Titanic Project

Daphney

2024-07-26

Titanic Project

Objective: This project involves building a predictive model to estimate the likelihood of survival for passengers on the Titanic using various factors such as passenger class, gender, age, and number of siblings/spouses aboard.

```
#Read in the data
train_data <- read.csv("train.csv", header = TRUE)
test_data <- read.csv("test.csv", header = TRUE)

#Load Packages
library(caret)

## Loading required package: ggplot2

## Loading required package: lattice

library(car)

## Loading required package: carData

library(MASS)
library(dplyr)

##
## Attaching package: 'dplyr'

## The following object is masked from 'package:MASS':
##
##     select

## The following object is masked from 'package:car':
##
##     recode

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

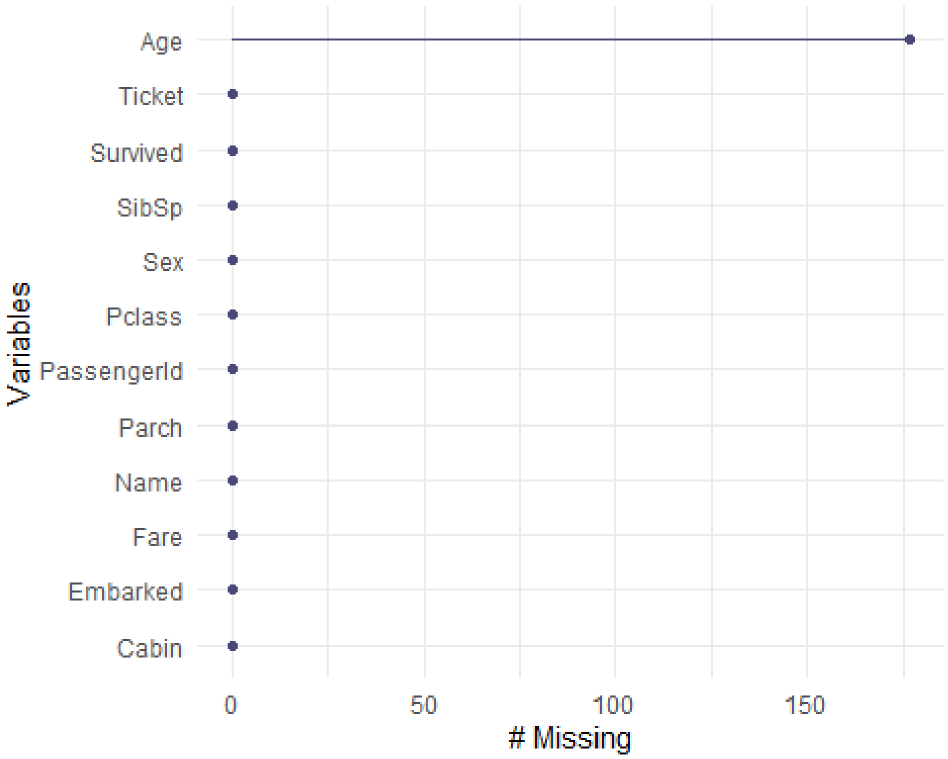
```
head(train_data, 3)

##   PassengerId Survived Pclass
## 1           1        0      3
## 2           2        1      1
## 3           3        1      3
##                                                  Name    Sex Age SibSp
Parch
## 1                             Braund, Mr. Owen Harris   male  22     1
0
## 2 Cumings, Mrs. John Bradley (Florence Briggs Thayer) female  38     1
0
## 3                              Heikkinen, Miss. Laina female  26     0
0
##             Ticket    Fare Cabin Embarked
## 1        A/5 21171  7.2500              S
## 2         PC 17599 71.2833   C85        C
## 3 STON/O2. 3101282  7.9250              S
```

```
summary(train_data)

##   PassengerId       Survived          Pclass          Name
##  Min.   :  1.0   Min.   :0.0000   Min.   :1.000   Length:891
##  1st Qu.:223.5   1st Qu.:0.0000   1st Qu.:2.000   Class :character
##  Median :446.0   Median :0.0000   Median :3.000   Mode  :character
##  Mean   :446.0   Mean   :0.3838   Mean   :2.309
##  3rd Qu.:668.5   3rd Qu.:1.0000   3rd Qu.:3.000
##  Max.   :891.0   Max.   :1.0000   Max.   :3.000
##
##      Sex                 Age            SibSp           Parch
##  Length:891         Min.   : 0.42   Min.   :0.000   Min.   :0.0000
##  Class :character   1st Qu.:20.12   1st Qu.:0.000   1st Qu.:0.0000
##  Mode  :character   Median :28.00   Median :0.000   Median :0.0000
##                     Mean   :29.70   Mean   :0.523   Mean   :0.3816
##                     3rd Qu.:38.00   3rd Qu.:1.000   3rd Qu.:0.0000
##                     Max.   :80.00   Max.   :8.000   Max.   :6.0000
##                     NA's   :177
##     Ticket               Fare           Cabin             Embarked
##  Length:891         Min.   :  0.00   Length:891         Length:891
##  Class :character   1st Qu.:  7.91   Class :character   Class :character
##  Mode  :character   Median : 14.45   Mode  :character   Mode  :character
##                     Mean   : 32.20
##                     3rd Qu.: 31.00
##                     Max.   :512.33
##
```

Pre-processing

```
#check for NA values
any(is.na(train_data))

## [1] TRUE
```

```
sum(is.na(train_data))

## [1] 177

colSums(is.na(train_data))

## PassengerId    Survived      Pclass        Name         Sex         Age
##           0           0           0           0           0         177
##       SibSp       Parch      Ticket        Fare       Cabin    Embarked
##           0           0           0           0           0           0

#View them
library(naniar)

## Warning: package 'naniar' was built under R version 4.4.1

gg_miss_var(train_data)
```



```
#Check for blanks and spaces
any(train_data == "")

## [1] TRUE

any(train_data == " ")

## [1] NA

sum(which(train_data$Embarked == ""))
```

```
## [1] 892
```

```
sum(which(train_data$Cabin == ""))
```

```
## [1] 304484
```

```
#Convert blanks to NA
train_data[train_data == ""] <- NA
colSums(is.na(train_data))
```

```
## PassengerId    Survived      Pclass        Name         Sex         Age
##           0           0           0           0           0         177
##       SibSp       Parch      Ticket        Fare       Cabin    Embarked
##           0           0           0           0         687           2
```

```
#Drop unecessary columns
train_data <- train_data[-c(9,11)]
head(train_data,2)
```

```
##   PassengerId Survived Pclass
## 1           1        0      3
## 2           2        1      1
##                                                  Name    Sex Age SibSp
Parch
## 1                             Braund, Mr. Owen Harris   male  22     1
0
## 2 Cumings, Mrs. John Bradley (Florence Briggs Thayer) female  38     1
0
##      Fare Embarked
## 1  7.2500        S
## 2 71.2833        C
```

```
#Imputation using median
train_data$Age[is.na(train_data$Age)] <- median(train_data$Age, na.rm = TRUE)
train_data$Embarked[is.na(train_data$Embarked)] <-
median(train_data$Embarked, na.rm = TRUE)
any(is.na(train_data))
```

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

```
sum(is.na(train_data))
```

```
## [1] 0
```

Thoughts:

Significant amount of blanks in the cabin column, so I decided to drop this column from my analysis.*

Small amount of NA in Embarked column, and moderate amount in age column. Used imputation using the medians to populate the missing values*

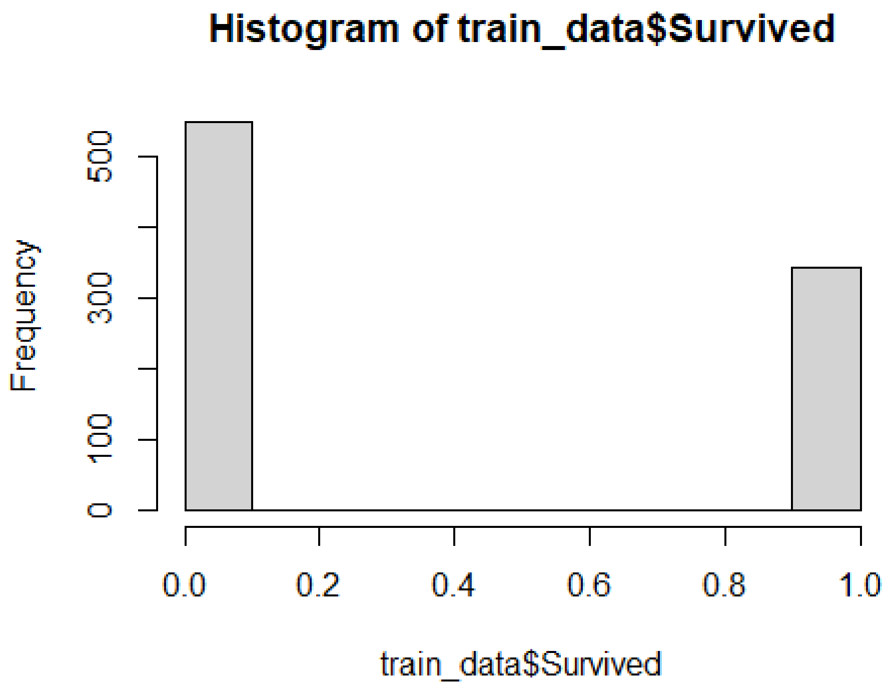
```
#Use last name instead of full name
train_data$Name <- sub(",.*", "", train_data$Name)
colnames(train_data)[colnames(train_data) == "Name"] <- "Last_Name"

unique_last_name <- unique(train_data$Last_Name)
length(unique_last_name)
```

```
## [1] 667
```

```
#Convert data type. Categorical to numerical
train_data$Last_Name <- as.factor(train_data$Last_Name)
train_data$Sex <- as.factor(train_data$Sex)
train_data$Embarked <- as.factor(train_data$Embarked)
```

Exploratory Data Analysis (EDA)

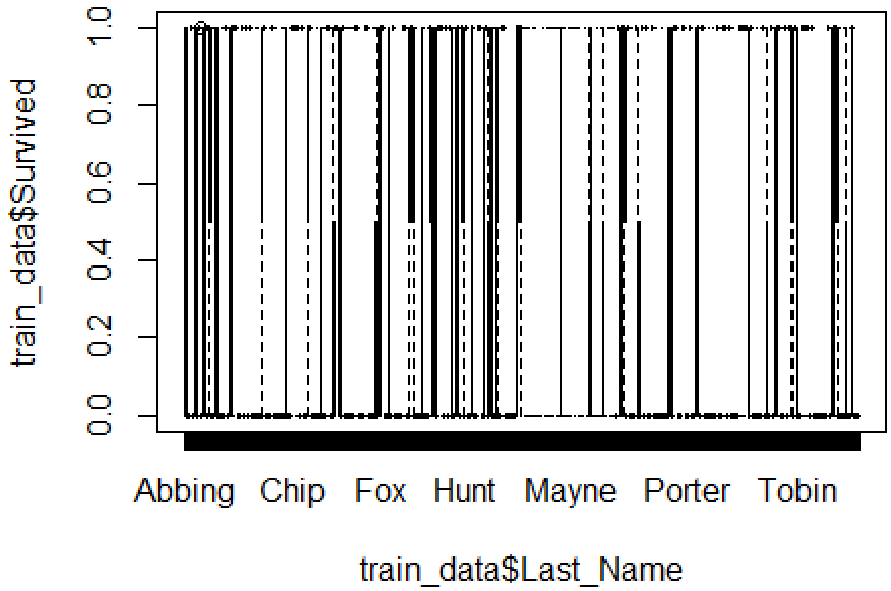
```
hist(train_data$Survived)
```



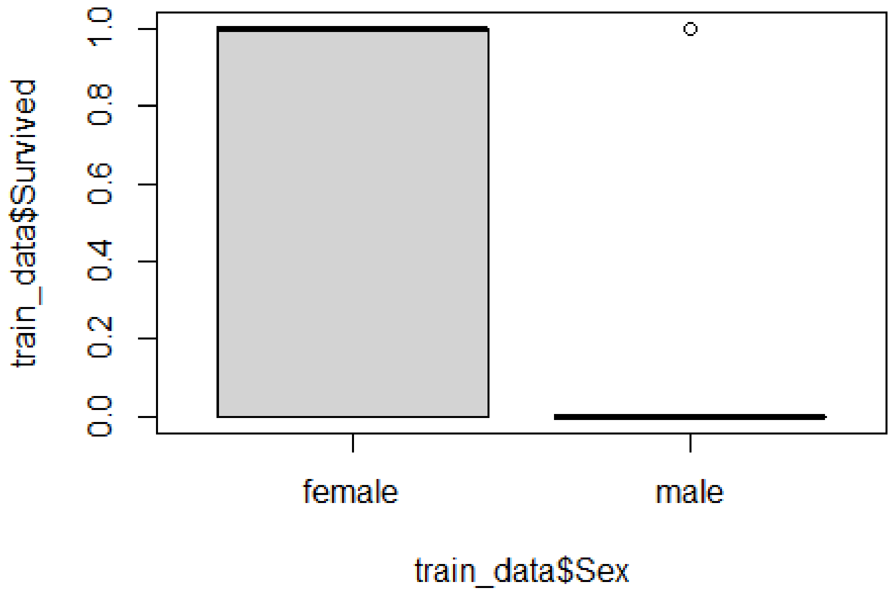
Interpretation:

Skewed distribution visualized in the dependent variable due to it being binary. Linear regression is not recommended for this analysis since the assumptions will be violated, however we can still gain some insight from it. Logistic regression recommended.

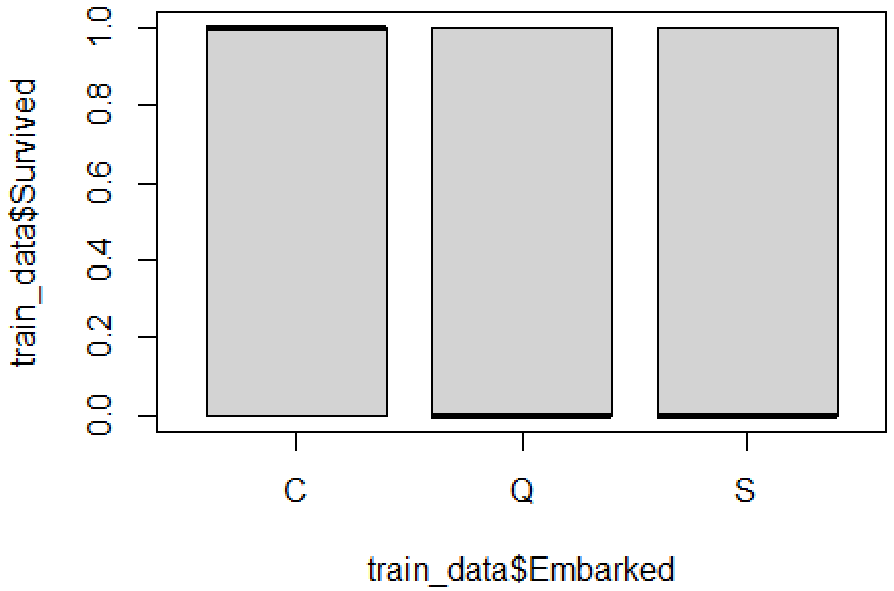
```
#boxplots of categorical data
boxplot(train_data$Survived ~ train_data$Last_Name)
```



```r
boxplot(train_data$Survived ~ train_data$Sex)
```



```r
boxplot(train_data$Survived ~ train_data$Embarked)
```



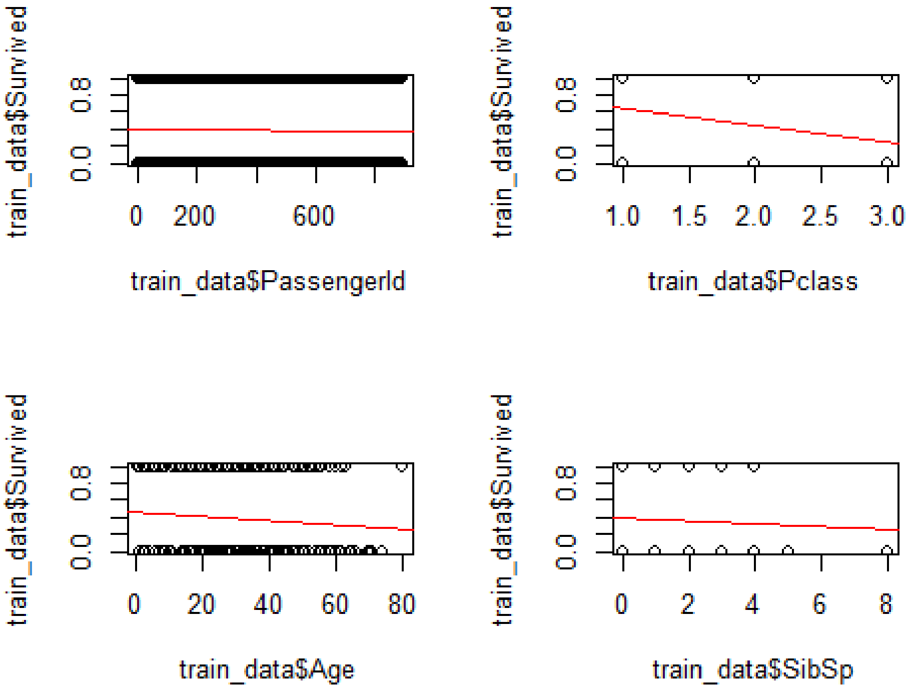
Thoughts:

The box plots for gender show a significant difference between the medians, indicating gender may be a key predictor for survival. It shows females had a much higher survival rate than males with no visible outliers. Males had a much lower rate of survival, however there are a few outliers visible.*

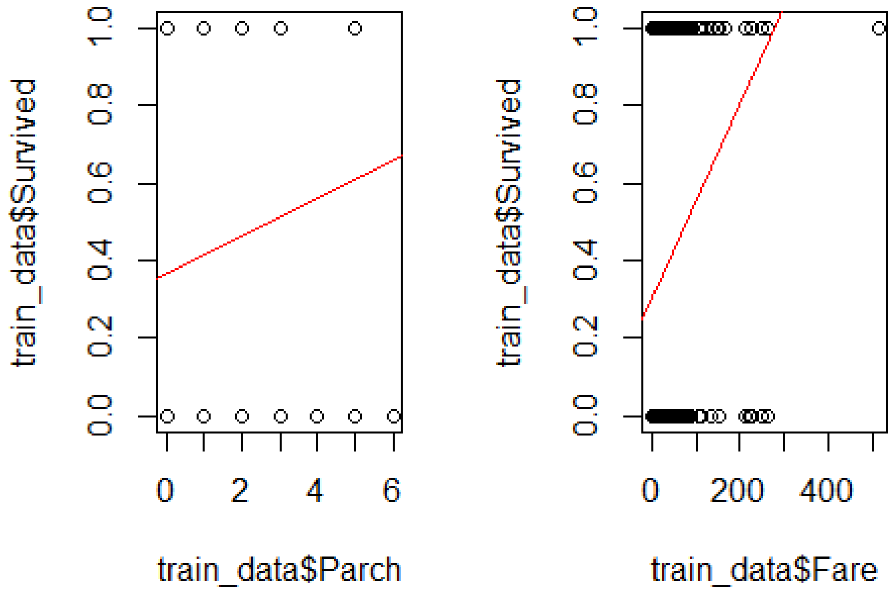
However, the box plot for the medians for Embarked appear to show no differences in medians, indicating embarkation location may not be a significant factor influencing survival.*

Limited interpretability for box plot of last names due to the large number of last names.*

```r
#Scatterplots of numerical data
par(mfrow=c(2,2))
plot(train_data$PassengerId, train_data$Survived)
abline(lm(Survived ~ PassengerId, data = train_data), col = "red")

plot(train_data$Pclass, train_data$Survived)
abline(lm(Survived ~ Pclass, data = train_data), col = "red")

plot(train_data$Age, train_data$Survived)
abline(lm(Survived ~ Age, data = train_data), col = "red")

plot(train_data$SibSp, train_data$Survived)
abline(lm(Survived ~ SibSp, data = train_data), col = "red")
```



```r
par(mfrow = c(1,2))
plot(train_data$Parch, train_data$Survived)
abline(lm(Survived ~ Parch, data = train_data), col = "red")

plot(train_data$Fare, train_data$Survived)
abline(lm(Survived ~ Fare, data = train_data), col = "red")
```



Thoughts:

Binary dependent variable do not show linear relationships well. Logistic regression indicated.

```
#Correlations
train_data <- train_data[-11]
cor(train_data[-c(4,5,10)])

##                PassengerId     Survived      Pclass         Age       SibSp
## PassengerId  1.000000000 -0.005006661 -0.03514399  0.03421211 -0.05752683
## Survived    -0.005006661  1.000000000 -0.33848104 -0.06491042 -0.03532250
## Pclass      -0.035143994 -0.338481036  1.00000000 -0.33989833  0.08308136
## Age          0.034212112 -0.064910420 -0.33989833  1.00000000 -0.23329633
## SibSp       -0.057526834 -0.035322499  0.08308136 -0.23329633  1.00000000
## Parch       -0.001652012  0.081629407  0.01844267 -0.17248195  0.41483770
## Fare         0.012658219  0.257306522 -0.54949962  0.09668842  0.15965104
##                    Parch        Fare
## PassengerId -0.001652012  0.01265822
## Survived     0.081629407  0.25730652
## Pclass       0.018442671 -0.54949962
## Age         -0.172481954  0.09668842
## SibSp        0.414837699  0.15965104
## Parch        1.000000000  0.21622494
## Fare         0.216224945  1.00000000
```

Interpretation:

Moderately strong correlation between class and fare price. Will check for multicolinearity later on in my Analysis.

Model Building

```
#Linear regression model
linear_model <- lm(Survived ~ ., data = train_data)
#options(max.print = 10000)
options(max.print = 50)
summary(linear_model)

## 
## Call:
## lm(formula = Survived ~ ., data = train_data)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -0.5817  0.0000  0.0000  0.0000  0.9725 
## 
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)   
## (Intercept)                       1.360e+00  4.660e-01   2.918  0.00390 **
## PassengerId                      -1.110e-04  8.858e-05  -1.254  0.21134   
## Pclass                           -3.020e-01  1.152e-01  -2.621  0.00938 **
## Last_NameAbbott                   2.212e-01  4.125e-01   0.536  0.59234   
## Last_NameAbelson                  1.969e-01  4.348e-01   0.453  0.65109   
## Last_NameAdahl                   -1.263e-01  4.690e-01  -0.269  0.78790   
## Last_NameAdams                   -1.476e-01  4.692e-01  -0.315  0.75336   
## Last_NameAhlin                   -4.545e-01  4.754e-01  -0.956  0.34015   
## Last_NameAks                      4.904e-01  4.722e-01   1.039  0.30010   
## Last_NameAlbimona                 1.099e+00  5.005e-01   2.196  0.02913 * 
##  [ reached getOption("max.print") -- omitted 666 rows ]
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.33 on 215 degrees of freedom
## Multiple R-squared:  0.8889, Adjusted R-squared:  0.5401 
## F-statistic: 2.548 on 675 and 215 DF,  p-value: 6.174e-15
```

Impression:

Sex, Age, And Class were significant predictors. Also, certain Last names such as Abbott and Moubarek appear to be significant predictors of survival.

```
#Logistic regression model
logistic_model <- glm(Survived ~ ., data = train_data, family = "binomial")

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

summary(logistic_model)
```

```
##
## Call:
## glm(formula = Survived ~ ., family = "binomial", data = train_data)
##
## Coefficients:
##                                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)                    -4.540e+00  4.820e+04   0.000  0.99992
## PassengerId                    -2.844e-04  1.708e-03  -0.167  0.86772
## Pclass                         -3.867e+00  2.009e+00  -1.925  0.05428 .
## Last_NameAbbott                 2.027e+01  4.820e+04   0.000  0.99966
## Last_NameAbelson                1.903e+01  4.820e+04   0.000  0.99969
## Last_NameAdahl                 -1.202e+00  6.816e+04   0.000  0.99999
## Last_NameAdams                 -1.558e+00  6.816e+04   0.000  0.99998
## Last_NameAhlin                 -4.357e+00  6.816e+04   0.000  0.99995
## Last_NameAks                    3.811e+01  6.816e+04   0.001  0.99955
## Last_NameAlbimona               4.633e+01  6.816e+04   0.001  0.99946
##  [ reached getOption("max.print") -- omitted 666 rows ]
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1186.655  on 890  degrees of freedom
## Residual deviance:   91.647  on 215  degrees of freedom
## AIC: 1443.6
##
## Number of Fisher Scoring iterations: 21
```

Interpretation:

The most significant predictor was Sex. Additionally, Age, class and SibSp (number of siblings or spouses aboard) were also significant predictors of survival at an alpha of 0.05.

Model Evaluation

```
AIC(logistic_model)
```

```
## [1] 1443.647
```

AIC measures model quality and penalizes for more parameters in the model to discourage over fitting. It's only used for comparison with other models

- Null deviance measures the fit of the model with no predictors to give an idea of the baseline level of error for predictions from the model.
- Residual deviance measures the fit of the model with the predictors.

There is a big difference between the two numbers.The large reduction in deviance suggests that the model fits the data well.

Additional model building

```
#multiple logistic models
logistic_model2 <- glm(Survived ~ Last_Name + Pclass + Sex + Age + SibSp +
Fare, data = train_data, family = binomial)

logistic_model3 <- glm(Survived ~ Pclass + Sex + Age + SibSp + Fare +
Embarked, data = train_data, family = binomial)

logistic_model4 <- glm(Survived ~ Pclass + Sex + Age + SibSp, data =
train_data, family = binomial)

summary(logistic_model2)

##
## Call:
## glm(formula = Survived ~ Last_Name + Pclass + Sex + Age + SibSp +
##     Fare, family = binomial, data = train_data)
##
## Coefficients:
##                                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)                     -5.916e+00  4.820e+04   0.000  0.99990
## Last_NameAbbott                  1.957e+01  4.820e+04   0.000  0.99968
## Last_NameAbelson                 1.730e+01  4.820e+04   0.000  0.99971
## Last_NameAdahl                  -1.128e+00  6.816e+04   0.000  0.99999
## Last_NameAdams                  -1.517e+00  6.816e+04   0.000  0.99998
## Last_NameAhlin                  -4.049e+00  6.816e+04   0.000  0.99995
## Last_NameAks                     3.767e+01  6.816e+04   0.001  0.99956
## Last_NameAlbimona                4.345e+01  6.816e+04   0.001  0.99949
## Last_NameAlexander              -1.515e+00  6.816e+04   0.000  0.99998
## Last_NameAlhomaki               -2.081e+00  6.816e+04   0.000  0.99998
##  [ reached getOption("max.print") -- omitted 662 rows ]
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1186.655  on 890  degrees of freedom
## Residual deviance:   95.439  on 219  degrees of freedom
## AIC: 1439.4
##
## Number of Fisher Scoring iterations: 21

summary(logistic_model3)

##
## Call:
## glm(formula = Survived ~ Pclass + Sex + Age + SibSp + Fare +
##     Embarked, family = binomial, data = train_data)
##
## Coefficients:
```

```
##             Estimate Std. Error z value Pr(>|z|)    
## (Intercept)  5.238685   0.560174   9.352  < 2e-16 ***
## Pclass      -1.114277   0.142318  -7.829 4.90e-15 ***
## Sexmale     -2.696427   0.195408 -13.799  < 2e-16 ***
## Age         -0.038887   0.007808  -4.981 6.34e-07 ***
## SibSp       -0.346875   0.105802  -3.279  0.00104 ** 
## Fare         0.001593   0.002281   0.698  0.48511    
## EmbarkedQ   -0.035081   0.379696  -0.092  0.92639    
## EmbarkedS   -0.410511   0.236402  -1.736  0.08248 .  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
##     Null deviance: 1186.66  on 890  degrees of freedom
## Residual deviance:  786.12  on 883  degrees of freedom
## AIC: 802.12
## 
## Number of Fisher Scoring iterations: 5
```

```r
summary(logistic_model4)
```

```
## 
## Call:
## glm(formula = Survived ~ Pclass + Sex + Age + SibSp, family = binomial, 
##     data = train_data)
## 
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)    
## (Intercept)  5.177025   0.477338  10.846  < 2e-16 ***
## Pclass      -1.175654   0.120073  -9.791  < 2e-16 ***
## Sexmale     -2.739477   0.193984 -14.122  < 2e-16 ***
## Age         -0.039553   0.007761  -5.096 3.47e-07 ***
## SibSp       -0.354433   0.103392  -3.428 0.000608 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
##     Null deviance: 1186.66  on 890  degrees of freedom
## Residual deviance:  791.23  on 886  degrees of freedom
## AIC: 801.23
## 
## Number of Fisher Scoring iterations: 5
```

Model evaluation

```r
cat("AIC for Logistic full model:", AIC(logistic_model), "\n")
```

```
## AIC for Logistic full model: 1443.647
```

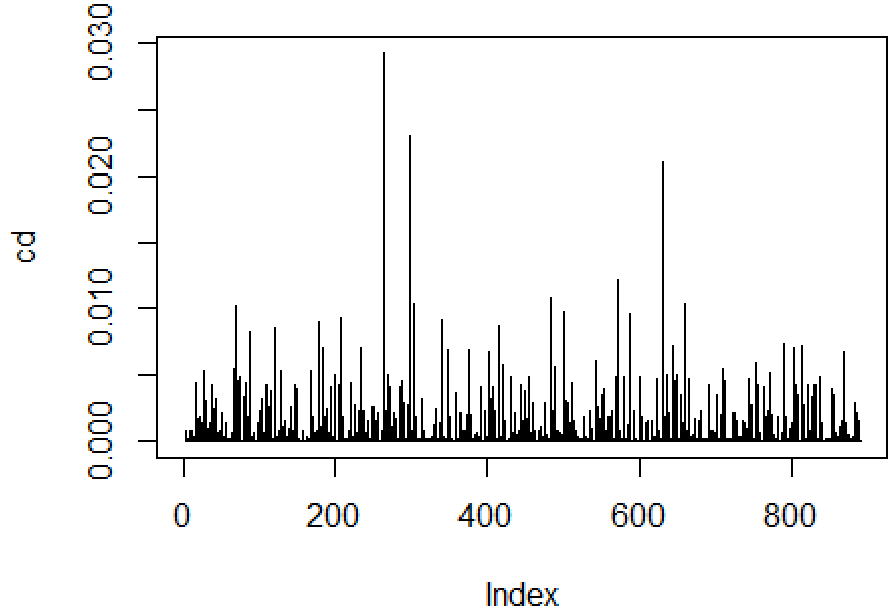
```r
cat("AIC for Logistic model2:", AIC(logistic_model2), "\n")
```

```
## AIC for Logistic model2: 1439.439

cat("AIC for Logistic model3:", AIC(logistic_model3), "\n")

## AIC for Logistic model3: 802.116

cat("AIC for Logistic model4:", AIC(logistic_model4), "\n")

## AIC for Logistic model4: 801.2303
```

Impression:

Model 4 is the best model because it has the lowest AIC value of 801.23. Model 3 is slightly worse with an AIC of 802.12.

```
#Outliers
cd <- cooks.distance(logistic_model4)
plot(cd, type = "h")
abline(h = 1, col = "red")
```



No significant outliers at threshold of 1.

```
#Multicolinearity
vif(logistic_model4)

##   Pclass      Sex      Age    SibSp
## 1.306536 1.142573 1.285274 1.123126
```

There is no multicollinearity observed among the predicting variables. The VIF values are all close to 1, which is ideal

Model Validation

```
##### Prepare the test data #####

head(test_data, 2)

##   PassengerId Pclass                             Name    Sex  Age SibSp
Parch
## 1         892      3                 Kelly, Mr. James   male 34.5     0
0
## 2         893      3 Wilkes, Mrs. James (Ellen Needs) female 47.0     1
0
##   Ticket   Fare Cabin Embarked
## 1 330911 7.8292              Q
## 2 363272 7.0000              S

#Check for missing values
sum(is.na(test_data))

## [1] 87

colSums(is.na(test_data))

## PassengerId      Pclass        Name         Sex         Age       SibSp
##           0           0           0           0          86           0
##       Parch      Ticket        Fare       Cabin    Embarked
##           0           0           1           0           0

#Check for blanks
any(test_data == "")

## [1] TRUE

any(train_data == " ")

## [1] FALSE

which(test_data == "")

##  [1] 3763 3764 3765 3766 3767 3768 3769 3770 3771 3772 3773 3774 3776 3778
3779
## [16] 3780 3781 3782 3783 3784 3785 3786 3788 3790 3792 3793 3794 3795 3796
3798
## [31] 3799 3800 3801 3802 3803 3805 3806 3808 3810 3812 3814 3815 3817 3818
3819
## [46] 3821 3823 3824 3825 3826
##  [ reached getOption("max.print") -- omitted 277 entries ]
```

```
#convert blanks to NA
test_data[test_data == ""] <- NA
any(test_data == "")

## [1] NA

colSums(is.na(test_data))

## PassengerId      Pclass        Name         Sex         Age       SibSp
##           0           0           0           0          86           0
##       Parch      Ticket        Fare       Cabin    Embarked
##           0           0           1         327           0

#Imputation
test_data$Age <- median(test_data$Age, na.rm = TRUE)
test_data$Fare <- median(test_data$Fare, na.rm = TRUE)
colSums(is.na(test_data))

## PassengerId      Pclass        Name         Sex         Age       SibSp
##           0           0           0           0           0           0
##       Parch      Ticket        Fare       Cabin    Embarked
##           0           0           0         327           0

#change column name and split it
test_data$Name <- sub(",.*", "", test_data$Name)
colnames(test_data)[colnames(test_data) == "Name"] <- "Last_Name"

#drop columns
test_data <- test_data[-c(8,10)]

#convert categorical
test_data$Last_Name <- as.factor(test_data$Last_Name)
test_data$Sex <- as.factor(test_data$Sex)
test_data$Embarked <- as.factor(test_data$Embarked)

#verify the structure is the same
str(test_data)

## 'data.frame':    418 obs. of  9 variables:
##  $ PassengerId: int  892 893 894 895 896 897 898 899 900 901 ...
##  $ Pclass     : int  3 3 2 3 3 3 3 2 3 3 ...
##  $ Last_Name  : Factor w/ 352 levels "Abbott","Abelseth",..: 176 344 235
349 154 316 74 49 4 91 ...
##  $ Sex        : Factor w/ 2 levels "female","male": 2 1 2 2 1 2 1 2 1 2
...
##  $ Age        : num  27 27 27 27 27 27 27 27 27 27 ...
##  $ SibSp      : int  0 1 0 0 1 0 0 1 0 2 ...
##  $ Parch      : int  0 0 0 0 1 0 0 1 0 0 ...
##  $ Fare       : num  14.5 14.5 14.5 14.5 14.5 ...
##  $ Embarked   : Factor w/ 3 levels "C","Q","S": 2 3 2 3 3 3 2 3 1 3 ...
```

```
str(train_data)

## 'data.frame':    891 obs. of  10 variables:
##  $ PassengerId: int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Survived   : int  0 1 1 1 0 0 0 0 1 1 ...
##  $ Pclass     : int  3 1 3 1 3 3 1 3 3 2 ...
##  $ Last_Name  : Factor w/ 667 levels "Abbing","Abbott",..: 74 137 256 203
12 414 383 468 297 431 ...
##  $ Sex        : Factor w/ 2 levels "female","male": 2 1 1 1 2 2 2 2 1 1
...
##  $ Age        : num  22 38 26 35 35 28 54 2 27 14 ...
##  $ SibSp      : int  1 1 0 1 0 0 0 3 0 1 ...
##  $ Parch      : int  0 0 0 0 0 0 0 1 2 0 ...
##  $ Fare       : num  7.25 71.28 7.92 53.1 8.05 ...
##  $ Embarked   : Factor w/ 3 levels "C","Q","S": 3 1 3 3 3 2 3 3 3 1 ...
```

###Prediction

```
#prediction on test data
test_data$Survival_prediction <- predict(logistic_model4, newdata =
test_data, type = "response")

#convert to binary
test_data$Survival_class_prediction <- ifelse(test_data$Survival_prediction >
0.5, 1, 0)
```

Survival predictions for the test data

###Validation

```
#Compare with actual results
actual_survival <- read.csv("gender_submission.csv", header = TRUE)


#combine the data
merge_data <- merge(test_data, actual_survival, by = "PassengerId")

#confusion matrix
matrix <- confusionMatrix(as.factor(merge_data$Survival_class_prediction),
as.factor(merge_data$Survived))

print(matrix)

## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 231   5
##          1  35 147
##
##                Accuracy : 0.9043
##                  95% CI : (0.872, 0.9308)
##     No Information Rate : 0.6364
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.8016
##
##  Mcnemar's Test P-Value : 4.533e-06
##
##             Sensitivity : 0.8684
##             Specificity : 0.9671
##          Pos Pred Value : 0.9788
##          Neg Pred Value : 0.8077
##              Prevalence : 0.6364
##          Detection Rate : 0.5526
##    Detection Prevalence : 0.5646
##       Balanced Accuracy : 0.9178
##
##        'Positive' Class : 0
##
```

```
#Accuracy of prediction
cat("Accuracy ratio:", matrix$overall['Accuracy'])
```

```
## Accuracy ratio: 0.9043062
```

```
cat("Accuracy:", (matrix$overall['Accuracy'])*100, "%")
```

```
## Accuracy: 90.43062 %
```

Interpretation:

My logistic regression model correctly classified 90.43% of the cases in my test data, which indicates good performance.

```
#create csv file of predictions
predictions_df <- test_data[, c("PassengerId", "Survival_class_prediction")]
write.csv(predictions_df, file = "titanic_predictions.csv", row.names =
FALSE)
```