Project Proposal: Titanic Survival Model

Prepared for: OMIS 472 and Team Leada

Prepared by: Jaleel Savoy

November 2, 2016

Proposal number: 000-0001

INTRODUCTION

Context

On the 15th of April in 1912, the RMS Titanic, a British passenger liner, sank in the North Atlantic Ocean after colliding into an iceberg. The Titanic was the largest ship during its time, and it carried 2,224 people during its first and last voyage. The accident is recorded in history as the most infamous and deadliest commercial maritime disaster during peacetime, which resulted in 1,500 deaths due to the lack of lifeboats.

Statement of the Problem

There were many people aboard the vessel and it would be beneficial to identify the major contributing factors to survival chances of the various passengers.

Purpose

The purpose of this machine learning project is to identify the characteristics that greatly affected passengers chance of survival. With that information, a model will be built that is able to accurately predict which passengers would survive and what passengers would die. Initial assumption, based upon the social norm of the time, is that class, sex, and age are the most important factors in determining survival.

Research Method

To perform the necessary analysis of the Titanic data, provided by Kaggle.com, I used the R language, the RStudio IDE, and various packages in R to improve its default capabilities.

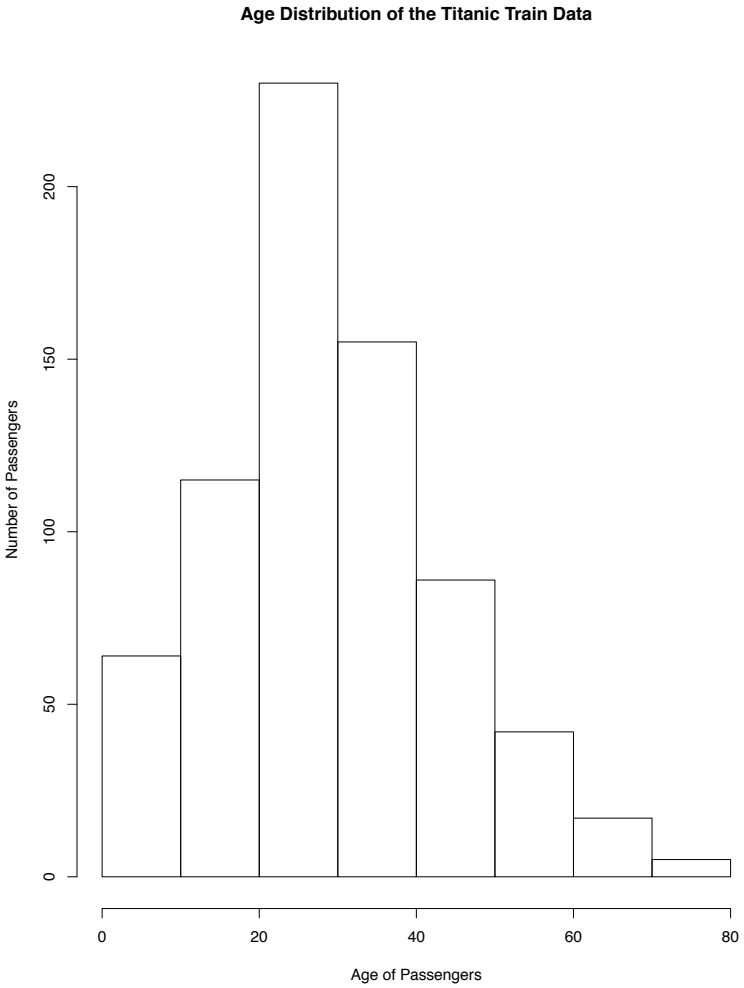
Scope and limitations

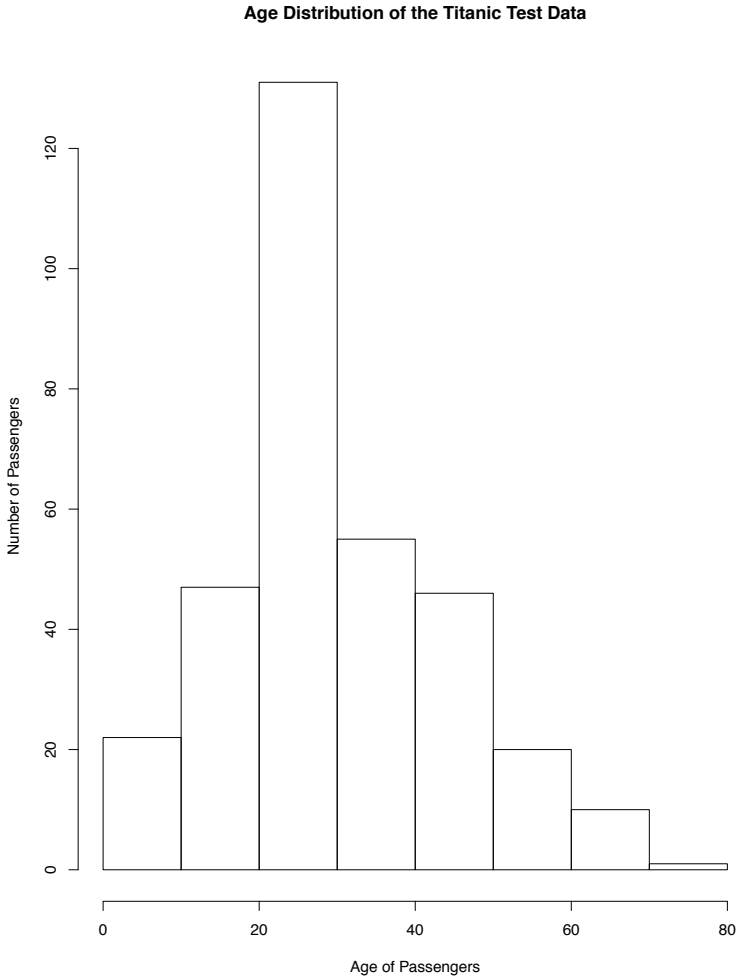
The project includes an analysis of all easily observed and recorded data about the passengers. Some data that may have been useful to the analysis cannot be used simply because it was not recorded.

COMPANY NAME

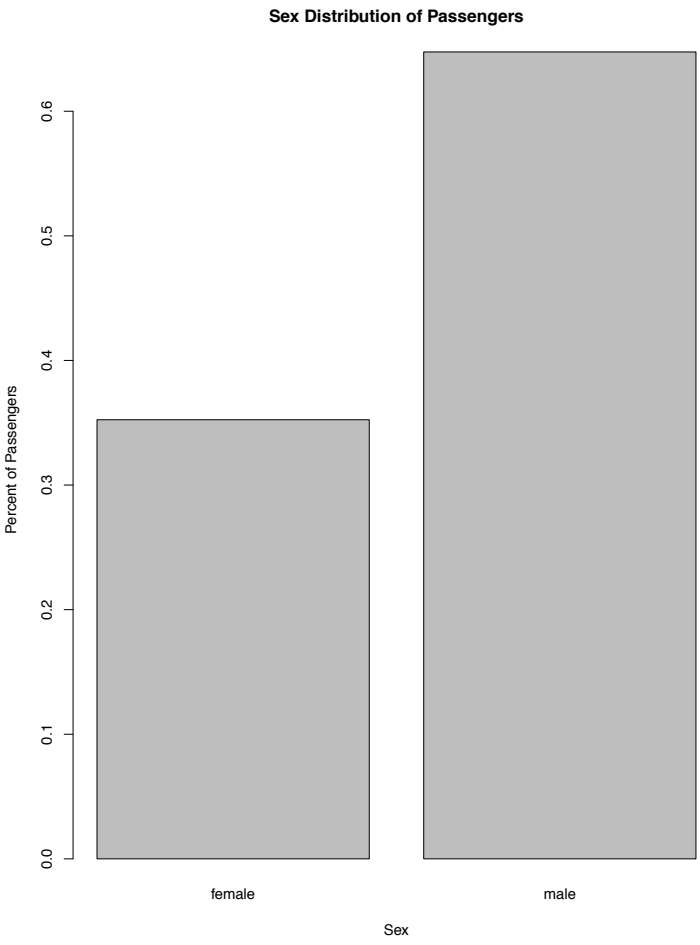
Exploratory Data Visualizations (Train dataset on the left, Test dataset on the right)

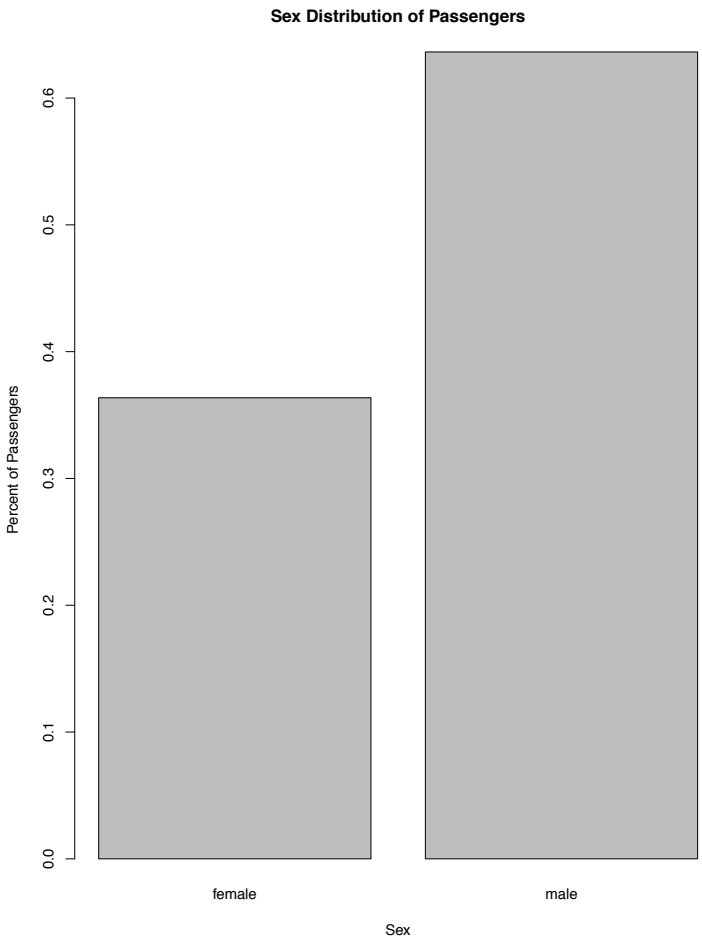
1) *Age*



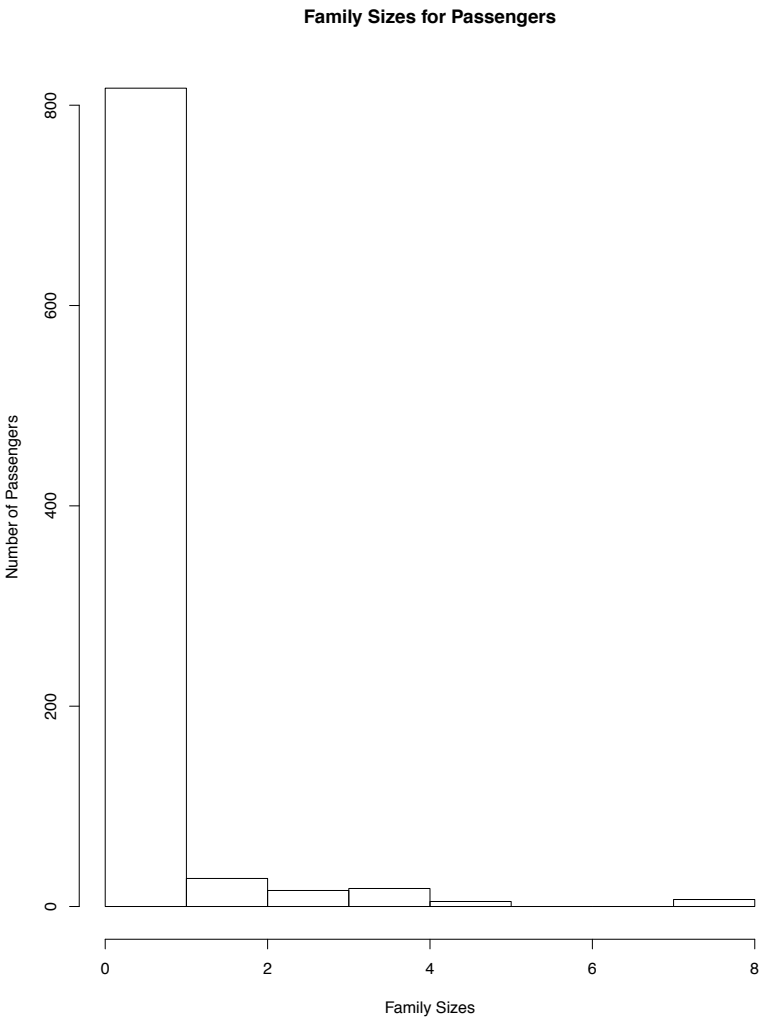


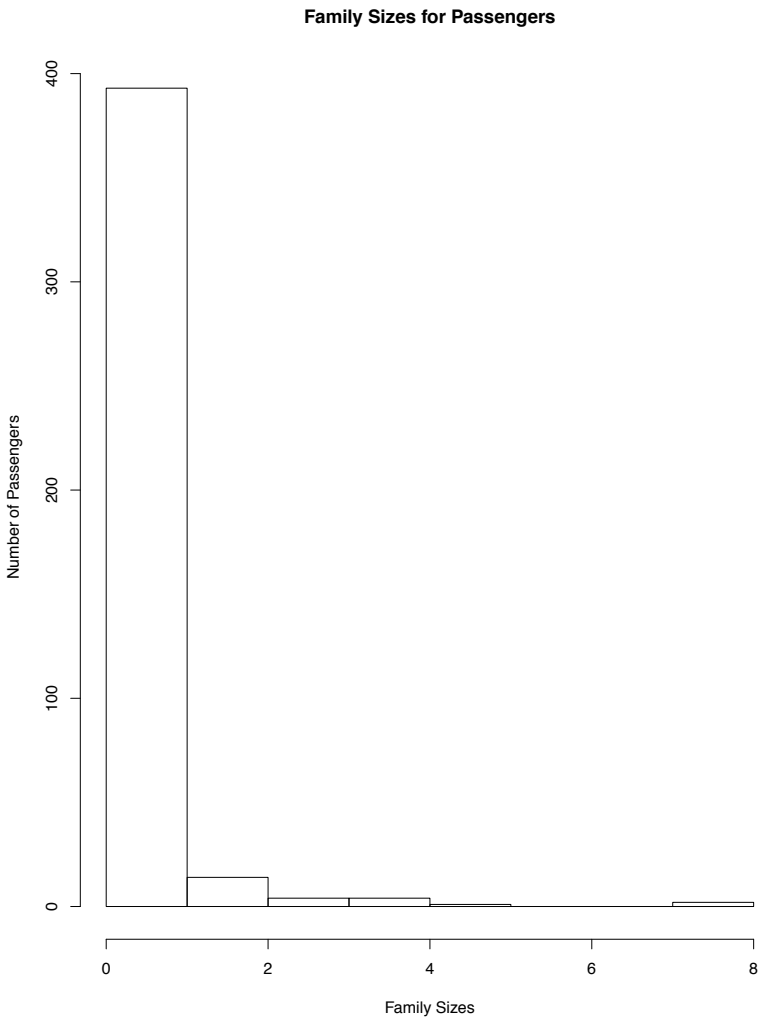
2) *Sex*


Sex Distribution of Passengers
Percent of Passengers
0.0
0.1
0.2
0.3
0.4
0.5
0.6
female
male
Sex

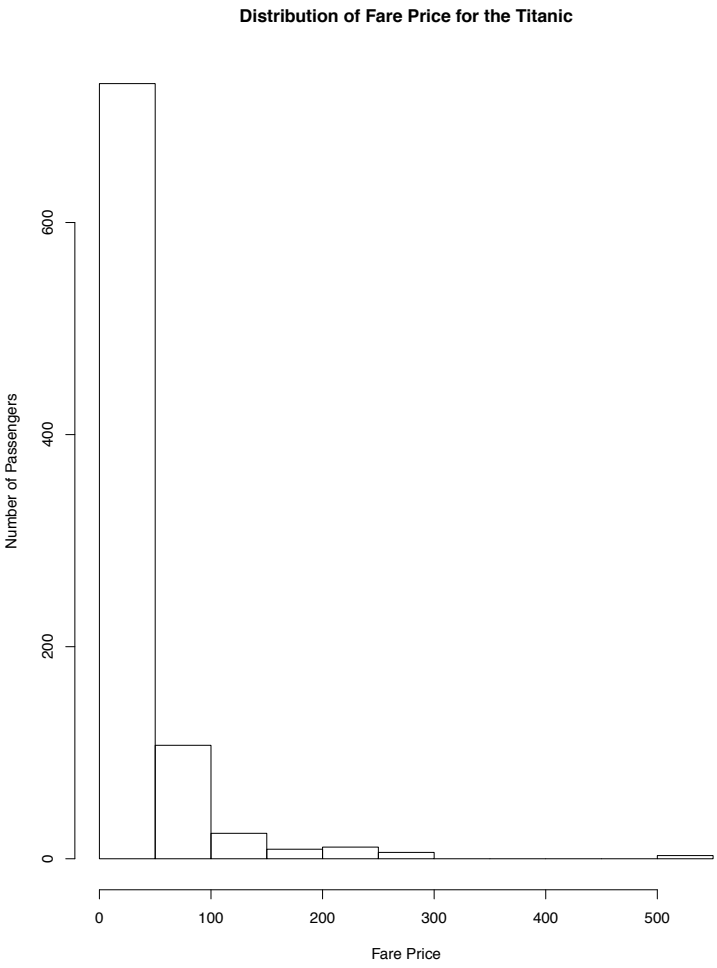

Sex Distribution of Passengers
Percent of Passengers
0.0
0.1
0.2
0.3
0.4
0.5
0.6
female
male
Sex

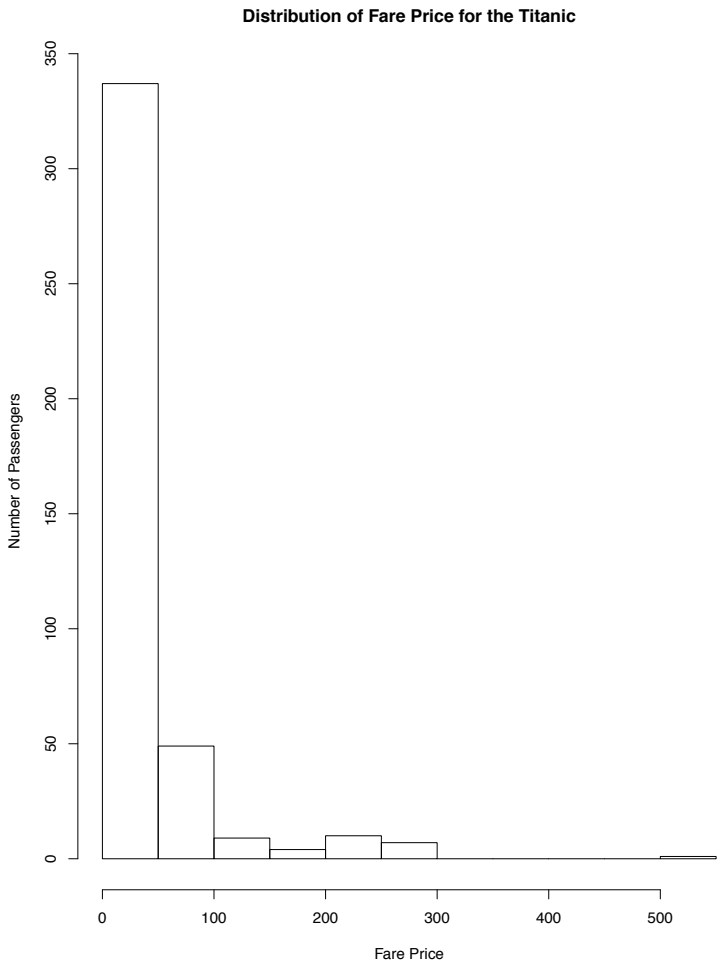
3) *Number of Spouses and Siblings*





4) *Fare Price*


Distribution of Fare Price for the Titanic
Number of Passengers
0
200
400
600
0
100
200
300
400
500
Fare Price

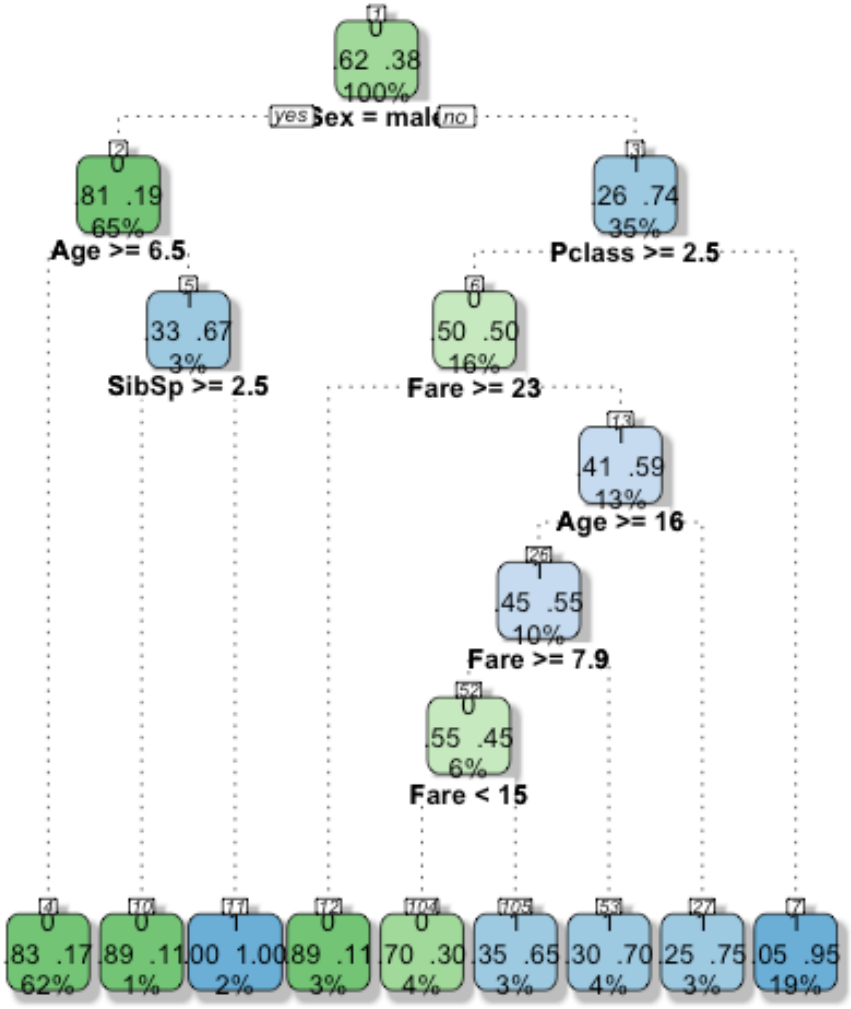

Distribution of Fare Price for the Titanic
Number of Passengers
0
50
100
150
200
250
300
350
0
100
200
300
400
500
Fare Price

Methods and Analysis

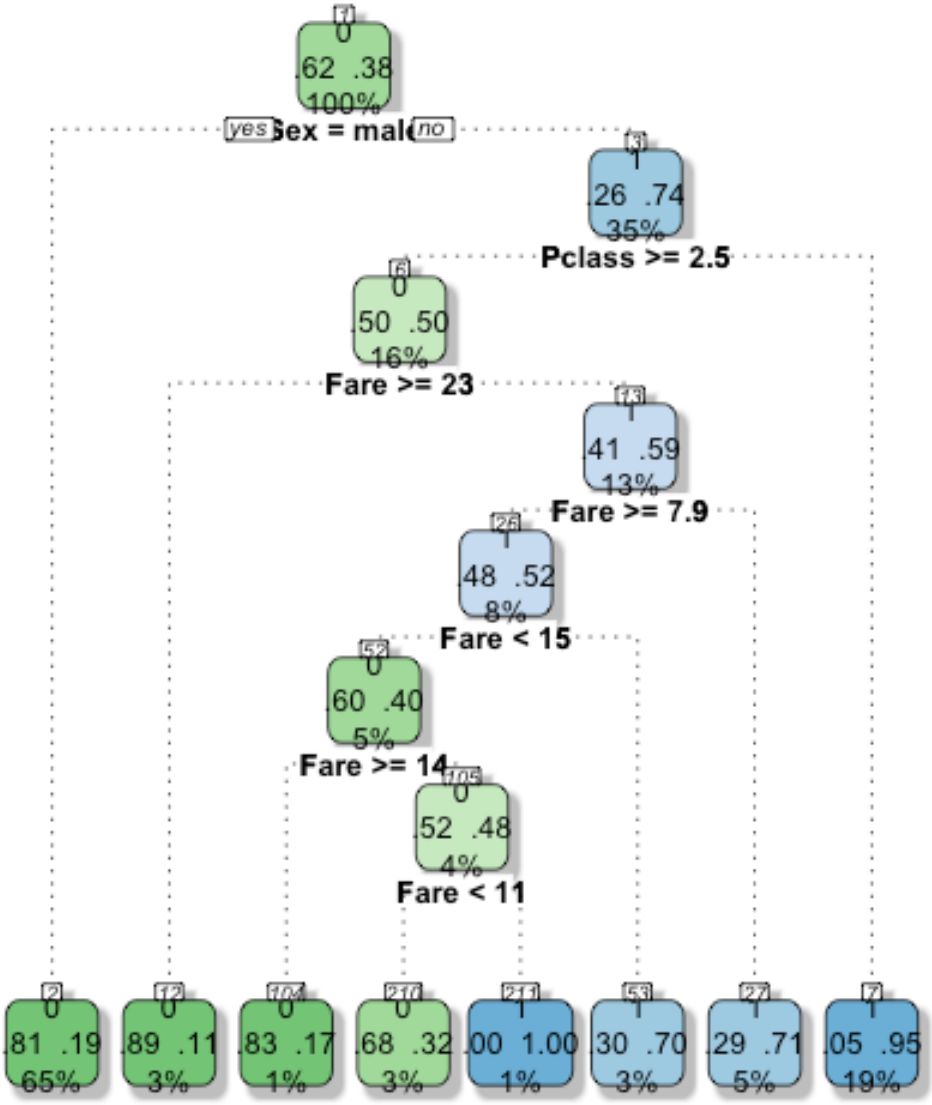
In order to get my results I used the *rpart* package in R; Rpart is an acronym for Recursive Partioning. The *rpart* library allows users to perform classifications and create simple, two-stage regression trees. The benefit of choosing this method to create the model is that it is easily digestible for readers of all backgrounds.

To aid in my visualization of the data, I loaded in the necessary components to use the FancyRpartPlot function.

After loading the necessary packages, I created my first model. I set the response variable to "Survived" (in which YES = 1 and NO = 0) and the explanatory variables to: Sex, Fare, Pclass, Age, and Fare. This is the resulting regression tree, which, when validated with the test data, accurately predicted passenger fate 75.77% of the time:



After examining that model and identifying ways to simplify, while improving, the model, I created a second model and tested it in comparison to the first model. It far outperformed the first model and when validated with the test data it was able to accurately predict the fate of passengers 77.512% of the time. This model was slightly different in that the explanatory variables were changed to: Sex, Pclass, Fare. I reduced the unnecessary variables and experienced a significant increase in the model's performance. Here is the second model:



Conclusions

Initial assumptions predicted Age, Sex, and Class as the major factors that determined survival; although the model built based on this assumption performed well, there another models that performed better. That model used Sex, Class, and Fare.

With both models, and especially the second one, the fate of a passenger can accurately be predicted with the necessary data.

Code:

```
#titanic project
getwd()
setwd("/Users/JandGComputerHome/Desktop/Jaleel Folder/titanic project")
#objectives:
#perform exploratory analysis on data with visualizations
#build a predictive model from scratch
#execute variable selection and creation to strengthen the predictive accuracy of your model
#communicate and present technical work

#importance of project:
#predictive modeling --> future of data science
#machine learning can take free response question return multiple choice answer
#with machine learning: highly targeted product recommendations, self-driving cars, and much more

#schema for the data:
#PassengerId = unique ID for each passenger
#Survived: 0 = Died, 1 = Survived
#Pclass: a proxy for passenger class (1 is the highest and 3 is the lowest)
#SibSp: sum total of the number of siblings or spouses aboard with passenger
#Fare: ticket price
#Embarked: Port departed from (CherBourg, Queenstown, Southhampton)

#I will load in both datasets
testdata = read.csv("test.csv")
traindata = read.csv("train.csv")

#find the missing age values in the train dataset:
sum(is.na(traindata$Age))

#find the average age of all passengers in the train dataset
mean(traindata$Age, na.rm = T) #had to remove the NA values to perform the maths function (only works on
numeric values or boolean values that have a corresponding numeric)


#EXPLORE THE DATA USING VISUALIZATION:
#the first tasks is to get a better understanding of the datasets
#this will aid in creating a predictive model

#create a visualizations (for train and test datasets) to show distribution of fare prices, age, sex, or family size
among the datasets?
hist(traindata$Fare, main = "Distribution of Fare Price for the Titanic", xlab = "Fare Price", ylab = "Number of
Passengers")
hist(testdata$Fare, main = "Distribution of Fare Price for the Titanic", xlab = "Fare Price", ylab = "Number of
Passengers")
#the overwhelming majority of passengers paid less than 100 currency units for their fare
hist(traindata$Age, xlab = "Age of Passengers", ylab = "Number of Passengers", main = "Age Distribution of the
Titanic Train Data")
#there are a decent amount of passengers under 20, majority of passengers are 20-40, some 40-60, very few
60-80
hist(testdata$Age, xlab = "Age of Passengers", ylab = "Number of Passengers", main = "Age Distribution of the
Titanic Test Data")
```

```
#incredible amount of passengers between 20-30, majority of passengers between 20-40, some under 20 and about the same amount between 40-60, very few 60-80
help("barplot")
barplot(prop.table(table(traindata$Sex)), main = "Sex Distribution of Passengers", xlab = "Sex", ylab = "Percent of Passengers")
#overwhelming male at ~65%
barplot(prop.table(table(testdata$Sex)), main = "Sex Distribution of Passengers", xlab = "Sex", ylab = "Percent of Passengers")
#about the same as the train data, but slightly higher female percentage and slightly less male percentage
hist(testdata$SibSp, main = "Number of Spouses and Siblings for Passengers", xlab = "Family Sizes", ylab = "Number of Passengers")
#nearly everyone only had 0 family members, very few had 1 - 4, almost none had 4-6, some passengers had 7 family members
hist(traindata$SibSp, main = "Number of Spouses and Siblings for Passengers", xlab = "Family Sizes", ylab = "Number of Passengers")
#very similar to the other chart but slightly more representation of family sizes 2-4

#I believe, based on social norms of the time period, that the strongest deciding factors for survival where: age, then sex, and lastly pclass
  #the groups that survived were females of high to mid Pclasses, or young females of low class, and adolescent males and some high class adult males

#Visualizing the data showed me an overview of the demographics for the passengers onboard the Titanic
  #it also helped me consider which explanatory variables would be best for the modeling

#BUILD A MODEL:
library('rpart')
names(traindata)
glm(Survived ~ Pclass + Sex + Age + SibSp + Fare, data = traindata, family = "binomial")
train_model = rpart( Survived ~ Pclass + Sex + Age + SibSp + Fare, data = traindata, method = "class", control = rpart.control(minsplit = 10))
plot(train_model, margin = .05, main = "Model to Predict Titanic Survival")
text(train_model)
fancyRpartPlot(train_model)
train_model
#very comprehensive model, I think it is a very good one
  #I predict that female passengers will have around a 73% chance of survival, opposed to ~18% for males
    #males younger than 6.5 had a 66% chance of survival
    #around 93-95% chance of survival for females of a Pclass lower than 2.5
      #for low class females younger than 38.5 a survival rate of around 50-53%
    #males older than 6.5 and with a Pclass lower than 1.5, a survival rate of around 33-35%

help(predict)

names(traindata)
train_subset = traindata[30:871, ]
subset = traindata[1:20, -2]
cv_model = rpart( Survived ~ Pclass + Sex + Age + SibSp + Fare, data = train_subset, method = "class", control = rpart.control(minsplit = 10))

predictions = predict(train_model, traindata, type = "class")
predictions = as.vector(predictions)
predictions
```

```
#cross-validate
cv_predictions1 = predict(cv_model, subset, type = "class")
cv_predictions1 = as.vector(cv_predictions1)
cv_predictions1

subset$predictedSurived <- NA
subset$predictedSurived <- predictions1

prop.table(table(traindata$Survived[1:20] == subset$predictedSurived))
#this shows that the model is right 80% of the time when cross-validated with this subset

FirstSubmission = data.frame(PassengerId = testdata$PassengerId, Survived = predictions)
FirstSubmission
write.csv(FirstSubmission, file = "FirstPredictionSubmission.csv", row.names = F)
#to increase my score I added Fare and SibSp to the model: score of .77512
printcp(train_model)

traindata$FamilySize <- NA
traindata$FamilySize = traindata$SibSp + traindata$Parch

testdata$FamilySize <- NA
testdata$FamilySize = testdata$SibSp + testdata$Parch

glm(Survived ~ Pclass + Sex + Fare, data = traindata, family = "binomial")
train_model2 = rpart( Survived ~ Pclass + Sex + Fare, data = traindata, method = "class", control =
rpart.control(minsplit = 10))
plot(train_model2, margin = .05, main = "Model to Predict Titanic Survival")
text(train_model2)
train_model2

predictions2 = predict(train_model2, testdata, type = "class")
predictions2 = as.vector(predictions2)
predictions2

LastSubmission = data.frame(PassengerId = testdata$PassengerId, Survived = predictions2)
LastSubmission
write.csv(LastSubmission, file = "LastPredictionSubmission.csv", row.names = F)
#to increase my score I added simplified the model, but it returned the same score as the last model: .77512
printcp(train_model)

library(rpart)
install.packages("RColorBrewer")
library(RColorBrewer)
install.packages("rpart.plot")
library(rpart.plot)
install.packages("rattle")
library(rattle)
fancyRpartPlot(train_model2)


#trying a new model
traindata$FamilySize <- NA
```

```
traindata$FamilySize = traindata$SibSp + traindata$Parch

testdata$FamilySize <- NA
testdata$FamilySize = testdata$SibSp + testdata$Parch

thirdModel = rpart(Survived ~ Sex + Fare + Pclass + FamilySize,data = traindata, method = "class", control =
rpart.control(minsplit = 10))
plot(thirdModel, margin = .05, main = "Model to Predict Titanic Survival")
text(thirdModel)
thirdModel
fancyRpartPlot(thirdModel)

cv_model2 = rpart( Survived ~ Sex + Fare + Pclass, data = train_subset, method = "class", control =
rpart.control(minsplit = 10))

cv_predictions2 = predict(cv_model2, subset, type = "class")
cv_predictions2 = as.vector(cv_predictions2)
cv_predictions2

subset$predictedSurived2 <- NA
subset$predictedSurived2 <- cv_predictions2

prop.table(table(traindata$Survived[1:20] == subset$predictedSurived2))
#still 80% correct predictions when cross-validate with the same small subset

predictions3 = predict(thirdModel, testdata, type = "class")
predictions3 = as.vector(predictions3)
predictions3

NextLastSubmission = data.frame(PassengerId = testdata$PassengerId, Survived = predictions3)
LastSubmission
write.csv(NextLastSubmission, file = "NextLastPredictionSubmission.csv", row.names = F)
#to increase my score I simplified the model, but it returned the same score as the last model: .77512

summary(traindata)
```