Title: Titanic

Group Number: 56

First Name	Last Name	Online Students? (Y or N)	Monday or Tuesday	Shared with ITMD 525? (Y or N)
Dhruva	Juloori	N	Tuesday	N
Lakshmi Anuhya	Koduri	N	Tuesday	N
Rajesh	Azmeera	Y	Monday	N

Note, if you are an online student or remote student from India, you belong to Monday class

INDEX

Table of Contents

1. Introduction and Motivations

On April 15, 1912, Titanic sank after colliding with an iceberg, killing 1502 out of 2224 passengers and crew. This sensational tragedy shocked the international community and led to better safety regulations for ships. Data Analysis is the process of systematically applying statistical and/or logical techniques to describe and illustrate, condense and recap, and evaluate data. Hence, the main Focus of this project is to analyze and interpret the survival rate of passengers from a given set of data by applying classification techniques. Also, we complete the analysis of what sorts of people were likely to survive. In this project, we apply the tools of machine learning to predict which passengers survived the tragedy. The whole project was implemented and visualized in R programming. Ultimately, the entire project is based on the percentage of passengers correctly predicted, which is also known simply as accuracy. Titanic is a very common topic among each of us and knowing the survival rate in passenger class accurately was a very interesting factor, therefore this motivated us to collect and data and therefore analyze, interpret it.

2. Data Description

We chose our dataset from KAGGLE where in the data was taken and further analyzed. The total number of records were about 1224. The is training set should be used to build your machine learning models. For the training set, an outcome was provided which was also known as the ground truth for each passenger. Our model was based on features like passenger's gender and class. The data set was used to check how well model performs on unseen data. In the test set, ground truth for each passenger was not provided and hence we predicted these outcomes. For each passenger in the data set, logistic regression model we trained to predict whether they survived the sinking of the Titanic. The basic data dictionary and the variables were declared as follows,

1. Port of Embarkation: C = Cherbourg, Q = Queenstown, S = Southampton
2. Sex: M = Male, F = Female
3. Survived: Y = Yes, N = No

ATTRIBUTES	TYPE	DESCRIPTION
Survived	Categorical	Survival
PassengerId	Integer	Passenger Number
Pclass	Integer	Ticket class
Sex	Categorical	Gender
Age	Numerical	Age in Years
SibSp	Integer	Siblings/Spouse
Parch	Integer	Parents/Children
Ticket	Integer	Ticket number
Fare	Numerical	Ticket Fare
Embarked	Categorical	Port of Embarkation

Fig 1: Attributes, Type, Description

The picture describes various attributes, attribute type and their description.

Link: https://www.kaggle.com/c/titanic/data

3. Research Problems and Solutions

Prediction of Survival rate on the Titanic, left thousands of passenger's dead. As shipwreck led to such loss of life that lack of many life jackets, boats etc. It is as predicted, that a set of people could survive more than others. To support our hypothesis, we use an existing data set which contains the case statuses (Survived = Yes or No), provided by OFLC and a set of feature-parameters where the cases are filed. We then analyse and interpret the factors that affect various cases statuses like, Survived or Not Survived

Feature selection that is which feature is more significant while classifying Survived. This will be a major research problem

1.Complete analysis of what sort of people were likely to survive

2.Tuning of various parameters of each model will be a research problem (logistic regression model with evaluations)

3.Apply Random forest classification Algorithm

Approach for logistic regression:

1. Load data, and run numerical and graphical summaries
2. Split the data into training and testing
3. Fit a logistic regression model using training data
4. Use the fitted model to do predictions for the test data
5. Create a confusion matrix, and compute the misclassification rate

4. Model Learning

4.1. Data Processing

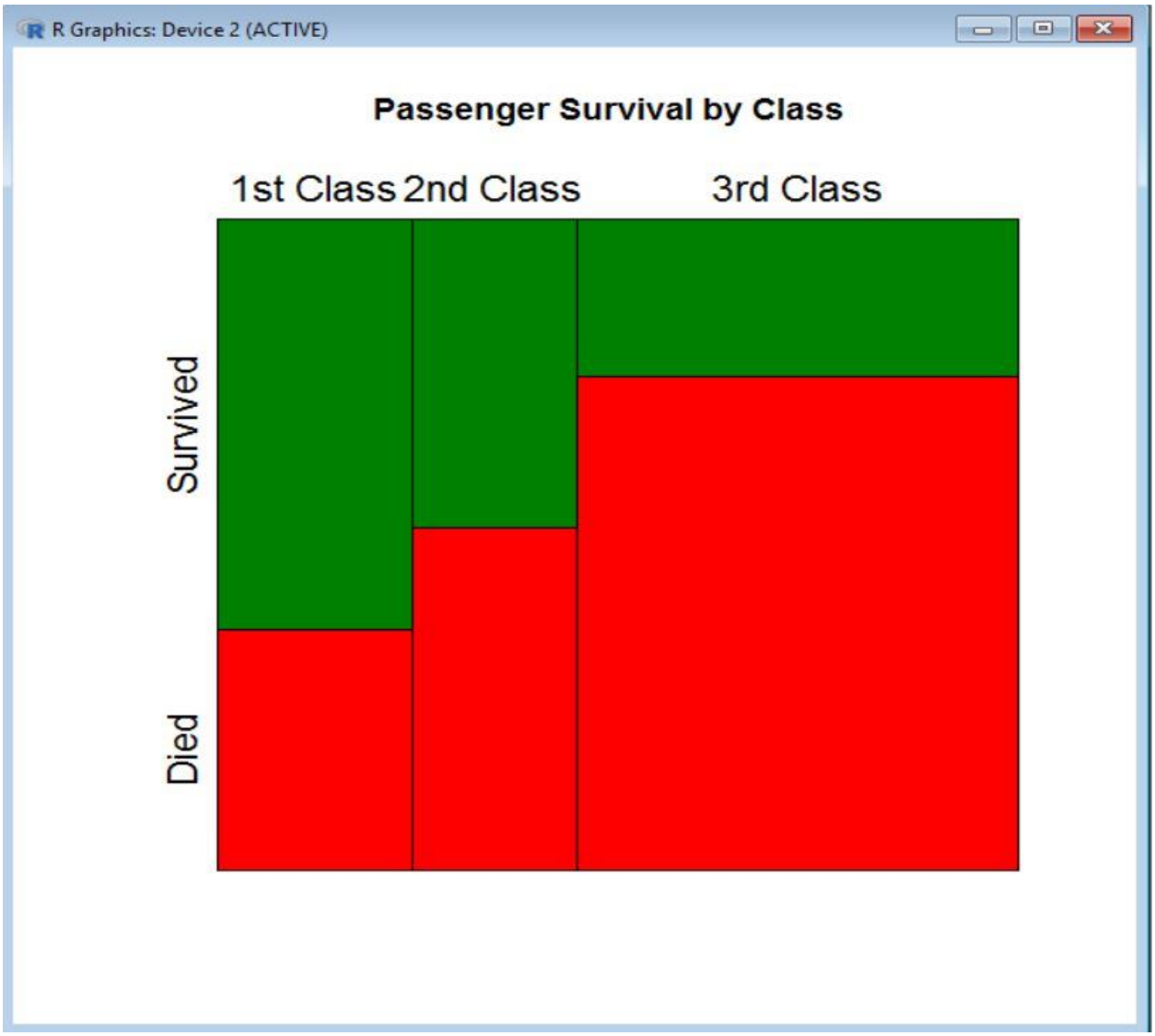
After Analyzing the data, there were few missing values that have been found. The missing values present in the column age were replaced with the mean value and also, the missing values were filled in the Fare by replacing with the mean value. The column “Passenger Name” has been removed from the data set as it does not have any significance on survival.

4.2. Data Analytics Tasks and Processes

4.2.1 Visualizations to analyze the data

We interpret data, through these visual plots to find the number passengers who survived and in particular if they were mwn or women who were on board.

1) Mosaic Plot of Survival Rate vs Passenger Rate by class



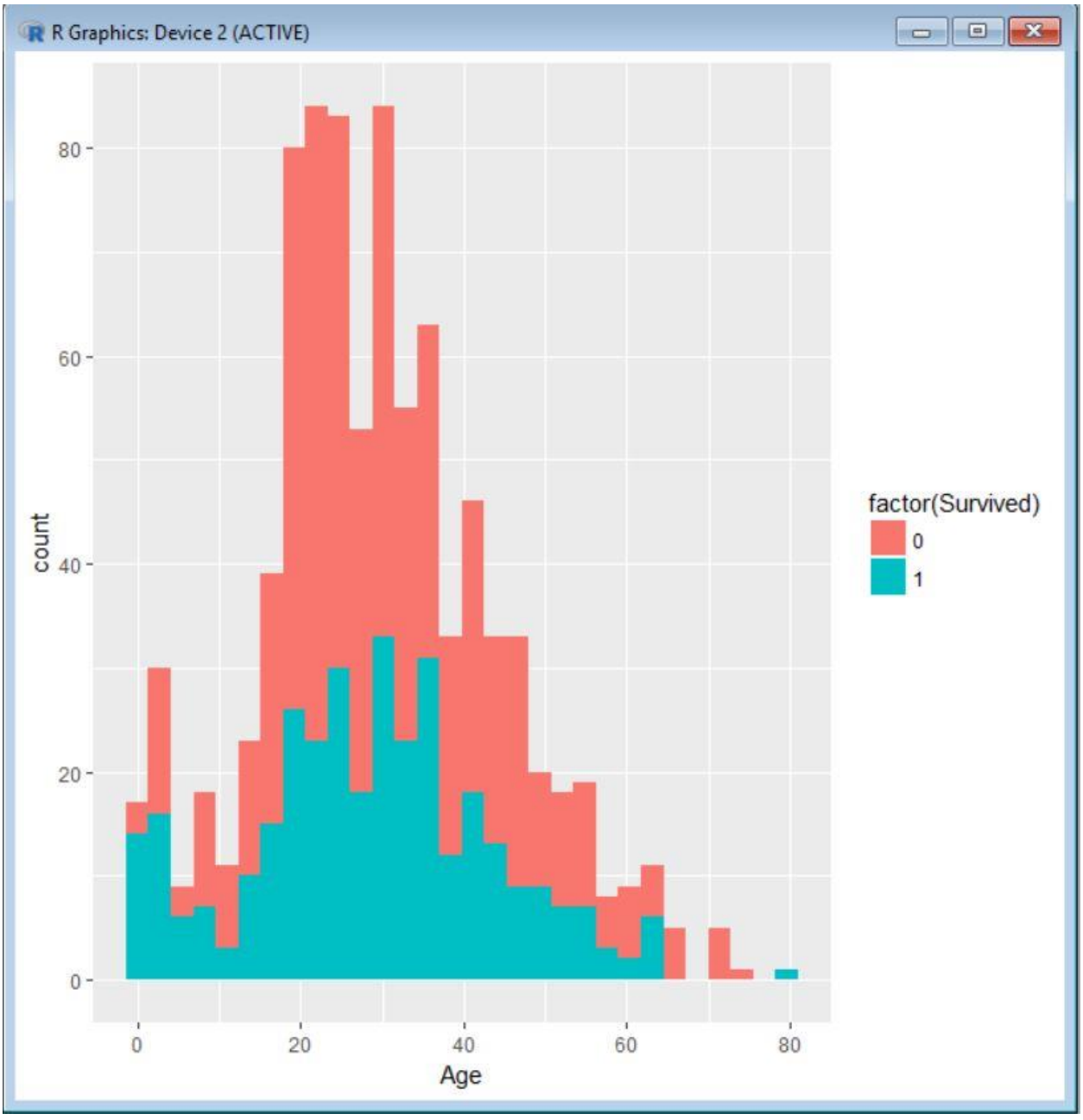
Representation:

Red: Died

Green: Survived

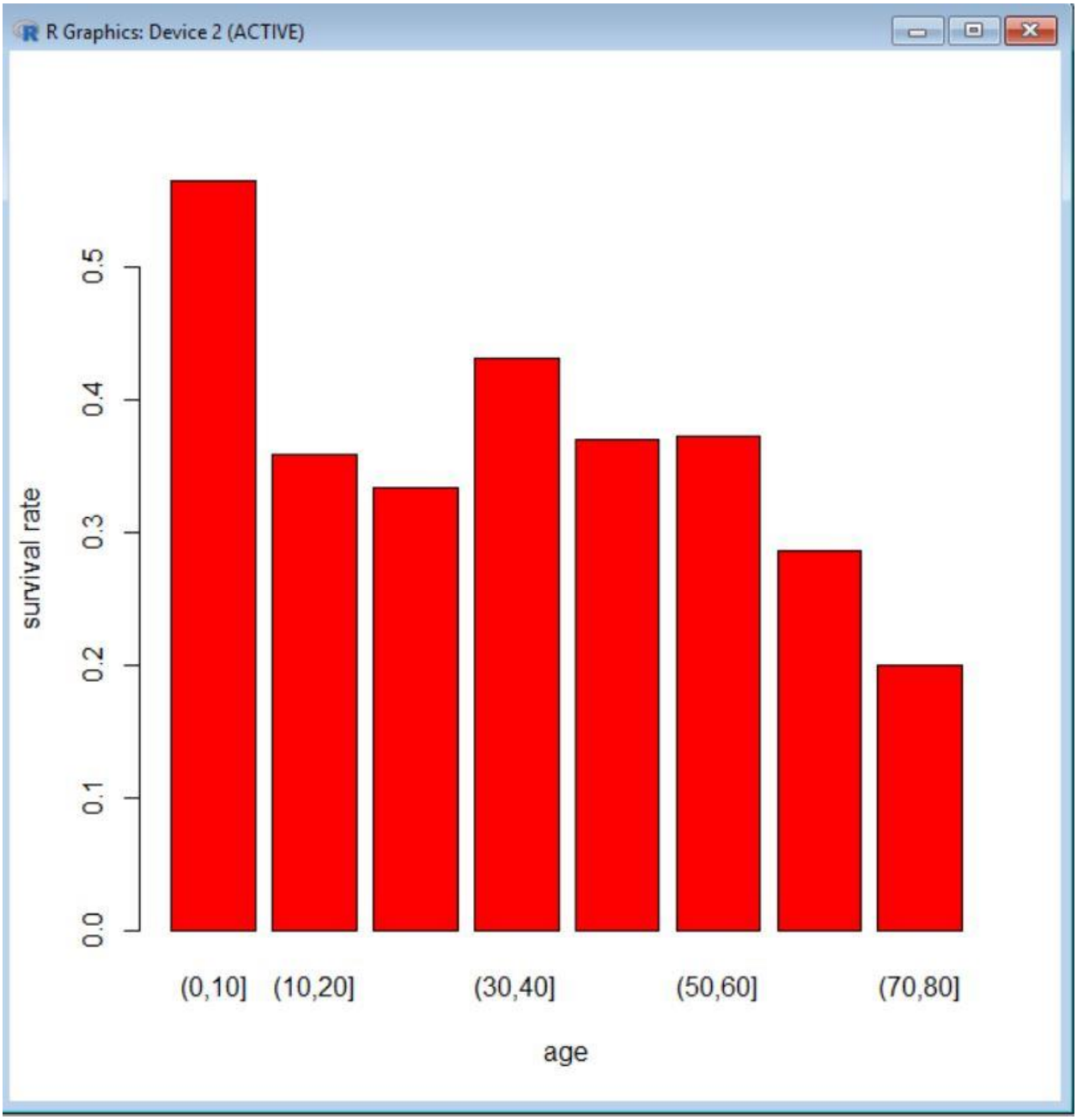
Mosaic plot shows that First class passengers could survive more than second and 3rd class passengers. In other words, third class passengers survival rate is lowest.

2) GGPLOT + HISTOGRAM FOR SURVIVAL RATE V/S AGE:



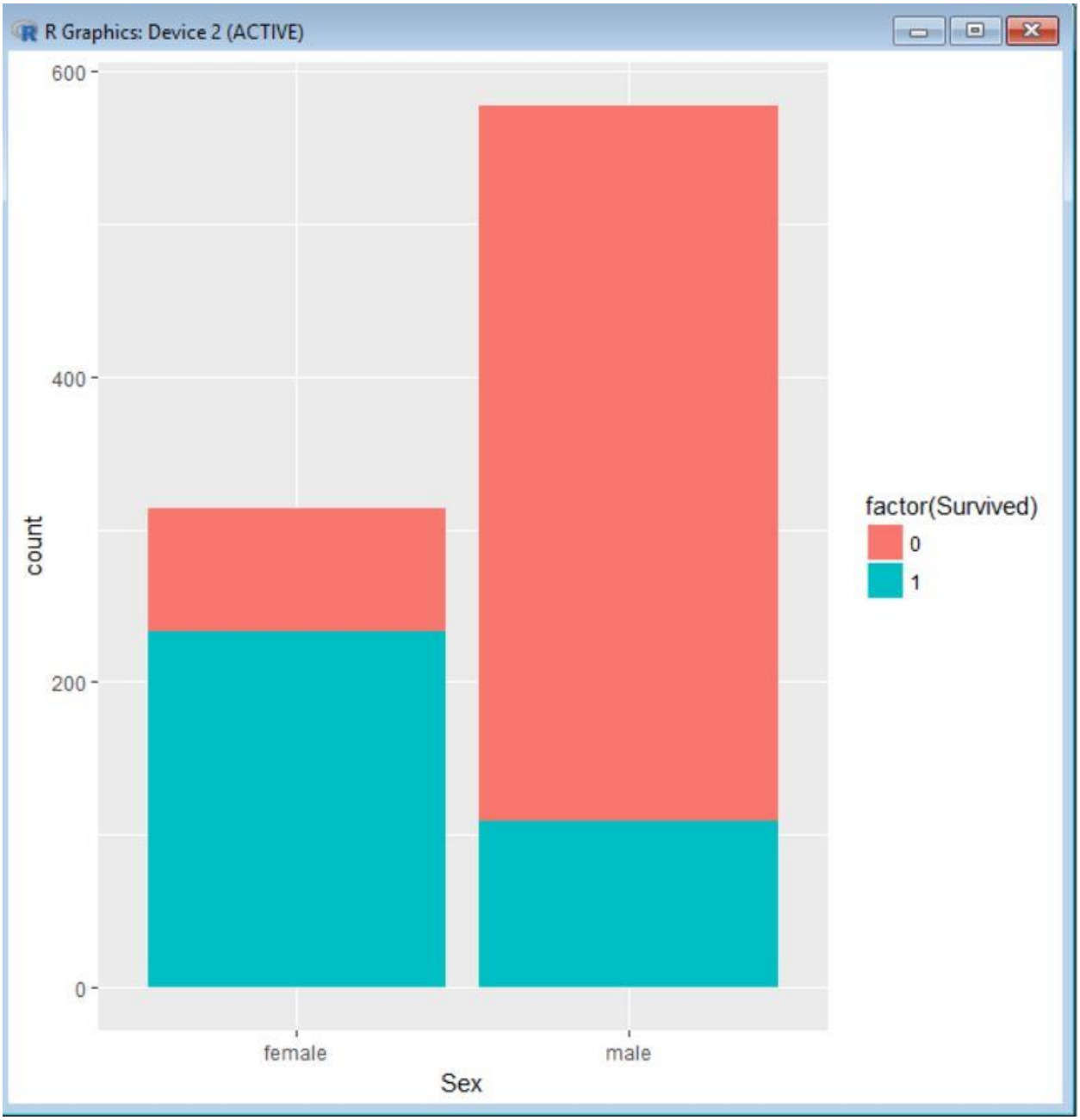
From the above plot we conclude that people who lie between the range 20-40 could survive more than people whose aged is above 40.

3)BAR PLOT FOR AGE V/S SURVIVAL RATE:



From the plot we observe that as age increases the survival rate decreases and also, as mentioned above 70-80 age group people could merely survive.

4) GGPLOT + HISTOGRAM FOR SEX V/S SURVIVAL RATE:



From the this plot we interpret that more number of female passengers survive than male passengers travelling onboard.

4.2.2 Regression Models

The regression models used in the analysis are:

1) Backward Elimination
2) Step-wise Regression
3) Random Forest Algorithm

- Backward Elimination:

The backward elimination technique starts from the full model including all independent effects. Then effects are deleted one by one until a stopping condition is satisfied. At each step, the effect showing the smallest contribution to the model is deleted. In traditional implementations of backward elimination, the contribution of an effect to the model is assessed by using a statistic. At any step, the predictor producing the least significant statistic is dropped and the process continues until all effects remaining in the model have statistics significant at a stay significance level (SLS). Here the p – value is chosen as the metric.

```
fit = glm(as.factor(Survived) ~ ., family = binomial(), data = train.data
```

summary(fit)

```
Call:
glm(formula = as.factor(Survived) ~ ., family = binomial(), data = train.data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.5817  -0.6437  -0.4003   0.6408   2.4744

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  1.610e+01  5.354e+02   0.030   0.9760
PassengerId -7.419e-05  3.856e-04  -0.192   0.8474
Pclass      -9.581e-01  1.708e-01  -5.608 2.05e-08 ***
Sexmale     -2.621e+00  2.203e-01 -11.895  < 2e-16 ***
Age         -2.772e-02  8.786e-03  -3.155   0.0016 **
SibSp       -2.768e-01  1.180e-01  -2.346   0.0190 *
Parch       -1.076e-01  1.311e-01  -0.821   0.4118
Ticket      -7.831e-04  3.794e-04  -2.064   0.0390 *
Fare         2.974e-03  2.882e-03   1.032   0.3020
EmbarkedC   -1.133e+01  5.354e+02  -0.021   0.9831
EmbarkedQ   -1.130e+01  5.354e+02  -0.021   0.9832
EmbarkedS   -1.169e+01  5.354e+02  -0.022   0.9826
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 952.58  on 711  degrees of freedom
Residual deviance: 639.19  on 700  degrees of freedom
AIC: 663.19

Number of Fisher Scoring iterations: 12
```

From the above analysis, we can interpret that the variable 'Embarked' as least significance on dependent variable as the p – value is more than 0.05

fit1 = glm(as.factor(Survived) ~ PassengerId + Pclass + as.factor(Sex) + Age + SibSp + Parch + Ticket + Fare, family = binomial(), data = train.data)

summary(fit1)

```
Call:
glm(formula = as.factor(Survived) ~ PassengerId + Pclass + Sex +
    Age + SibSp + Parch + Ticket + Fare, family = binomial(),
    data = train.data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.6417  -0.6359  -0.4155   0.6268   2.4329

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  4.481e+00  6.160e-01   7.274 3.49e-13 ***
PassengerId -8.226e-05  3.839e-04  -0.214  0.83033
Pclass      -9.385e-01  1.667e-01  -5.629 1.81e-08 ***
Sexmale     -2.663e+00  2.182e-01 -12.207  < 2e-16 ***
Age         -2.761e-02  8.732e-03  -3.162  0.00156 **
SibSp       -2.972e-01  1.180e-01  -2.518  0.01180 *
Parch       -1.261e-01  1.302e-01  -0.969  0.33260
Ticket      -7.671e-04  3.775e-04  -2.032  0.04213 *
Fare         3.850e-03  2.870e-03   1.341  0.17980
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 952.58  on 711  degrees of freedom
Residual deviance: 642.11  on 703  degrees of freedom
AIC: 660.11

Number of Fisher Scoring iterations: 5
```

From the above analysis, we can interpret that the variable 'PassengerId' as least significance on dependent variable as the p – value is more than 0.05

fit2 = glm(as.factor(Survived) ~ Pclass + Sex + Age + SibSp + Parch + Ticket + Fare, family = binomial(), data = train.data)

summary(fit2)

```
Call:
glm(formula = as.factor(Survived) ~ Pclass + Sex + Age + SibSp +
    Parch + Ticket + Fare, family = binomial(), data = train.data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.6378  -0.6306  -0.4155   0.6247   2.4339

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  4.4468805  0.5948533   7.476 7.69e-14 ***
Pclass      -0.9380300  0.1667396  -5.626 1.85e-08 ***
Sexmale     -2.6652521  0.2180398 -12.224  < 2e-16 ***
Age         -0.0276943  0.0087249  -3.174   0.0015 **
SibSp       -0.2964746  0.1178878  -2.515   0.0119 *
Parch       -0.1278519  0.1300142  -0.983   0.3254
Ticket      -0.0007658  0.0003775  -2.029   0.0425 *
Fare         0.0038573  0.0028761   1.341   0.1799
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 952.58  on 711  degrees of freedom
Residual deviance: 642.16  on 704  degrees of freedom
AIC: 658.16

Number of Fisher Scoring iterations: 5
```

From the above analysis, we can interpret that the variable 'Parch' as least significance on dependent variable as the p – value is more than 0.05.

fit3 = glm(as.factor(Survived) ~ Pclass + Sex + Age + SibSp + Ticket + Fare, family = binomial(), data = train.data)

summary(fit3)

```
Call:
glm(formula = as.factor(Survived) ~ Pclass + Sex + Age + SibSp +
    Ticket + Fare, family = binomial(), data = train.data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.6779  -0.6286  -0.4133   0.6303   2.4383

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  4.4356090  0.5923686   7.488 7.00e-14 ***
Pclass      -0.9569224  0.1650665  -5.797 6.74e-09 ***
Sexmale     -2.6237429  0.2131577 -12.309  < 2e-16 ***
Age         -0.0274875  0.0087018  -3.159  0.00158 **
SibSp       -0.3281841  0.1141878  -2.874  0.00405 **
Ticket      -0.0007542  0.0003768  -2.001  0.04536 *
Fare         0.0032977  0.0027656   1.192  0.23311
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 952.58  on 711  degrees of freedom
Residual deviance: 643.15  on 705  degrees of freedom
AIC: 657.15

Number of Fisher Scoring iterations: 5
```

From the above analysis, we can interpret that the variable 'Fare' as least significance on dependent variable as the p – value is more than 0.05.

fit4 = glm(as.factor(Survived) ~ Pclass + Sex + Age + SibSp + Ticket, family = binomial(), data = train.data)

summary(fit4)

```
Call:
glm(formula = as.factor(Survived) ~ Pclass + Sex + Age + SibSp +
    Ticket, family = binomial(), data = train.data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.5906  -0.6438  -0.4104   0.6231   2.4405

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  4.7806743  0.5202793   9.189  < 2e-16 ***
Pclass      -1.0722836  0.1356065  -7.907 2.63e-15 ***
Sexmale     -2.6374407  0.2125975 -12.406  < 2e-16 ***
Age         -0.0287067  0.0086209  -3.330 0.000869 ***
SibSp       -0.2980400  0.1104885  -2.697 0.006987 **
Ticket      -0.0006554  0.0003666  -1.788 0.073848 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 952.58  on 711  degrees of freedom
Residual deviance: 644.66  on 706  degrees of freedom
AIC: 656.66

Number of Fisher Scoring iterations: 5
```

From the above analysis, we can interpret that the variable 'Ticket' as least significance on dependent variable as the p – value is more than 0.05.

fit5 = glm(as.factor(Survived) ~ Pclass + Sex + Age + SibSp, family = binomial(), data = train.data)

Summary(fit5)

```
Call:
glm(formula = as.factor(Survived) ~ Pclass + Sex + Age + SibSp,
    family = binomial(), data = train.data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.5299  -0.6626  -0.4247   0.6339   2.3572

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  4.663283   0.512970   9.091  < 2e-16 ***
Pclass      -1.141962   0.130495  -8.751  < 2e-16 ***
Sexmale     -2.626275   0.211640 -12.409  < 2e-16 ***
Age         -0.029446   0.008562  -3.439 0.000584 ***
SibSp       -0.303761   0.108899  -2.789 0.005281 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 952.58  on 711  degrees of freedom
Residual deviance: 647.87  on 707  degrees of freedom
AIC: 657.87

Number of Fisher Scoring iterations: 5
```

From the above analysis, we can conclude that the above fit is the best model in backward regression as the p-value for each independent variable is less than 0.05

- Step-wise Regression:

The Step wise regression, works by comparing the AIC improvements from dropping each candidate variable, and adding each candidate variable between the upper and lower bound repressor sets supplied, from the current model, and by dropping or adding the one variable that leads to the best AIC improvement.

```
> step(fit, direction = "both")
Start:  AIC=639.81
as.factor(Survived) ~ PassengerId + Pclass + Sex + Age + SibSp +
    Parch + Ticket + Fare + Embarked

             Df Deviance    AIC
- PassengerId  1   617.92 637.92
- Parch        1   618.02 638.02
- Fare         1   618.11 638.11
- Embarked     2   621.58 639.58
<none>             617.81 639.81
- Ticket       1   620.35 640.35
- SibSp        1   625.89 645.89
- Age          1   640.35 660.35
- Pclass       1   647.86 667.86
- Sex          1   823.94 843.94

Step:  AIC=637.92
as.factor(Survived) ~ Pclass + Sex + Age + SibSp + Parch + Ticket +
    Fare + Embarked

             Df Deviance    AIC
- Parch        1   618.11 636.11
- Fare         1   618.21 636.21
- Embarked     2   621.72 637.72
<none>             617.92 637.92
- Ticket       1   620.50 638.50
+ PassengerId  1   617.81 639.81
- SibSp        1   626.17 644.17
- Age          1   640.45 658.45
- Pclass       1   648.07 666.07
- Sex          1   824.12 842.12
```

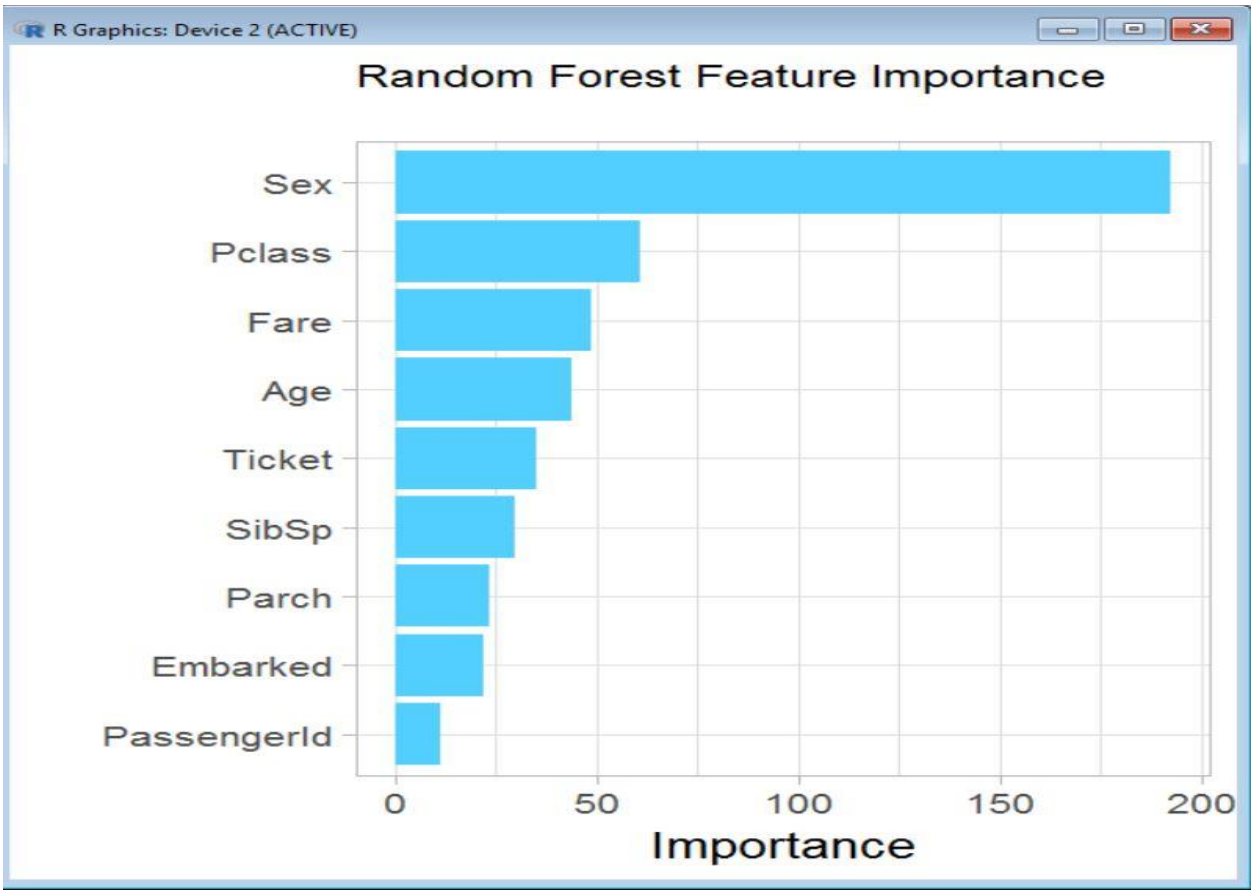
```
Step:  AIC=636.11
as.factor(Survived) ~ Pclass + Sex + Age + SibSp + Ticket + Fare +
    Embarked

             Df Deviance    AIC
- Fare        1   618.31 634.31
- Embarked    2   621.95 635.95
<none>            618.11 636.11
- Ticket      1   620.64 636.64
+ Parch       1   617.92 637.92
+ PassengerId 1   618.02 638.02
- SibSp       1   627.80 643.80
- Age         1   640.55 656.55
- Pclass      1   649.57 665.57
- Sex         1   831.44 847.44

Step:  AIC=634.31
as.factor(Survived) ~ Pclass + Sex + Age + SibSp + Ticket + Embarked

             Df Deviance    AIC
<none>            618.31 634.31
- Ticket      1   620.65 634.65
- Embarked    2   622.71 634.71
+ Fare        1   618.11 636.11
+ Parch       1   618.21 636.21
+ PassengerId 1   618.22 636.22
- SibSp       1   627.87 641.87
- Age         1   641.05 655.05
- Pclass      1   669.79 683.79
- Sex         1   835.06 849.06

Call:  glm(formula = as.factor(Survived) ~ Pclass + Sex + Age + SibSp +
    Ticket + Embarked, family = binomial(), data = train.data)

Coefficients:
(Intercept)       Pclass      Sexmale          Age        SibSp       Ticket
  5.6133328   -1.0107647   -2.8795257   -0.0399525   -0.3423583   -0.0005786
  EmbarkedQ    EmbarkedS
 -0.2239210   -0.5352442

Degrees of Freedom: 711 Total (i.e. Null);  704 Residual
Null Deviance:      949.9
Residual Deviance: 618.3        AIC: 634.3
```

From the above step wise analysis, we can conclude that the variables ‘Pclass’,’Sex’,’Age’,’SibSp’,’Ticket’ have significant effect on dependent variable.

- Random Forest Algorithm:

  The Random Forest algorithm is one of the most widely used machine learning algorithm for classification. It is also be used for regression models (Continuous target variable) but it mainly performs well on classification model. This algorithm, is operated by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. It gives an estimate of what variables are important in the classification. Also, It has an effective method for estimating missing data and maintains accuracy when a large proportion of the data are missing. The, advantage by using this algorithm is ,it reduces the chances of overfitting. High Model Performance and Accuracy are observed by using this algorithm.

5) The Feature Importance Plot:



From the above plot we can infer that the variable ‘Sex’ has the highest significance on independent variable.

5. Evaluations and Results

5.1. Evaluation Methods

We have divided the data into training (80%) and testing (20%) by hold-out validation. To find out the best model between Stepwise Regression and Backward Elimination model's we checked for the value of Pseudo R square.

- Pseudo R^2:
  Unlike linear regression with ordinary least squares estimation, there is no R2 statistic which explains the proportion of variance in the dependent variable that is explained by the predictors. However, there are a number of pseudo R2 metrics that could be of value. Most notable is McFadden's R2, which is defined as 1−[ln(LM)/ln(L0)] where ln(LM) is the log likelihood value for the fitted model and ln(L0) is the log likelihood for the null model with only an intercept as a predictor. The measure ranges from *0* to just under *1*, with values closer to zero indicating that the model has no predictive power.

Pseudo R^2 value for Backward Elimination Best Model:

```
> fit5 = glm(as.factor(Survived) ~ Pclass + Sex + Age + SibSp, family = binomial(), data = train.data)
> 1-logLik(fit5)/logLik(nullmod)
'log Lik.' 0.4735938 (df=5)
```

Pseudo R^2 value for Step-wise Regression Best Model:

```
> fit6 = glm(as.factor(Survived) ~ Pclass + Sex + Age + SibSp + Ticket + Embarked, family = binomial(), data = train.data)
> nullmod = glm(Survived~1,data=data,family=binomial())
> 1-logLik(fit6)/logLik(nullmod)
'log Lik.' 0.4789454 (df=8)
```

From the above values, we can see that Pseudo R^2 value for Stepwise Regression Best Model is Higher than Backward Elimination Best Model, So we take Stepwise model for prediction:

```
> Prediction = predict(fit6,test.data)
> Prediction
        240         449         139         244         112          89
-1.51754478  0.68596642 -1.81891123 -2.14058854  0.78549953  2.52953400
        795           6         392         737         603         796
-2.00499719 -2.09428776 -1.90913608 -0.61226745  0.09983819 -1.39647117
        129         248         545         322         815         161
 0.31362558  1.87149793 -1.19370969 -2.07958192 -2.24186501 -2.68859671
        339         562         525         840         851         288
-2.77981786 -2.52425744 -2.05530880  0.04341197 -2.92882428 -1.90816706
         18           1         494          19         872         883
-1.03371352 -2.46771039 -1.65602713  0.16863911  1.85916386  0.70921386
        679         184         312         698         776         682
-0.31930482 -0.82264248  2.72920578  0.51365659 -1.75841563 -0.09961360
        325          69         189         687         742         303
-5.52440708 -0.64758151 -2.95947944 -3.15732766 -0.43407375 -2.00458351
        573          35         615         179         784         669
-0.57289481 -0.55049161 -2.39184722 -1.05693880 -2.77448225 -2.72975412
        547          63         559         488         885         688
 1.59572982 -0.99801918  2.25927410 -0.88523308 -2.23379159 -1.81361464
        110         731         641         808         151         790
 0.10616954  3.39347938 -1.88110439  0.96094115 -2.09538309 -0.71833458
        210         397         852         480          27         540
-0.25770608  0.47642970 -3.58699727  1.57243194 -2.02355832  3.08225852
        421          10         548         635         520         782
-2.18773907  1.85446428 -1.39837754  0.04406260 -2.25656974  2.83646146
        195          31         857         203         308         365
 1.87407965 -0.71384977  2.02526930 -2.21700769  2.62424970 -2.64190911
        222         665         193         623         167         609
-0.90883751 -2.50156154  0.44867638 -2.10742034  2.88240939  1.25264412
         99         759         501         838         414         757
 1.61433485 -2.35455485 -1.66373213 -2.25962413 -1.06838658 -2.14064600
          5         675         564         133         418         384
-2.42361305 -1.07005846 -2.38501556 -0.54671085  2.06135036  2.25950678
        600         612         469         464         214          17
-1.31995639 -2.37863739 -2.21232672 -1.60894023 -1.05303774 -2.98246957
        111         390         890         655         264         144
-0.40023951  1.74054981  0.20388980  0.87917534 -0.37039620 -1.87682517
        809         867         544         626         359         728
-1.34129894  0.98386192 -1.55483605 -1.28067333  0.62286215  0.48803635
        142         789         388         676         420         590
```

We are supposed to get either “Yes” or “No” but we got numerical values. How do we interpret them?

As these values are negative, we can interpret that these are the values of log(odds)

$$\log\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 x$$

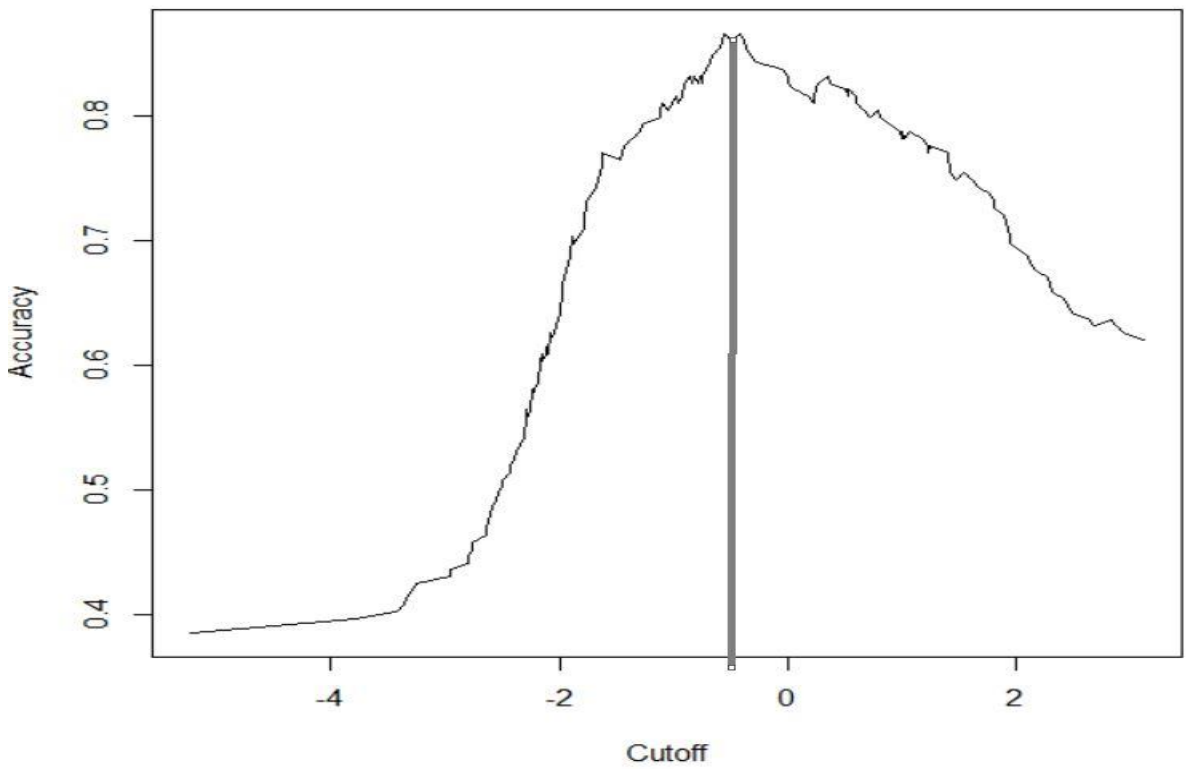
the logistic regression model is linear in log(odds). To interpret these values, we need to find the threshold value for the predicted values. If the values are greater than the threshold value they can be considered as “Yes” or “No”.

Prediction = predict(fit6,test.data)

Pred = prediction(Prediction,test.data$Survived)

acc.perf = performance(Pred, measure = "acc")

plot(acc.perf)



From the plot we can observe that the optimal threshold value will be as -0.421 as the accuracy of the prediction is high at that point.

```
> Prediction[Prediction > -0.421] = "YES"
> Prediction[Prediction < -0.421] = "NO"
```

And then computing misclassification rate for accuracy by creating confusion matrix

- Confusion Matrix:

In the field of machine learning and specifically the problem of statistical classification, a confusion matrix, also known as an error matrix, is a specific table layout that allows visualization of the performance of an algorithm, typically a supervised learning one

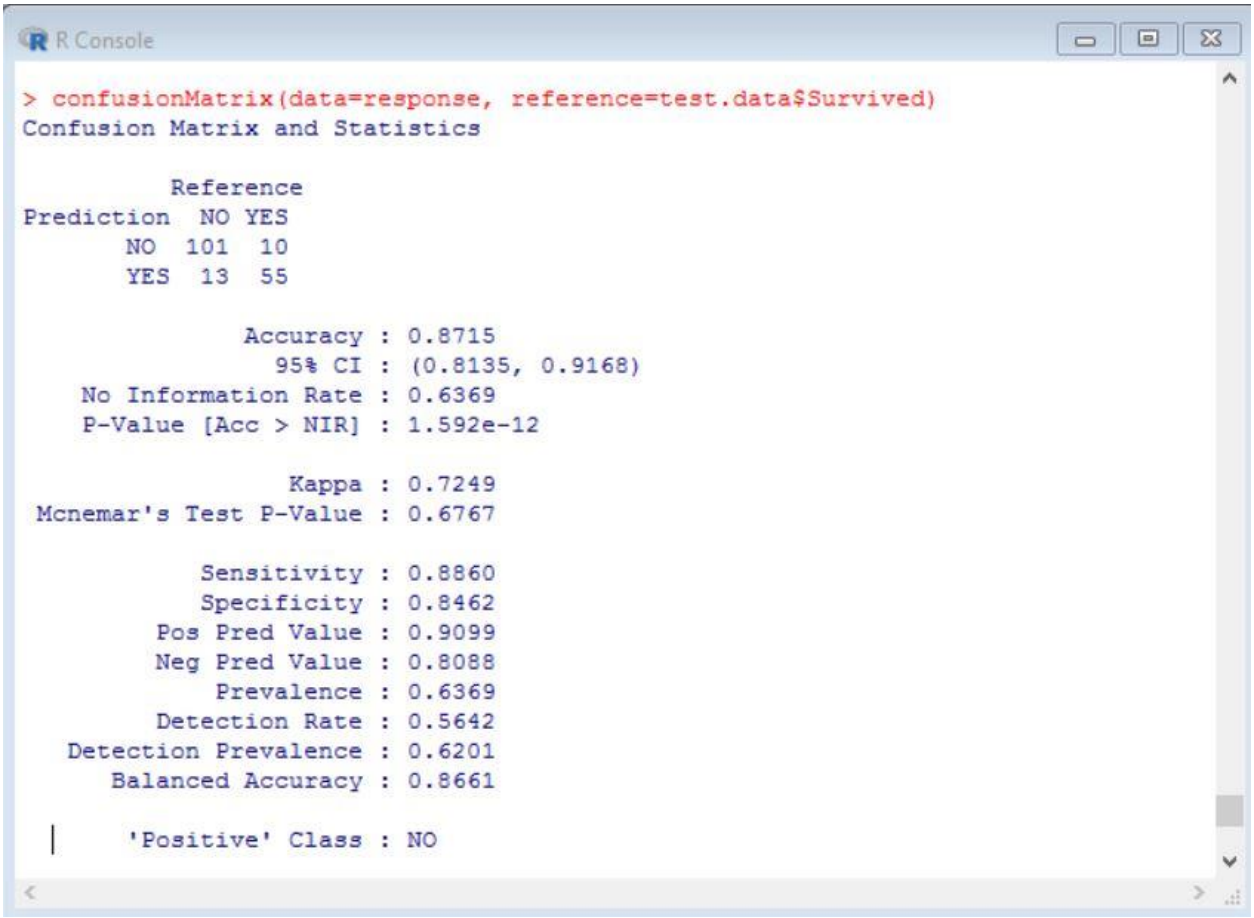
(in unsupervised learning it is usually called a matching matrix). Each column of the matrix represents the instances in a predicted class while each row represents the instances in an actual class (or vice versa)

```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 106  35
         1   4  34

               Accuracy : 0.7821
                 95% CI : (0.7144, 0.8402)
    No Information Rate : 0.6145
    P-Value [Acc > NIR] : 1.268e-06

                  Kappa : 0.4981
 Mcnemar's Test P-Value : 1.556e-06

            Sensitivity : 0.9636
            Specificity : 0.4928
         Pos Pred Value : 0.7518
         Neg Pred Value : 0.8947
             Prevalence : 0.6145
         Detection Rate : 0.5922
   Detection Prevalence : 0.7877
      Balanced Accuracy : 0.7282

       'Positive' Class : 0
```

From the above picture, we can analyze that the accuracy of the model is 78.21% and at 95% C.I, Accuracy may vary from 71.44% to 84.02%

Now by, computing the miss-classification rate for the random Forest fit

```
fit7 = randomForest(as.factor(Survived) ~ ., data = train.data, importance = TRUE, ntree=2000)
response = predict(fit7, test.data)
confusionMatrix(data=response, reference=test.data$Survived)
```



From the above picture, we can analyze that the accuracy of the model is 87.15% and at 95% C.I, Accuracy may vary from 81.35% to 91.68%. From the above evaluation, we conclude that random forest fit gives highest accuracy for the model, and hence it has high predictive power.

5.2. Results and Findings

These are some of the results and Findings, that we conclude from our analysis,

1) The accuracy for the random Forest fit is 87.15% and accuracy for the stepwise fit is 78.21%

2) Pseudo R square value for Stepwise fit is 0.473 and for Backward Regression fit is 0.478

3)Potentially Stepwise fit may have overfitting problem

4)From another perspective, our data is too small. The results by hold-out evaluation are probably not that reliable

5) Threshold value has been calculated to interpret the numerical values of prediction

6) Common metric used to determine the accuracy of random Forest fir and logistic fit is done by creating confusion Matrix

7) dev.off() is used for to delete the previous plot outputs.

8) We need to further use N-fold cross validation to provide reliable evaluations.

6. Conclusions and Future Work

6.1. Conclusions

1) From the above analysis and visualizations, we conclude that people who were young could survive more than the people who were old. And also, as the age increased survival rate decreased.

2)Also, by comparing the accuracies of random forest fit and stepwise fit we come to a conclusion that Random forest fit has the highest accuracy.

3)The Stepwise Model has the best Pseudo R square value compare to Backward regression model so it has high predictive power.

6.2. Limitations

1) Facts like families sink or swim together, were difficult to interpret.

2) It was difficult to interpret whether a particular passenger survived or not. That is, Identifying the passenger by Name.

3) Missing values were filled by calculating the mean value and sometimes the prediction imputations with this data may not be accurate.

4) Large sets of data, couldn't be handled in R as it throws an error "Vector size exceeded" or "Cant allocate Vector size of 450 GB"

6.3. Potential Improvements or Future Work

Accuracy of the models, can be improved by using machine leaning algorithms such as SVM(Support Vector Machine) and other classification algorithms. Also, by increasing the size of the dataset we can further interpret many facts such as "survival rate by name of the passenger" and "Survival Rate by Families" and many more.