance
   -10.6513      0.0055
```

$$\hat{p}(X = 1000) = \frac{e^{-10.6513 + 0.0055 \cdot \mathbf{2000}}}{1 + e^{-10.6513 + 0.0055 \cdot \mathbf{2000}}} =$$

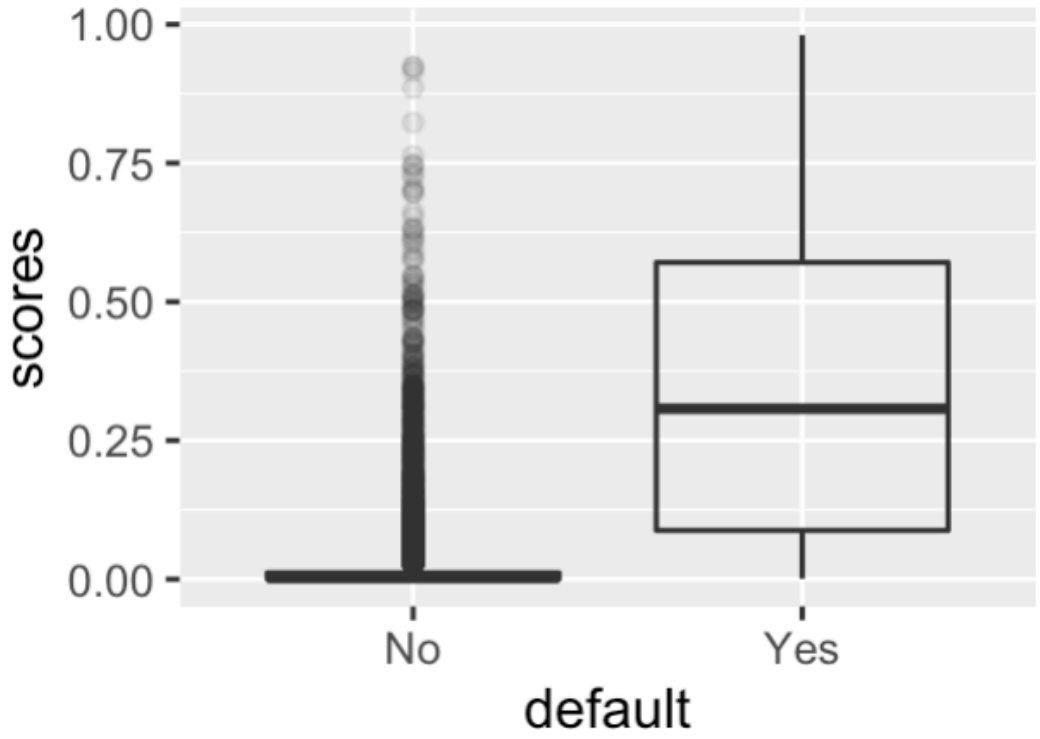
```
top <- exp(glm_fit$coefficients[1] +
             glm_fit$coefficients[2] * 2000)
p_hat_1000 <- top / (1 + top)

p_hat_1000
```

# Generating Predicted Probabilities For All Observations in Our Dataset

- We say we “score” the model when we assign predicted probabilities for each observation in our datasetd

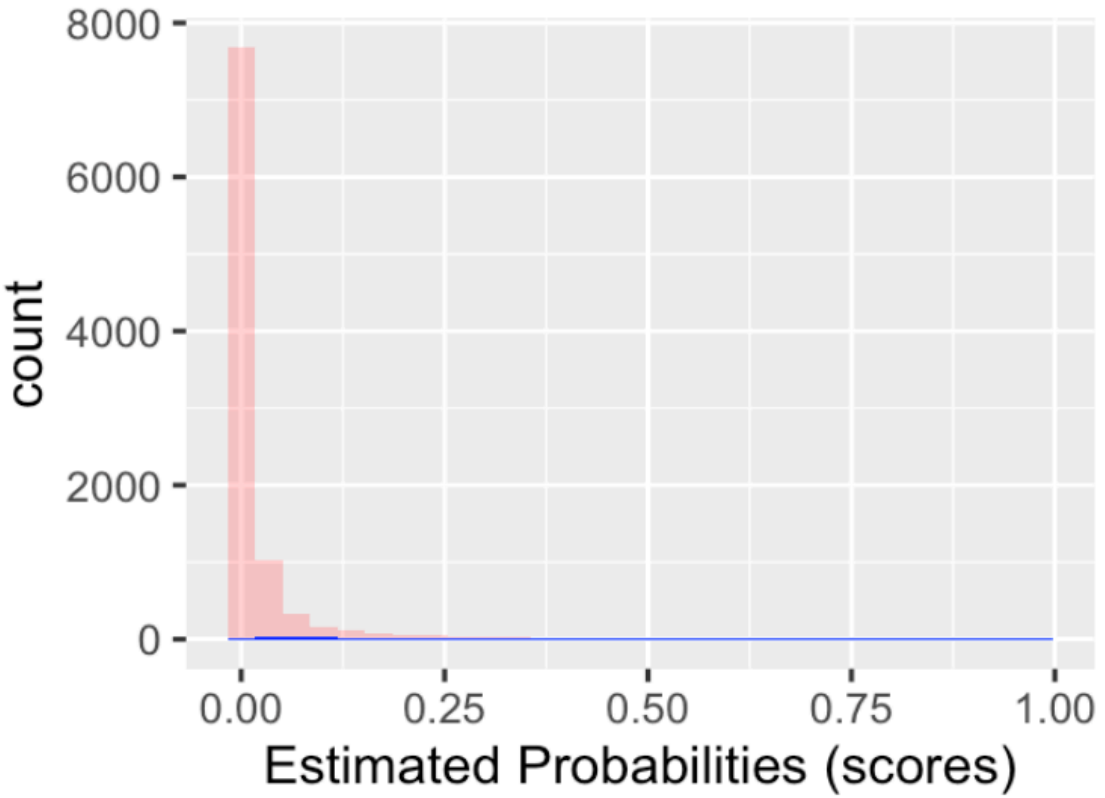
```
scores <- predict(glm_fit,
                  type = "response")
```

- **Note type = “response”!**
  Otherwise what happens?



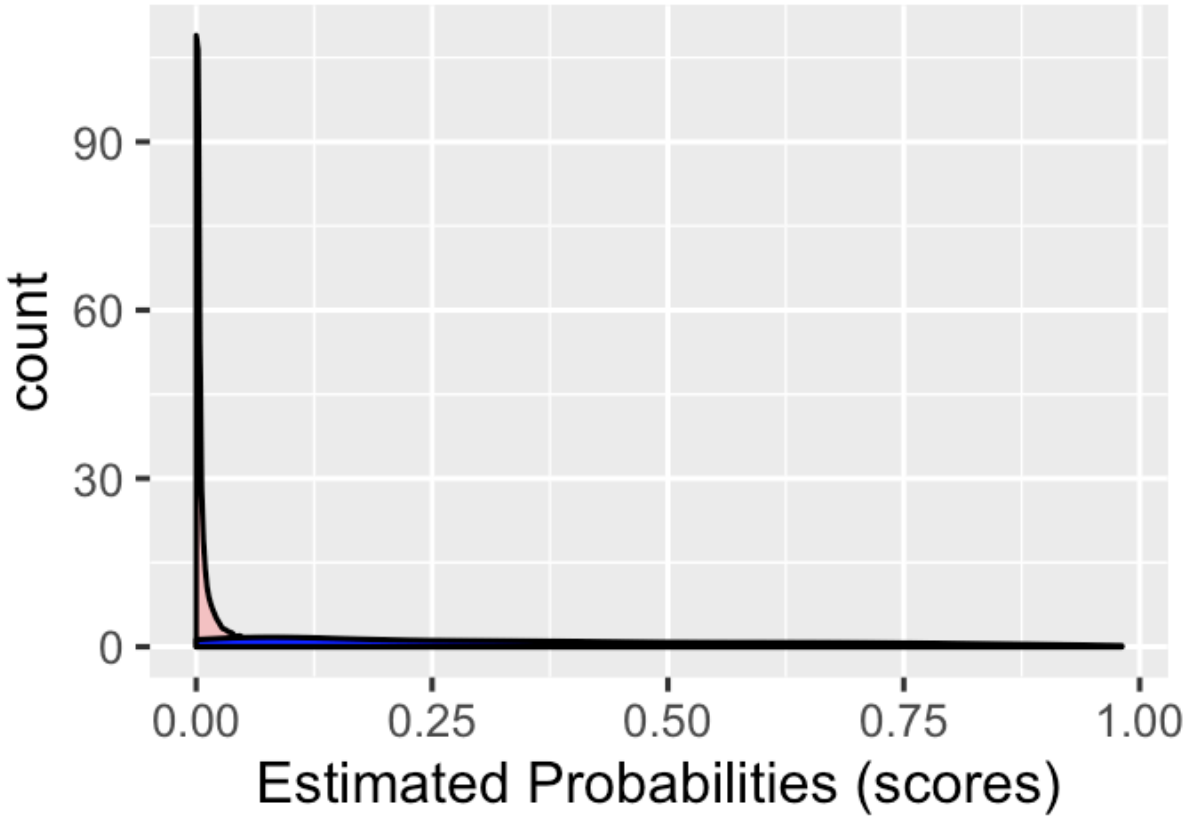
# Histogram of Scores by Default Status

```
ggplot(preds_DF, aes(x = scores_mod1)) +
  geom_histogram(data = preds_DF %>%
                   filter(default == "No"),
                 fill = "red", alpha = 0.2) +
  geom_histogram(data = preds_DF %>%
                   filter(default == "Yes"),
                 fill = "blue", alpha = 1) +
  labs(x = "Estimated Probabilities (scores)",
       y = "count")
```



# Density of Scores by Default Status

```
ggplot(preds_DF, aes(x = scores_mod1)) +
  geom_density(data = preds_DF %>%
                 filter(default == "No"),
               fill = "red", alpha = 0.2) +
  geom_density(data = preds_DF %>%
                 filter(default == "Yes"),
               fill = "blue", alpha = 1) +
  labs(x = "Estimated Probabilities (scores)",
       y = "count")
```



# Expressing Ratio of Probabilities: Odds Ratio

- Armed with P(Y=1) and P(Y=0) we know probabilities for each of these events
- A useful expression is the odds ratio, or the ratio of events occurring

$$\frac{p(Y=1|X)}{p(Y=0|X)} =$$

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

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

$$= e^{\beta_0+\beta_1 X}$$

# Example odds ratio: pr(Y = “rain” | X)

$$\frac{p(Y = rain|X)}{1 - p(Y = rain|X)} =$$

$$= \frac{0.2}{0.8} = \frac{1}{4}$$

Boston, MA 10 Day Weather

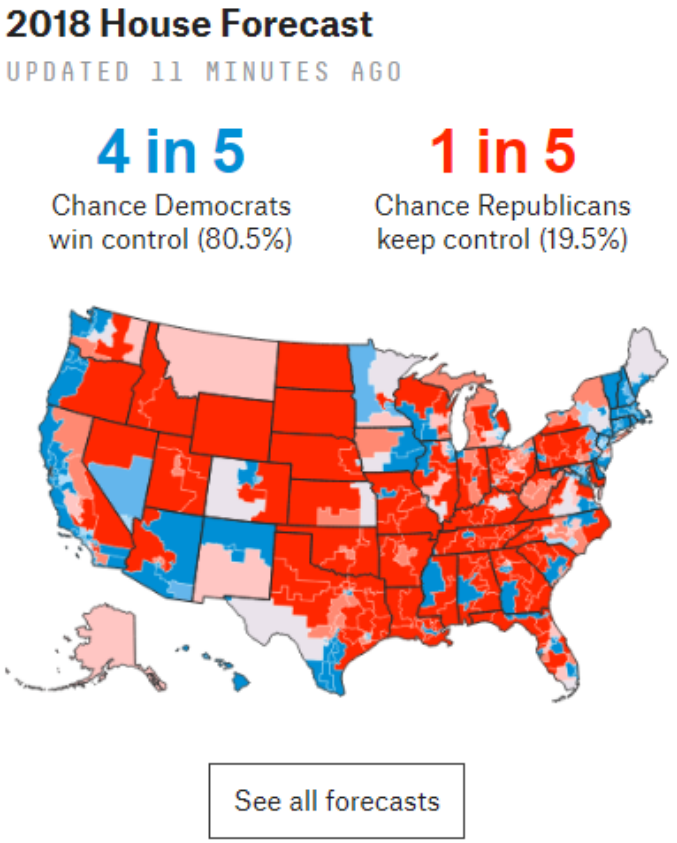
12:31 pm EDT12:29 pm EDT Print

| DAY | DESCRIPTION | HIGH / LOW | PRECIP |
|---|---|---|---|
| TODAY SEP 19 | Cloudy | 64°/58° | 0% |
| THU SEP 20 | Partly Cloudy | 65°/58° | 20% |

# Example odds ratio: pr("Dems win house"|X)

$$\frac{p(Y = \text{“}Dems\ win\text{”}|X)}{1 - p(Y = \text{“}Dems\ win\text{”}|X)} =$$

$$= \frac{0.8}{0.2} = 4$$



# Log Odds Ratio

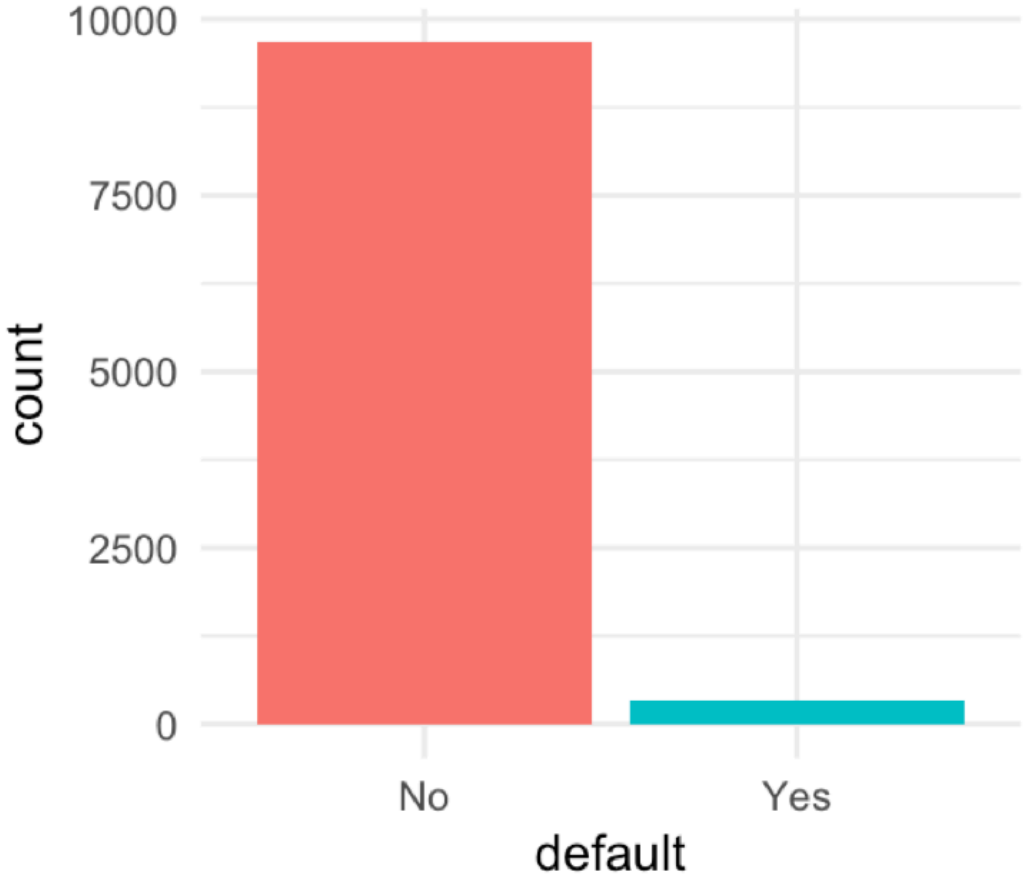
$$\frac{p(Y=1|X)}{p(Y=0|\,X)} = e^{\beta_0+\beta_1 X}$$

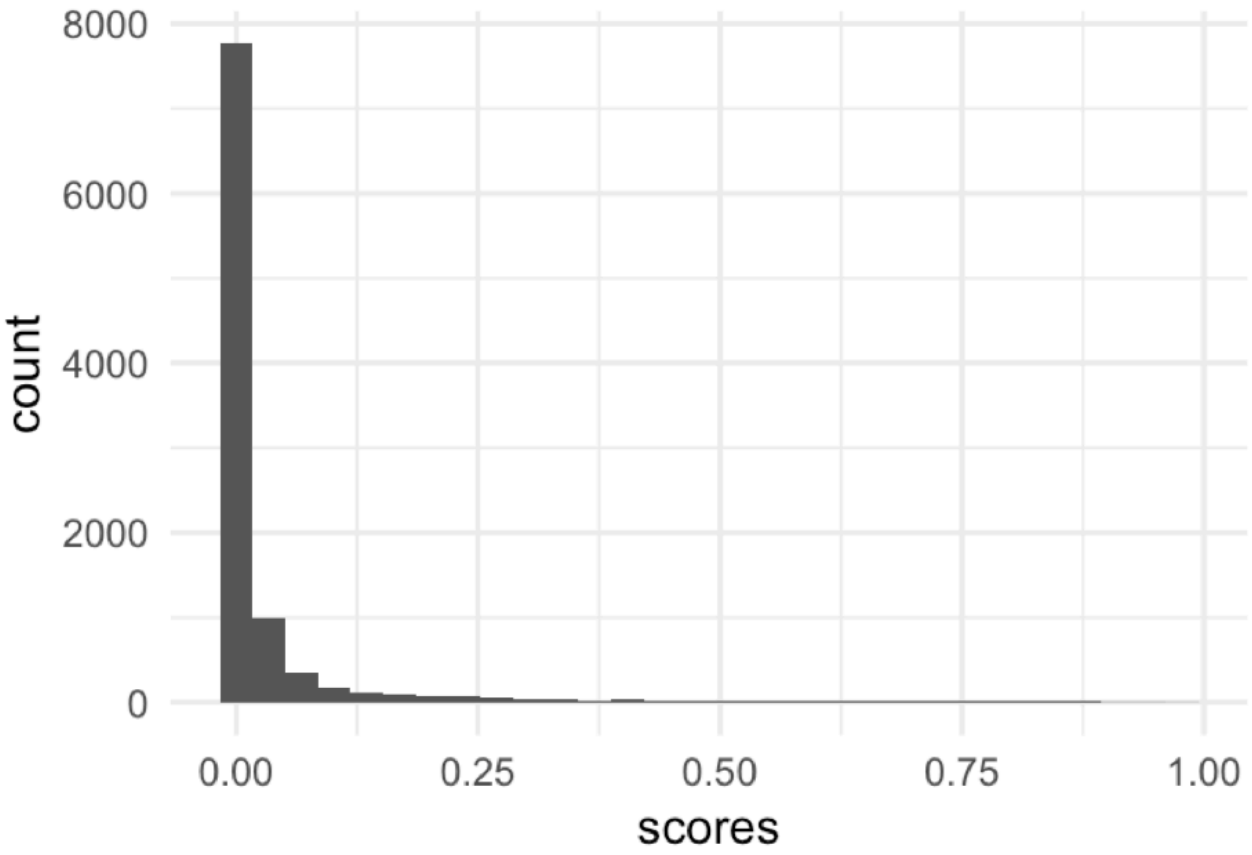
$$log\left(\frac{p(Y=1|X)}{p(Y=0|\,X)}\right) = \log\left(e^{\beta_0+\beta_1 X}\right)$$

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

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

# From probabilities to classes?



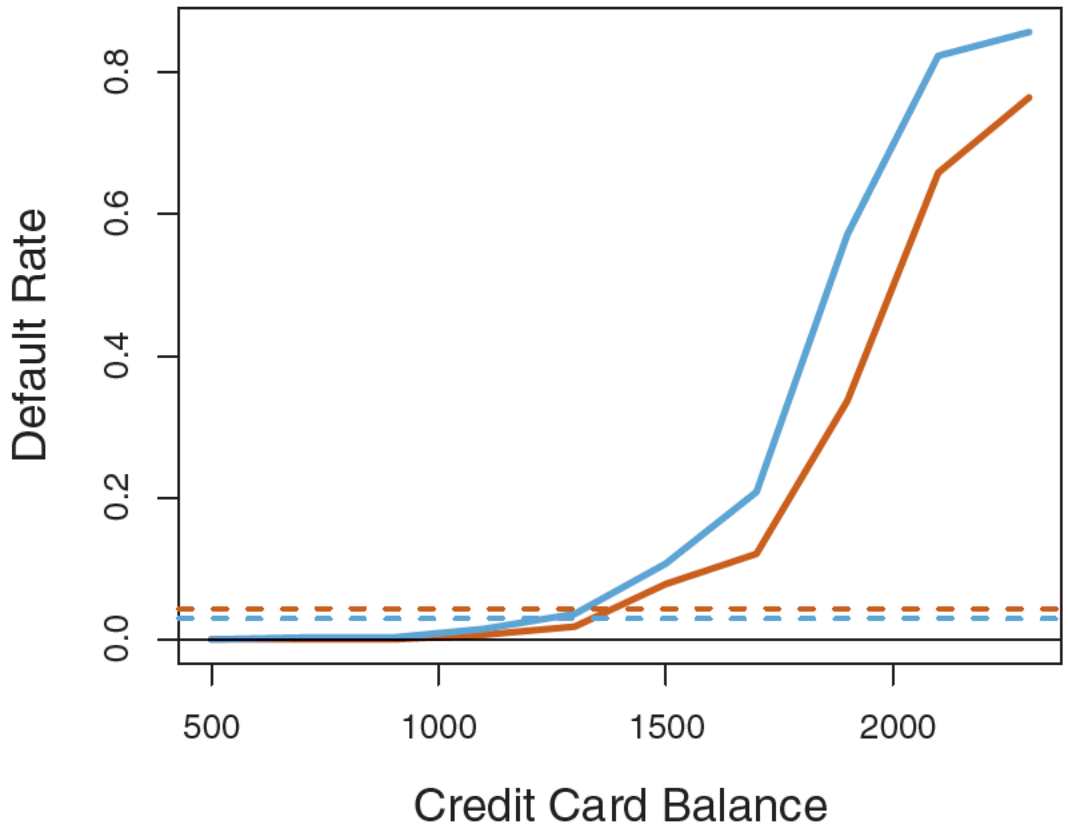



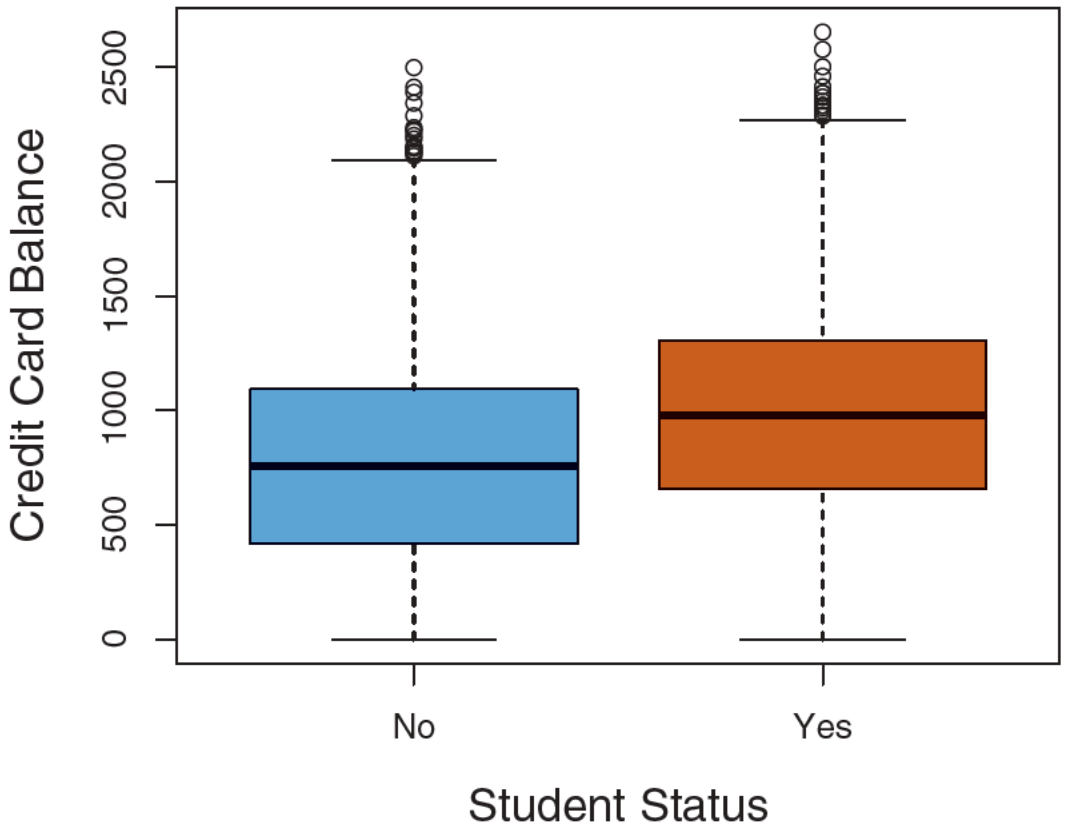
# Adding More Variables

```
Call:
glm(formula = default ~ student + balance + income, family = binomial
    data = Default)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.4691  -0.1418  -0.0557  -0.0203   3.7383

Coefficients:
                Estimate   Std. Error z value Pr(>|z|)
(Intercept) -10.869045196  0.492255516 -22.080  < 2e-16 ***
studentYes   -0.646775807  0.236252529  -2.738  0.00619 **
balance       0.005736505  0.000231895  24.738  < 2e-16 ***
income        0.000003033  0.000008203   0.370  0.71152
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

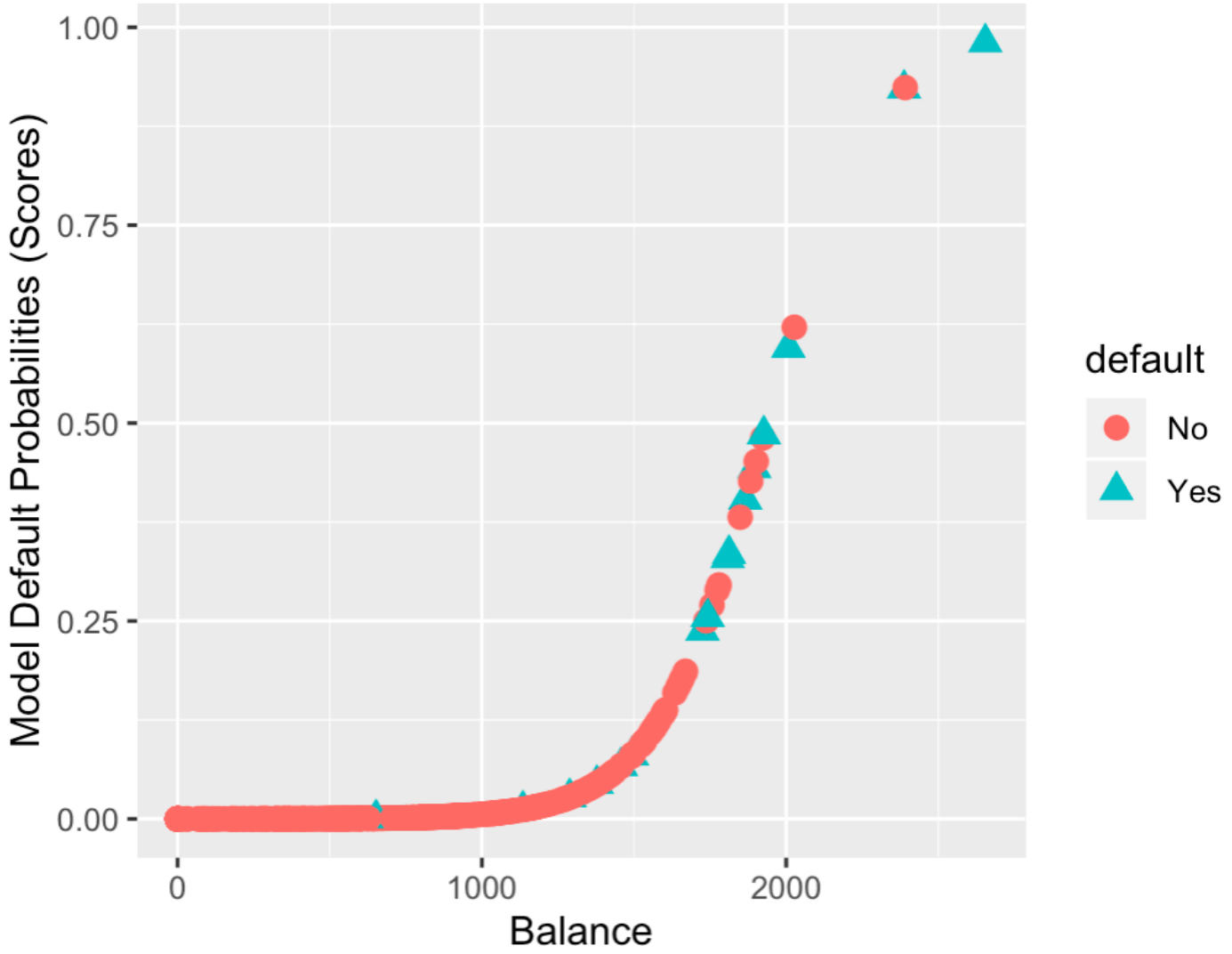
Interpret coefficients as impact on log odds!

# Student as “confounder”



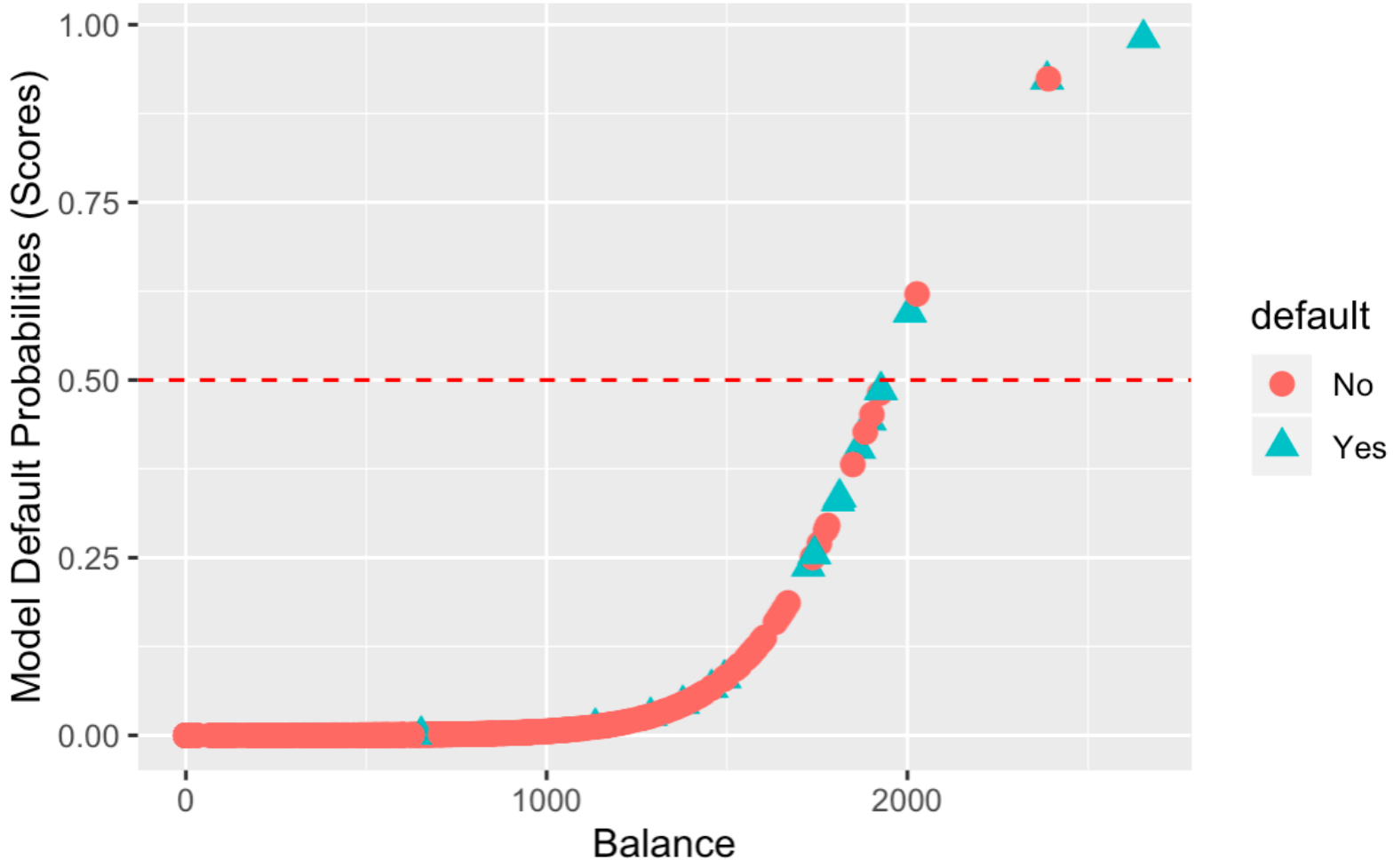
# Plotting Model Probabilities Against Balance by Default Status



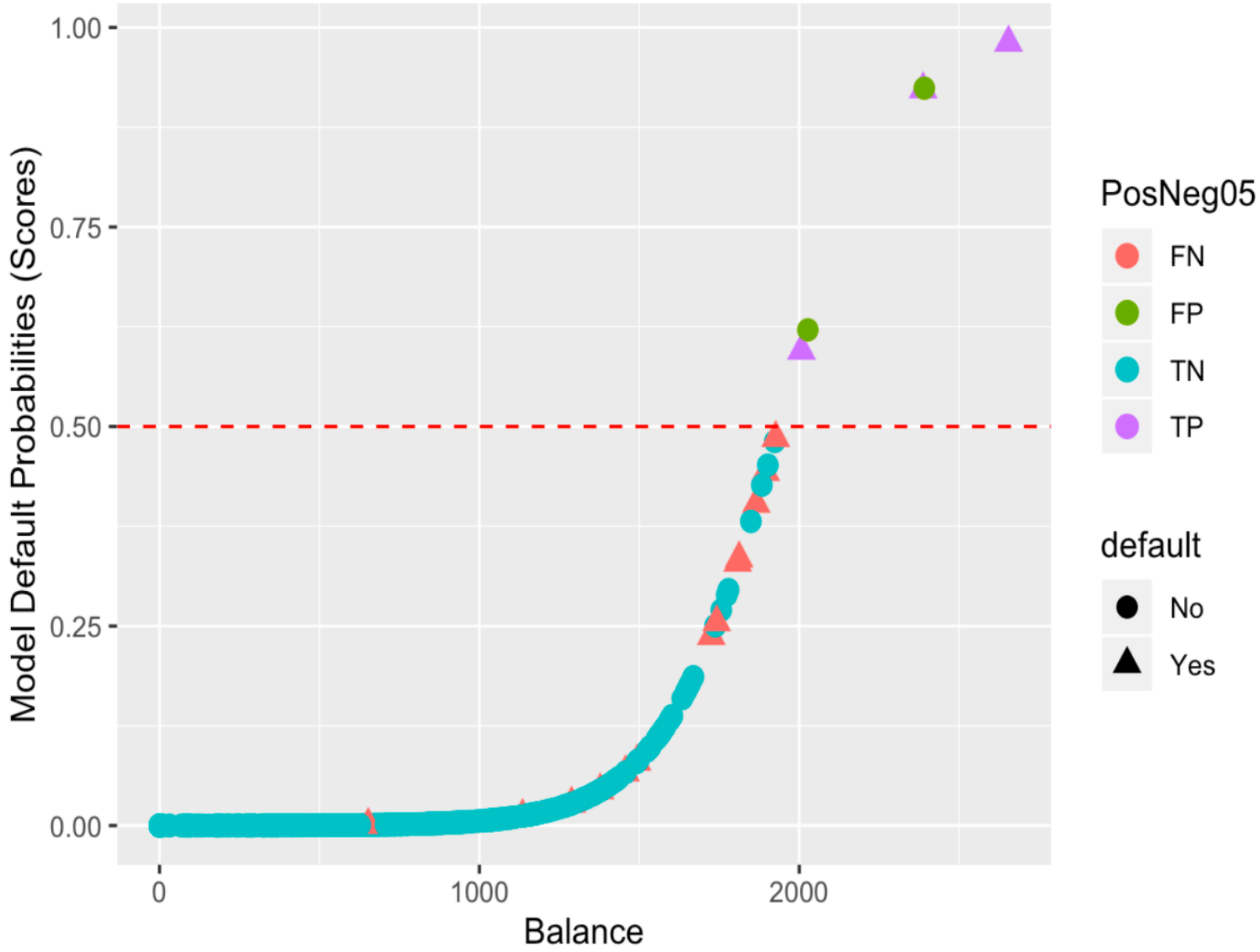
# “Confusion” Matrix

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

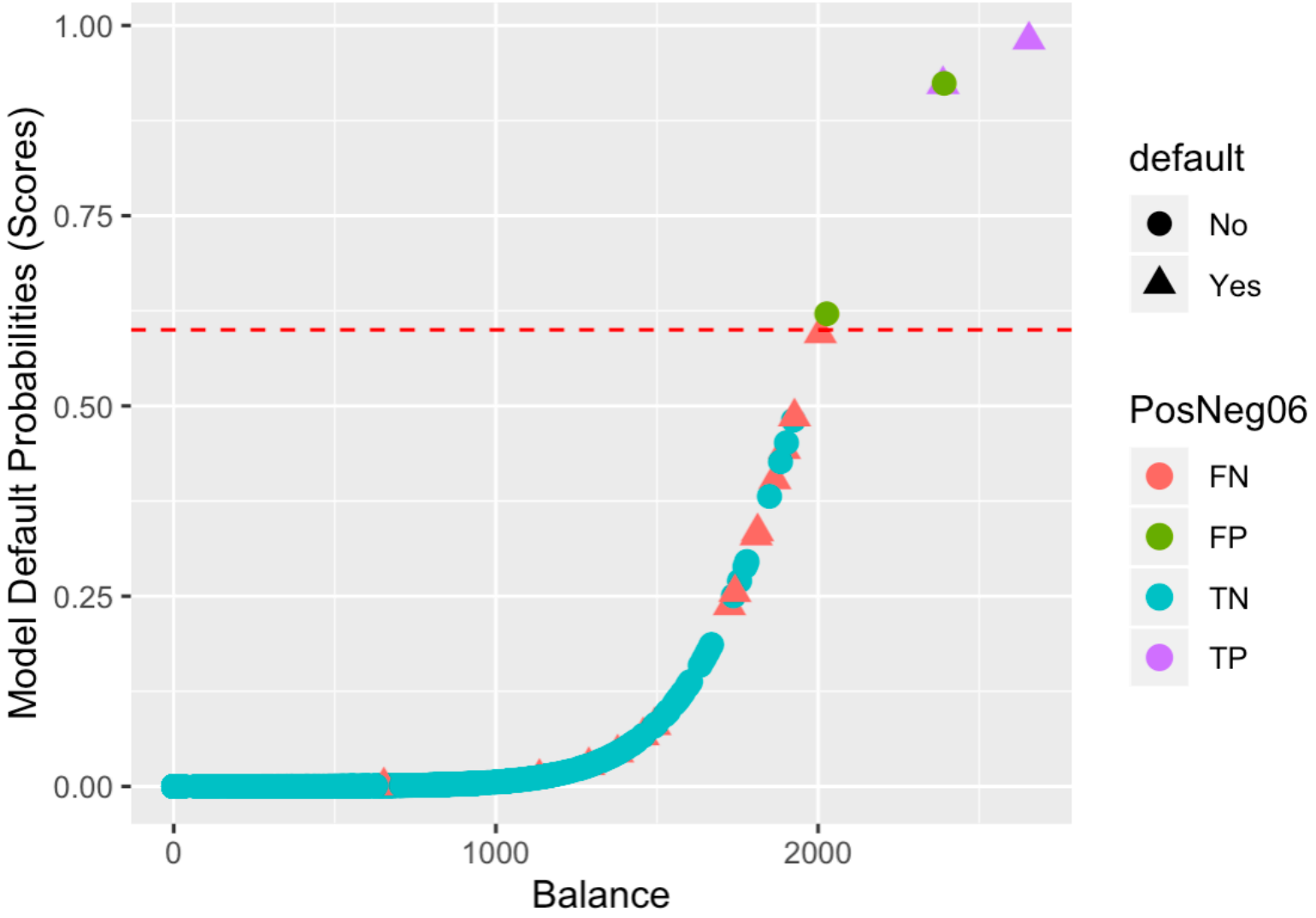
# Assigning Class=Default to $\hat{p} > 0.5$



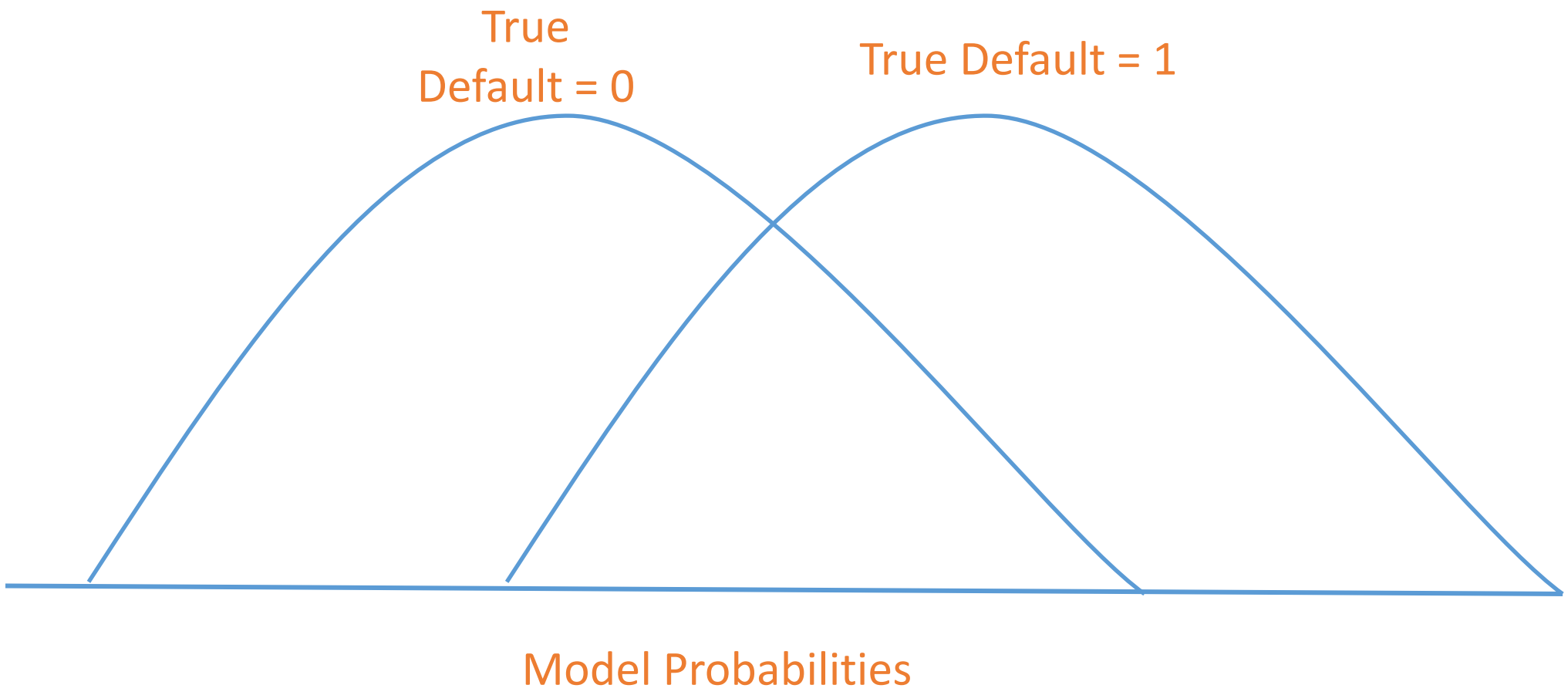
# Assigning Class=Default to $\hat{p} > 0.5$



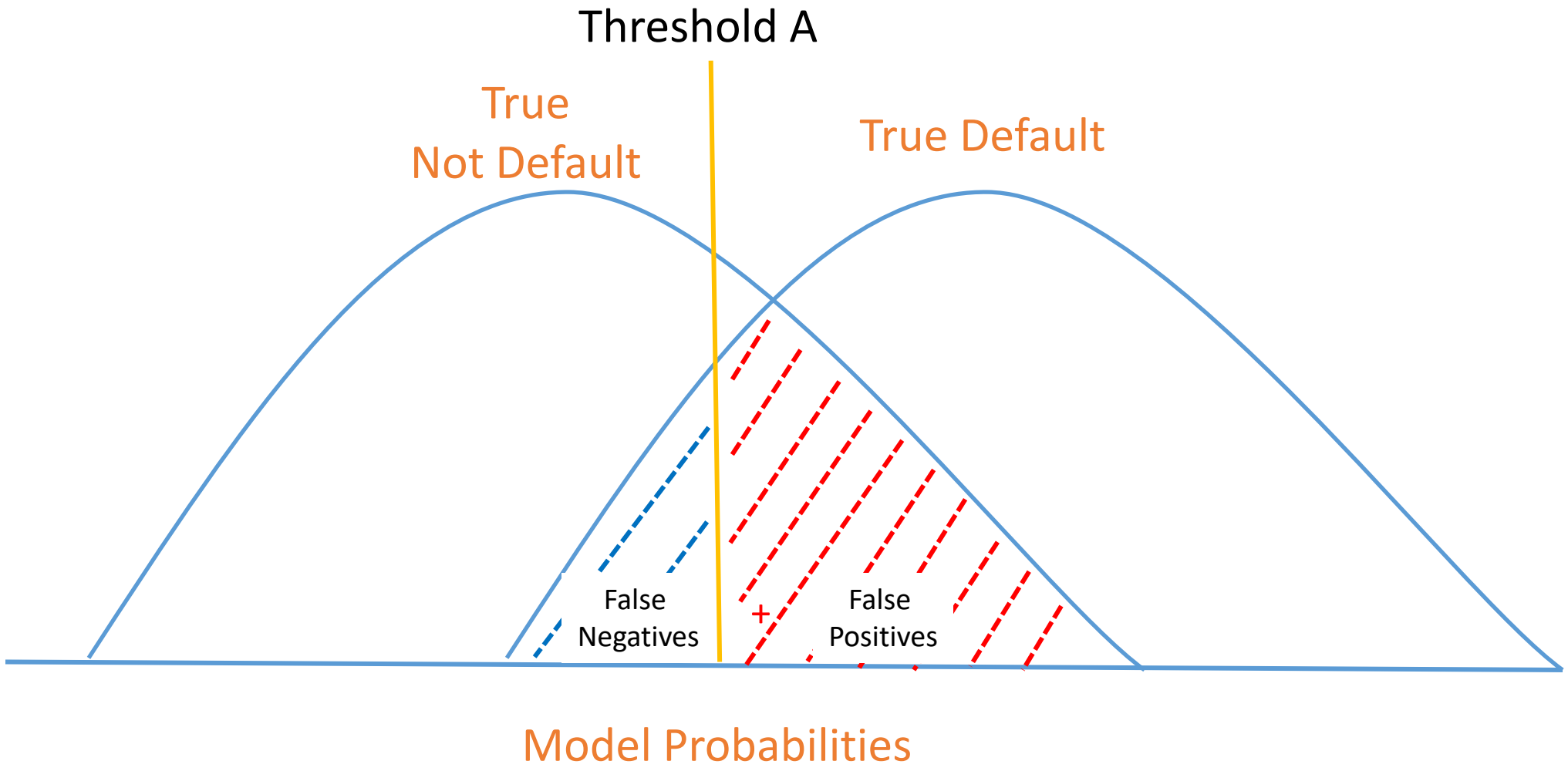
# Assigning Class=Default to $\hat{p} > 0.5$



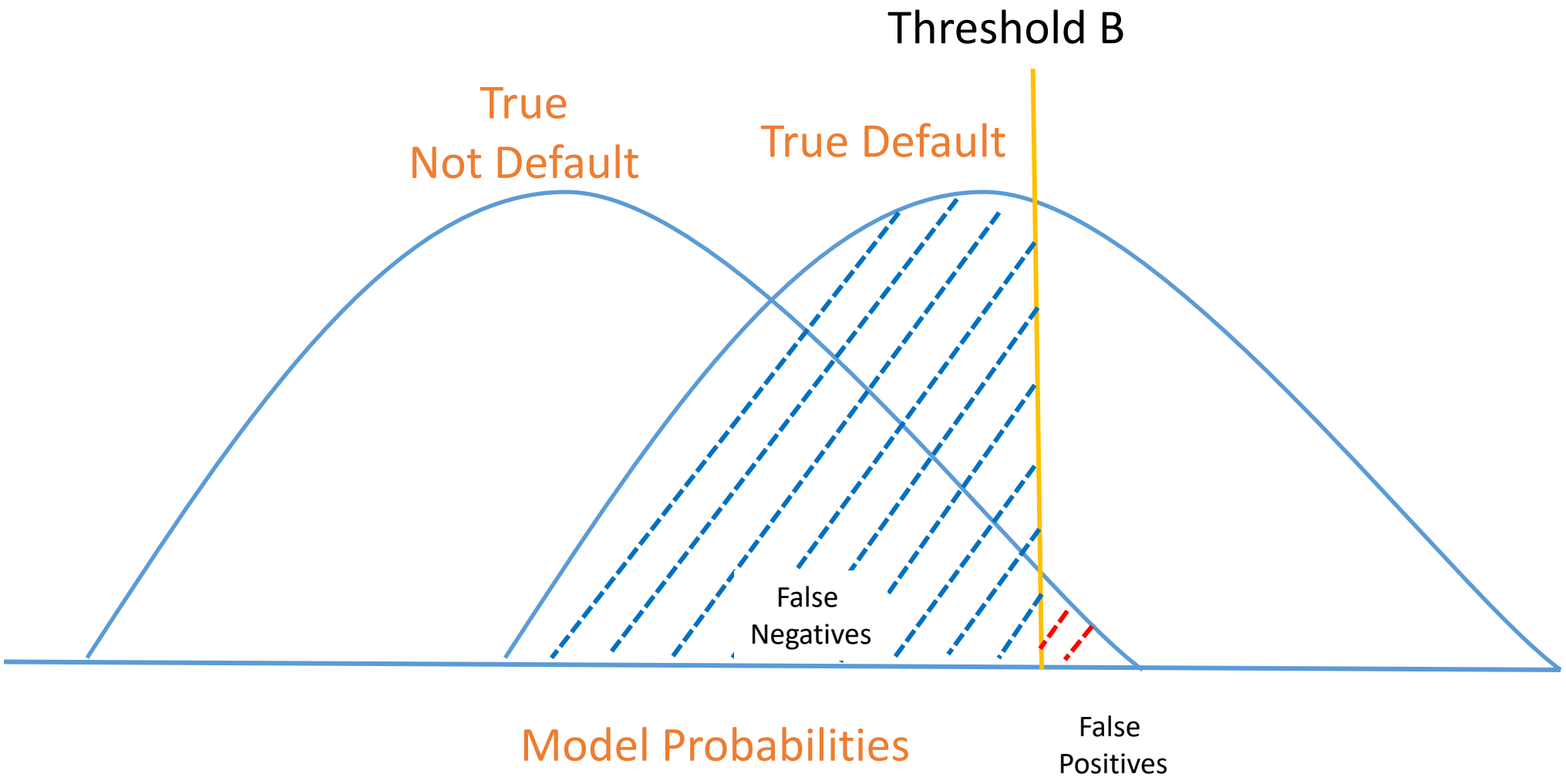
# Choosing Probability Cutoff to Assign Class



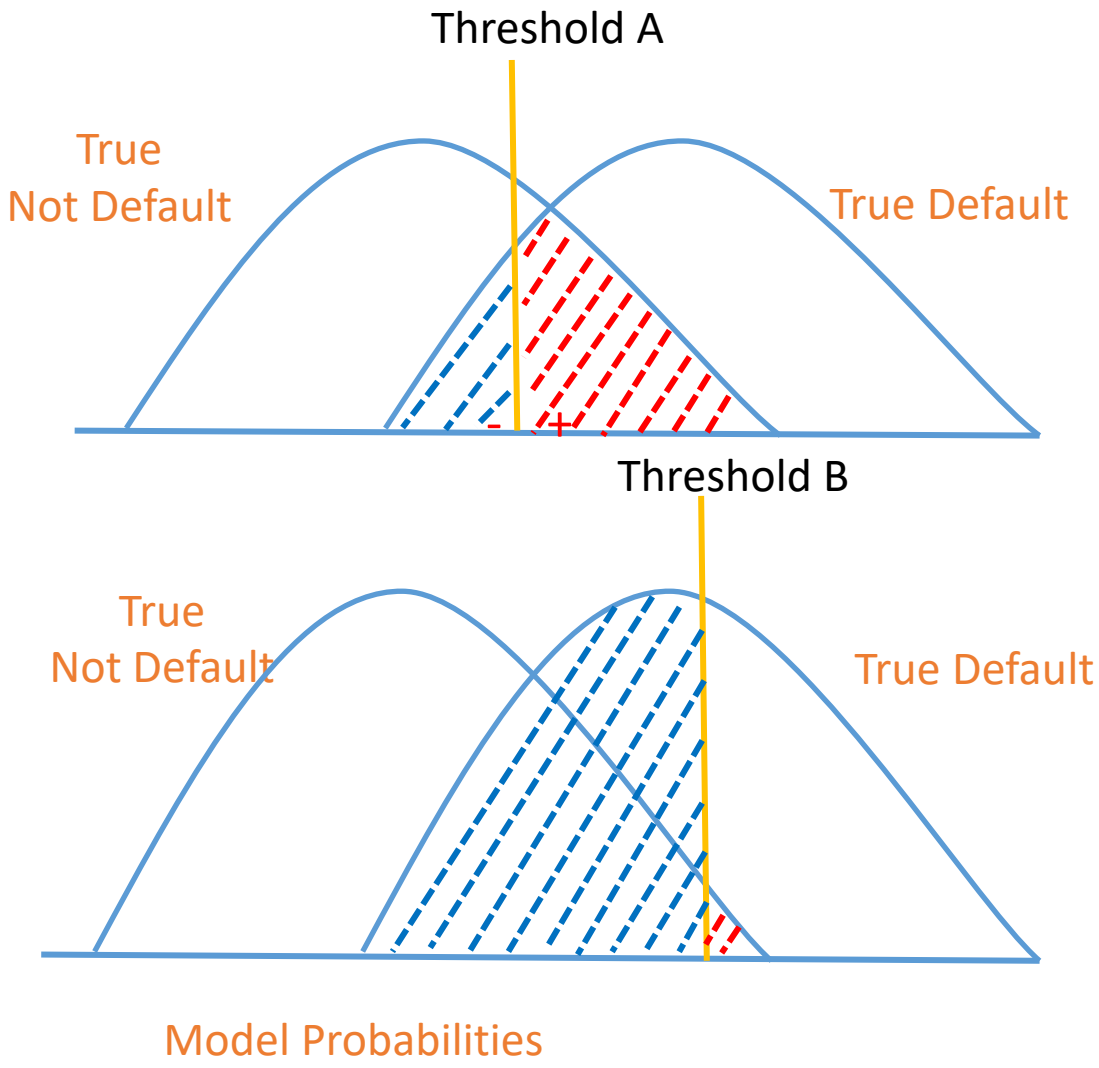
# Threshold A: Moderate Threshold



# Threshold B: Higher Threshold



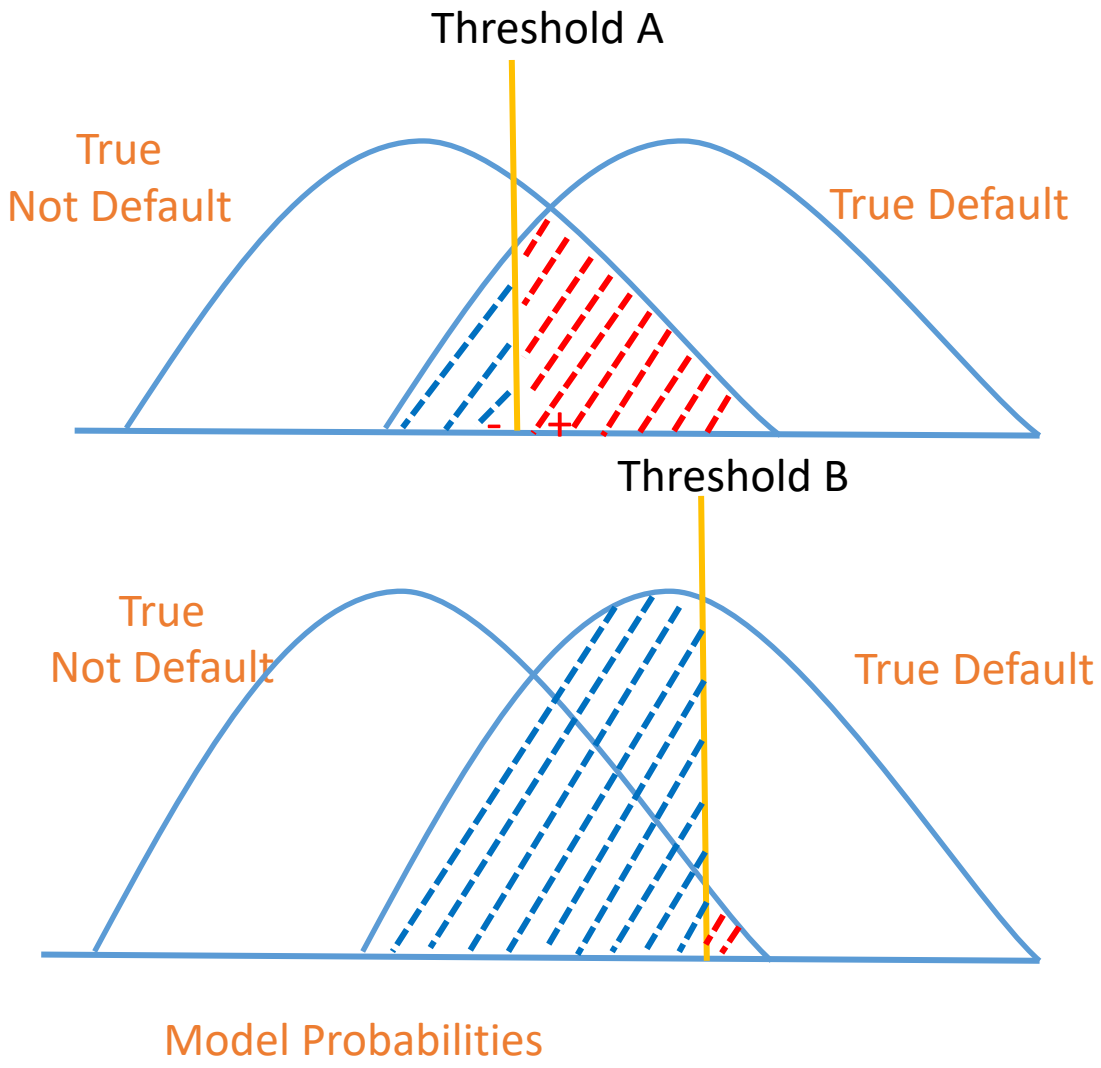
# Comparing cutoffs:




Many false positives

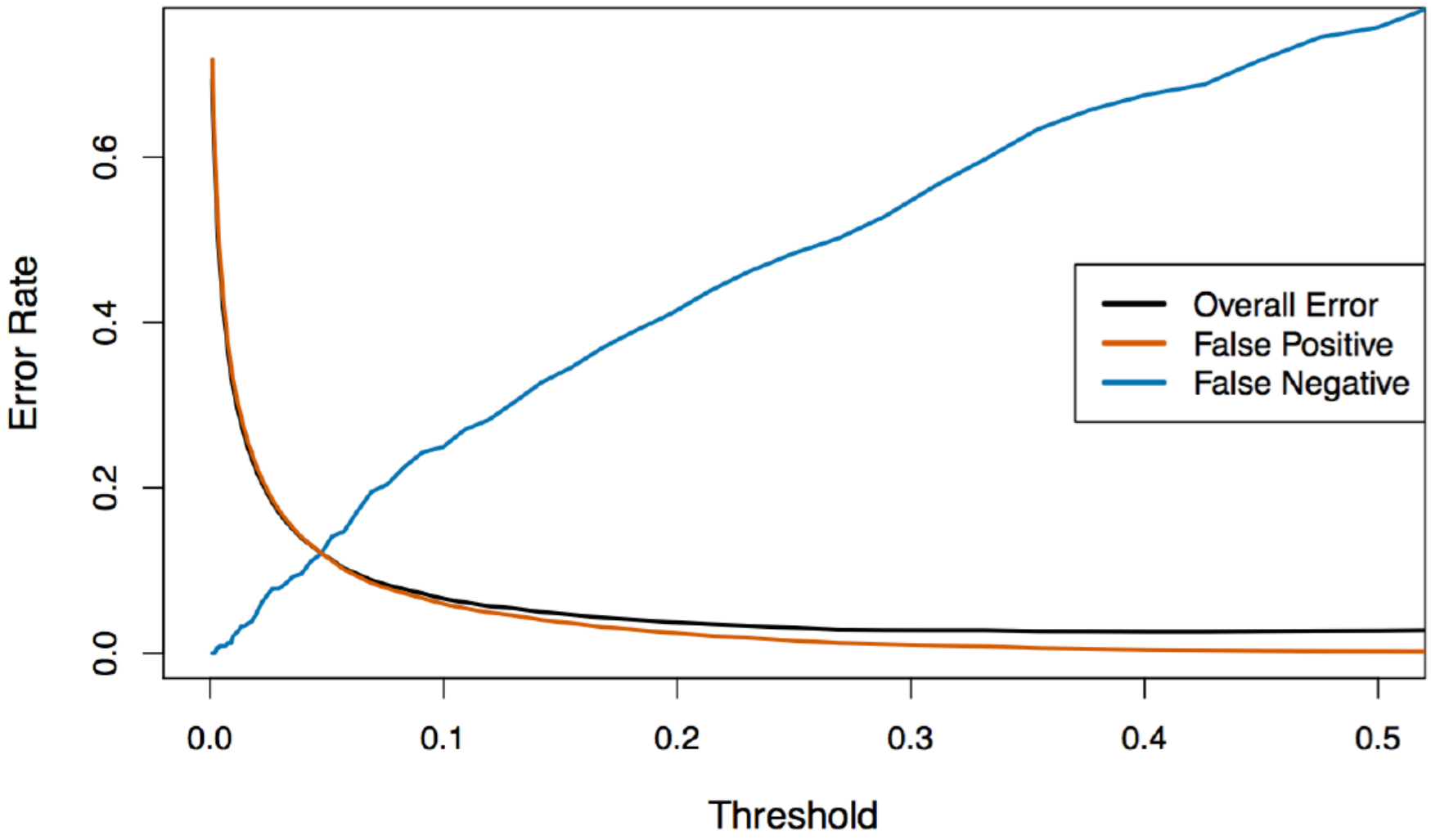
Few false positives

# 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.
- Some common choices
  - Maximize Accuracy (equal weighting of FPs and FNs)
  - Threshold p_hat > 0.5

# Threshold choice affects TP and TNs



# Accuracy and mis-classification rate

| | | True default status | | |
|---|---|---|---|---|
| | | **No** | **Yes** | |
| **Predicted default status** | **No** | TN = 9,627 | FN = 228 | $N^* = 9855$ |
| | **Yes** | FP = 40 | TP = 105 | $P^* = 145$ |
| | | N = 9,667 | P = 333 | |

- Accuracy *"How often is my classifier correct?"*
  - (TP+TN)/Total = (9,627 + 105) / 10,000 = 97.3%
- Mis-classification rate "*How often is the classifier wrong?*"
  - (FP + FN)/Total = (40 + 228) / 10,000 = 2.68%
- No Information Rate "*Everyone is the majority class*"
  - (FP + FN)/Total = (40 + 228) / 10,000 = 2.68%

# Sensitivity and Specificity

| | | True default status | | |
|---|---|---|---|---|
| | | **No** | **Yes** | |
| **Predicted default status** | **No** | TN = 9,627 | FN = 228 | $N^* = 9855$ |
| | **Yes** | FP = 40 | TP = 105 | $P^* = 145$ |
| | | N = 9,667 | P = 333 | |

- **Sensitivity:** True positive rate (aka 1 – power or recall)
  - TP/P = 105 / 333 = 31.5%
- **Specificity:** True negative rate
  - TN/N = 9627 / 9667= 99.5%

- **False positive rate** (aka Type I error, 1 - Specificity)
  - FP/N =

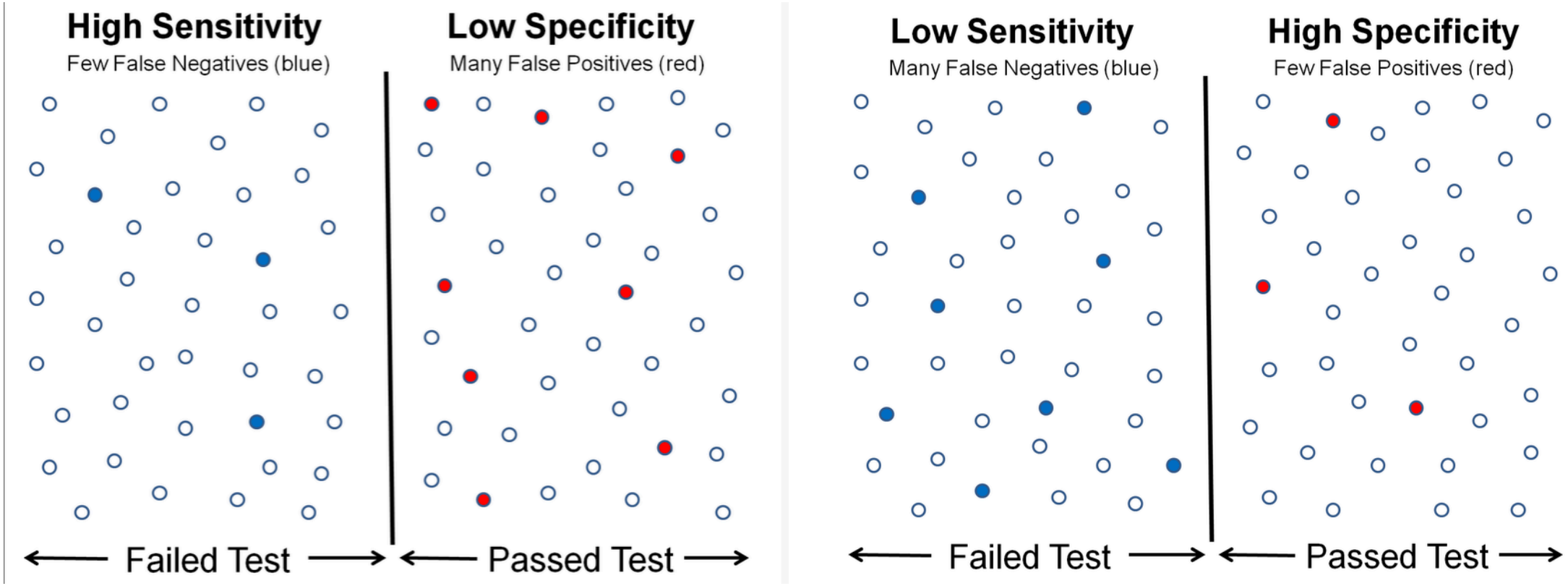
# Confusion Matrices in R

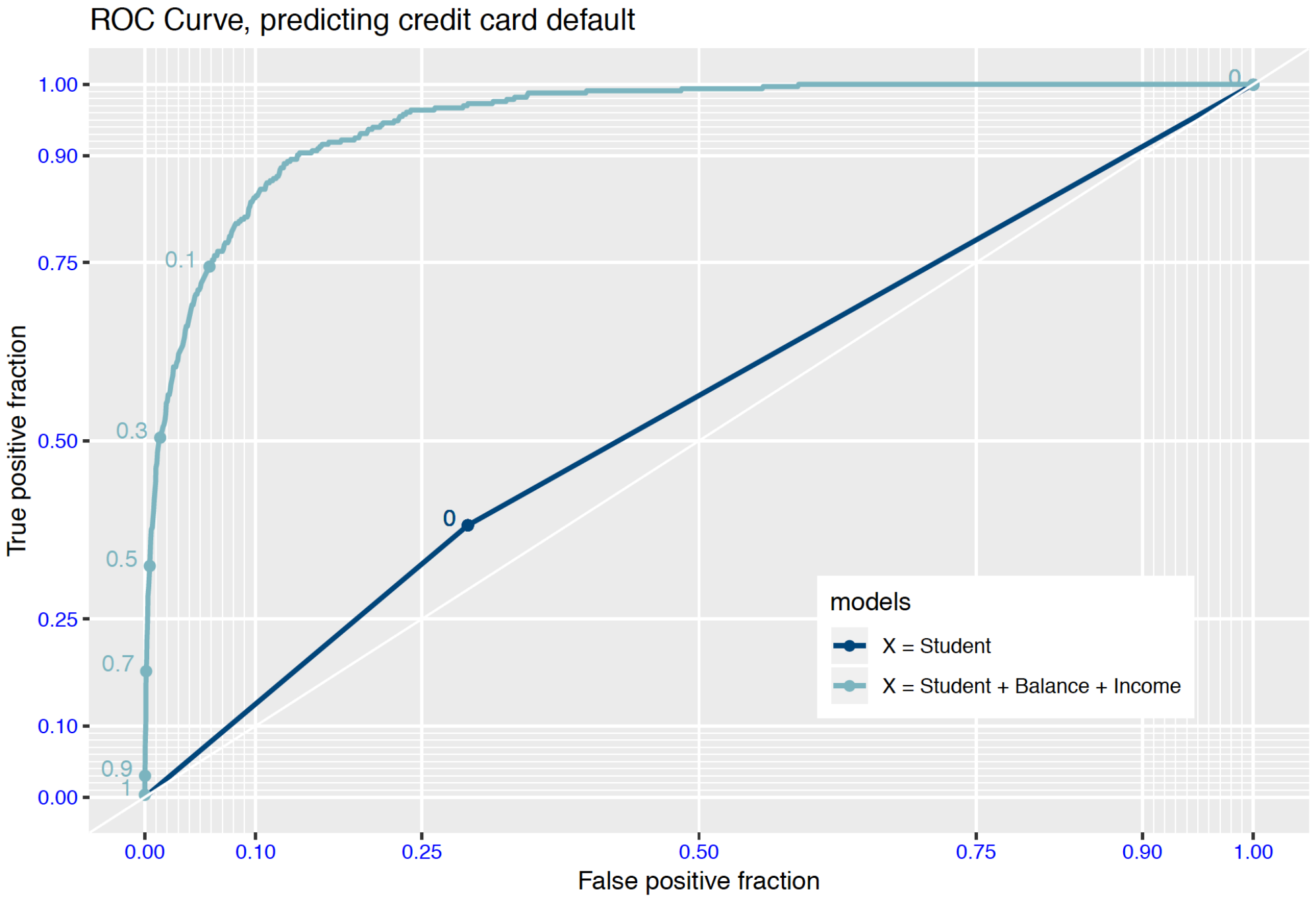
# F1 and Kappa Statistic

| | | True default status | | |
|---|---|---|---|---|
| | | **No** | **Yes** | |
| **Predicted default status** | **No** | TN = 9,627 | FN = 228 | $N^*$ = 9855 |
| | **Yes** | FP = 40 | TP = 105 | $P^*$= 145 |
| | | N = 9,667 | P = 333 | |

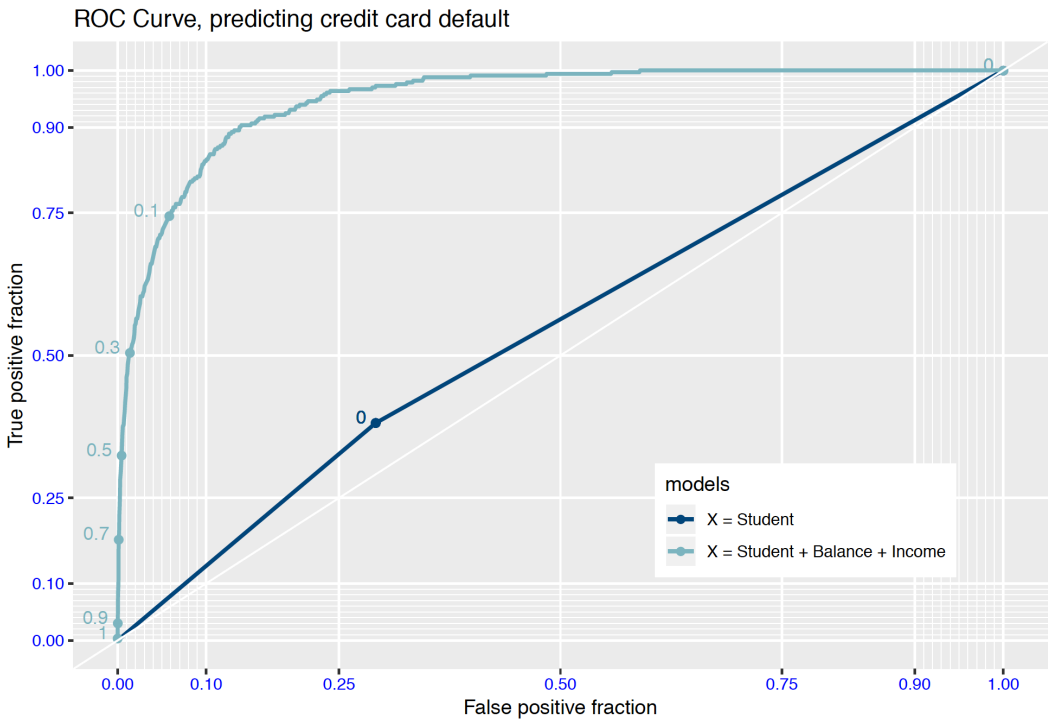
- F1 score
  - Harmonic mean of precision and recall
  - $F1 = 2\frac{precision * recall}{precision + recall}$
- Kappa statistic:
  - Takes into account accuracy that would have been generated by random chance
  - Kappa $\frac{O-E}{1-E} = \frac{observed\ accuracy - expected\ accuracy}{1 - expected\ accuracy}$
  - Higher kappa – better than random chance

# Sensitivity and Specificity



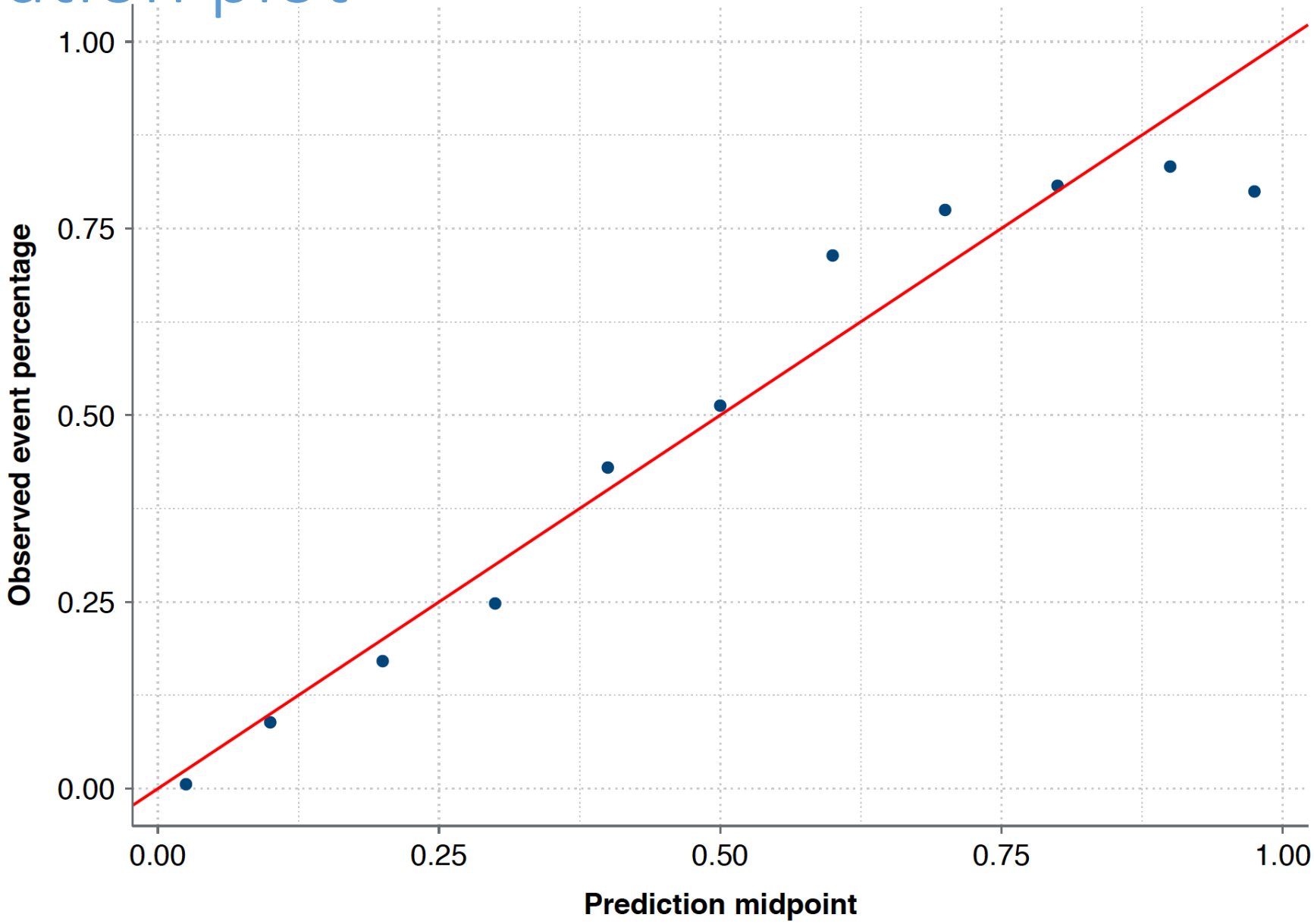

ROC Curve, predicting credit card default
True positive fraction
False positive fraction
1.00
0.90
0.75
0.50
0.25
0.10
0.00
0.00
0.10
0.25
0.50
0.75
0.90
1.00
0
0.1
0.3
0.5
0.7
0.9
1
0
models
X = Student
X = Student + Balance + Income

# AUC = Area Under Curve




- AUC = Area Under Curve
-

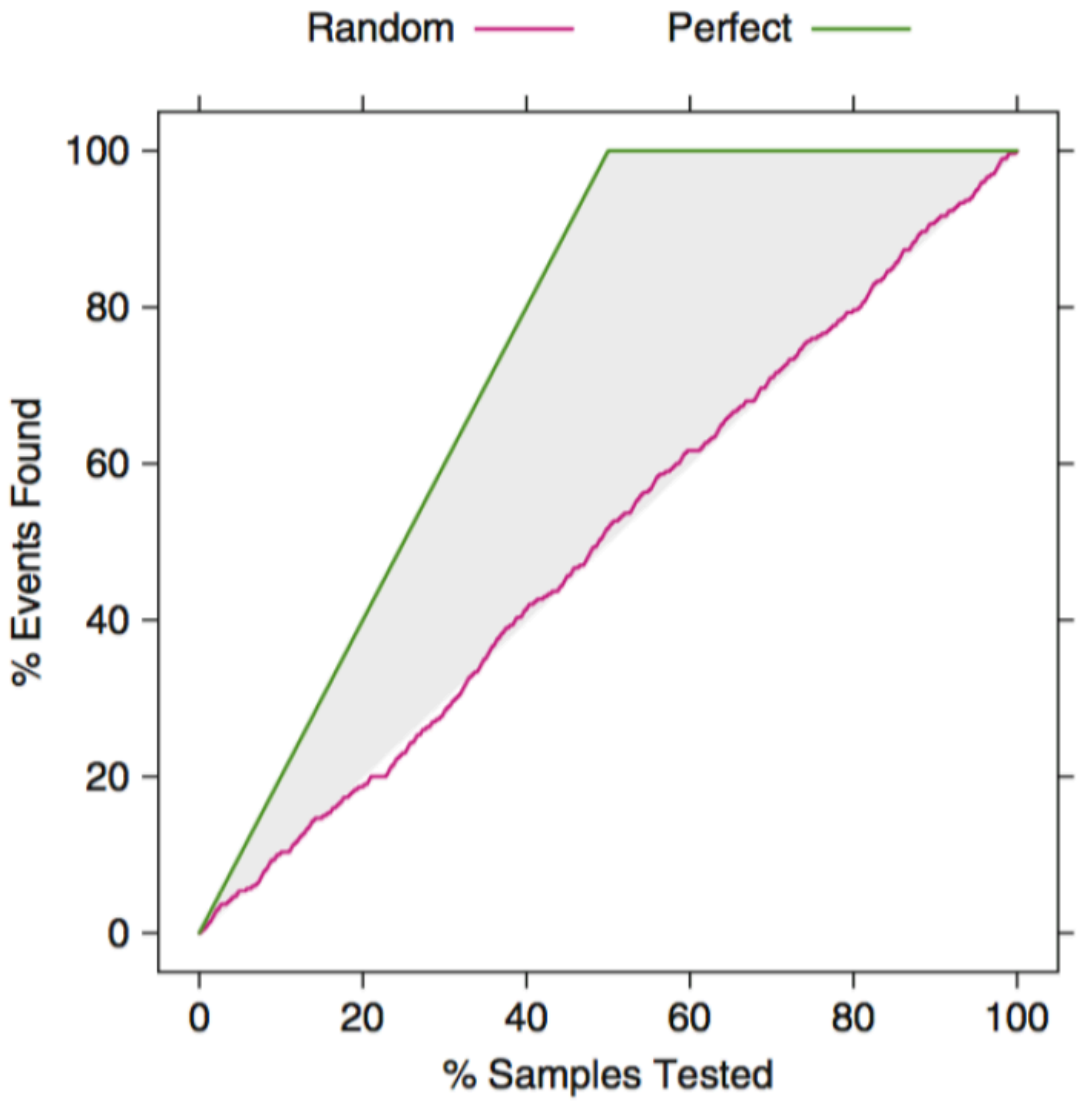
# Calibration plot



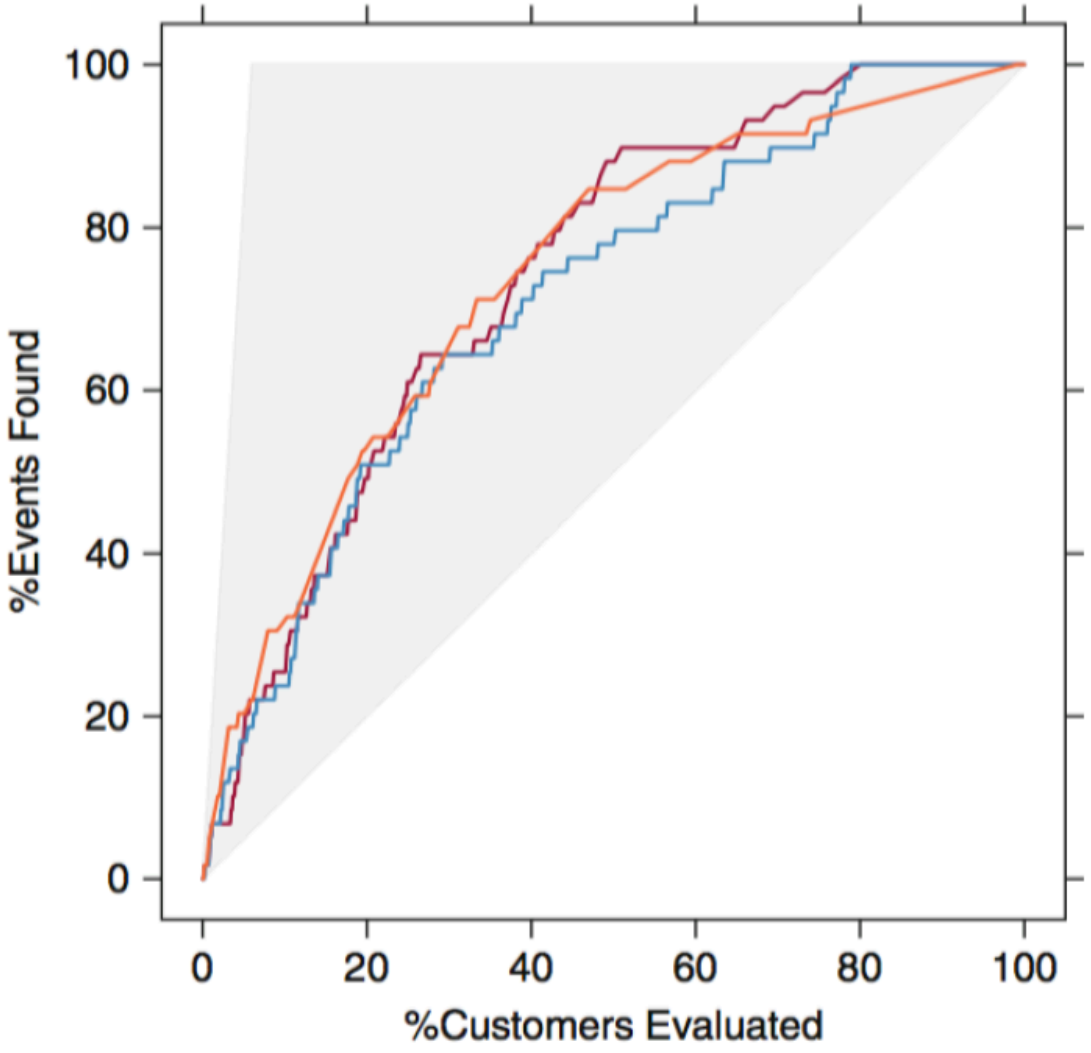
# Lift Chart

- To construct a lift chart, use any method to get predicted probabilities
- Then order observations by probabilities
- For each predicted, probability, count whether event actually occurred

# Lift Chart


Random
Perfect
100
80
60
40
20
0
% Events Found
0
20
40
60
80
100
% Samples Tested

# Lift Chart

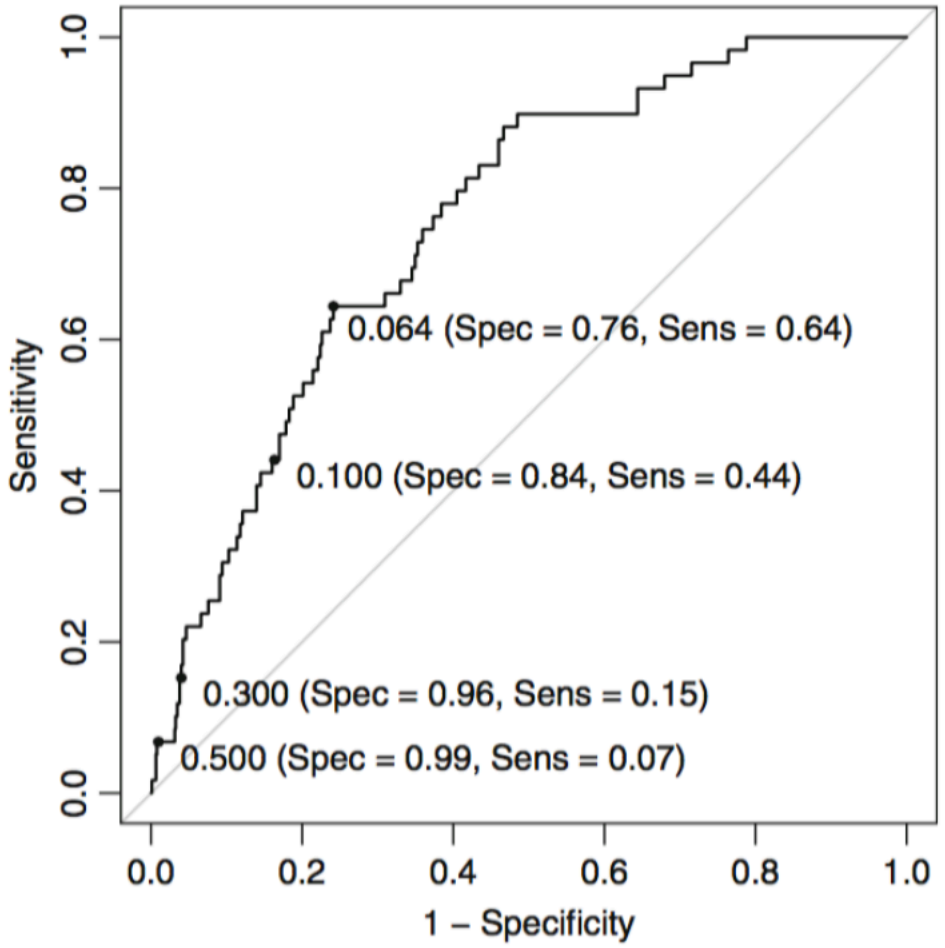

100
80
60
40
20
0
%Events Found
0
20
40
60
80
100
%Customers Evaluated

# Severe class imbalance

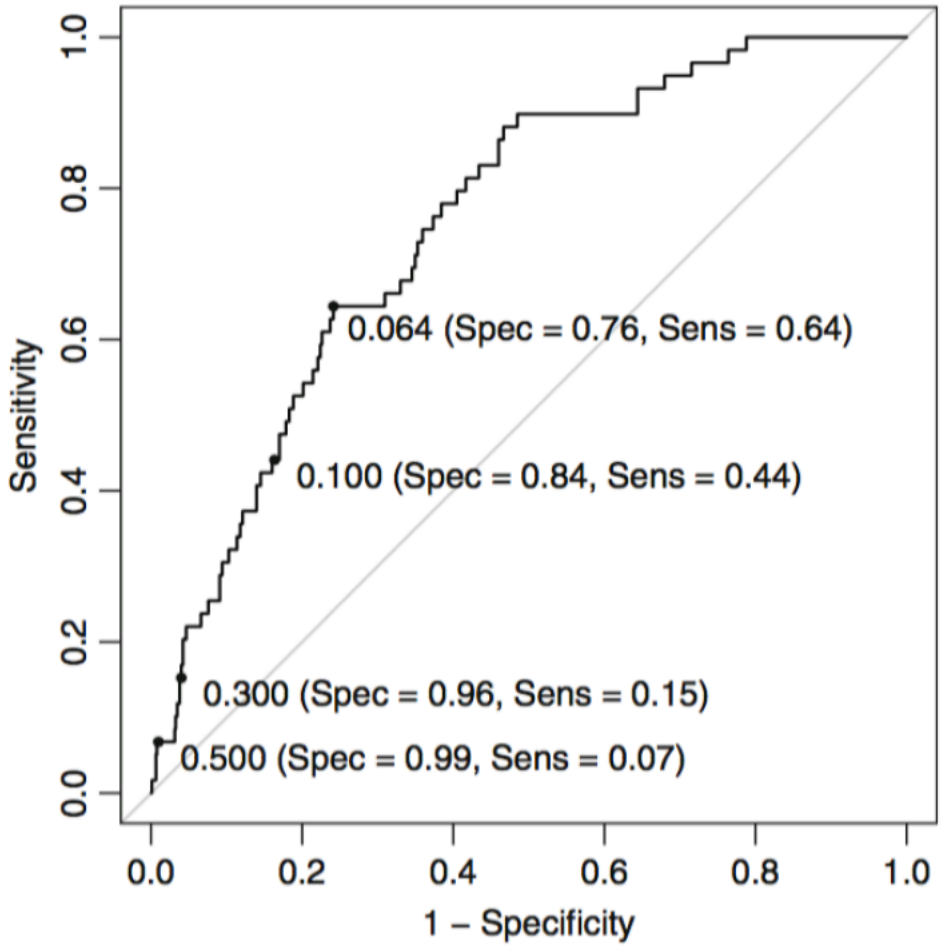
Table 16.1: Results for three predictive models using the evaluation set

| Model | Accuracy | Kappa | Sensitivity | Specificity | ROC AUC |
|---|---|---|---|---|---|
| Random forest | 93.5 | 0.091 | 6.78 | 99.0 | 0.757 |
| FDA (MARS) | 93.8 | 0.024 | 1.69 | 99.7 | 0.754 |
| Logistic regression | 93.9 | 0.027 | 1.69 | 99.8 | 0.727 |

# Remedy 1: alternate score cutoff



# Remedy 1: alternate score cutoff



# Remedies 2,3&4 under-sample majority class, oversample minority class, SMOTE interpolate minority class

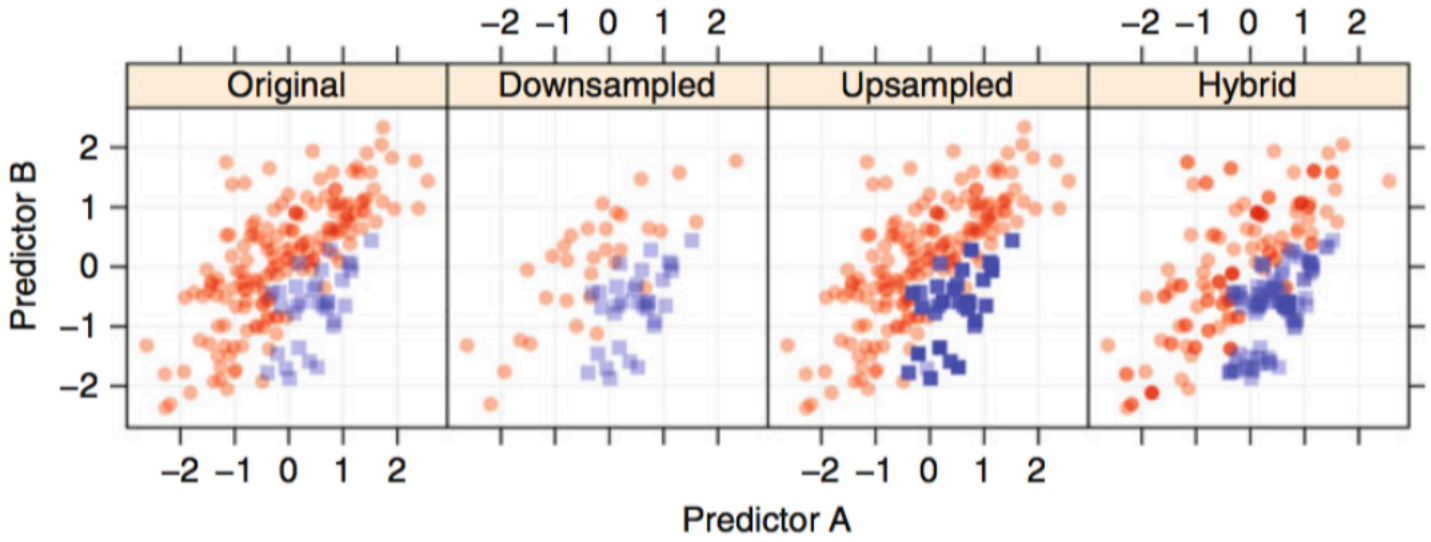



Fig. 16.3: *From left to right*: The original simulated data set and realizations of a down-sampled version, an up-sampled version, and sampling using SMOTE where the cases are sampled and/or imputed

# Remedy 5: Use calibration plot to change model

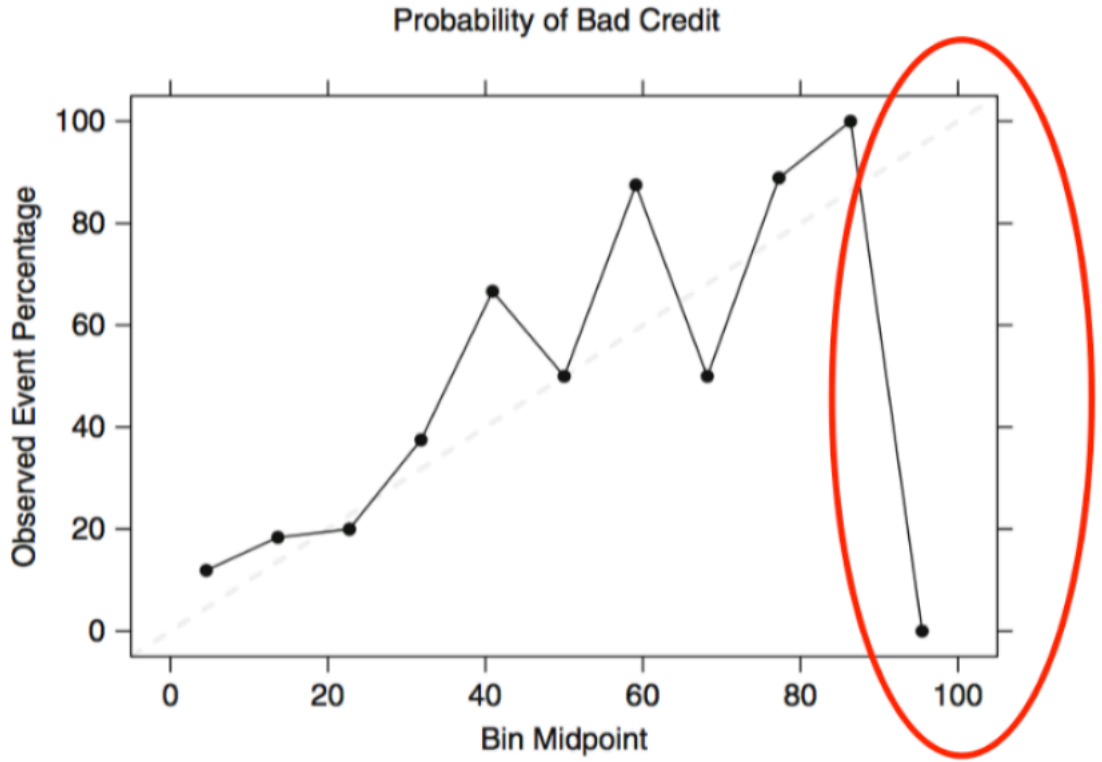



Fig. 11.3: *Top*: Histograms for a set of probabilities associated with bad credit. The two panels split the customers by their true class. *Bottom*: A calibration plot for these probabilities