



STOR 320 Modeling V

Lecture 19

Yao Li

Department of Statistics and Operations Research

UNC Chapel Hill

Introduction

- Now We Consider
 - Categorical Response (Outcome) Variable
 - Numerical/Categorical Explanatory Variables
- Focus is on Classification
- Read Chapter 4 in ISLR

Introduction

- Basic Case: Binary Response
 - Variable Has Two Possible Outcomes
 - Typically, Yes or No Responses to a Question
 - Example
 - $Y =$ Who Will Win the 2024 Presidential Election?
 - $Y =$ Did You Pass Your STOR 320 Class?
 - $Y =$ What Factors Influence the Admission into Graduate School?

Scenario

- Question: Are Students Who Get Good Grades Likely to be Admitted to Graduate School?
 - Y = Would the Student be Admitted to a Graduate School?
 - X = College GPA
- Why is Linear Regression Inappropriate?

$$P(\text{Admission}|X) = \beta_0 + \beta_1 X$$

Problem Setting

- Bernoulli Random Variable

$$Y = \begin{cases} 1 & \text{if Yes} \\ 0 & \text{if No} \end{cases}$$
$$p = E(Y) = P(Y = 1)$$

- Sample n Students

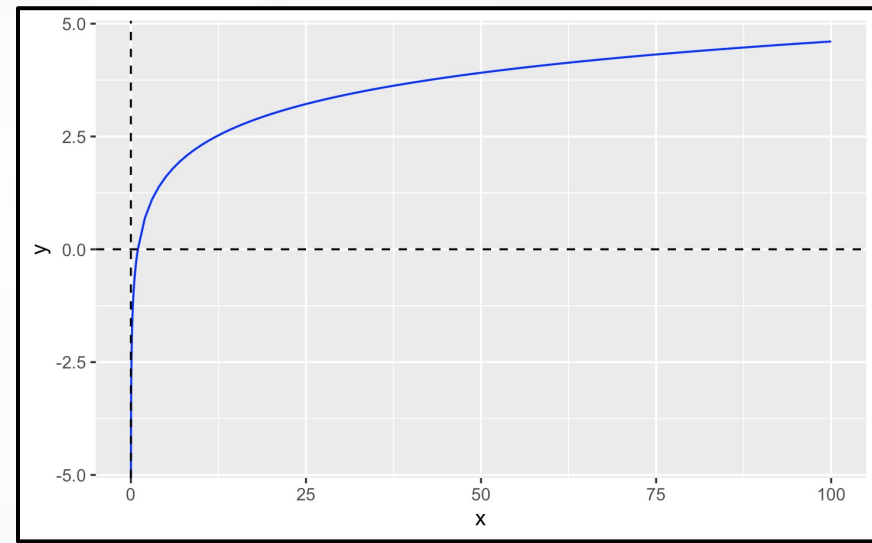
$$Y' = \sum Y_i \sim \text{Binomial}(n, p)$$

$$\hat{p} = \frac{\sum y_i}{n}$$

Estimated Probability that a Student Would
be Admitted to a Graduate School

- Analyze the Effect of X on p : $p = E(Y|X) \neq \beta_0 + \beta_1 X$

Logit Link



- Modeling the Mean

- Logit Link Function

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



Odds of Admission

- Understanding Odds
 - Odds of Admission = 1
 - Odds of Admission < 1
 - Odds of Admission > 1

Model Construction

- Solving for $\frac{p}{1-p}$

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

$$\frac{p}{1-p} = e^{\beta_0 + \beta_1 X} \quad \longrightarrow$$

Odds of Admission Given
the Student's GPA

- Solving for p

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

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

$$p = \frac{e^{\beta_0 + \beta_1 X}}{1 + e^{\beta_0 + \beta_1 X}} \quad \longrightarrow$$

Probability of Admission Given the
Student's GPA

Logistic Regression for Classification

- Recall: $Y = \begin{cases} 1 & \text{if Yes} \\ 0 & \text{if No} \end{cases}$

- After Getting Data, We Estimate

- $\hat{\beta}_0$

- $\hat{\beta}_1$

- $\hat{p} = \frac{e^{\hat{\beta}_0 + \hat{\beta}_1 X}}{1 + e^{\hat{\beta}_0 + \hat{\beta}_1 X}} \rightarrow$

Estimated Probability of Admission Given the Student's GPA

- Two Scenarios

- $\hat{p} < 0.5 \rightarrow \hat{Y} = 0$

- $\hat{p} > 0.5 \rightarrow \hat{Y} = 1$

Evaluating the LR Model

- Two Methods
 - Leave Out Data Intentionally
 - Use Cross-Validation
- Positives and Negatives
 - True Positive = Predicted an Admission and the Student Got Admitted
 - False Positive = Predicted an Admission and the Student Didn't Get Admitted
 - False Negative = Predicted a Student Wouldn't be Admitted and They Did Get Admitted
 - True Negative = Predicted a Student Wouldn't be Admitted and They Didn't Get Admitted

Confusion Matrix

- Confusion Matrix

	Predicted	
Actual	<i>Will be Admitted</i>	<i>Won't be Admitted</i>
<i>Admission</i>	n_{11}	n_{12}
<i>Isn't Admitted</i>	n_{21}	n_{22}

- Sensitivity:

$$n_{11}/(n_{11} + n_{12})$$

- Specificity:

$$n_{22}/(n_{21} + n_{22})$$

- False Positive Rate:

$$n_{21}/(n_{21} + n_{22})$$

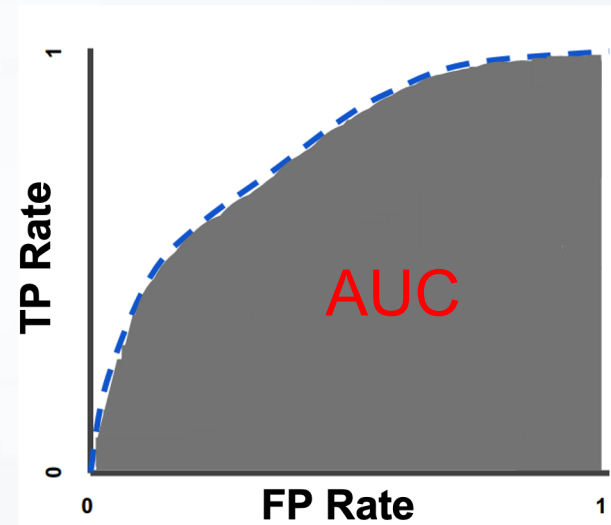
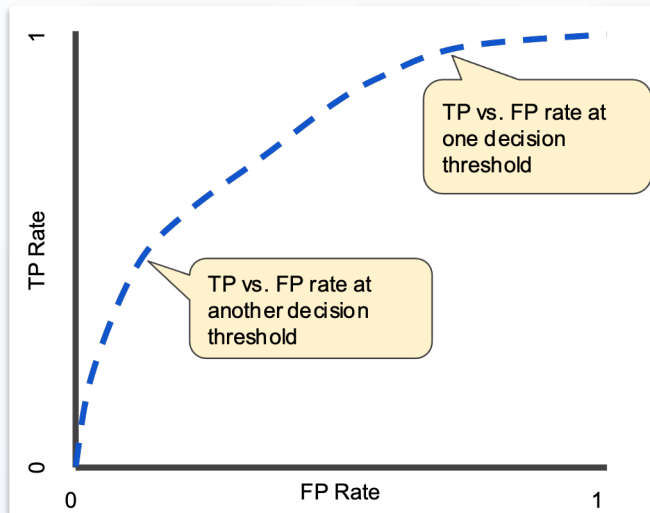
- False Negative Rate:

$$n_{12}/(n_{11} + n_{12})$$

Area Under ROC Curve

	Predicted	
Actual	Will be Admitted	Won't be Admitted
Admission	n_{11}	n_{12}
Isn't Admitted	n_{21}	n_{22}

- True Positive Rate (Sensitivity): $n_{11}/(n_{11} + n_{12})$
- False Positive Rate: $n_{21}/(n_{21} + n_{22})$



Titanic: Data

- Titanic Survival Data

```
> library(titanic)
```

- Response Variable

$$Y = \begin{cases} 1 & \text{if Survived} \\ 0 & \text{if Did Not Survive} \end{cases}$$

- Explanatory Variables
 - Passenger Class
 - Sex
 - Age
 - Siblings/Spouses Aboard
 - Parents/Children Aboard
 - Passenger Fare
 - Port of Embarkation

Titanic: Data

- Titanic Survival Data (Continued)
 - Selecting Variables of Interest

```
> TRAIN=titanic_train[,c(2,3,5,6,7,8,10,12)]  
> TEST=titanic_test[,c(2,4,5,6,7,9,11)]
```

- Glimpse of Data

```
glimpse(TRAIN)
```

```
## Observations: 891  
## Variables: 8  
## $ Survived <int> 0, 1, 1, 1, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 1, 0, 1, ...  
## $ Pclass <int> 3, 1, 3, 1, 3, 3, 1, 3, 3  
## $ Sex <chr> "male", "female", "female"  
## $ Age <dbl> 22, 38, 26, 35, 35, NA, 5  
## $ SibSp <int> 1, 1, 0, 1, 0, 0, 0, 3, 0  
## $ Parch <int> 0, 0, 0, 0, 0, 0, 0, 1, 2  
## $ Fare <dbl> 7.2500, 71.2833, 7.9250,  
## $ Embarked <chr> "S", "C", "S", "S", "S",
```

```
glimpse(TEST)
```

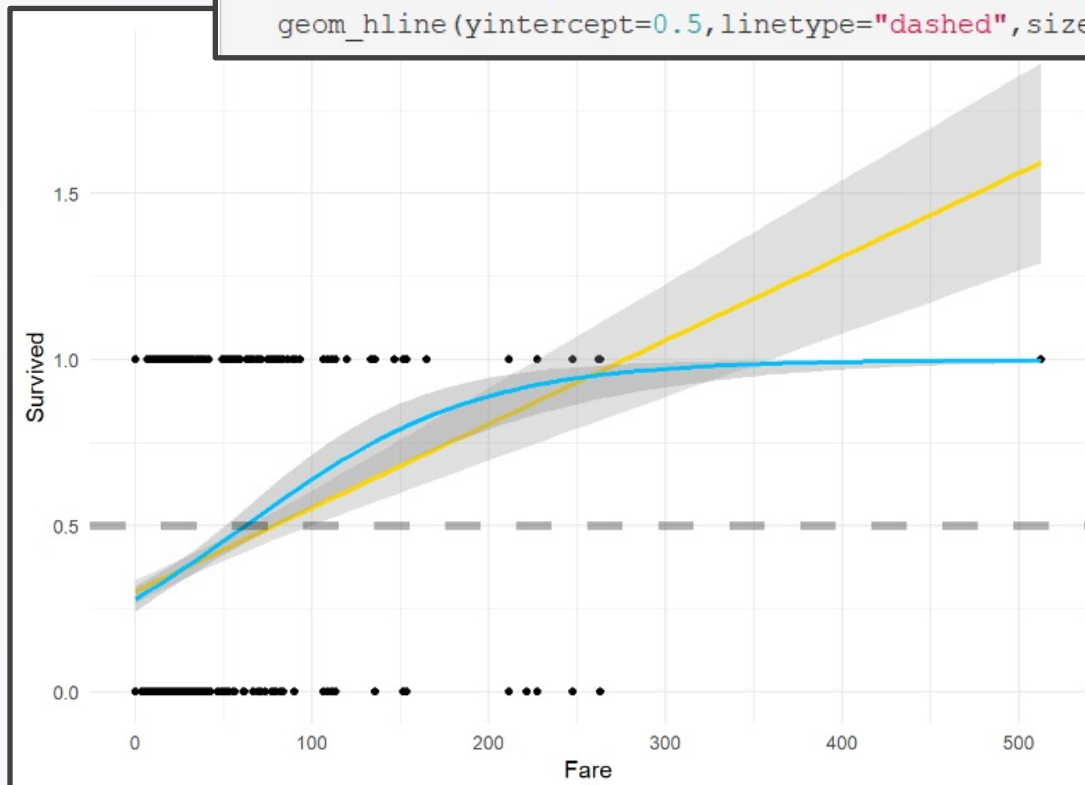
Problem?

```
## Observations: 418  
## Variables: 7  
## $ Pclass <int> 3, 3, 2, 3, 3, 3, 3, 2, 3, 3, 3, 1, 1, 2, 1, 2, 2, 3, ...  
## $ Sex <chr> "male", "female", "male", "male", "female", "male", "...  
## $ Age <dbl> 34.5, 47.0, 62.0, 27.0, 22.0, 14.0, 30.0, 26.0, 18.0, ...  
## $ SibSp <int> 0, 1, 0, 0, 1, 0, 0, 1, 0, 2, 0, 0, 1, 1, 1, 1, 0, 0, ...  
## $ Parch <int> 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...  
## $ Fare <dbl> 7.8292, 7.0000, 9.6875, 8.6625, 12.2875, 9.2250, 7.62...  
## $ Embarked <chr> "Q", "S", "Q", "S", "S", "S", "Q", "S", "C", "S", "S"...
```

Visualization: Survival vs. Fare

- Visualizing the Data

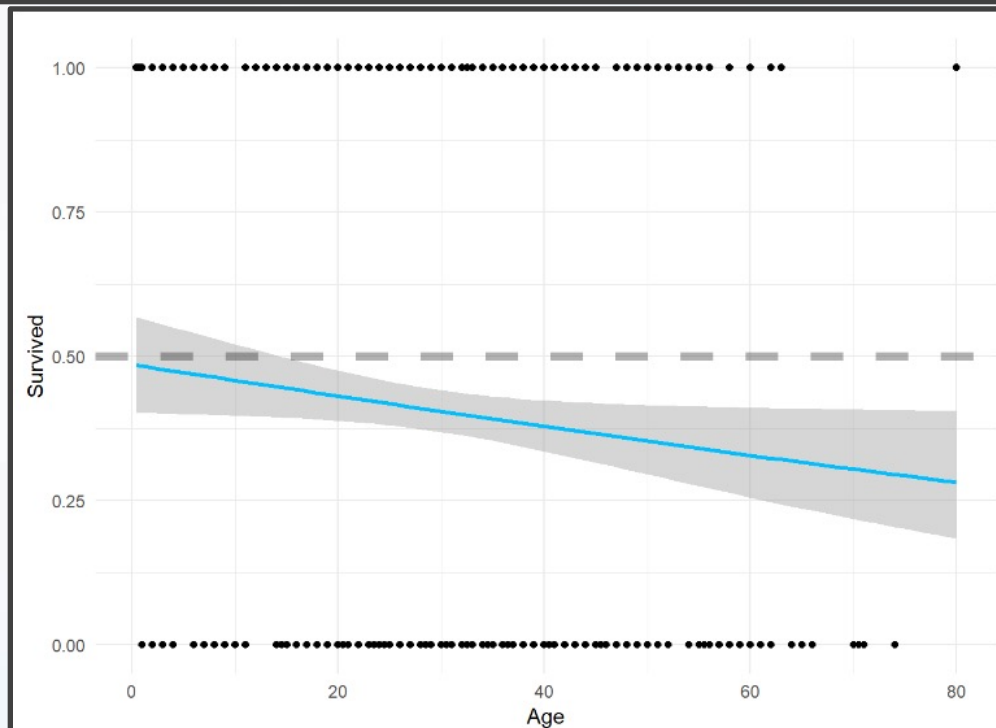
```
ggplot(TRAIN) + geom_point(aes(x=Fare,y=Survived)) + theme_minimal() +  
  geom_smooth(aes(x=Fare,y=Survived),method="lm",alpha=0.3,color="gold") +  
  geom_smooth(aes(x=Fare,y=Survived),method="glm",  
              method.args=list(family="binomial"),color="deepskyblue1") +  
  geom_hline(yintercept=0.5,linetype="dashed",size=2,alpha=0.3)
```



Visualization: Survival vs. Age

- Visualizing the Data (Continued)

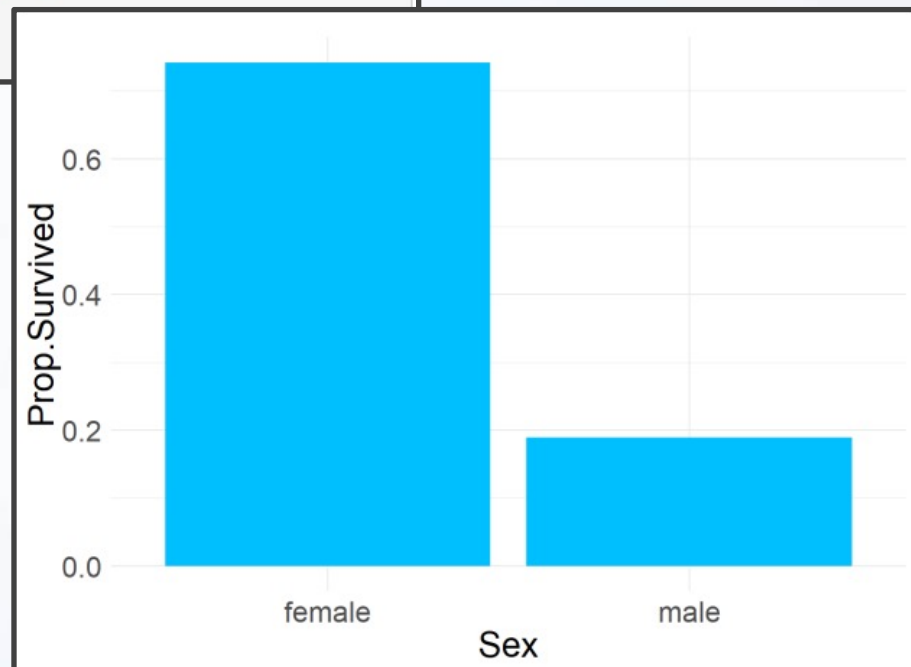
```
ggplot(TRAIN) + geom_point(aes(x=Age,y=Survived)) + theme_minimal() +  
  geom_smooth(aes(x=Age,y=Survived),method="glm",  
              method.args=list(family="binomial"),color="deepskyblue1") +  
  geom_hline(yintercept=0.5,linetype="dashed",size=2,alpha=0.3)
```



Visualization: Survival vs. Sex

- Visualizing the Data (Continued)

```
TRAIN %>%  
  mutate(Sex=factor(Sex)) %>%  
  group_by(Sex) %>%  
  summarize(Prop.Survived=mean(Survived)) %>%  
  ggplot() +  
  geom_bar(aes(x=Sex,y=Prop.Survived),  
           stat="Identity",fill="deepskyblue1") +  
  theme_minimal() +  
  theme(text=element_text(size=20))
```



Data Splitting

- Logistic Regression Models
 - Split Training Set Up

```
> set.seed(216)
> sample.in=sample(1:dim(TRAIN)[1],
                  size=floor(0.8*dim(TRAIN)[1]))
> TRAIN.IN=TRAIN[sample.in,
                 c("Survived","Fare","Sex","Age")]
> TRAIN.OUT=TRAIN[-sample.in,
                  c("Survived","Fare","Sex","Age")]
```

- Modeling the Probability of Survival Given the Ticket Fare, the Sex of the Passenger, and the Age of the Passenger

Model 1

- Logistic Regression Models (Cont.)
 - Including 3-Way Interaction

```
logmod1=glm(Survived~.^3,family="binomial",data=TRAIN.IN)  
tidy(logmod1)[,c("term","estimate","p.value")]
```

```
## # A tibble: 8 x 3  
##   term                estimate p.value  
##   <chr>                <dbl>   <dbl>  
## 1 (Intercept)          1.16    0.0254  
## 2 Fare                -0.0156  0.265  
## 3 Sexmale             -1.91    0.00314  
## 4 Age                 -0.0380  0.0636  
## 5 Fare:Sexmale         0.0226  0.148  
## 6 Fare:Age             0.00175 0.00840  
## 7 Sexmale:Age          0.0118  0.623  
## 8 Fare:Sexmale:Age    -0.00169 0.0147
```

Model 2

- Logistic Regression Models (Cont.)
 - Only 2-Way Interactions

```
logmod2=glm(Survived~.*.,family="binomial",data=TRAIN.IN)  
tidy(logmod2)[,c("term","estimate","p.value")]
```

```
## # A tibble: 7 x 3  
##   term          estimate p.value  
##   <chr>          <dbl>   <dbl>  
## 1 (Intercept)    0.311    0.453  
## 2 Fare           0.0161   0.0926  
## 3 Sexmale       -0.849    0.0924  
## 4 Age            0.000682 0.961  
## 5 Fare:Sexmale  -0.0151   0.0681  
## 6 Fare:Age       0.000253 0.229  
## 7 Sexmale:Age   -0.0343   0.0333
```

Model 3

- Logistic Regression Models (Cont.)
 - No Way Interactions

```
logmod3=glm(Survived~.,family="binomial",data=TRAIN.IN)  
tidy(logmod3)[,c("term","estimate","p.value")]
```

```
## # A tibble: 4 x 3  
##   term          estimate p.value  
##   <chr>          <dbl>   <dbl>  
## 1 (Intercept)    0.901 6.84e- 4  
## 2 Fare           0.0125 1.68e- 5  
## 3 Sexmale        -2.22 1.34e-26  
## 4 Age            -0.0106 1.51e- 1
```

Predictions

- Getting Predictions

```
TRAIN.OUT2 = TRAIN.OUT %>%  
  mutate(p1=predict(logmod1,newdata=TRAIN.OUT,type="response"),  
         p2=predict(logmod2,newdata=TRAIN.OUT,type="response"),  
         p3=predict(logmod3,newdata=TRAIN.OUT,type="response")) %>%  
  select(Survived,p1,p2,p3) %>%  
  mutate(S1=ifelse(p1<0.5,0,1),  
         S2=ifelse(p2<0.5,0,1),  
         S3=ifelse(p3<0.5,0,1))  
head(TRAIN.OUT2,15)
```

##	Survived	p1	p2	p3	S1	S2	S3
## 1	0	0.1679674	0.1631565	0.1695469	0	0	0
## 2	0	NA	NA	NA	NA	NA	NA
## 3	1	0.7028675	0.6456134	0.7441205	1	1	1
## 4	1	0.7739275	0.7629271	0.6503765	1	1	1
## 5	0	0.3543259	0.3635900	0.2734311	0	0	0
## 6	1	0.1780810	0.1743017	0.1799857	0	0	0
## 7	1	NA	NA	NA	NA	NA	NA
## 8	0	0.5379343	0.6426473	0.6450425	1	1	1
## 9	0	NA	NA	NA	NA	NA	NA
## 10	0	0.2241130	0.2324596	0.1908923	0	0	0


Why?

Predictions

- Getting Predictions

```
TRAIN.OUT3=na.omit(TRAIN.OUT2)
head(TRAIN.OUT3,20)
```

##	Survived		p1	p2	p3	S1	S2	S3
## 1	0	0.16796737	0.16315653	0.1695469	0	0	0	
## 3	1	0.70286747	0.64561340	0.7441205	1	1	1	
## 4	1	0.77392753	0.76292710	0.6503765	1	1	1	
## 5	0	0.35432593	0.36359002	0.2734311	0	0	0	
## 6	1	0.17808100	0.17430173	0.1799857	0	0	0	
## 8	0	0.53793429	0.64264728	0.6450425	1	1	1	
## 10	0	0.22411295	0.23245962	0.1908923	0	0	0	



```
mean(TRAIN.OUT3$S1==TRAIN.OUT3$S2)
```

```
## [1] 0.993007
```

```
mean(TRAIN.OUT3$S2==TRAIN.OUT3$S3)
```

```
## [1] 1
```

What Do You Notice About the Predictions?

Predictions

- Getting Predictions

```
TRAIN.OUT4=TRAIN.OUT3 %>% select(-p2,-S2)  
head(TRAIN.OUT4,8)
```

##	Survived		p1	p3	S1	S3
## 1	0	0.1679674	0.1695469	0	0	
## 3	1	0.7028675	0.7441205	1	1	
## 4	1	0.7739275	0.6503765	1	1	
## 5	0	0.3543259	0.2734311	0	0	
## 6	1	0.1780810	0.1799857	0	0	
## 8	0	0.5379343	0.6450425	1	1	
## 10	0	0.2241130	0.1908923	0	0	
## 11	1	0.9907016	0.7929174	1	1	



Where Do You See Error?

Evaluation

- Evaluating Results
 - Helpful Modifications

```
TRAIN.OUT5 = TRAIN.OUT4 %>%
  select(-p1,-p3) %>%
  mutate(Survived=factor(Survived),S1=factor(S1),S3=factor(S3)) %>%
  mutate(Survived=fct_recode(Survived,"Survived"="1","Died"="0"),
         S1=fct_recode(S1,"Will Survive"="1","Will Die"="0"),
         S3=fct_recode(S3,"Will Survive"="1","Will Die"="0")) %>%
  mutate(Survived=factor(Survived,levels=c("Survived","Died")),
         S1=factor(S1,levels=c("Will Survive","Will Die")),
         S3=factor(S3,levels=c("Will Survive","Will Die")))

head(TRAIN.OUT5)
```

```
##   Survived      S1      S3
## 1     Died    Will Die    Will Die
## 2  Survived Will Survive Will Survive
## 3  Survived Will Survive Will Survive
## 4     Died    Will Die    Will Die
## 5  Survived    Will Die    Will Die
## 6     Died Will Survive Will Survive
```


Evaluation: Confusion Matrix

- Evaluating Results (Continued)
 - Confusion Matrix
 - Including 3-Way Interactions

```
RESULTS1=table(TRAIN.OUT5$Survived,TRAIN.OUT5$S1) %>%  
  prop.table()  
print(RESULTS1)
```

```
##  
##           Will Survive  Will Die  
## Survived  0.25174825  0.11188811  
## Died      0.06293706  0.57342657
```

- No Way Interactions

```
RESULTS3=table(TRAIN.OUT5$Survived,TRAIN.OUT5$S3) %>%  
  prop.table()  
print(RESULTS3)
```

```
##  
##           Will Survive  Will Die  
## Survived  0.25874126  0.10489510  
## Died      0.06293706  0.57342657
```

Evaluation: Rates

- Evaluating Results (Continued)
 - Error Statistics

- Code

```
ERROR.RESULTS = tibble(  
  Model=c("3 Way", "No Way"),  
  Sensitivity=c(RESULTS1[1,1]/sum(RESULTS1[1,]), RESULTS3[1,1]/sum(RESULTS3[1,])),  
  Specificity=c(RESULTS1[2,2]/sum(RESULTS1[2,]), RESULTS3[2,2]/sum(RESULTS3[2,])),  
  FPR=c(RESULTS1[2,1]/sum(RESULTS1[2,]), RESULTS3[2,1]/sum(RESULTS3[2,])),  
  FNR=c(RESULTS1[1,2]/sum(RESULTS1[1,]), RESULTS3[1,2]/sum(RESULTS3[1,]))  
)  
print(ERROR.RESULTS)
```

- Results

Model	Sensitivity	Specificity	FPR	FNR
<chr>	<dbl>	<dbl>	<dbl>	<dbl>
3 Way	0.692	0.901	0.0989	0.308
No Way	0.712	0.901	0.0989	0.288

Evaluation: Package

- Evaluating with ROCit and caret Package

```
> library(ROCit)
```

```
> library(caret)
```

- Generate Confusion Matrix with caret
 - Data: Prediction
 - Reference: Response
 - Input: factor

```
````{r}
confusionMatrix(as.factor(TRAIN.OUT4$S1),as.factor(TRAIN.OUT4$Survived),positive='1')
confusionMatrix(as.factor(TRAIN.OUT4$S3),as.factor(TRAIN.OUT4$Survived),positive='1')
````
```

Caret Output

- Model 1:

```
Confusion Matrix and Statistics

      Reference
Prediction 0  1
0      82 16
1       9 36

      Accuracy : 0.8252
      95% CI   : (0.7528, 0.8836)
No Information Rate : 0.6364
P-Value [Acc > NIR] : 0.0000005904

      Kappa : 0.611

Mcnemar's Test P-Value : 0.2301

      Sensitivity : 0.6923
      Specificity : 0.9011
Pos Pred Value : 0.8000
Neg Pred Value : 0.8367
Prevalence : 0.3636
Detection Rate : 0.2517
Detection Prevalence : 0.3147
Balanced Accuracy : 0.7967

'Positive' Class : 1
```

- Model 3:

```
Confusion Matrix and Statistics

      Reference
Prediction 0  1
0      82 15
1       9 37

      Accuracy : 0.8322
      95% CI   : (0.7606, 0.8894)
No Information Rate : 0.6364
P-Value [Acc > NIR] : 0.0000002115

      Kappa : 0.6282

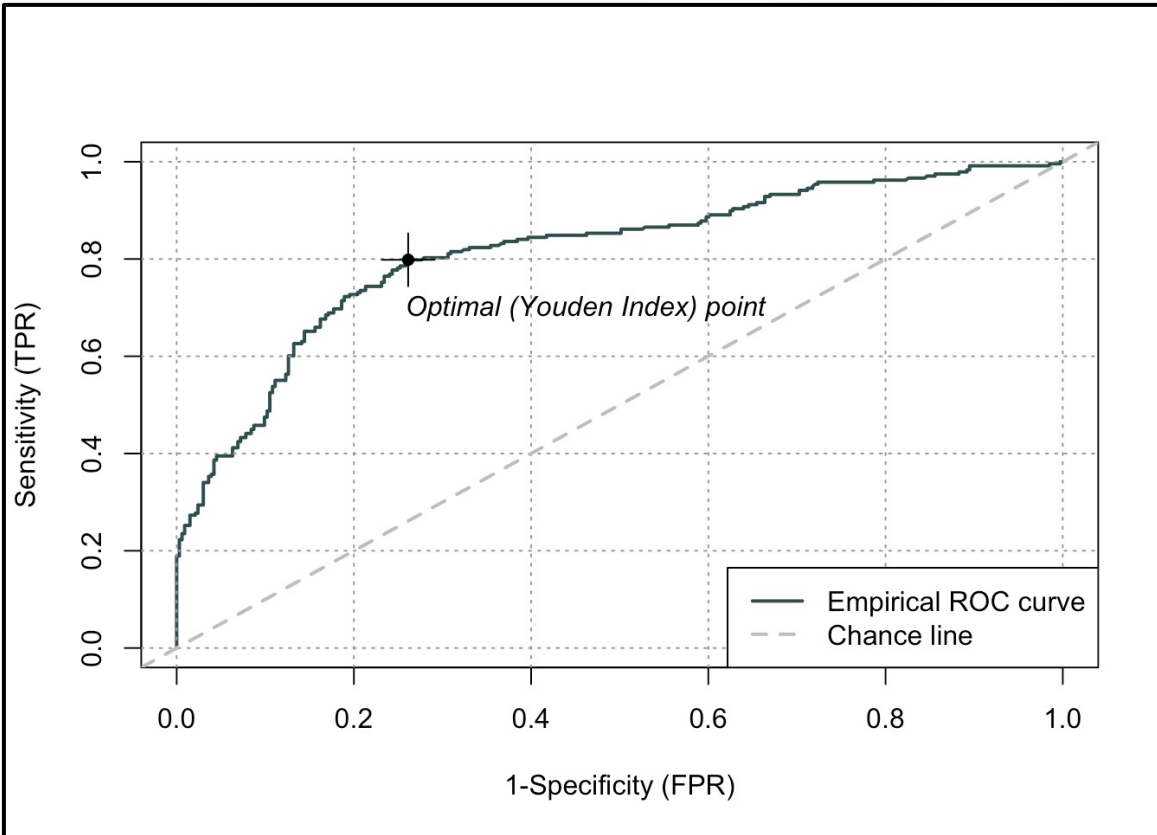
Mcnemar's Test P-Value : 0.3074

      Sensitivity : 0.7115
      Specificity : 0.9011
Pos Pred Value : 0.8043
Neg Pred Value : 0.8454
Prevalence : 0.3636
Detection Rate : 0.2587
Detection Prevalence : 0.3217
Balanced Accuracy : 0.8063

'Positive' Class : 1
```

ROC Curve: Model 1

```
logmod1_roc = rocit(score = logmod1$fitted.values, class = logmod1$y, negref=0)  
plot(logmod1_roc)
```

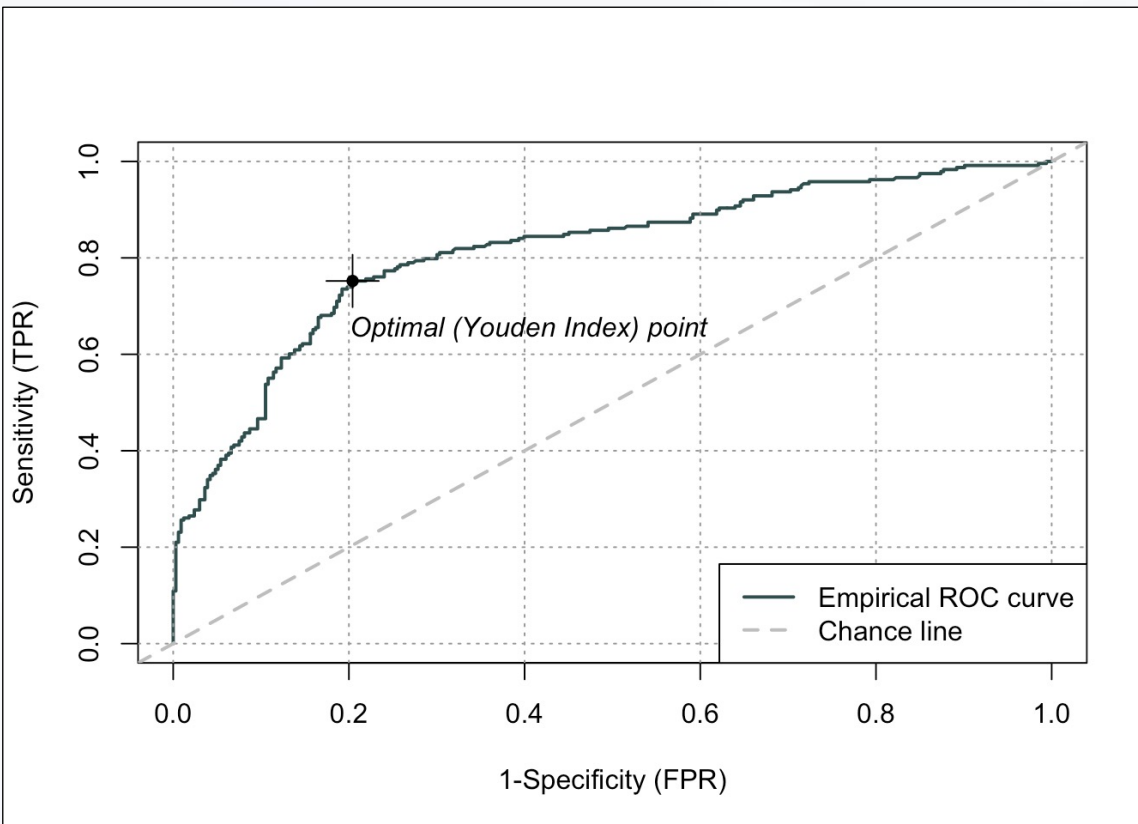


```
summary(logmod1_roc)
```

```
##  
## Method used: empirical  
## Number of positive(s): 238  
## Number of negative(s): 333  
## Area under curve: 0.8146
```

ROC Curve: Model 2

```
logmod2_roc = rocit(score = logmod2$fitted.values, class = logmod2$y, negref=0)  
plot(logmod2_roc)
```



```
summary(logmod2_roc)
```

```
##  
## Method used: empirical  
## Number of positive(s): 238  
## Number of negative(s): 333  
## Area under curve: 0.813
```