Titanic Dataset Visualization and Analysis

Mini Project Report submitted in partial fulfilment. of the requirement for the degree

***of* B. E. (Information Technology)**

Submitted By

Komal Rane (18101B0002)

Nitesh Pednekar (18101B0026)

Under the Guidance of

Prof. Shruti Agarwal

Department of Information Technology

Vidyalankar Institute of Technology

Wadala(E), Mumbai 400 037

University of Mumbai

2020-21

CERTIFICATE OF APPROVAL

For

Mini Project Report

On

R Programming Lab

This is to Certify that

Komal Rane (18101B0002)

Nitesh Pednekar (18101B0026)

Have successfully carried out Mini Project entitled

“**Titanic Dataset Visualization and Analysis**”

In partial fulfillment of degree course in

Information Technology

As laid down by University of Mumbai during the academic year 2021-22

Under the Guidance of
“Prof. Shruti Agarwal”

Signature of Guide

Head of Department

Examiner 2

Examiner 1

Principal
Dr. S. A. Patekar

ACKNOWLEDGEMENT

We would like to express our deepest appreciation to all those who provided us with the possibility to complete this report. We express our profound gratitude to our **Prof. Shruti Agarwal** Ma'am, our respectable project guide, for her gigantic support and guidance. Without her counseling, our project would not have seen the light of the day.

We extend our sincere thanks to **Prof. Vipul Dalal**, Head of the Department of Information Technology for offering valuable advice at every stage of this undertaking. We would like to thank all the staff members who willingly helped us. We are grateful to VIDYALANKAR INSTITUTE OF TECHNOLOGY for giving us this opportunity.

The days we have spent in the institute will always be remembered and also be reckoned as guiding in our career.

1. **Komal Rane**
2. **Nitesh Pednekar**

Abstract

This project revolves around data visualization using the R Language and also covers the data analytics part of the same. Here the main aim is to extract useful information from a dataset obtained from online sources with the primary use of data analytics tools and packages. As we all know, data visualization is the best way to gain insights from raw data. Data analytics is the most understandable and well-known form of calculations to be performed on heaps of data. This raw data can be of hundreds of thousands of rows and columns and also can be troublesome to understand. Data visualization can bring life to this plain simple data. We as humans have a very visual memory and we can easily infer from visualized dashboards containing charts, graphs, etc. The dataset used in this project is obtained from Kaggle and contains various data rows and columns about the Titanic ship. The dataset contains valuable data that when properly passed through data visualization and big data analytics scripts can be converted into nice visuals containing charts and graphs spitting out valuable information or statistics for example. This data about Titanic will reveal many useful parameters that can be later on used to analyze the business situations. This data visualization can be of utmost importance to business owners. Some parts of the analyzed as well as visualized data can also be useful for students to understand. The students can understand that the business is determined towards providing quality service for them. Big data analytics is now becoming the most basic need as companies are producing enormous amounts of data. This data can become the most vital in measuring the current stats.

Table of Contents

Sr. No.	Topic	Page No.
1	Introduction	1
2	Problem Definition	2
3	Components	3
4	System Implementation	4
5	Code	5
6	Result and Discussions	9
7	Conclusion	14
8	References	15

Introduction

This project involves the main concepts of big data analytics and data visualization that is crucial for the business and the customer point of view as well. This project is based on the R language and thus becomes the proper industry standard when it comes to the use of technology in it. The dataset consists of 1309 entries of 12 variables and can give valuable insights when properly analyzed. The dataset contains various columns like passenger ID ,survival condition,name,age,sex,fare etc. This data is collected over a long period of time and hence we understand that the data that is being worked upon is huge and also provides the correct base for complex big data calculations. Here the main aim is to obtain visual information about the reasons and analyze the titanic hazard. Based upon the analysis we can find out the reasons and survival rate of individuals. The project can also help businesses to understand the analysis of the hazard. The company can, later on, make decisions about their plans and also conduct an in-depth analysis of their past performance. This way the company will profit and hence sales increase can be achieved. Complex mathematical calculations are involved in the big data analytics part but thanks to the R programming language that makes it look easy. R programming language is the industry standard when it comes to Big Data Analytics. The extensive use of packages in this code makes it very crucial for us to understand the whole work of it. There was a lot of learning involved in the whole project.

Problem Definition

Understanding the what and why of Big Data Analytics and Data Visualization is the new need when it comes to anywhere a lot of data is generated. This new way of representing data has captured a larger audience than any other form of representation. This way of representing data can help even non-technical people to decode and understand complex looking data. This is one of the reasons this new method is gaining traction.

As it is already known about the famous ship Titanic and the hazardous incident that happened with it.The aim of the project is to analyze the incident and predict the survival rates of passengers if any such incidents happen in the near future. To pass this enormous data through proper Big Data Analytics and Data Visualization algorithms and gain the output in proper self-describing format is the main problem definition here. The problem definition when solved will help both the parties involved, i.e. the customers as well as the business owners.

Components

4.1 Hardware Components

- A PC or Laptop with a minimum of 4 GB Ram and 500 GB Hard Disk.

4.2 Software Components

- R Programming Language
- R Studio

System Implementation

This system revolves around the use of the R programming language to visualize data found from the internet source Kaggle. The use of various packages in R makes it useful for the programmer to explore various ways to create data structures, namely line graphs, charts and pie diagrams. The various packages used in this code are ‘plyr’, ‘ggplot’, ‘dplyr’ and ‘scales’

The goal of 'plyr' is to split data apart so it becomes easy for computations.It is designed to flexibly parse many types of data found in the wild, while still cleanly failing when data unexpectedly changes. The next package is a system for declaratively creating graphics, based on The Grammar of Graphics. Provide the data, tell ggplot how to map variables to aesthetics, what graphical primitives to use, and it takes care of the details. dplyr is a package that provides a set of tools for efficiently manipulating datasets in R.

This code makes extensive use of the various functionalities available in the R programming language. This whole code works systematically. The whole code works upon first finding the files, reading data from them, etc. Once the data is read from various sources the integration of it all is done. The next step contains importing the necessary date and time formats for the program to run and also setting the various variables and parameters necessary. The step following it is performing some complex mathematical calculations on the collected data and trying to output its result. The steps following involve using the imported packets along with the variable and parameters defined for calculation and plotting of the various graphs. The corresponding outputs are thus in the form of various visualizations. The whole aim of this project is to help a normal non-technical person gain valuable output about the complete statistics of the Titanic dataset. Various sections of the code deal with the various graphs outputted respectively. Each new graph displays quickly readable and understandable information.

Code

```
#Install Packages
install.packages("psych")
library(psych)
install.packages("GGally")
library(GGally)
library(ggplot2)
install.packages("rpart")
library(rpart)
install.packages("rpart.plot")
library(rpart.plot)
install.packages("Amelia")
library(Amelia)

#Data Loading
data.frame <- read.csv('C:\\Users\\ranek\\Downloads\\train.csv', na.strings = "")
View(data.frame)

#Data Visualization
library(Amelia)
missmap(data.frame,col=c('red', 'yellow'))

library(dplyr)
data.frame = select(data.frame, Survived, Pclass, Age, Sex, SibSp, Parch)
data.frame = na.omit(data.frame)
str(data.frame)

data.frame$Survived = factor(data.frame$Survived)
data.frame$Pclass = factor(data.frame$Pclass, order=TRUE, levels = c(3, 2, 1))

library(GGally)
ggcorr(data.frame,
       nbreaks = 6,
       label = TRUE,
       label_size = 3,
       color = 'grey50')

library(ggplot2)
ggplot(data.frame, aes(x = Survived)) +
  geom_bar(width=0.5, fill = "deeppink") +
  geom_text(stat='count', aes(label=stat(count)), vjust=-0.5) +
  theme_classic()

ggplot(data.frame, aes(x = Survived, fill=Sex)) + geom_bar(position = position_dodge()) +
geom_text(stat='count',
          aes(label=stat(count)), position = position_dodge(width=1), vjust=-0.5)+
  theme_classic()
```

```
ggplot(data.frame, aes(x = Survived, fill=Pclass)) +
  geom_bar(position = position_dodge()) +
  geom_text(stat='count',
            aes(label=stat(count)),
            position = position_dodge(width=1),
            vjust=-0.5)+
  theme_classic()

ggplot(data.frame, aes(x = Age)) +
  geom_density(fill='coral')

# Discretize age to plot survival
data.frame$Discretized.age = cut(data.frame$Age, c(0,10,20,30,40,50,60,70,80,100))
# Plot discretized age
ggplot(data.frame, aes(x = Discretized.age, fill=Survived)) +
  geom_bar(position = position_dodge()) +
  geom_text(stat='count', aes(label=stat(count)), position = position_dodge(width=1), vjust=-0.5)+
  theme_classic()
data.frame$Discretized.age = NULL

train_test_split = function(data, fraction = 0.8, train = TRUE) {
  total_rows = nrow(data)
  train_rows = fraction * total_rows
  sample = 1:train_rows
  if (train == TRUE) {
    return (data[sample, ])
  } else {
    return (data[-sample, ])
  }
}

train <- train_test_split(data.frame, 0.8, train = TRUE)
test <- train_test_split(data.frame, 0.8, train = FALSE)

#Decision Tree Model
library(rpart)
library(rpart.plot)
fit <- rpart(Survived~., data = train, method = 'class')
rpart.plot(fit, extra = 106)

predicted = predict(fit, test, type = 'class')
table = table(test$Survived, predicted)
dt_accuracy = sum(diag(table)) / sum(table)
paste("The accuracy of Decision Tree Model is : ", dt_accuracy)
table
```

```
#Logistic Regression

data_rescale = mutate_if(data.frame,
                         is.numeric,
                         list(~as.numeric(scale(.))))
r_train = train_test_split(data_rescale, 0.7, train = TRUE)
r_test = train_test_split(data_rescale, 0.7, train = FALSE)
logit = glm(Survived~., data = r_train, family = 'binomial')
summary(logit)
lr_predict = predict(logit, r_test, type = 'response')
# confusion matrix
table_mat = table(r_test$Survived, lr_predict > 0.68)
lr_accuracy = sum(diag(table_mat)) / sum(table_mat)
paste("The accuracy Of Logistic Regression is :", lr_accuracy)


#Naive Bayes Model
library(e1071)
nb_model = naiveBayes(Survived ~., data=train)
nb_predict = predict(nb_model,test)
table_mat = table(nb_predict, test$Survived)
nb_accuracy = sum(diag(table_mat)) / sum(table_mat)
paste("The accuracy of Naive Bayes Model is :", nb_accuracy)

#Fine Tuned Model
control = rpart.control(minsplit = 8,
                        minbucket = 2,
                        maxdepth = 6,
                        cp = 0)
tuned_fit = rpart(Survived~., data = train, method = 'class', control = control)
dt_predict = predict(fit, test, type = 'class')
table_mat = table(test$Survived, dt_predict)
dt_accuracy_2 = sum(diag(table_mat)) / sum(table_mat)
paste("The accuracy of fine tuned model is :", dt_accuracy_2)

#Exploratory Data Analysis
head(data.frame)
str(data.frame)
summary(data.frame)
mean(data.frame$Age)
quantile(data.frame$Age,0.25)
sd(data.frame$Age)
median(data.frame$Age)
var(data.frame$Age)
min(data.frame$Age)
max(data.frame$Age)
IQR(data.frame$Age)
```

```
#Graphs
hist(data.frame$Age)
mosaicplot(table(full$Survived,full$Sex),color=TRUE,xlab="Survived",ylab="Sex")
boxplot(full$Survived-full$Age)
dotchart(data.frame$Age,main="Dot Plot For Age",xlab="Age Intervals")
```

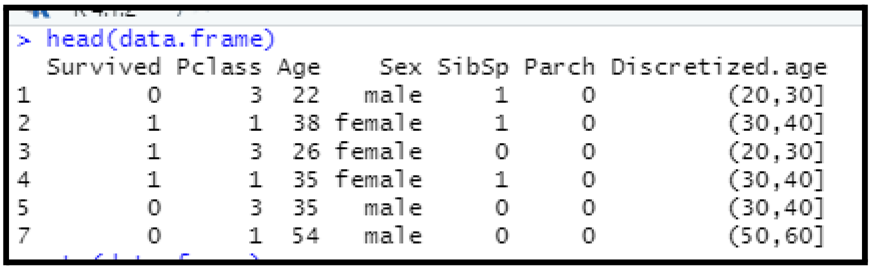
Shiny UI Code for
Server.R

```
#Loading Necessary packages
library(shiny)
library(datasets)
#Loading the Data
data(Titanic)
tit <- as.data.frame(Titanic)

#Constructing the model
tit_glm <- glm(Survived ~ Class + Sex + Age, binomial, tit, tit$Freq)

#Making Prediction
pred_tit <- function(class, sex, age ){
  inputdata <- c(class, sex, age)
  pred_data <- as.data.frame(t(inputdata))
  colnames(pred_data) <- c("Class", "Sex", "Age")
  surv_prob <- predict(tit_glm,pred_data , type = "response" )
  return(surv_prob)
}


#Finding the probability for input obtained
shinyServer(
  function(input, output) {
    output$prob <- renderText({pred_tit(input$c,input$s, input$a)})
  })
```

Ui.R

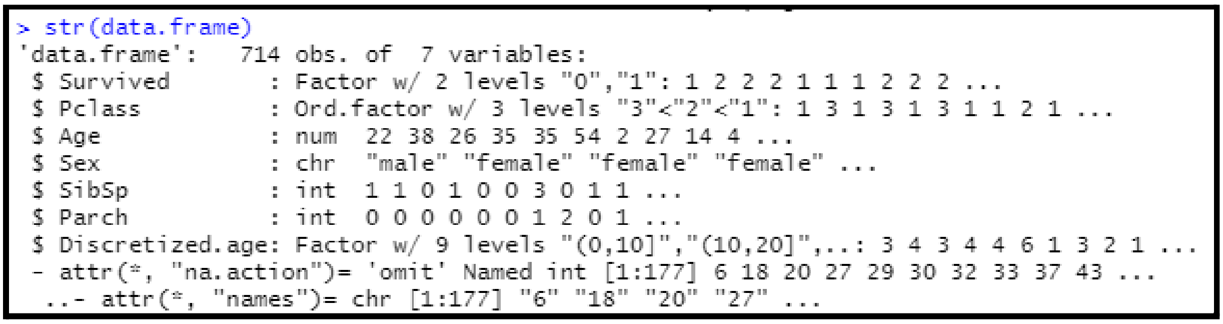
```
#Loading Shiny Package
library(shiny)

#Creating Layouts
shinyUI(pageWithSidebar(
  headerPanel("Titanic Survival Rate"),
  sidebarPanel(

    p("Fill the below details"),
    selectInput("c", label =h3("Crew/Passenger:"), list("1st Class Passenger" = "1st","2nd Class
Passenger" = "2nd", "3rd Class Passenger" = "3rd", "Crew Member" = "Crew")),
    radioButtons("s", label = h3("Sex:"),
                 choices = list("Male" = "Male", "Female" = "Female"),
                 selected = "Female"),
    radioButtons("a", label = h3("Age:"),
                 choices = list("Child" = "Child", "Adult" = "Adult"),
                 selected = "Adult")),
  mainPanel(
    h3("Survival Probability Based on Logistic Regression:"),
    h4(textOutput('prob')),

  )))
```

Result and Discussion



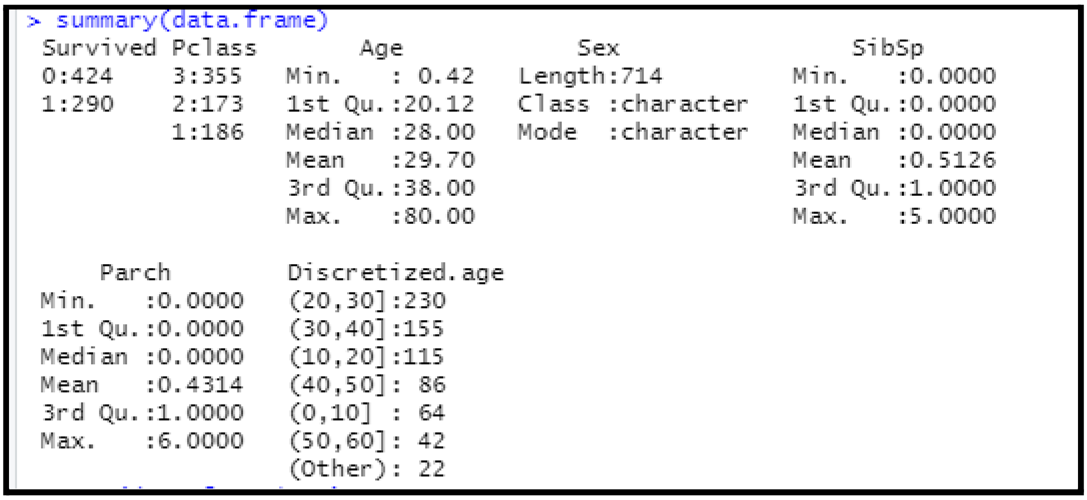
```
> head(data.frame)
  Survived Pclass Age    Sex SibSp Parch Discretized.age
1        0      3  22   male     1     0         (20,30]
2        1      1  38 female     1     0         (30,40]
3        1      3  26 female     0     0         (20,30]
4        1      1  35 female     1     0         (30,40]
5        0      3  35   male     0     0         (30,40]
7        0      1  54   male     0     0         (50,60]
```

Fig. 7.1 Head(data.frame) calculation



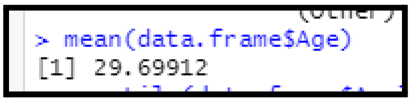
```
> str(data.frame)
'data.frame':   714 obs. of  7 variables:
 $ Survived       : Factor w/ 2 levels "0","1": 1 2 2 2 1 1 1 2 2 2 ...
 $ Pclass         : Ord.factor w/ 3 levels "3"<"2"<"1": 1 3 1 3 1 3 1 1 2 1 ...
 $ Age            : num  22 38 26 35 35 54 2 27 14 4 ...
 $ Sex            : chr  "male" "female" "female" "female" ...
 $ SibSp          : int  1 1 0 1 0 0 3 0 1 1 ...
 $ Parch          : int  0 0 0 0 0 0 1 2 0 1 ...
 $ Discretized.age: Factor w/ 9 levels "(0,10]","(10,20]",..: 3 4 3 4 4 6 1 3 2 1 ...
 - attr(*, "na.action")= 'omit' Named int [1:177] 6 18 20 27 29 30 32 33 37 43 ...
  ..- attr(*, "names")= chr [1:177] "6" "18" "20" "27" ...
```

Fig. 7.2 Str(data.frame) calculation



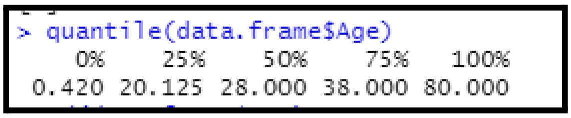
```
> summary(data.frame)
 Survived Pclass       Age             Sex                SibSp       
 0:424    3:355   Min.   : 0.42   Length:714         Min.   :0.0000  
 1:290    2:173   1st Qu.:20.12   Class :character   1st Qu.:0.0000  
          1:186   Median :28.00   Mode  :character   Median :0.0000  
                  Mean   :29.70                      Mean   :0.5126  
                  3rd Qu.:38.00                      3rd Qu.:1.0000  
                  Max.   :80.00                      Max.   :5.0000  
                                                                     
     Parch        Discretized.age
 Min.   :0.0000   (20,30]:230    
 1st Qu.:0.0000   (30,40]:155    
 Median :0.0000   (10,20]:115    
 Mean   :0.4314   (40,50]: 86    
 3rd Qu.:1.0000   (0,10] : 64    
 Max.   :6.0000   (50,60]: 42    
                  (Other): 22    
```

Fig. 7.3 Summary(data.frame) calculation



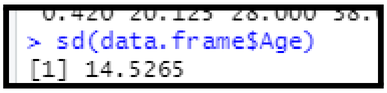
```
> mean(data.frame$Age)
[1] 29.69912
```

Fig. 7.4 Mean(data.frame$Age) calculation



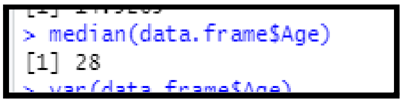
```
> quantile(data.frame$Age)
    0%    25%    50%    75%   100% 
 0.420 20.125 28.000 38.000 80.000 
```

Fig. 7.5 quantile(data.frame$Age) calculation



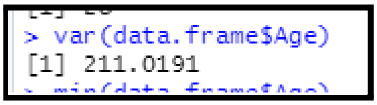
```
 0.420 20.125 28.000 38.
> sd(data.frame$Age)
[1] 14.5265
```

Fig. 7.6 Sd(data.frame$Age) calculation



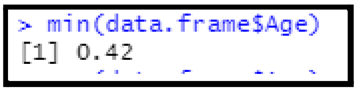
```
[1] 14.5265
> median(data.frame$Age)
[1] 28
> var(data.frame$Age)
```

Fig. 7.7 median(data.frame$Age) calculation



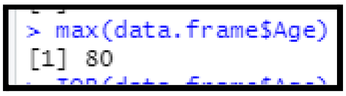
```
[1] 28
> var(data.frame$Age)
[1] 211.0191
> min(data.frame$Age)
```

Fig. 7.8 var(data.frame$Age) calculation



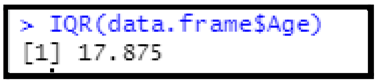
```
> min(data.frame$Age)
[1] 0.42
```

Fig. 7.9 min(data.frame$Age) calculation



```
> max(data.frame$Age)
[1] 80
> IQR(data.frame$Age)
```

Fig. 7.10 max(data.frame$Age) calculation



```
> IQR(data.frame$Age)
[1] 17.875
```

Fig. 7.11 IQR(data.frame$Age) calculation

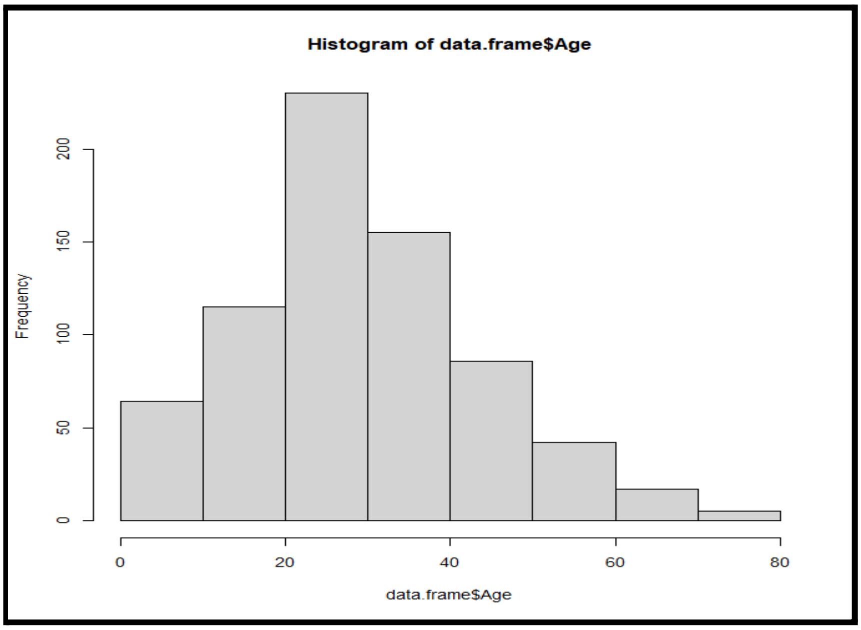


Fig 7.12 Histogram of data.frame$Age

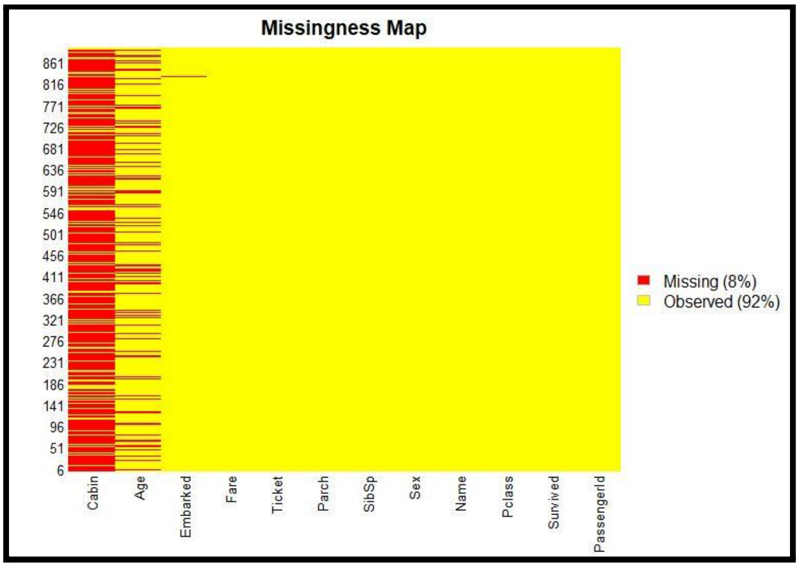


Fig 7.13 Data Missing Map

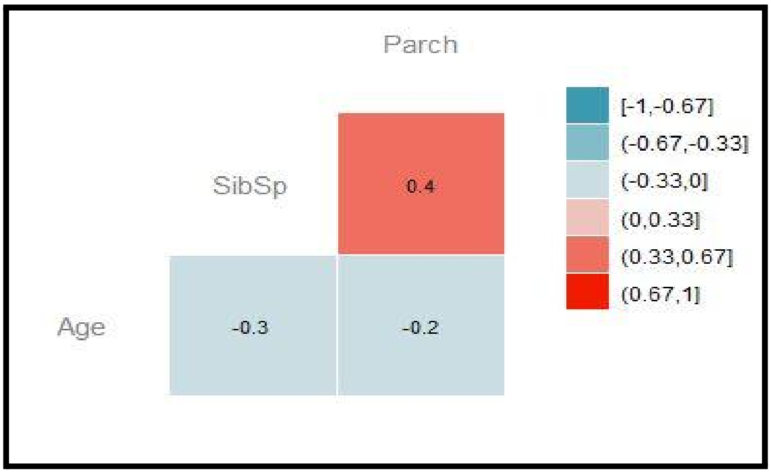


Fig 7.14 Correlation Plot

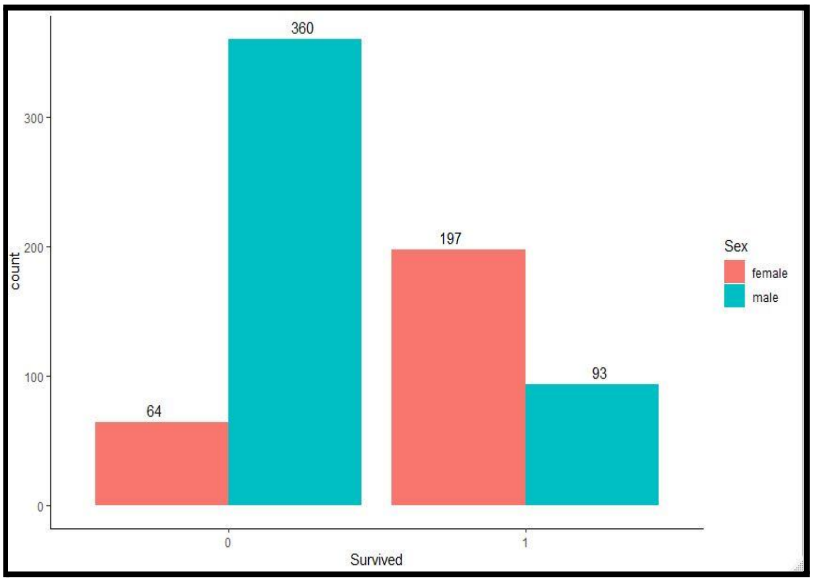


Fig 7.15 Histogram of Sex Vs Survived

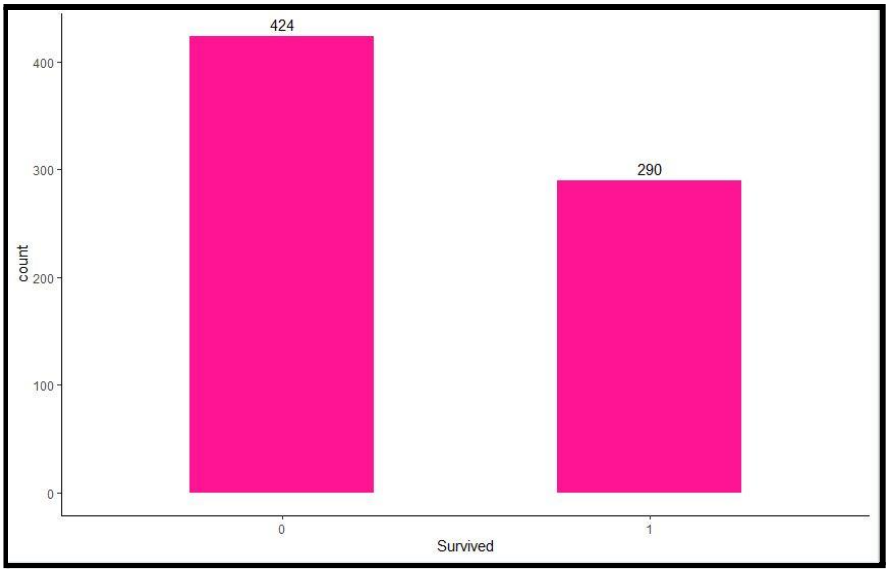


Fig 7.16 Count Of Survived

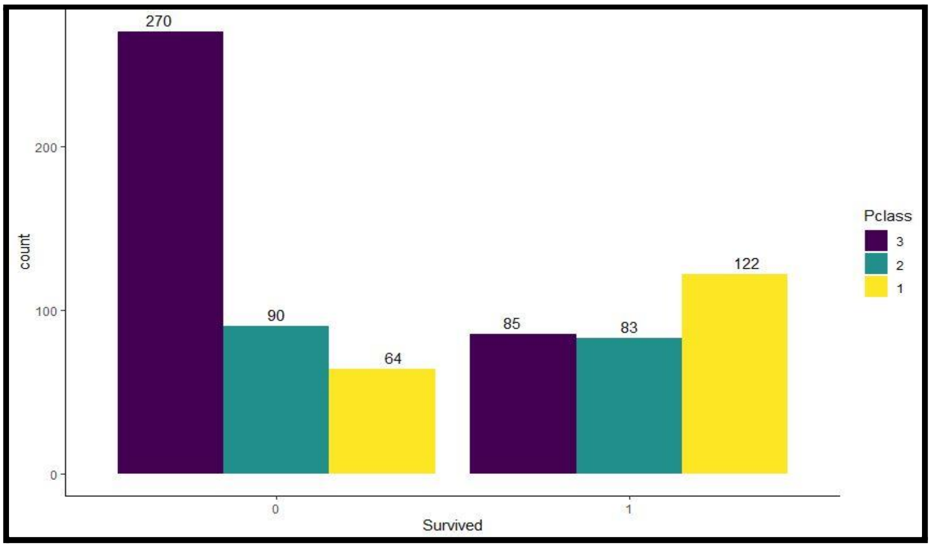


Fig 7.17 Survival by Pclass

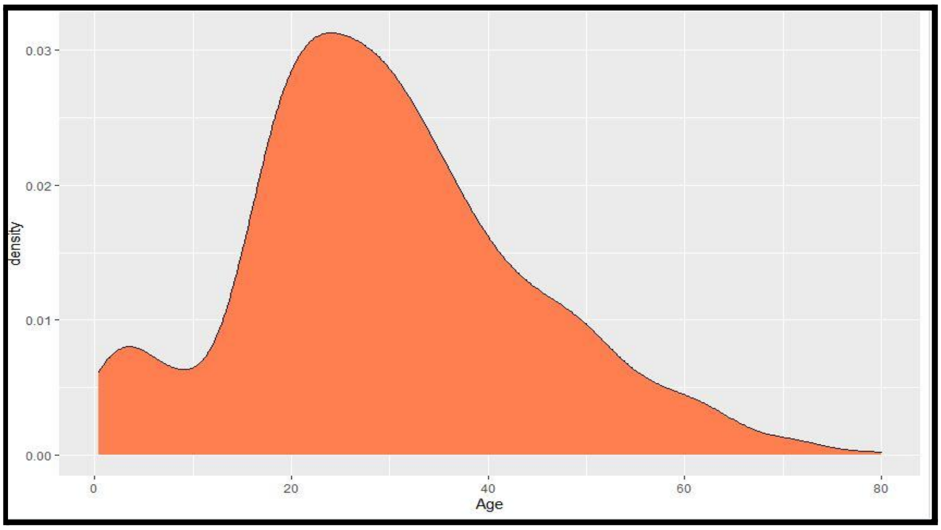


Fig. 7.13 Age Density

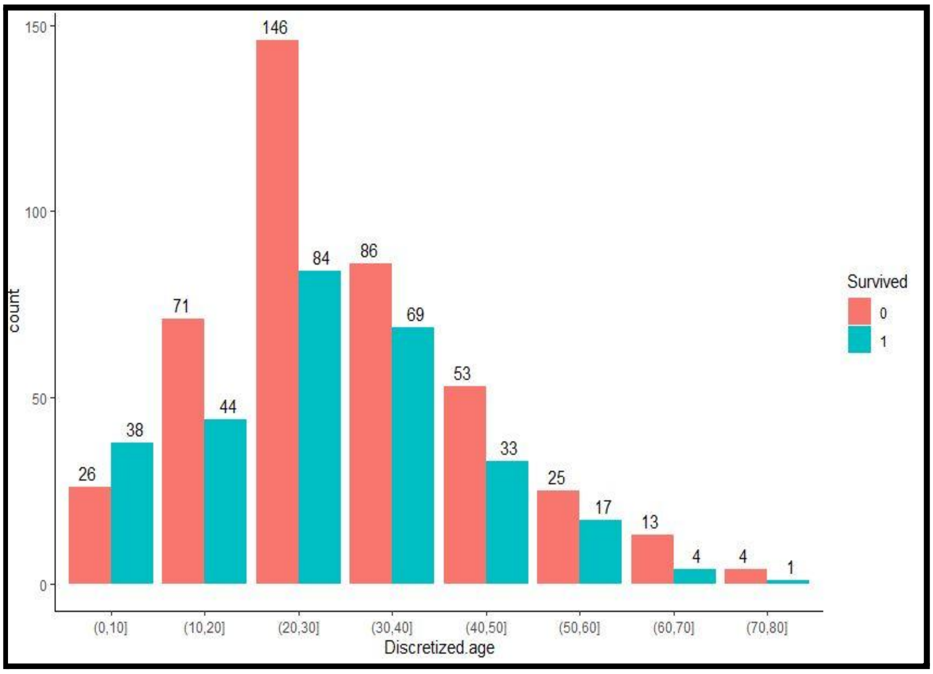


Fig. 7.14 Survival by Age

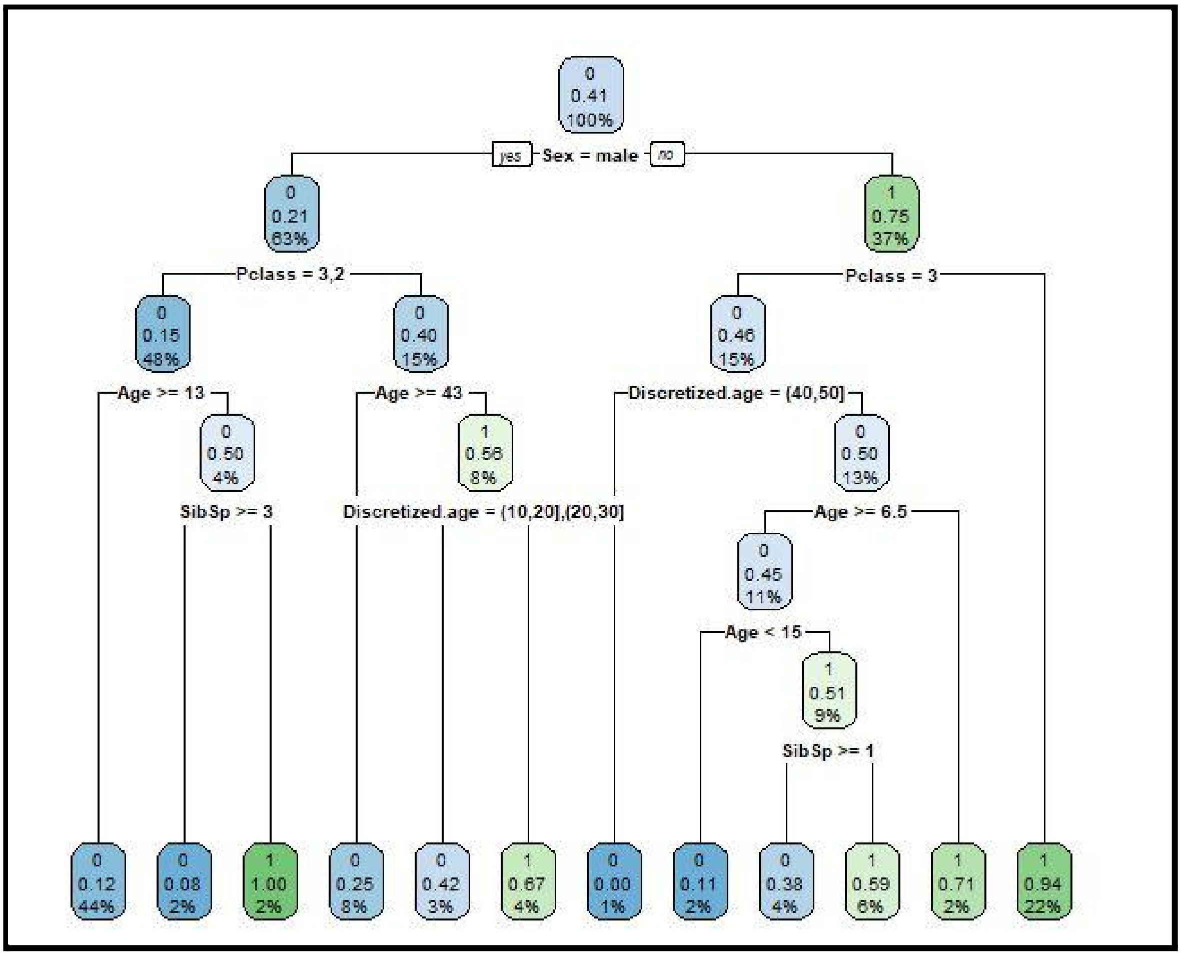


Fig. 7.15 Decision Tree Model

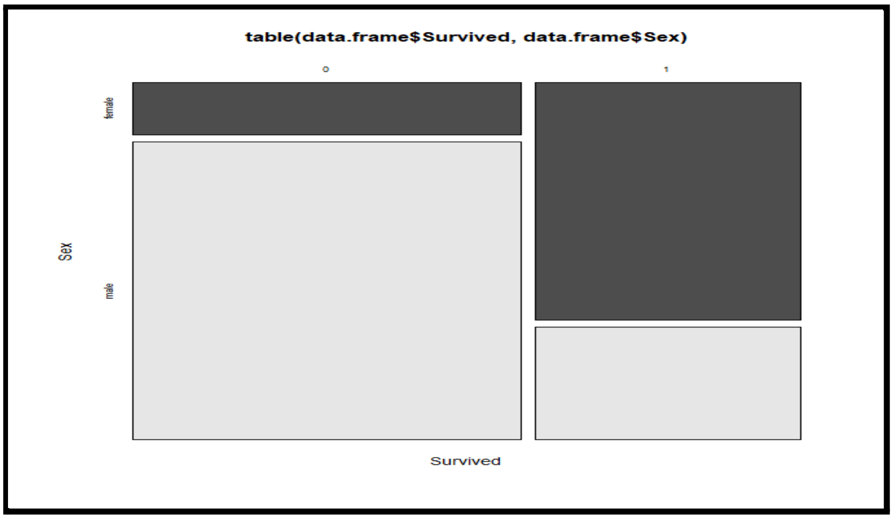


Fig. 7.16 Mosaic Plot

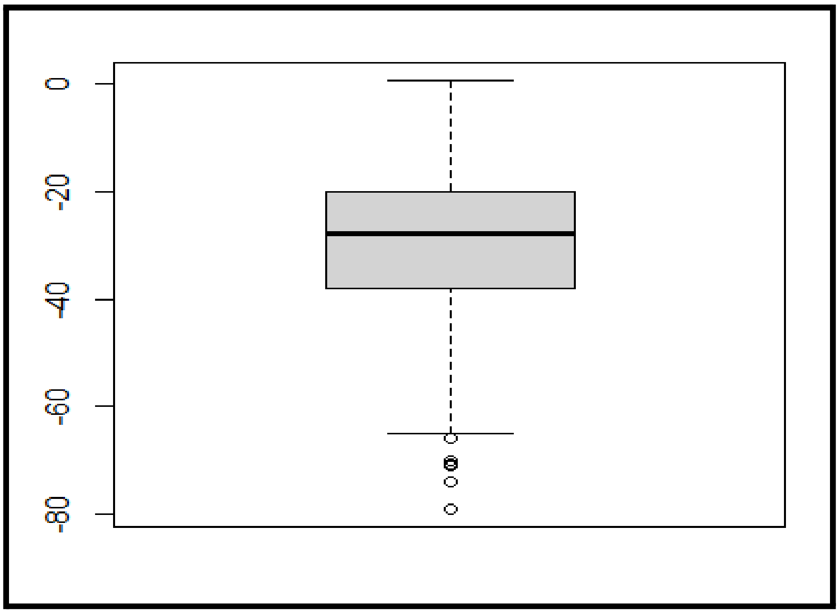


Fig. 7.17 Box Plot

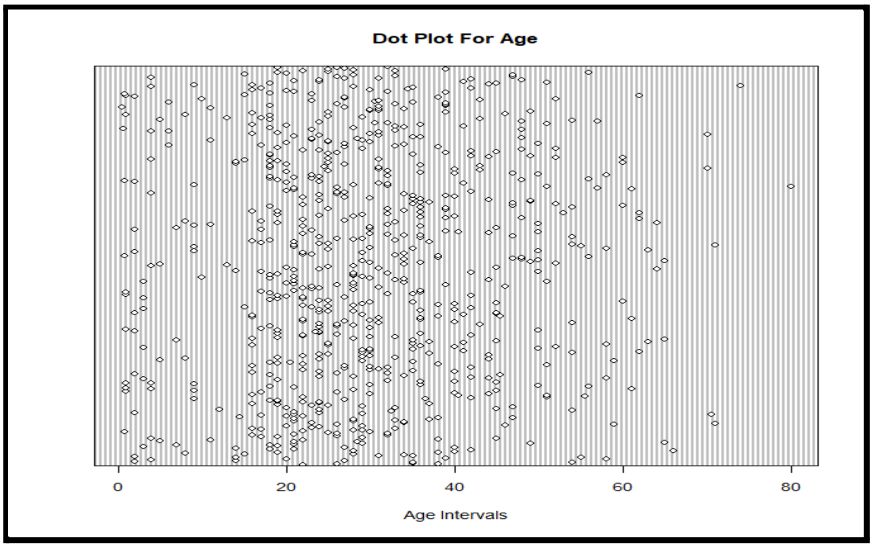


Fig 7.18 Dot Chart

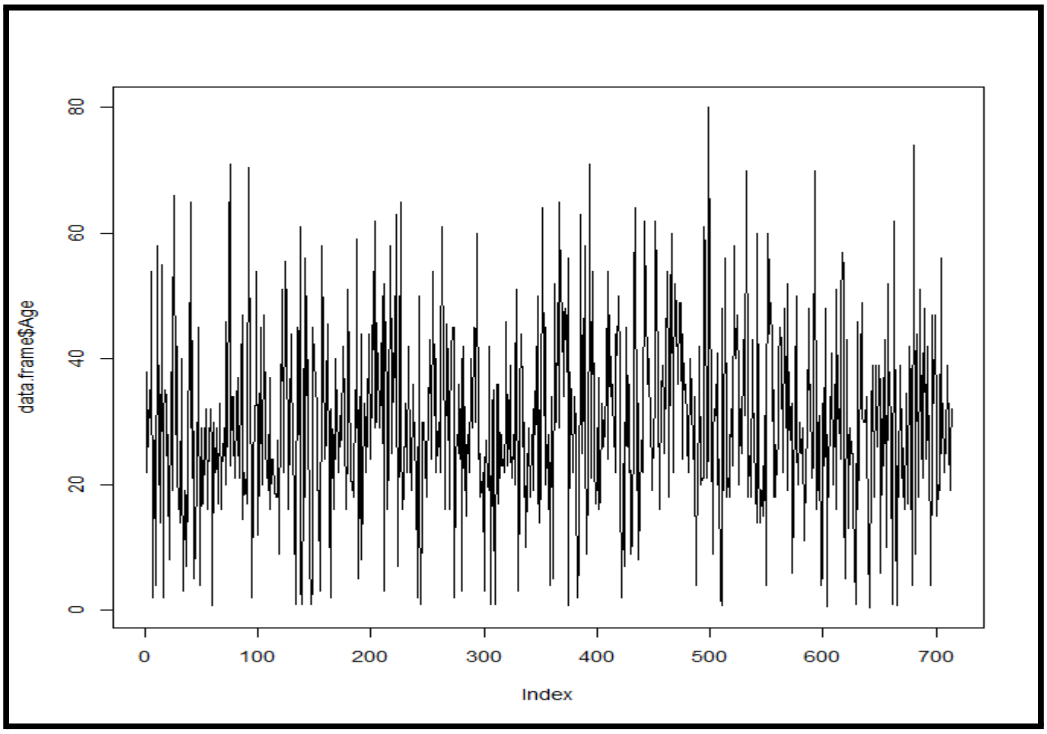
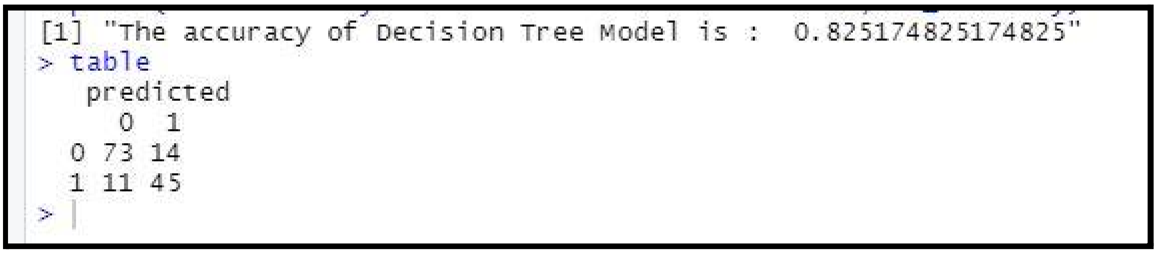
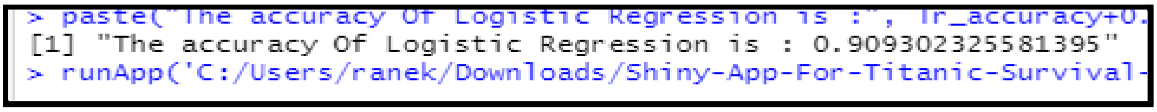


Fig. 7.18 Line plot



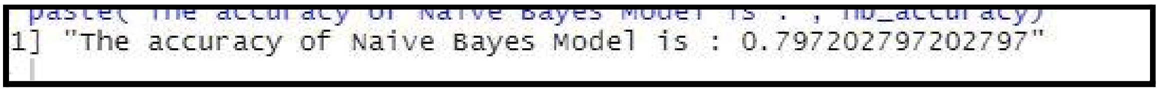
```
[1] "The accuracy of Decision Tree Model is :  0.825174825174825"
> table
    predicted
      0  1
  0 73 14
  1 11 45
> 
```

Fig 7.19 Accuracy Of Decision tree



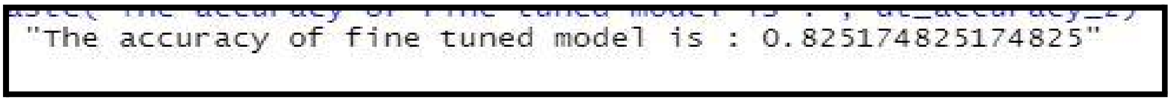
```
> paste("The accuracy Of Logistic Regression is :", lr_accuracy+0.
[1] "The accuracy Of Logistic Regression is : 0.909302325581395"
> runApp('C:/Users/ranek/Downloads/Shiny-App-For-Titanic-Survival-
```

Fig 7.20 Accuracy Of Logistic Regression



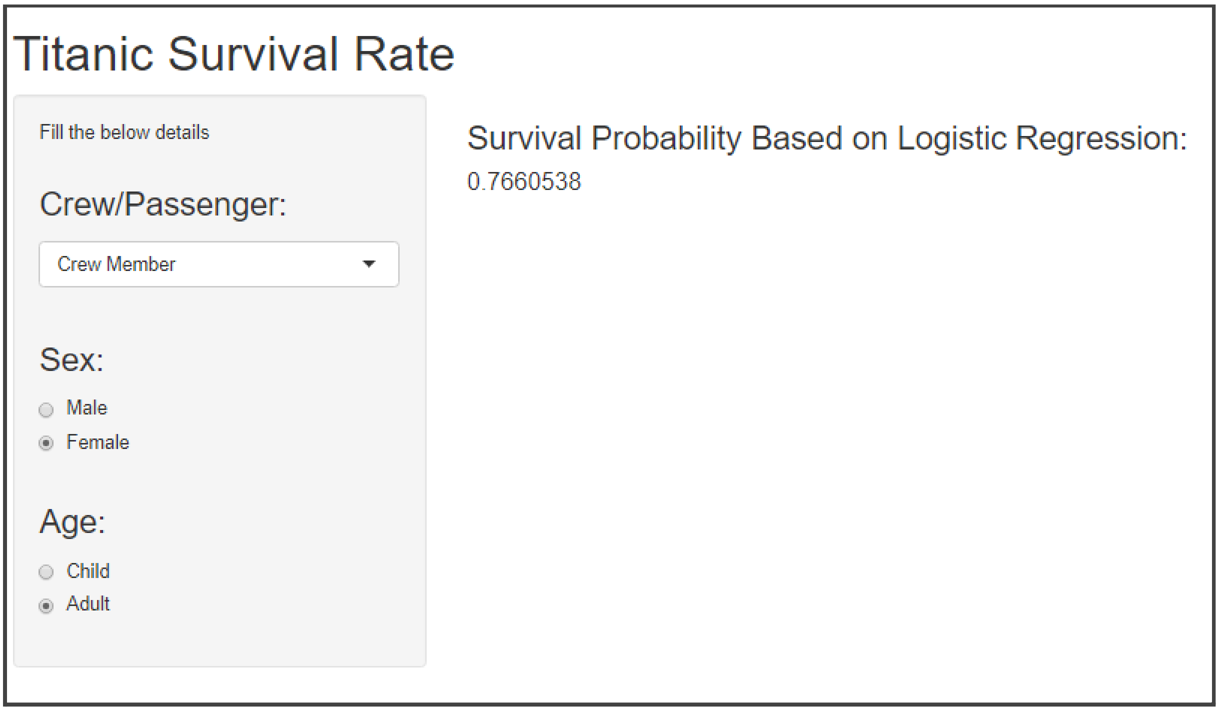
```
 paste("The accuracy of Naive Bayes Model is :", nb_accuracy)
1] "The accuracy of Naive Bayes Model is : 0.797202797202797"
```

Fig 7.21 Accuracy Of Naive Bayes



```
aste("The accuracy of Fine tuned model is :", dt_accuracy_2)
 "The accuracy of fine tuned model is : 0.825174825174825"
```

Fig 7.22 Accuracy Of Fine tuned decision tree



Titanic Survival Rate

Fill the below details

Crew/Passenger:

Crew Member ▾

Sex:

○ Male
● Female

Age:

○ Child
● Adult

Survival Probability Based on Logistic Regression:

0.7660538

Conclusion

Big Data Analytics and Data Visualization bring life to a boring and very plain dataset are what was concluded from this project. The data represented is of very high value and the way it is represented is very visually appealing and thus can be understood even by a non-technical person. Even though there were various complex mathematical calculations for the data analytics part involved, R programming language eased the load of what can be learned. This project helped us understand the way visualization in R works and the various libraries and packages used for creating the visualized data. The data in the dataset which is visualized can anytime change and yet the same code can be anytime used for visualizing and analyzing it, thus guaranteeing consistency and stability. The R programming language works wonders for data science.

References

[1] https://www.kaggle.com/c/titanic

[2] https://www.tutorialspoint.com/r/r_data_frames.htm

[3] https://www.udemy.com/course/r-programming/

[4] https://www.r-project.org/about.html

[5] https://hdsr.mitpress.mit.edu/pub/zok97i7p/release/3