getwd() # Checking to see if we are in the correct working directory where our files are saved

library(dplyr) # loading packages for analysis

library(rpart) # loading packages for analysis

library(ggplot2) # loading packages for analysis

library(caTools) # loading packages for analysis

library(zoo) # loading the package zoo

library(gridExtra) # loading a package, which allows multiple ggplot2 sibe by side

library(randomForest) # Loading Random Forest Pack

# Reading in the files

gender\_submission <- read.csv("gender\_submission.csv") # reading in the file gender submission

test <- read.csv("test.csv") # reading in the file test

train <- read.csv("train.csv") # reading in the file train

View(train)

# Examining the train dataset

str(train) # looking at the structure of the data.frame object

dim(train) # looking at the dimensions of the data.frame object

# Combining the test and train dataset

train$Istrainset <- TRUE

test$Istrainset <- FALSE

head(train)

head(test)

test$Survived <- NA

head(test)

names(test)

names(train)

titanic.full <- rbind(train, test)

titanic.full[titanic.full$Embarked=='', "Embarked"] <- 'S'

table(titanic.full$Embarked)

#Assessing the Distribution of Age

age\_plot <- ggplot(train, aes(x=Age)) +

geom\_histogram(aes(y=..density..),

binwidth=2,

colour="black", fill="white") +

geom\_density(alpha=.2, fill="red") +

ggtitle("Age Distribution for the full Titanic Dataset with Mean (Dottet) and Median (Line) indicated")

age\_plot + geom\_vline(xintercept = 29.88114, linetype="dotted",

color = "blue", size=1)

# Calculating Average Ages per Class to find the most appropriate replacement for Age per Group (Mean or Median for all)

first\_class\_subset <- titanic.full[which(titanic.full$Pclass == "1"), ] #subsetting for 1st class

second\_class\_subset <- titanic.full[which(titanic.full$Pclass == "2"), ] #subsetting for 2nd class

third\_class\_subset <- titanic.full[which(titanic.full$Pclass == "3"), ] #subsetting for 3rd class

# Calculating the mean for each class subset

Avg\_age\_first <- mean(na.omit(first\_class\_subset$Age)) # Calculating Averages for the various classes

Avg\_age\_second <- mean(na.omit(second\_class\_subset$Age)) # Calculating Averages for the various classes

Avg\_age\_third <- mean(na.omit(third\_class\_subset$Age)) # Calculating Averages for the various classes

# Calculating the median for each class subset

Median\_age\_first <- median(na.omit(first\_class\_subset$Age)) # Calculating Median for the various classes

Median\_age\_second <- median(na.omit(second\_class\_subset$Age)) # Calculating Median for the various classes

Median\_age\_third <- median(na.omit(third\_class\_subset$Age)) # Calculating Median for the various classes

## It appears, that the different subsets have different age groups! Let us substitute the missing NA Vals for the means per class!

x <- is.na(titanic.full) #Checking to see how many NAs are in the first class subset, this can be replicated per subset by adjusting the code

summary(x)

first\_class\_subset[is.na(first\_class\_subset$Age), "Age"]<-"38.2334" # Filling the NA values in the first class subset with the mean of the subset

second\_class\_subset[is.na(second\_class\_subset$Age), "Age"]<-"29.8776" # Filling the NA values in the first class subset with the mean of the subset

third\_class\_subset[is.na(third\_class\_subset$Age), "Age"]<-"25.1406" # Filling the NA values in the first class subset with the mean of the subset

## Combining the subsets together again as the new titanic.full dataset

titanic.clean <- rbind(first\_class\_subset, second\_class\_subset, third\_class\_subset)

titanic.clean[is.na(titanic.full$Fare), "Fare"] <- "33.29548"

x <- is.na(titanic.clean)

summary(x)

# Clean empty fares and replace with mean

a <- na.omit(titanic.clean$Fare)

a <- as.numeric(a)

mean(a)

titanic.clean[is.na(titanic.clean$Fare), "Fare"] <- "33.3149"

table(is.na(titanic.clean$Fare))

# Casting certain variables into other classes, as needed

as.factor(titanic.clean$Pclass) -> titanic.clean$Pclass

as.factor(titanic.clean$Sex) -> titanic.clean$Sex

as.factor(titanic.clean$Embarked) -> titanic.clean$Embarked

str(titanic.clean)

# Could we use Bootstrapping to increase the accuracy of our model?

# Splitting Dataset

train <- titanic.clean[titanic.clean$Istrainset==TRUE, ]

test <- titanic.clean[titanic.clean$Istrainset== FALSE, ]

str(train$Istrainset)

# Model 1: Using a Random Forest Model to Predict the Outcome

survived.equation <- "Survived ~ Pclass + Age + SibSp + Parch + Fare + Embarked"

survived.formula <- as.formula(survived.equation)

install.packages("randomForest")

randomForest(formula = survived.formula, data = train, ntree = 500, mtry = 3, nodesizes = 0.01 \* nrow(test)) -> titanic.model

features.equation <- "Pclass + Age + SibSp + Parch + Fare + Embarked"

Survived <- predict(titanic.model, newdata = test)

Survived <- round(Survived, digits = 0)

head(Survived)

PassengerID <- test$PassengerId

as.data.frame(PassengerID) -> output.df

output.df$Survived <- Survived

length(Survived)

tail(output.df)

write.csv(output.df, file = "Globe\_Data\_Science\_Team", row.names = FALSE)

# Using a Multiple Linear Regression Model to Predict Survival Rates

mod\_regress <- lm(Survived ~ Pclass + Sex + Age + SibSp + Parch + Embarked, data = train) #Setting up the initial model

summary(mod\_regress)

improved.model <- step(mod\_regress, direction = "backward") # Using step-wise backward regression to optimize our model

summary(improved.model) #looking at the output of the regression analysis

predict(improved.model, test) -> result\_regress # predicting the survival rate on the test dataset

round(result\_regress, digits = 0) -> result\_regress

head(result\_regress)

output <- cbind(PassengerID = test$PassengerId, Survived = result\_regress)

output <- as.data.frame(output)

output <- sort(output$PassengerID)

head(output)

str(output)

dim(output)

write.csv(output, file = "Submission with Multiple Regression", row.names = F)

cbind(Actual = train\_clean$Survived, Predicted = result\_regress) -> Final\_Data # binding the variable Survived of the original dataset and predicted from test into one data.frame

as.data.frame(Final\_Data, check.rows= FALSE)-> Final\_Data # transforming the matrix to a data.frame

head(Final\_Data)

# Finding theb RMSE (Root Mean Squared Error)

(Final\_Data$Actual- Final\_Data$Predicted) -> error

cbind(Final\_Data, error) -> Final\_Data

head(Final\_Data)

tail(Final\_Data)

summary(Final\_Data)

Final\_Data <- na.omit(Final\_Data)

RMSE <- sqrt(mean(Final\_Data$error^2))

RMSE

plot(Final\_Data$error)

# Creating Subsets of the test data

first\_class\_subset <- train[which(train$Pclass == "1"), ] #subsetting for 1st class

second\_class\_subset <- train[which(train$Pclass == "2"), ] #subsetting for 2nd class

third\_class\_subset <- train[which(train$Pclass == "3"), ] #subsetting for 3rd class

survived\_subset <- train[which(train$Survived == "1"), ] #subsetting for survived

died\_subset <- train[which(train$Survived == "0"), ] #subsetting for dies

male\_passengers <- train[which(train$Sex == "male"), ] #subsetting for gender = male

female\_passengers <- train[which(train$Sex == "female"), ] #subsetting for gender = female

embarked\_ <- train[which(train$Embarked == ""), ] #subsetting for embarkement location

embarked\_C <- train[which(train$Embarked == "C"), ] #subsetting for embarkement location

embarked\_Q <- train[which(train$Embarked == "Q"), ] #subsetting for embarkement location

embarked\_S <- train[which(train$Embarked == "S"), ] #subsetting for embarkement location

Worst\_Surv\_Chance <- train[which(train$Pclass == "3"), ]

Worst\_Surv\_Chance <- Worst\_Surv\_Chance[which(Worst\_Surv\_Chance$Sex == "male"), ]

Worst\_Surv\_Chance <- Worst\_Surv\_Chance[which(Worst\_Surv\_Chance$Embarked == "S"), ]

Worst\_Surv\_Chance <- Worst\_Surv\_Chance[which(Worst\_Surv\_Chance$Age < "18"), ]

Best\_Surv\_Chance <- train[which(train$Pclass == "1"), ]

Best\_Surv\_Chance <- Best\_Surv\_Chance[which(Best\_Surv\_Chance$Sex == "female"), ]

Best\_Surv\_Chance <- Best\_Surv\_Chance[which(Best\_Surv\_Chance$Embarked == "C"), ]

# Plotting the Subsets

par(mfrow=c(1, 3)) #specifies the parameter on the plotting tool in R (how many rows and collumns)

hist(Worst\_Surv\_Chance$Survived, breaks = 2, main = "Survival of passengers under the worst conditions",

xlab = "Survival", ylim = c(0, 50), sub = "(0 = Died) (1 = Survived)")

hist(Best\_Surv\_Chance$Survived, breaks = 2, main = "Survival of passengers under the best conditions",

xlab = "Survival", ylim = c(0, 50), sub = "(0 = Died) (1 = Survived)")

hist(train$Survived, breaks = 2, main = "Survival of passengers under normal conditions",

xlab = "Survival", ylim = c(0,450), sub = "(0 = Died) (1 = Survived)")

# Barplot

bp<- ggplot(df, aes(x="", y=value, fill=group))+

geom\_bar(width = 1, stat = "identity")

bp

# Histograms on Survival Rates

par(mfrow =c(2,1))

hist(male\_passengers$Survived, breaks = 2, main = "Survival of male passengers", xlab = "Survival", ylim = c(0, 350), sub = "(0 = Died) (1 = Survived)")

hist(female\_passengers$Survived, breaks = 2, main = "Survival of female passengers", xlab = "Survival", ylim = c(0, 350), sub = "(0 = Died) (1 = Survived)")

# Histograms on Survival Rates of the different classes

par(mfrow =c(1,3))

hist(first\_class\_subset$Survived, breaks = 2, main = "Survival of 1st class passengers", xlab = "Survival", ylim = c(0, 300), sub = "(0-0.5 = Died) (0.5-1.0 = Survived)")

hist(second\_class\_subset$Survived, breaks = 2, main = "Survival of 2nd class passengers", xlab = "Survival",ylim = c(0, 300),sub = "(0-0.5 = Died) (0.5-1.0 = Survived)")

hist(third\_class\_subset$Survived, breaks = 2, main = "Survival of 3rd class passengers", xlab = "Survival", ylim = c(0, 300),sub = "(0-0.5 = Died) (0.5-1.0 = Survived)")

# Histogram of Survival Rates based on the departing harbour

par(mfrow =c(1,3)) # Port of Embarkation (C = Cherbourg; Q = Queenstown; S = Southampton)

hist(embarked\_C$Survived, breaks = 2, main = "Survival of passengers embarked in Cherbourg", xlab = "Survival", sub = "(0 = Died) (1 = Survived)", ylim = c(0, 350)) # Interesting that the survival rate in C is much higher (could this be a richer port?)

hist(embarked\_Q$Survived, breaks = 2, main = "Survival of passengers embarked in Queenstown", xlab = "Survival",sub = "(0 = Died) (1 = Survived)",ylim = c(0, 350))

hist(embarked\_S$Survived, breaks = 2, main = "Survival of passengers embarked in Southampton", xlab = "Survival",sub = "(0 = Died) (1 = Survived)",ylim = c(0, 350))

# Histogram of Class Distribution based on the departing harbour

par(mfrow =c(1,3)) # Port of Embarkation (C = Cherbourg; Q = Queenstown; S = Southampton)

hist(embarked\_C$Pclass, breaks = 3, main = "Class Distribution of passengers embarked in Cherbourg", xlab = "Classes", sub = "(1 = First Class) (2 = Second Class) (3 = Third Class)", ylim = c(0, 350)) # Interesting that the survival rate in C is much higher (could this be a richer port?)

hist(embarked\_Q$Pclass, breaks = 3, main = "Class Distribution of passengers embarked in Queenstown", xlab = "Classes", sub = "(1 = First Class) (2 = Second Class) (3 = Third Class)", ylim = c(0, 350))

hist(embarked\_S$Pclass, breaks = 3, main = "Class Distribution of passengers embarked in Southampton" , xlab = "Classes", sub = "(1 = First Class) (2 = Second Class) (3 = Third Class)", ylim = c(0, 350))

# Histograms

# Creating an age distribution histogram of all passengers

ggplot(train, aes(x=Age)) +

geom\_histogram(aes(y=..density..),

binwidth=1,

colour="black", fill="white") +

geom\_density(alpha=.2, fill="red")

# Creating an age distribution histogram of all passengers who survived

dev.off()

ggplot(survived\_subset, aes(x=Age)) +

geom\_histogram(aes(y=..density..),

binwidth=1,

colour="black", fill="white") +

geom\_density(alpha=.2, fill="red")

# Creating an age distribution histogram of all passengers who died

ggplot(died\_subset, aes(x=Age)) +

geom\_histogram(aes(y=..density..),

binwidth=1,

colour="black", fill="white") +

geom\_density(alpha=.2, fill="red")

# comparring the three Charts

ageall<- ggplot(train, aes(x=Age)) +

geom\_histogram(aes(y=..density..),

binwidth=1,

colour="black", fill="white") +

geom\_density(alpha=.2, fill="red")+

ggtitle("Age distribution of all people")

agesurv <- ggplot(survived\_subset, aes(x=Age)) +

geom\_histogram(aes(y=..density..),

binwidth=1,

colour="black", fill="white") +

geom\_density(alpha=.2, fill="red")+

ggtitle("Age distribution of people who survived")

agedead <- ggplot(died\_subset, aes(x=Age)) +

geom\_histogram(aes(y=..density..),

binwidth=1,

colour="black", fill="white") +

geom\_density(alpha=.2, fill="red")+

ggtitle("Age distribution of people who died")

# All but two lifeboats were situated on the Boat Deck, the highest deck of Titanic.

# Second Officer Charles Lightoller working on the port side = (The port side is the side of the vessel which is to the left of an observer facing the bow, that is, facing forward towards the direction the vehicle is heading when underway, and the starboard is to the right of such an observer)

#Lightoller lowered lifeboats with empty seats if there were not any women and children waiting to board,

#while Murdoch only allowed a limited number of men to board if all the nearby women and children had already embarked.

#This had a significant effect on the survival rates of the men aboard Titanic, whose chances of survival came to depend on which side of the ship they tried to find lifeboat seats.

# What happenden during the night of the desaster (Watch the movie to reconstruct) -> need variable that classifies which side the cabins were on! (Can we reference the ticket numbers to the cabins? )

# No general alarm sounded, many of the lower deck passengers are unaware of lifeboats being lowered to the water

# timeline of desaster (watch at twice the speed ;) ): https://www.youtube.com/watch?v=rs9w5bgtJC8

# ticket\_number -> location on ship (left side is worst, right best) (do histogramns on survial rates on ship based on the side subsets)

# Lifeboats were at the top desk, which classes had the quickest access (if is is according to 1,2,3 keep them, else transform and use distance to lifeboat instead of class)

# Link to the deck layout of the ship: https://www.encyclopedia-titanica.org/titanic-deckplans/f-deck.html

# Other Analysis

# finding the most expensive ticket price and who bought it

max\_ticket\_price <- train[which(train$Fare == "512.3292"), ]

max\_ticket\_price <- as.numeric(train$Fare) # transforming the variable to numeric to calculate the max ticket price

class((max\_ticket\_price)) # checking the class

summary((max\_ticket\_price))