R_LOGISTIC_REGRESSION

Ulises Jose Bustamante Mora

2023-10-24

Importing the libraries

```
#install.packages("Amelia")
library(Amelia)
library(ggplot2)
```

Importing the dataset

```
df = read.csv("titanic.csv")
head(df)
```

```
##   PassengerId Survived Pclass
## 1           1        0      3
## 2           2        1      1
## 3           3        1      3
## 4           4        1      1
## 5           5        0      3
## 6           6        0      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
## 4        Futrelle, Mrs. Jacques Heath (Lily May Peel) female  35     1
0
## 5                             Allen, Mr. William Henry   male  35     0
0
## 6                                     Moran, Mr. James   male  NA     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
## 4           113803 53.1000  C123        S
## 5           373450  8.0500              S
## 6           330877  8.4583              Q
```

Knowing better the dataset

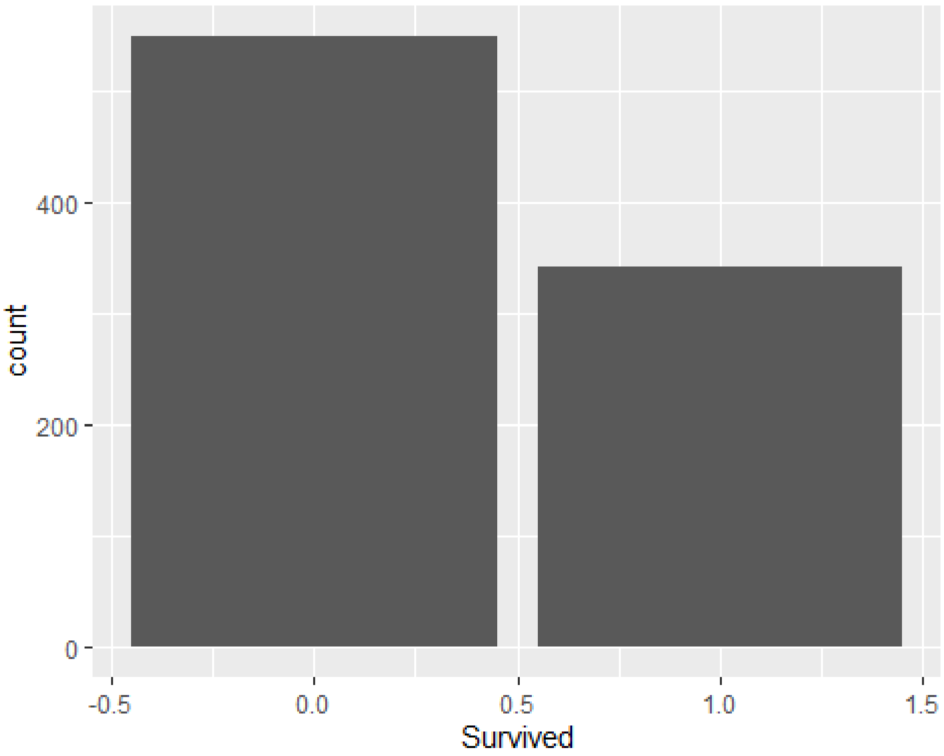
```
str(df)
```

```
## 'data.frame':    891 obs. of  12 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 ...
##  $ Name       : chr  "Braund, Mr. Owen Harris" "Cumings, Mrs. John
Bradley (Florence Briggs Thayer)" "Heikkinen, Miss. Laina" "Futrelle,
Mrs. Jacques Heath (Lily May Peel)" ...
##  $ Sex        : chr  "male" "female" "female" "female" ...
##  $ Age        : num  22 38 26 35 35 NA 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 ...
##  $ Ticket     : chr  "A/5 21171" "PC 17599" "STON/O2. 3101282"
"113803" ...
##  $ Fare       : num  7.25 71.28 7.92 53.1 8.05 ...
##  $ Cabin      : chr  "" "C85" "" "C123" ...
##  $ Embarked   : chr  "S" "C" "S" "S" ...
```

EDA

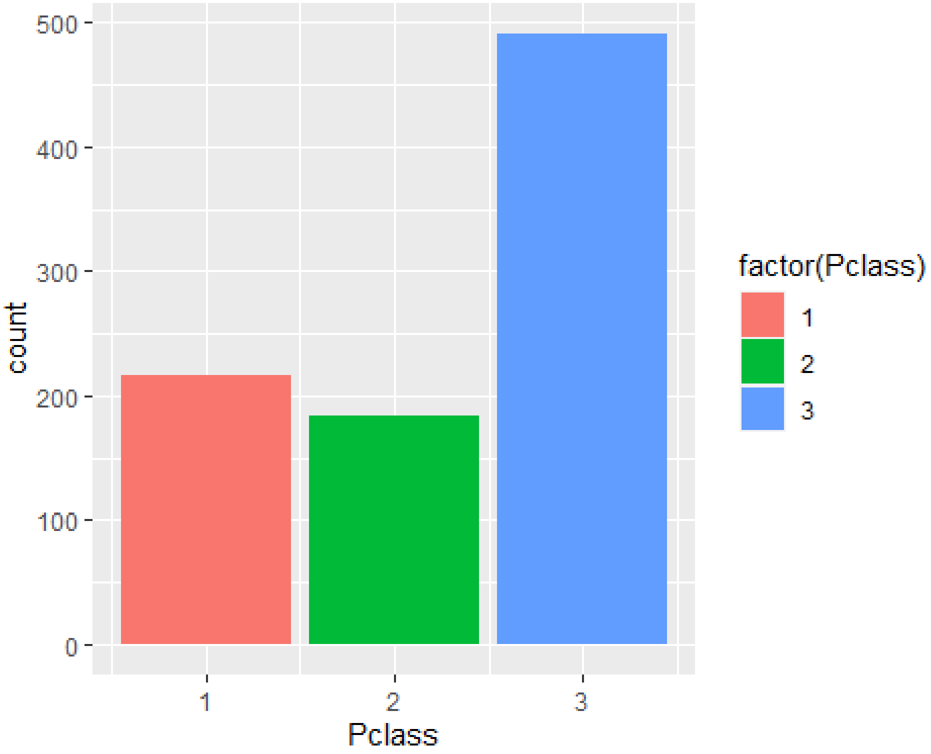
Counting how many people survived

```
ggplot(df, aes(Survived)) + geom_bar()
```



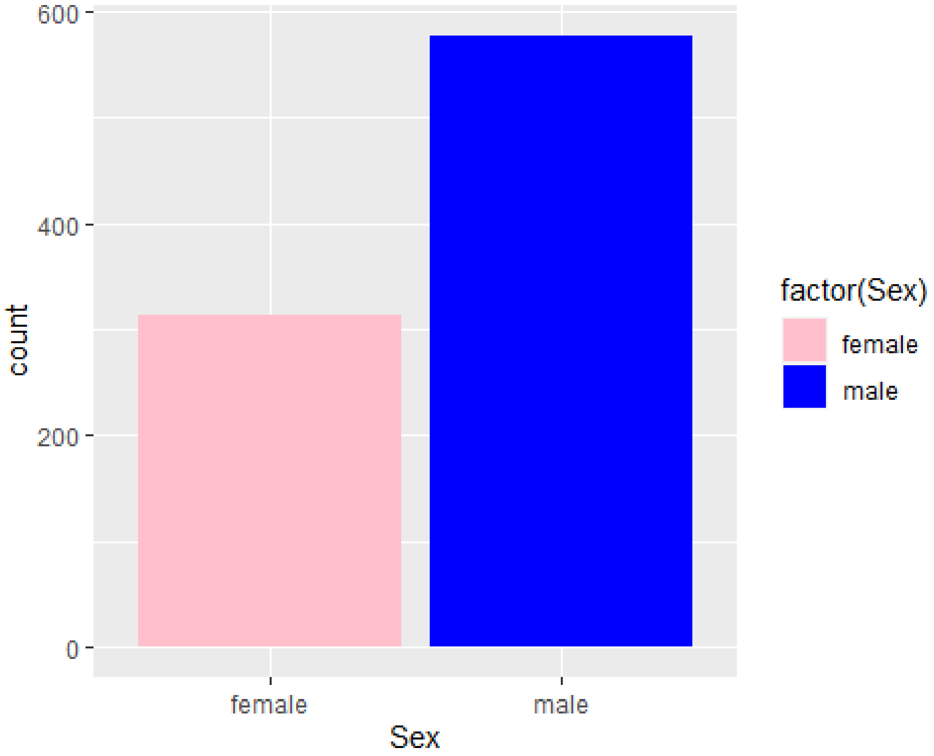
### Counting the people by ticket class

```
ggplot(df, aes(Pclass)) + geom_bar(aes(fill= factor(Pclass)))
```



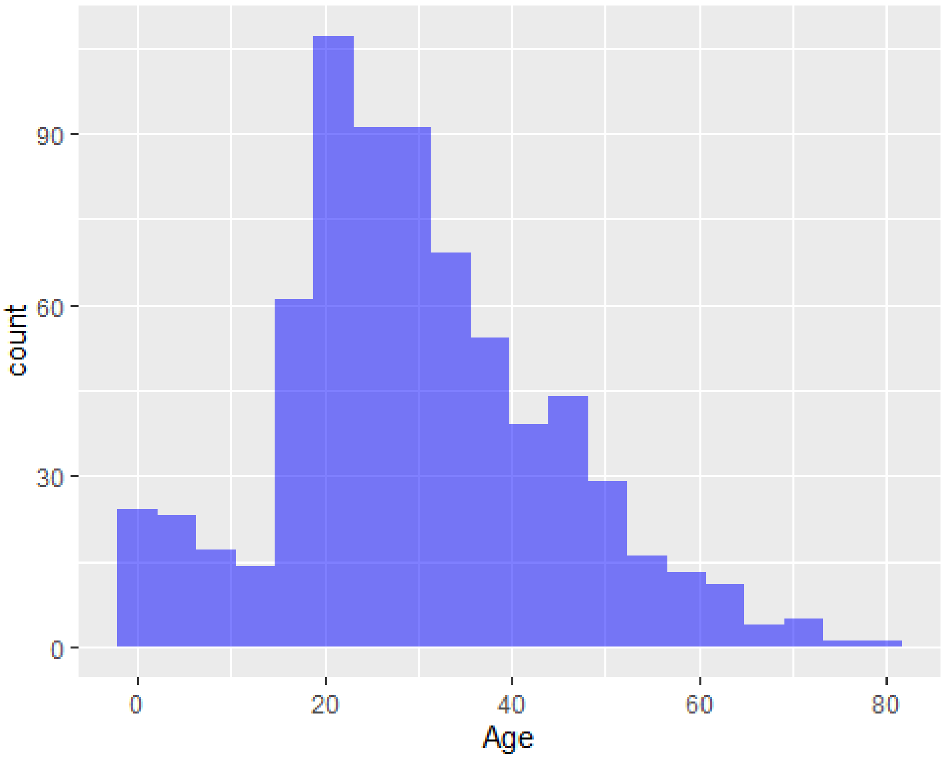
### Counting the people by sex

```r
ggplot(df, aes(Sex)) + geom_bar(aes(fill= factor(Sex))) +
  scale_fill_manual(values=c("pink", "blue"))
```



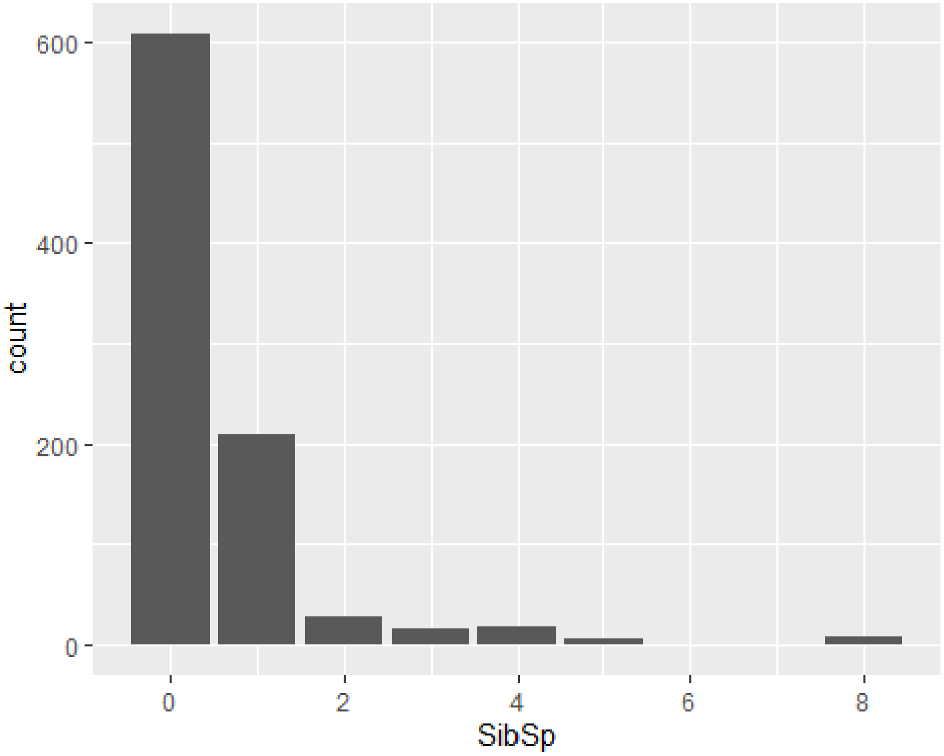
Data distribution of the age

```
ggplot(df, aes(Age)) + geom_histogram(bins = 20, alpha = 0.5, fill =
"blue")
```



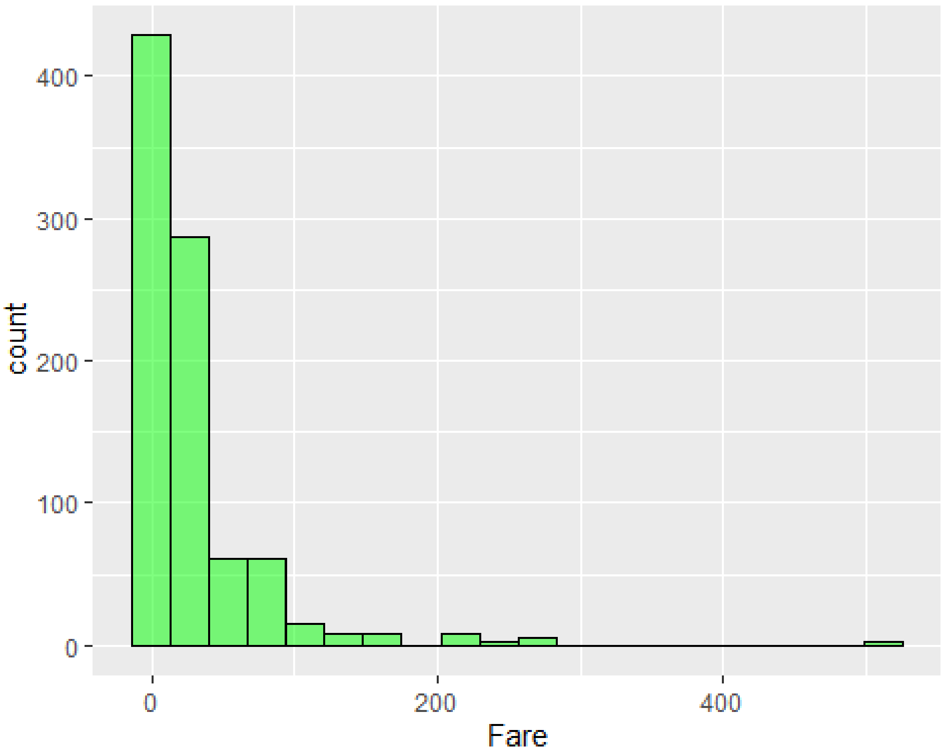
Counting people with siblings or spouse

```
ggplot(df, aes(SibSp)) + geom_bar()
```



Data distribution of the fare

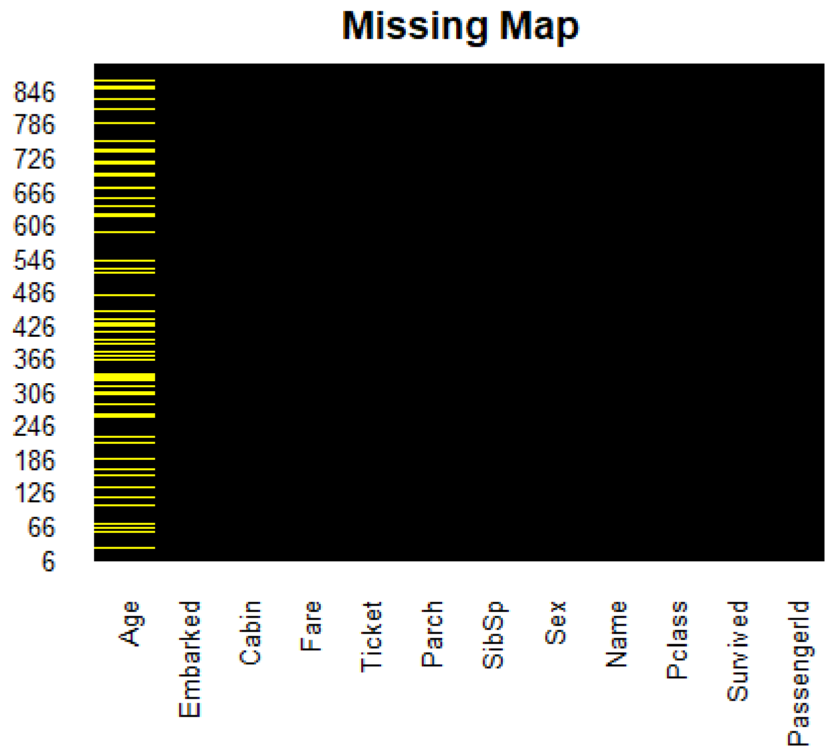
```r
ggplot(df, aes(Fare)) + geom_histogram(bins=20, alpha = 0.5, color =
"black", fill = "green")
```



ETL

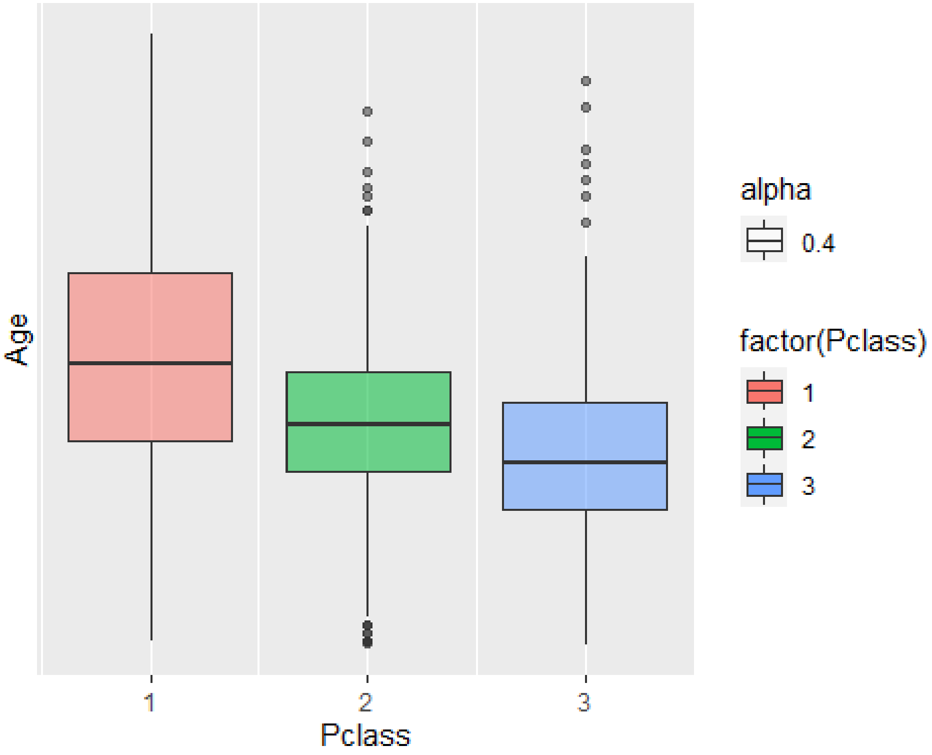
Working with missing values

```r
missmap(df, main = "Missing Map", col = c("yellow", "black"), legend = F)
```



Noting the mean of age per class

```r
g = ggplot(df, aes(Pclass, Age))
g = g + geom_boxplot(aes(group = Pclass, fill  = factor(Pclass), alpha = 0.4))
g = g + scale_y_continuous(breaks = seq(0,80,2) + theme_bw())
g
```



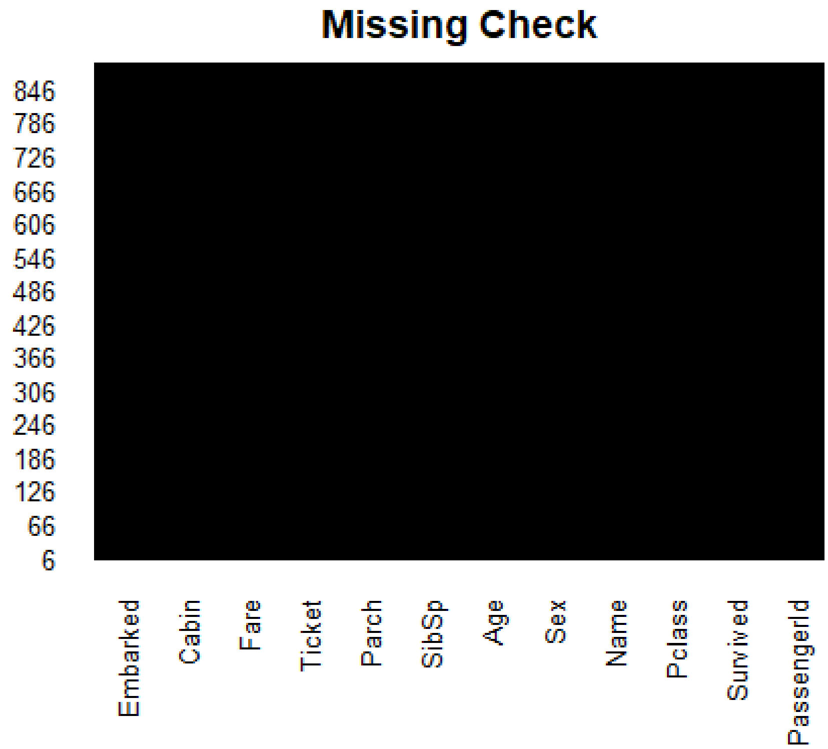
Modifying the value of NA by the mean depending of the ticket class

```r
impute_age = function (age,class){
   out <- age
   for (i in 1:length(age)){
     if (is.na(age[i])){
       if (class[i] == 1) {
         out[i] = 37
       }else if (class(i) == 2){
         out[i] = 29
       }else{
        out[i] = 24
       }
     }else{
      out[i]=age[i]
     }
   }
   return(out)
}
```

```r
fixed_ages = impute_age(df$Age, df$Pclass)
df$Age = fixed_ages
```

Checking the results of the NA values

```r
check_MP = missmap(df, main = "Missing Check", col = c("yellow",
"black"), legend = F)
```



```r
check_MP

## NULL
```

Creating the model

Removing variables that we are not going to use

```
library(dplyr)

## 
## Attaching package: 'dplyr'

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

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

df = select(df, -PassengerId, -Name, -Ticket, -Cabin)
head(df)

##   Survived Pclass    Sex Age SibSp Parch    Fare Embarked
## 1        0      3   male  22     1     0  7.2500        S
## 2        1      1 female  38     1     0 71.2833        C
## 3        1      3 female  26     0     0  7.9250        S
## 4        1      1 female  35     1     0 53.1000        S
## 5        0      3   male  35     0     0  8.0500        S
## 6        0      3   male  24     0     0  8.4583        Q
```

Training the model

```
library(caTools)
split = sample.split(df$Survived, SplitRatio = 0.7)
df_train = subset(df,split == T)
df_test = subset(df,split == F)

model = glm(Survived~. , family = binomial(link = "logit"), data =
df_train)
summary(model)

## 
## Call:
## glm(formula = Survived ~ ., family = binomial(link = "logit"), 
##     data = df_train)
## 
## Deviance Residuals: 
##     Min       1Q   Median       3Q      Max  
## -2.3816  -0.5451  -0.3854   0.5581   2.6185  
## 
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)    
## (Intercept)  18.531699 601.456914   0.031   0.9754    
## Pclass       -1.282808   0.185960  -6.898 5.26e-12 ***
## Sexmale      -2.962342   0.251831 -11.763  < 2e-16 ***
## Age          -0.054212   0.010388  -5.219 1.80e-07 ***
## SibSp        -0.281182   0.122378  -2.298   0.0216 *  
## Parch        -0.216286   0.157570  -1.373   0.1699    
## Fare          0.001897   0.002578   0.736   0.4618    
## EmbarkedC   -12.267253 601.456563  -0.020   0.9837    
## EmbarkedQ   -12.277886 601.456640  -0.020   0.9837    
## EmbarkedS   -12.691839 601.456530  -0.021   0.9832    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
##     Null deviance: 829.60  on 622  degrees of freedom
## Residual deviance: 514.95  on 613  degrees of freedom
## AIC: 534.95
## 
## Number of Fisher Scoring iterations: 13
```

Creating predictions

```
pre = predict(model, df_test, type = "response")
results = ifelse(pre>0.5,0,1)
```

Metrics

```
missClass = mean(results != df_test$Survived)
print(paste0("The accuracy of this model is ", missClass))

## [1] "The accuracy of this model is 0.764925373134328"
```

Confusion Matrix

```
table(df_test$Survived, pre > 0.5)

## 
##      FALSE TRUE
##   0    136   29
##   1     34   69
```