# Titanic ML

*Sangeeta Mondal*

# Objective

The sinking of the RMS Titanic is one of the most notable shipwrecks in history. After colliding with an iceberg, the Titanic had killed 1502 of 2224 passengers and crew on board. The main objective of this project is to build a model that predicts whether a passenger is going to survive based on different predictors such as age, gender and passenger class. The second objective of this project is understand which covariates are the most valuable in the prediction of survival. Machine learning algorithm Random Forest will be used to predict survival.

# Data Background

The data in this study was collected from 1309 randomly selected passengers on the Titanic. The outcome variable, 'Survival', was a dichotomous variable with 1 representing survival and 0 representing death. There were 12 total variables in the dataset. Certain covariates were not included in the analysis (discussed later). There were also missing observations in the dataset (discussed later).

# Preprocessing the Dataset

```
require(ggplot2)
require(knitr)
require(mice)
require(randomForest)
require(gmodels)
require(MLmetrics)
require(dplyr)
require(caTools)
require(gridExtra)
```

```
all<-read.csv('C:/Users/PC/Documents/Biostat 273/train.csv', stringsAsFactors = F)
```

```
all_levs<-apply(all,2,unique)
dim(all)
```

```
## [1] 891  12
```

```
#factor variables
all$Survived<-as.factor(all$Survived)
all$Sex<-as.factor(all$Sex)
all$Embarked<-as.factor(all$Embarked)

#numeric variables
all$PassengerId<-as.numeric(all$PassengerId)
all$Age<-as.numeric(all$Age)
all$SibSp<-as.numeric(all$SibSp)
all$Parch<-as.numeric(all$Parch)
all$Fare<-as.numeric(all$Fare)
all$Pclass<-as.numeric(all$Pclass)

#cabin and embarked have blanks -- fill in blanks with NAs
all$Cabin[all$Cabin == '']<-NA
all$Embarked[all$Embarked == '']<-NA
```

# Feature Engineering

In order to increase accuracy of prediction, new features that were potentially thought to be correlated to survival were constructed. Two new features were created. The first one was ‘Title’. The titles of the passengers were extracted and then were categorized into 5 groups: Miss, Mrs, Mr, Master, and Other Title. Essentially, ‘Title’ can be viewed as age-by-gender interaction term. The second feature created was an interaction term between sex and passenger class called ‘SexP’.

```
#Title
all$Title<- gsub('(.*, )|(\\..*)', '', all$Name)

other <- c('Dona', 'Lady', 'the Countess','Capt', 'Col', 'Don',
           'Dr', 'Major', 'Rev', 'Sir', 'Jonkheer') #collapse uncommon titles into on
e category

all$Title[all$Title == 'Mlle']<- 'Miss'
all$Title[all$Title == 'Ms']<- 'Miss'
all$Title[all$Title == 'Mme']<- 'Mrs'
all$Title[all$Title %in% other]<-'Other Title'
all$Title<-as.factor(all$Title)

#Interaction term between Sex and Pclass-->recode and then multiply terms
all$Sex_N<-ifelse(all$Sex == 'female', 1, 0)
all$SexP<-all$Sex_N * all$Pclass
```

The data will be split back into a training and test set.

```
#Split back into training and test
train<-all[1:891,]
test<-all[892:1309,]
```

For now, we will ignore the test set. The training test will be split 80/20.80% of the train data will be the observations that we train our algorithm with. The other 20% will serve as the validation set. Hence, the training set had 713 observations while the test set had 178 observations.

```
set.seed(307)
samp_d<-sample.split(train$Survived, SplitRatio = .8)
n_train<-subset(train, samp_d == TRUE)
n_test<-subset(train, samp_d == FALSE)
```

# Exploratory Analysis

Both the training and test sets were split into two groups. The first group contained quantitative variables (both discrete and continous types). The second group contained qualitative variables. In doing the analysis, certain variables were omitted: PassengerId, Name, Ticket, Cabin, and Sex_N. ‘PassengerID’ and ‘Name’was omitted because it was unique to every passenger and no important information could be analyzed from it. ’Ticket’ was less variant but still fairly unique to every passenger so it was also omitted. ‘Cabin’ had too much missing information so it was dropped as well. Finally, ‘Sex_N’ was dropped from the analysis as well because it was recoded from the ‘Sex’ variable; hence, it was redundant.

```
cont<-c('Age', 'SibSp', 'Parch', 'Fare')

train_cont<-n_train[,cont]
train_oth<-n_train[,!names(n_train) %in% cont]
train_oth<-train_oth[,!names(train_oth) %in% c('PassengerId',
                                               'Name',
                                               'Ticket',
                                               'Cabin',
                                               'Sex_N')]

test_cont<-n_test[,cont]
test_oth<-n_test[,!names(n_test) %in% cont]
test_oth<-test_oth[,!names(test_oth) %in% c('PassengerId',
                                               'Name',
                                               'Ticket',
                                               'Cabin',
                                               'Sex_N')]
```

```
#summary stats fxn-- quant variables
sumCont<-function(var){
  c(mean(var, na.rm = TRUE), median(var, na.rm = TRUE),
    sd(var, na.rm = TRUE), sum(is.na(var)))
}

#summary stats fxn-- qual vars
sumCat<-function(var){
  l<-c()
  for (level in sort(unique(var))){
    prop<-round(length(which(var == level))/length(var),2)
    l<-c(prop, l)
  }
  if (length(l)<5){
    l<-c(l, rep('NA', 5-length(l)))
  }
  l<-c(l, sum(is.na(var)))
  return(l)
}

#continuous vars
sumContTr<-apply(train_cont, 2, sumCont)
sumContTr<-t(round(sumContTr,2))
sumContTe<-apply(test_cont, 2, sumCont)
sumContTe<-t(round(sumContTe, 2))

allCont<-cbind(sumContTr, sumContTe)
colnames(allCont)<-c('Train: Mean', 'Train: Median', 'Train: SD', 'Train: Missing Valu
es', 'Test: Mean', 'Test: Median', 'Test: SD', 'Test: Missing Values')
kable(allCont, caption = 'Table 1. Summary Statistics of Quantitative Variables for Bo
th Training and Test Sets')
```

Table 1. Summary Statistics of Quantitative Variables for Both Training and Test Sets

| | Train: Mean | Train: Median | Train: SD | Train: Missing Values | Test: Mean | Test: Median | Test: SD | Test: Missing Values |
|---|---|---|---|---|---|---|---|---|
| Age | 29.62 | 28.00 | 14.87 | 136 | 30.02 | 29.00 | 13.03 | 41 |
| SibSp | 0.52 | 0.00 | 1.13 | 0 | 0.52 | 0.00 | 1.01 | 0 |
| Parch | 0.38 | 0.00 | 0.80 | 0 | 0.38 | 0.00 | 0.84 | 0 |
| Fare | 31.38 | 14.11 | 49.95 | 0 | 35.50 | 15.37 | 48.64 | 0 |

Based on table 1, there are no remarkable differences in the summary statistics between the training and test sets. The passengers in the test set did seem to have slightly higher fares. Furthermore, the large numbers of missing age values in both sets should be noted as well.

```
#categorical vars
sumCatTr<-apply(train_oth, 2, sumCat)
sumCatTr<-t(sumCatTr)

sumCatTe<-apply(test_oth, 2, sumCat)
sumCatTe<-t(sumCatTe)

allCat<-cbind(sumCatTr, sumCatTe)
colnames(allCat)<-c('Train: Cat 1', 'Cat 2', 'Cat 3', 'Cat 4','Cat 5', 'Missing Value
s', 'Test: Cat 1', 'Cat 2', 'Cat 3', 'Cat 4', 'Cat 5', 'Missing Values')

kable(allCat, caption = 'Table 2. Summary Statistics for the Qualitative Variables in
the Training and Test Dataset. If the variable did not have the max categories as list
ed in the table, it was assigned a NA value')
```

Table 2. Summary Statistics for the Qualitative Variables in the Training and Test Dataset. If the variable did not have the max categories as listed in the table, it was assigned a NA value

| | **Train: Cat 1** | **Cat 2** | **Cat 3** | **Cat 4** | **Cat 5** | **Missing Values** | **Test: Cat 1** | **Cat 2** | **Cat 3** | **Cat 4** | **Cat 5** | **Missing Values** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Survived | 0.38 | 0.62 | NA | NA | NA | 0 | 0.38 | 0.62 | NA | NA | NA | 0 |
| Pclass | 0.56 | 0.2 | 0.24 | NA | NA | 0 | 0.53 | 0.22 | 0.25 | NA | NA | 0 |
| Sex | 0.66 | 0.34 | NA | NA | NA | 0 | 0.61 | 0.39 | NA | NA | NA | 0 |
| Embarked | 0.73 | 0.08 | 0.19 | NA | NA | 1 | 0.7 | 0.1 | 0.2 | NA | NA | 1 |
| Title | 0.03 | 0.13 | 0.58 | 0.21 | 0.05 | 0 | 0.01 | 0.2 | 0.57 | 0.19 | 0.03 | 0 |
| SexP | 0.16 | 0.08 | 0.1 | 0.66 | NA | 0 | 0.19 | 0.1 | 0.11 | 0.61 | NA | 0 |

Table 2 describes the qualitative variables in the dataset. The proportion of passengers per level of each variable was tabulated. It can be seen that there are no major differences in the distributions of each variable. Furthermore, ‘Embarked’ has 2 missing values.

# Exploring Associations Between Survival and Other Covariates

```
#getting grid objects--Pclass, Sex, Embarked, SibSp, Parch
gridObject<-function(p){
  grid<-as.matrix(table(n_train[,p], n_train[,'Survived'], dnn = c(p, 'Survived')))
  n_grid<-grid/rowSums(grid)
  n_grid<-as.data.frame(n_grid)
  return(ggplot(data = n_grid, aes(x = n_grid[,1], y = Freq, fill = Survived))+
           geom_bar(stat="identity", position=position_dodge())+labs(x = p))
}

vars<-c('Pclass', 'Sex', 'Embarked', 'SibSp', 'Parch')
```

```
grid.arrange(gridObject(vars[1]), gridObject(vars[2]),
             gridObject(vars[3]), gridObject(vars[4]),
             gridObject(vars[5]),
             ncol = 2, nrow = 3)
```

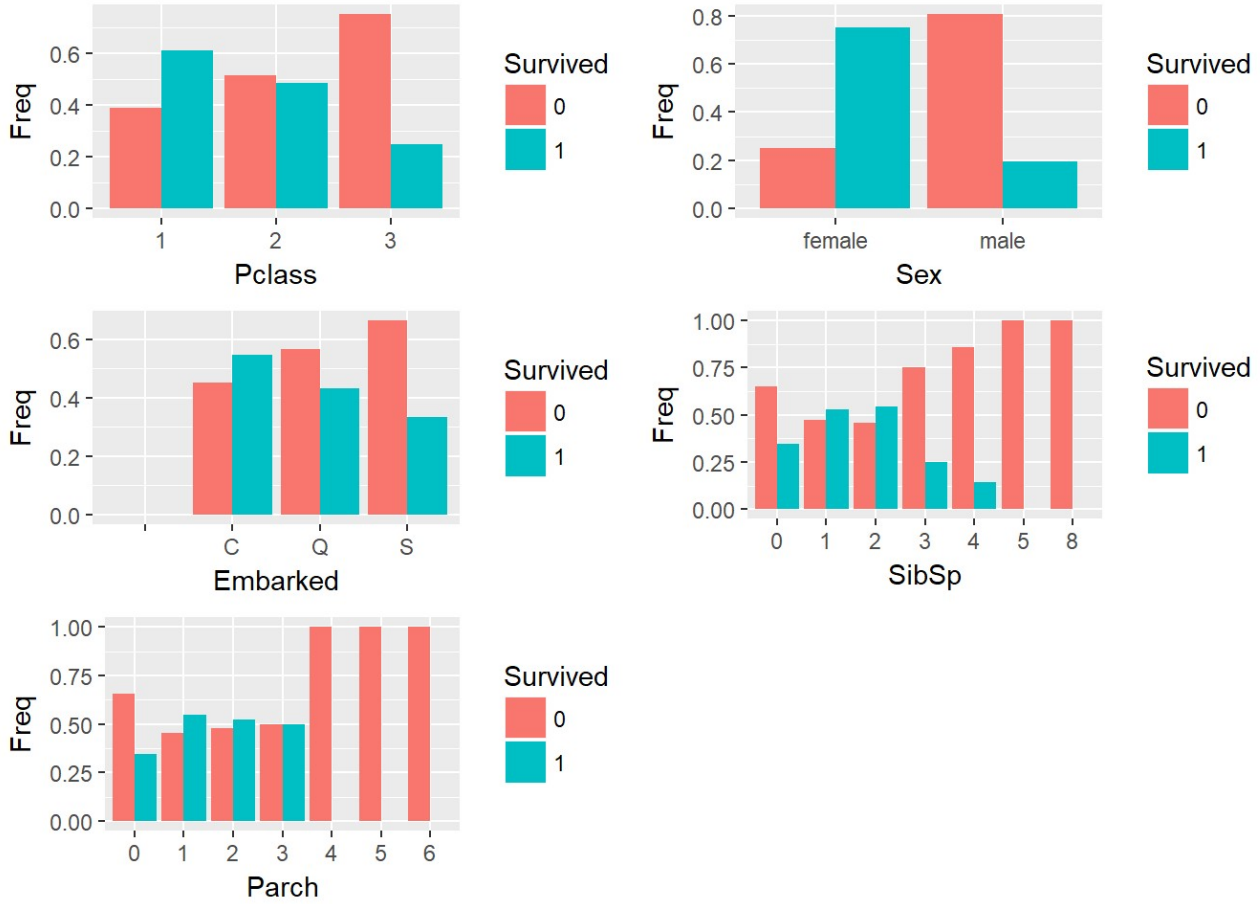
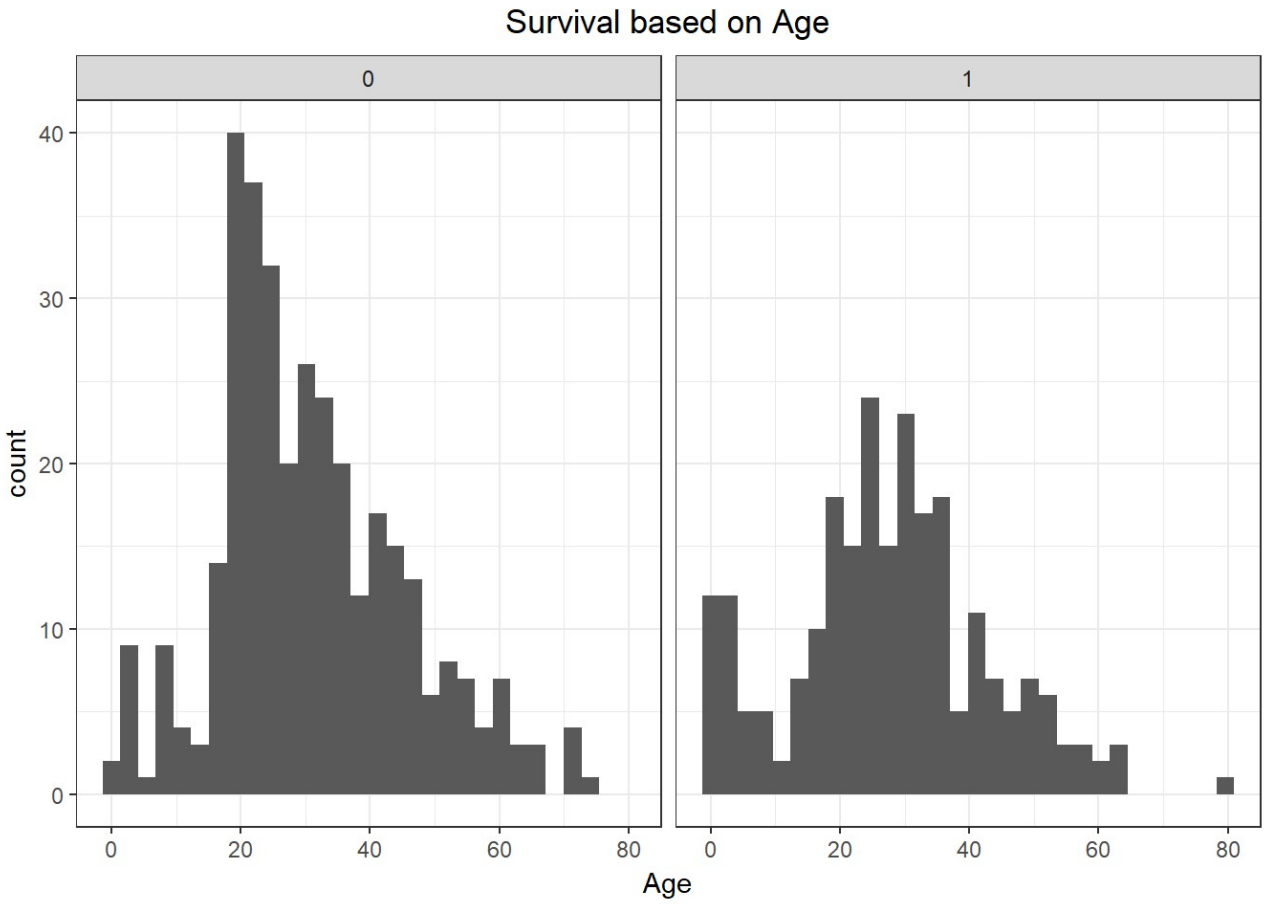


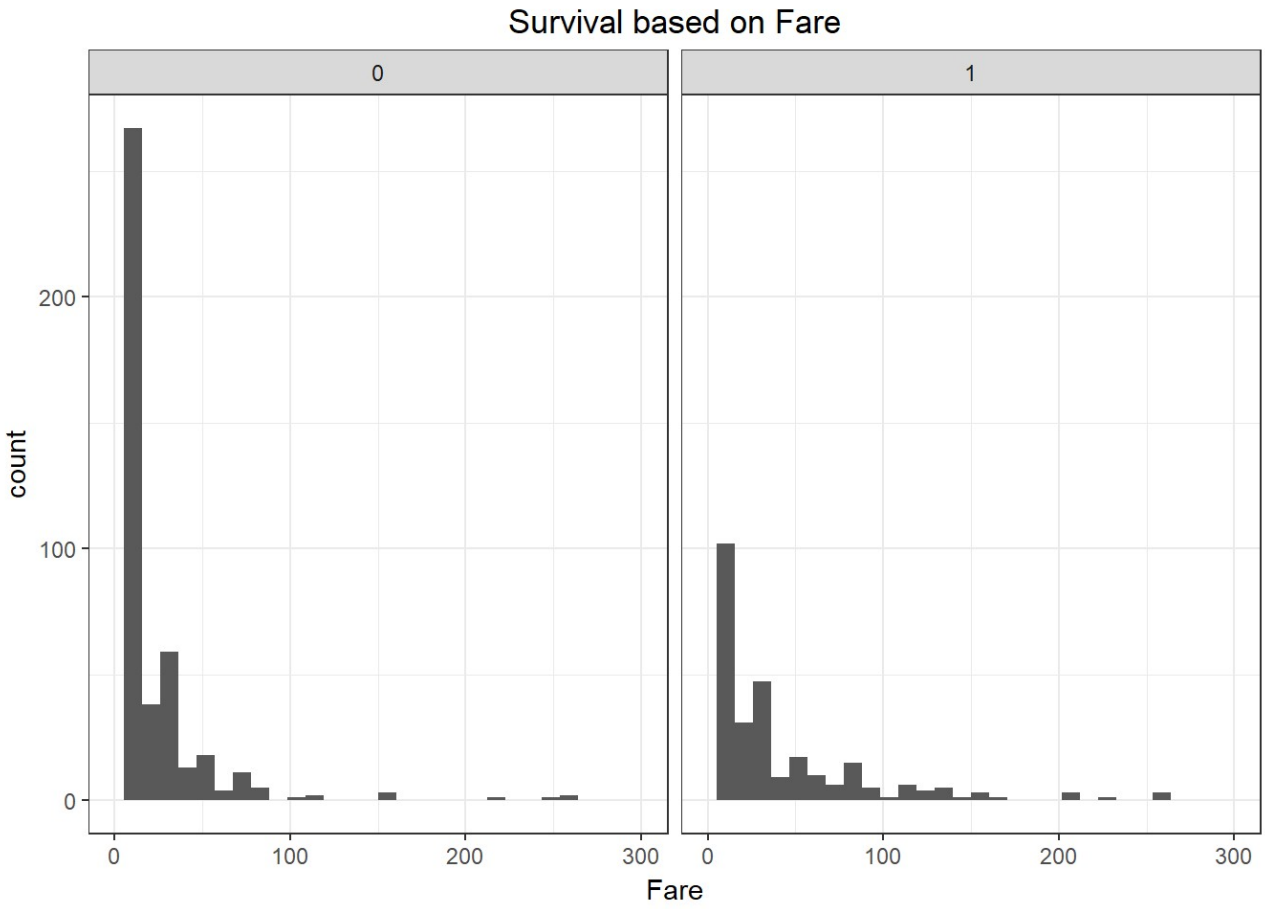

Figure 1. Bar plot that show distributions of different categories of predictor variables correlated to the different categories of Survival

The four variables that I wanted to observe relationships of Survival with were : Pclass (passenger class), Sex, Embarked (which port the passengers embarked from), SibSp (number of siblings and spouses on board the Titanic), and Parch (number of parents and children onboard the Titanic). Based on the graphs, the variables all had some sort of relationship with Survival. There were some several striking main differences: * The largest difference in the distribution of survival was observed for passengers with <5 siblings and spouses and >=5 siblings and spouses. Every passenger who had >5 siblings and spouses died. * Similarly, all passengers who had >=4 parents and children onboard, had died. * Finally, there was some pretty clear gender divides. Female passengers were much more likely to survive than male passengers.

```
#histogram distribution for age and survival
ggplot(n_train, aes(x = Age))+geom_histogram()+facet_grid(~Survived)+theme_bw()+ggtitl
e('Survival based on Age')+theme(plot.title = element_text(hjust = 0.5))
```




```
#Histogram for fare and survival
ggplot(n_train, aes(x = Fare))+geom_histogram()+facet_grid(~Survived)+xlim(0,300) +the
me_bw()+ggtitle('Survival based on Fare')+theme(plot.title = element_text(hjust = 0.
5))
```




Age based distributions show that younger passengers died at higher rates than older passengers; those who payed lower fares also died at higher rates than those who did not pay higher fares.

# Working with Missing Data

Missing data from 3 variables will be dealt with here: Embarked and Age. We will string by joining the data back together.

```
tog<-bind_rows(n_train, n_test)
```

## Embarked

Embarked has 2 missing values. The strategy here is to look at the median fare values for 1st class passengers at each port (Charbourg, Queenstown, and Southampton). Since the median fare of Charbourg 1st class passenger is closest to the $80 fare paid by passengers 62 and 830, we impute Charbourg as the missing embarked values.

```
tog[which(is.na(tog$Embarked)), c('Pclass', 'Fare')]
```

```
##     Pclass Fare
## 50       1   80
## 876      1   80
```

```
c<-median(tog$Fare[tog$Embarked == 'C'&tog$Pclass == 1], na.rm = TRUE)
q<-median(tog$Fare[tog$Embarked == 'Q'&tog$Pclass == 1], na.rm = TRUE)
s<-median(tog$Fare[tog$Embarked == 'S'&tog$Pclass == 1], na.rm = TRUE)
tot<-c(c,q,s);tot<-t(tot);colnames(tot)<-c('Fare Charbourg Passenger Class 1',
                                           'Fare Queenstown Passenger Class 1',
                                           'Fare Southampton Passenger Class 1')
kable(tot)
```

| Fare Charbourg Passenger Class 1 | Fare Queenstown Passenger Class 1 | Fare Southampton Passenger Class 1 |
|---:|---:|---:|
| 78.2667 | 90 | 52 |

```
tog$Embarked[c(50,876)]<- 'C'
```

### Age

Age had 263 missing values. MICE imputation was used to impute missing age values.

```
tog2<-tog[,c('Age', 'SibSp', 'Parch','Fare')] #disclude Pclass though
mice_imp<-mice(tog2, method = 'pmm', seed = 121) #Predictive mean matching
num_imp<-complete(mice_imp)
tog$Age<-num_imp$Age
```

```
#split back into test and train
n_train <- tog[1:713,]
n_test <- tog[714:891,]
```

# Random Forest

```
varwork<-c('Survived','Pclass', 'Age', 'SibSp', 'Parch', 'Fare', 'Embarked', 'Title',
'Sex_N', 'SexP')
inp<-n_train[,names(n_train) %in% varwork]

test_pred<-n_test[,names(n_test) %in% varwork]
```

```
#basic model
set.seed(311)
cross_v<-.70*length(inp$Survived) #OOB Error
base_for<- randomForest(y = inp$Survived, x = inp[,2:10],
                        ntree = 1000, sampsize = cross_v, cutoff = c(0.5,0.5))
```

```
base_for.resp<-predict(base_for, newdata = test_pred[,2:10])
```

The basic model was ran with 1000 tries. 30% of the data training set was set aside for cross validation. The cut off was specifically left to default settings. A passenger was assigned to survival only if 50% or more of the trees classified it as a survivor. Unless other specified, all default settings were used. For example, the number of predictors randomly sampled at each split was 3 (mtry); the minimum size of terminal nodes was 1 observation (node size).

```
#performance errors tabulated--classification accuracy, sensitivity (recall), ppv(prec
ision), f1 score, specificity
PET<-function(fore,p1, p2){
  a_score<-Accuracy(p1,p2)
  specificity_1<-1-fore$confusion[1,3]
  sensitivity_1<-1-fore$confusion[2,3]
  ppv<-fore$confusion[2,2]/(fore$confusion[2,2]+fore$confusion[1,2])
  f1<-(2*(sensitivity_1*ppv))/(sensitivity_1+ppv)
  return(c('Accuracy Score' = a_score, 'Sensitivity (Recall)' = sensitivity_1, 'PPV (p
recision)' = ppv, 'F1' = f1, 'Specificity' = specificity_1))
}

results<-PET(base_for, base_for.resp ,test_pred$Survived)
results.mat<-matrix(results, nrow = 1);colnames(results.mat)<-names(results)
results.df<-as.data.frame(results.mat)
kable(results.df)
```

| Accuracy Score | Sensitivity (Recall) | PPV (precision) | F1 | Specificity |
|---|---|---|---|---|
| 0.8370787 | 0.6970803 | 0.809322 | 0.7490196 | 0.8974943 |

Based on The sensitivity value was .70 while the specificity value was .90. There is a somewhat large difference between sensitivity and specificity and that model is skewed towards labeling passengers as dead. A series of models were built to find the most optimal cutoffs to increase the sensitivity of the model.

```
sen<-c()
set.seed(580)
for (i in seq(from = 0.05, to = .95, by = .05)){
  for_ob<-randomForest(y = inp$Survived, x = inp[,2:10], ntree = 1000, sampsize = cros
s_v, cutoff = c(i,1-i))
  sen2<-c(for_ob$confusion[,3])
  sen<-rbind(sen2,sen)

}

sensp <- 1-sen
```

```
youden <- sensp[,1] + sensp[,2] - 1
sensp <- cbind(sensp, youden)
colnames(sensp)<-c('Specificity', 'Sensitivity', 'Youden')
rownames(sensp)<-rev(seq(from = .05, to= .95, by = .05))
sensp <- as.data.frame(sensp)
```

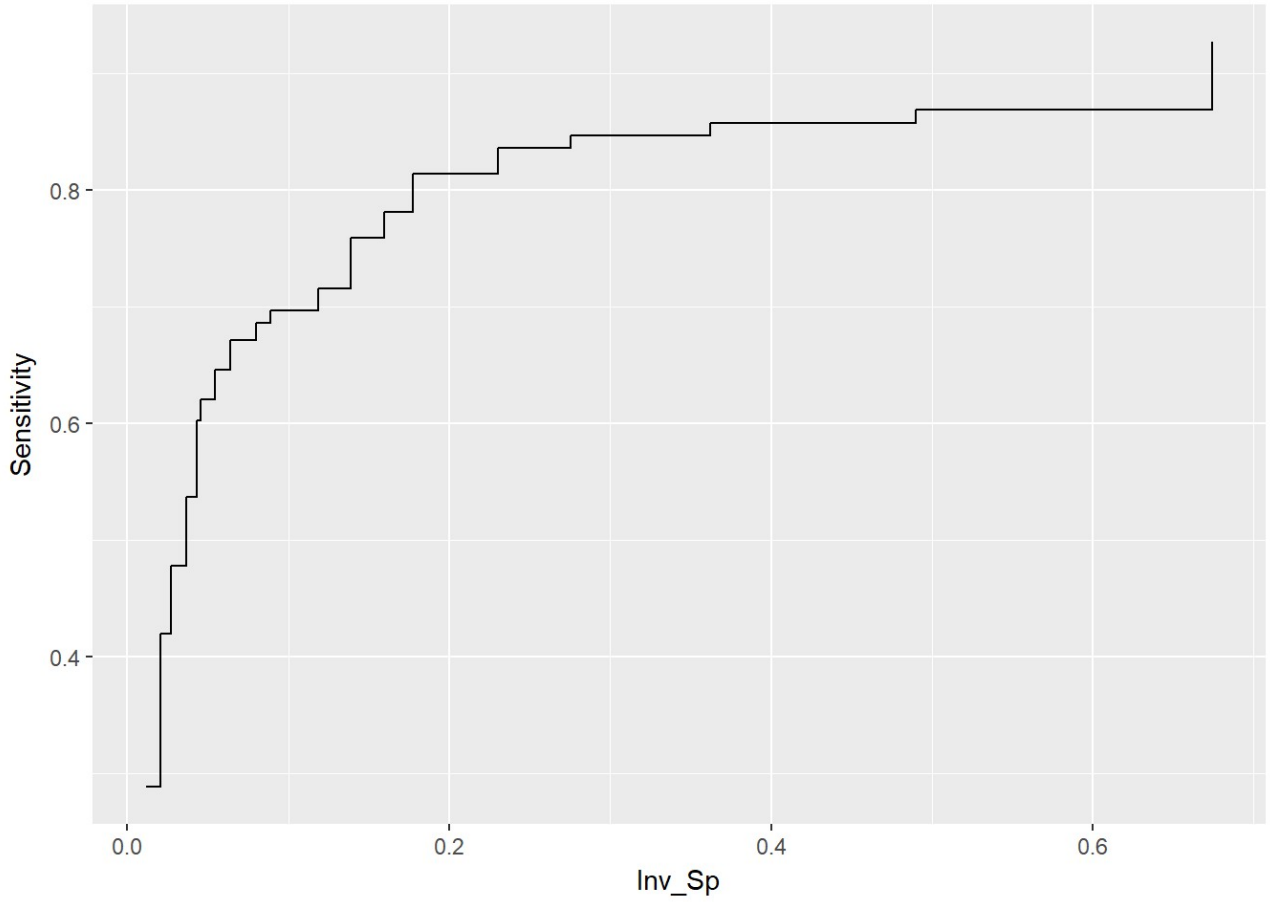
```
sensp$Inv_Sp <- 1-sensp$Specificity
kable(sensp)
```

| | Specificity | Sensitivity | Youden | Inv_Sp |
|---|---|---|---|---|
| 0.95 | 0.3257403 | 0.9270073 | 0.2527476 | 0.6742597 |
| 0.9 | 0.5102506 | 0.8686131 | 0.3788637 | 0.4897494 |
| 0.85 | 0.6378132 | 0.8576642 | 0.4954774 | 0.3621868 |
| 0.8 | 0.7243736 | 0.8467153 | 0.5710889 | 0.2756264 |
| 0.75 | 0.7699317 | 0.8357664 | 0.6056981 | 0.2300683 |
| 0.7 | 0.8223235 | 0.8138686 | 0.6361921 | 0.1776765 |
| 0.65 | 0.8405467 | 0.7810219 | 0.6215686 | 0.1594533 |
| 0.6 | 0.8610478 | 0.7591241 | 0.6201719 | 0.1389522 |
| 0.55 | 0.8815490 | 0.7153285 | 0.5968774 | 0.1184510 |
| 0.5 | 0.9111617 | 0.6970803 | 0.6082420 | 0.0888383 |
| 0.45 | 0.9202733 | 0.6861314 | 0.6064047 | 0.0797267 |
| 0.4 | 0.9362187 | 0.6715328 | 0.6077515 | 0.0637813 |
| 0.35 | 0.9453303 | 0.6459854 | 0.5913157 | 0.0546697 |
| 0.3 | 0.9544419 | 0.6204380 | 0.5748799 | 0.0455581 |

| | Specificity | Sensitivity | Youden | Inv_Sp |
|---|---|---|---|---|
| 0.25 | 0.9567198 | 0.6021898 | 0.5589096 | 0.0432802 |
| 0.2 | 0.9635535 | 0.5364964 | 0.5000499 | 0.0364465 |
| 0.15 | 0.9726651 | 0.4781022 | 0.4507673 | 0.0273349 |
| 0.1 | 0.9794989 | 0.4197080 | 0.3992069 | 0.0205011 |
| 0.05 | 0.9886105 | 0.2883212 | 0.2769316 | 0.0113895 |

We use the Youden Index to find the best cutoff point. In this way, both sensitivity and specificity are optimized. The maximum value of the Youden Index is when the cutoff is .7. The specificty is .82 and sensitivity is .81.

```
ggplot(data = sensp) +
  geom_step(mapping= aes(x = Inv_Sp, y =Sensitivity))
```




The AUC Curve is plotted above.

```
set.seed(7000)
mod1<-randomForest(y = inp$Survived, x = inp[,2:10],
                   ntree= 1000, importance = TRUE, sampsize = cross_v, cutoff = c(0.5,
0.5))
mod2<-randomForest(y = inp$Survived, x = inp[,2:10],
                   ntree= 1000, importance = TRUE, sampsize = cross_v, cutoff = c(0.7,
0.3))

mod1.exp<-predict(mod1, newdata = test_pred[,2:10])
mod2.exp<-predict(mod2, newdata = test_pred[,2:10])

metr_1<-PET(mod1, mod1.exp, test_pred$Survived)
metr_2<-PET(mod2, mod2.exp, test_pred$Survived)

tab<-rbind(metr_1, metr_2)
rownames(tab)<-c('Model 1- Cut Off: .5',
                 'Model 2- Cut Off: .7')

kable(tab)
```

| | Accuracy Score | Sensitivity (Recall) | PPV (precision) | F1 | Specificity |
|---|---|---|---|---|---|
| Model 1- Cut Off: .5 | 0.8426966 | 0.7007299 | 0.8170213 | 0.7544204 | 0.9020501 |
| Model 2- Cut Off: .7 | 0.8089888 | 0.8102190 | 0.7278689 | 0.7668394 | 0.8109339 |

When we compare models with different cutoffs, we observe a lower accuracy score. However, we see a increase in sensitivity and a slight increase in F1. More analysis should be carried out to find how specificity can be increased so that we can preserve accuracy. Accuracy may be improved by adding additional features.

```
### Variable Importance
mod2.imp<-cbind(names(mod2$importance[,3]), unname(mod2$importance[,3]))
mod2.imp<-data.frame(mod2.imp)
colnames(mod2.imp)<-c('Variable', 'Importance')
rankImportance <- mod2.imp %>%
  mutate(Rank=dense_rank(desc(Importance)))
rankImportance<-rankImportance[order((rankImportance[,3]), decreasing = FALSE),]

kable(rankImportance)
```

| | Variable | Importance | Rank |
|---|---|---|---|
| 7 | Title | 0.0822319280208774 | 1 |

| | Variable | Importance | Rank |
|---|---|---|---|
| 9 | SexP | 0.0633532501767582 | 2 |
| 8 | Sex_N | 0.059137487471227 | 3 |
| 1 | Pclass | 0.0407881691200827 | 4 |
| 5 | Fare | 0.0357654222188014 | 5 |
| 2 | Age | 0.0216636123134159 | 6 |
| 3 | SibSp | 0.0167042009027494 | 7 |
| 6 | Embarked | 0.00639096210121474 | 8 |
| 4 | Parch | 0.00224198360147465 | 9 |

Variable importance was assessed as well. We see that Title and SexP (Sex*PassengerClass) were the most important variables.