**Appendix**

This section holds the experimental results and necessary code needed for understanding the proposed paper.

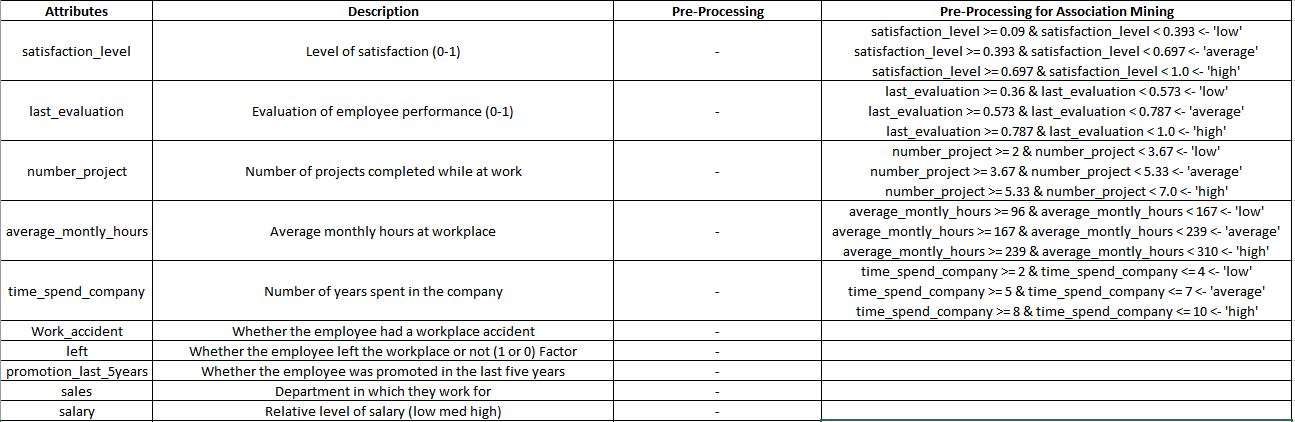


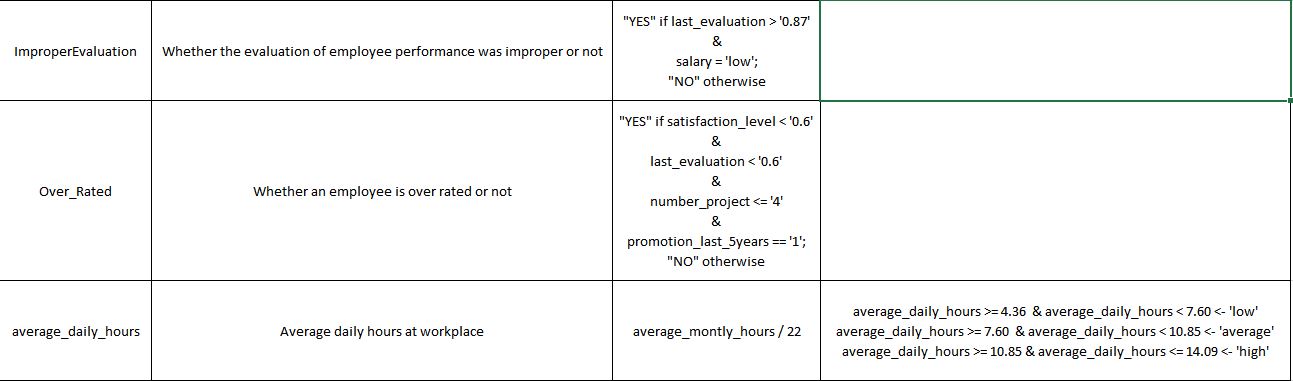
Dataset

***Pre-Processing***

***Table:***

**2.a**

****

****

***Codes***

**4.a**

# Read the input file

> HR<-read.csv("HR\_comma\_sep.csv")

> HR

> summary(HR)

# Data Pre-processing

# Employees with evaluation > 0.8 and salary as low

> mut1 <- HR$last\_evaluation >'0.87' & HR$salary=='low'

> HR[mut1, "ImproperEvaluation"] <- "Yes"

> HR

> HR[!mut1, "ImproperEvaluation"] <- "No"

> HR

# Employees with satisfaction <0.6, evaluation <0.6, number of projects <4 and got promoted

> mut2 <- HR$satisfaction\_level <'0.6' & HR$ last\_evaluation <'0.6'& HR$number\_project <='4' & HR$ promotion\_last\_5years =='1'

> HR[mut2, "Over\_Rated"] <- "Yes"

> HR

> HR[!mut2, "Over\_Rated"] <- "No"

> HR

# Calculate the average daily hours of every employee

> HR$average\_daily\_hours<-HR$average\_montly\_hours/22

> HR

# Round the average daily hours to two decimal places

> HR$average\_daily\_hours<-round(HR$average\_daily\_hours,digits=2)

> HR

> write.csv(HR, file = "foo1.csv", row.names = F)

# Converting Sales,salary ,promotion\_last\_5years, time\_spend\_company, and number\_project to factors

> Fsales<-as.factor(HR$sales)

> Fsalary<-as.factor(HR$salary)

> Fpromotion\_last\_5years<-as.factor(HR$promotion\_last\_5years)

> Ftimespent<-as.factor(HR$time\_spend\_company)

> Fnumber\_project<-as.factor(HR$number\_project)

> Fsalary<-ordered(HR$salary,levels=c("low","medium","high"))

**4.b**

# Data Exploration

> library(ggplot2)

> library(gridExtra)

# Analyze Salary:

> CSalary<-table(HR$salary)

> CSalary

# Interaction between sales and salary:

> psales<-ggplot(HR,aes(x=Fsales))+geom\_bar(fill="#FF00FF")+coord\_flip()

> psalary<-ggplot(HR,aes(x=Fsalary))+geom\_bar(fill="#FF00FF")+coord\_flip()

> psales

> psalary

> psales\_left<-ggplot(HR,aes(x=Fsales,fill=as.factor(left)))+geom\_bar(position="fill")+coord\_flip()+scale\_fill\_brewer(palette="PiYG")

> psalary\_left<-ggplot(HR,aes(x=Fsalary,fill=as.factor(left)))+geom\_bar()+coord\_flip()+scale\_fill\_brewer(palette="PiYG")

> psales\_left

> psalary\_left

> psalary\_sales<-ggplot(HR,aes(x=Fsales,fill=Fsalary))+geom\_bar(position="fill")+scale\_fill\_brewer(palette="PiYG")+coord\_flip()

>grid.arrange(psales,psalary,psales\_left,psalary\_left,psalary\_sales,ncol=2)

**4.c**

# Analyse number of people promoted

> Cpromoted<-table(HR$promotion\_last\_5years)

> Cpromoted

# Interaction between people promoted and salary

> ppromoted<-ggplot(HR,aes(x=Fpromotion\_last\_5years,fill=as.factor(Fsalary)))+geom\_bar(position="fill")

> ppromoted

**4.d**

# Analyze Promotion in last 5 years and Over rated employees

> pOver<-ggplot(HR,aes(x=Fpromotion\_last\_5years,fill=as.factor(Over\_Rated)))+geom\_bar(position="fill")

> pOver

**4.e**

# Analyse time spent at the company

> Ctimespent<-table(HR$time\_spend\_company)

> Ctimespent

**4.f**

# Interaction between the number of employees promoted and time spent at the company

> ppromoted\_timespent<-ggplot(HR,aes(x=Ftimespent,fill=as.factor(Fpromotion\_last\_5years)))+geom\_bar()

> ppromoted\_timespent

**4.g**

# Interaction between the number of projects and the employees who left

> pprojects\_left<-ggplot(HR,aes(x=Fnumber\_project,fill=as.factor(left)))+geom\_bar()

> pprojects\_left

**4.h**

# Interaction between satisfaction and ImproperEvaluation

> psatisfacttion\_Overrated<-ggplot(HR,aes(x=satisfaction\_level,fill=as.factor(ImproperEvaluation)))+geom\_bar()

> psatisfacttion\_Overrated

**4.i**

# Interaction between Satisfaction levels and Salary

> ggplot(HR, aes(x = Fsalary, y = satisfaction\_level, fill = factor(left), colour = factor(left))) +

+ geom\_boxplot(outlier.colour = "black") + xlab("Salary") + ylab("Satisfacion level")

**4.j**

# Interaction between Time Spent in Company and Salary

> ggplot(HR, aes(x = Fsalary, y = time\_spend\_company, fill = factor(left), colour = factor(left))) +

+ geom\_boxplot(outlier.colour = NA) + xlab("Salary") + ylab("time\_spend\_company")

**4.k**

# Analyse only the employees who have left the company

> Cleft<-table(HR$left)

> Cleft

# Analyze based on employees whose average daily hours greater than 8

> Overwork <- subset(HR, average\_daily\_hours > 8 , select=c(left, salary))

> Overwork

> plot(Overwork)

**4.l**

# Visualization based on different departments and salary paid

> pdepart\_Salary<-ggplot(HR,aes(x=Fsalary,fill=as.factor(Fsales)))+geom\_bar()

> pdepart\_Salary

***Figures***

**4.a**

|  |
| --- |
|  |
| **Fig.4.a** |

**4.b**

|  |
| --- |
|  |
| **Fig.4.b** |

**4.c**

|  |
| --- |
|  |
| **Fig.4.c** |

**4.d**

|  |
| --- |
|  |
| **Fig.4.d** |

**4.e**

|  |
| --- |
|  |
| **Fig.4.e** |

**4.f**

|  |
| --- |
|  |
| **Fig.4.f** |

**4.g**

|  |
| --- |
|  |
| **Fig.4.g** |

**4.h**

|  |
| --- |
|  |
| **Fig.4.h** |

**4.i**

|  |
| --- |
|  |
| **Fig.4.i** |

**4.j**

|  |
| --- |
|  |
| **Fig.4.j** |

***Data Mining***

***5.1 Association***

***Codes:***

**5.1.a**

# Generating rules for the whole dataset

> rulea <- apriori(ppdata, parameter = list(minlen = 3, support = 0.1, confidence = 0.5))

> quality(rulea) <- round(quality(rulea), digits = 3)

> rulea.sorted <- sort(rulea, by = "confidence")

# To plot scatter plot

> plot(rulea)

**5.1.b**

# Part A: Analysis of the characteristics of employees who are still working {left = 0}.

> Rule1 <- apriori(ppdata, parameter = list(minlen = 3, support = 0.1, confidence = 0.5),

appearance = list(rhs = c("left=0"),

lhs = c("satisfaction\_level=high", "satisfaction\_level=low", "satisfaction\_level=average",

"average\_daily\_hours=average", "average\_daily\_hours=low", "average\_daily\_hours=high",

"last\_evaluation=high", "last\_evaluation=low", "last\_evaluation=average",

"number\_project=average", "number\_project=low", "number\_project=high",

"salary=medium", "salary=low", "salary=high",

"ImproperEvaluation=No", "ImproperEvaluation=Yes",

"Over\_Rated=No", "Over\_Rated=Yes"),

default = "none"))

> quality(Rule1) <- round(quality(Rule1), digits = 3)

> Rule1.sorted <- sort(Rule1, by = "confidence")

# To plot scatter plot

> plot(Rule1)

**5.1.c**

# Plotting distribution graph for Rule1

> plot(Rule1, method = "graph", control = list(type = "items"))

# Plotting Parallel co-ordinate plots

> plot(Rule1, method="paracoord", control=list(reorder=TRUE))

**5.1.d**

# Saving the rules to csv for manual analysis

> Rule1\_csv <- as(Rule1.sorted, "data.frame")

> write.csv(Rule1\_csv, "Rule1.csv")

**5.1.e**

# Part B: Analysis of the characteristics of employees who have left the company {left = 1}

> Rule2 <- apriori(ppdata, parameter = list(minlen = 3, support = 0.06, confidence = 0.5),

appearance = list(rhs = c("left=1"),

lhs = c("satisfaction\_level=high", "satisfaction\_level=low", "satisfaction\_level=average",

"average\_daily\_hours=average", "average\_daily\_hours=low", "average\_daily\_hours=high",

"last\_evaluation=high", "last\_evaluation=low", "last\_evaluation=average",

"number\_project=average", "number\_project=low", "number\_project=high",

"salary=medium", "salary=low", "salary=high",

"ImproperEvaluation=No", "ImproperEvaluation=Yes",

"Over\_Rated=No", "Over\_Rated=Yes"),

default = "none"))

> quality(Rule2) <- round(quality(Rule2), digits = 3)

> Rule2.sorted <- sort(Rule2, by = "confidence")

**5.1.f**

# Plotting distribution graph for Rule2

> plot(Rule2, method = "graph", control = list(type = "items"))

# Plotting Parallel co-ordinate plots

> plot(Rule2, method="paracoord", control=list(reorder=TRUE))

**5.1.g**

# Saving the rules to csv for manual analysis

> Rule2\_csv <- as(Rule2.sorted, "data.frame")

> write.csv(Rule2\_csv, "Rule2.csv")

***Figures***

**5.1.a**

Fig. 5.1.a is a scatter plot of rulea where each point is plotted for its corresponding support, confidence and lift values.

|  |
| --- |
| C:\Users\sachi\AppData\Local\Microsoft\Windows\INetCache\Content.Word\rulea_plot1.png |
| **Fig. 5.1.a** |

**5.1.b**

Shows the scatter plot for Rule1.

|  |
| --- |
| C:\Users\sachi\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Rule1_plot1.png |
| **5.1.b** |

**Fig. 5.1.c** is the distribution plot of Rule1 and **Fig. 5.1.d** is the parallel co-ordinate plot for Rule1.

|  |  |
| --- | --- |
| C:\Users\sachi\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Rule1_plot2.png | C:\Users\sachi\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Rule1_plot3.png |
| **Fig. 5.1.c** | **Fig. 5.1.d** |

**Fig. 5.1.e** Shows the scatter plot for Rule2.

|  |
| --- |
| C:\Users\sachi\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Rule2_plot1.png |
| **Fig. 5.1.e** |

**Fig. 5.1.f** is the distribution plot of Rule2 and **Fig. 5.1.g** is the parallel co-ordinate plot for Rule2.

|  |  |
| --- | --- |
| C:\Users\sachi\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Rule2_plot2.png | C:\Users\sachi\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Rule2_plot3_2.png |
| **Fig. 5.1.f** | **Fig. 5.1.g** |

***Tables***

**5.1.a**

Table 5.1.a shows some of the rules generated with its corresponding support, confidence and lift values.

|  |
| --- |
| C:\Users\sachi\AppData\Local\Microsoft\Windows\INetCache\Content.Word\rulea.png |
| **Table. 5.1.a** |

**5.1.b**

Table 5.1.b shows some of the rules with high support, confidence and lift values for RHS of {left=0}.

|  |
| --- |
| C:\Users\sachi\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Rule1.png |
| **Table. 5.1.b** |

**5.1.c**

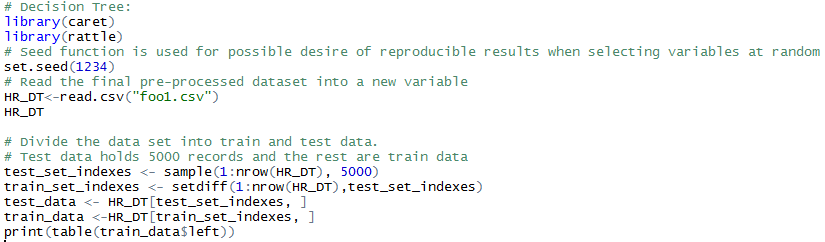
Table 5.1.c shows compilation of some of the rules with high support, confidence and lift values for RHS of {left=1}.

|  |
| --- |
| C:\Users\sachi\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Rule2.png |
| **Table. 5.1.c** |

**5.2 Decision Tree**

***Code***

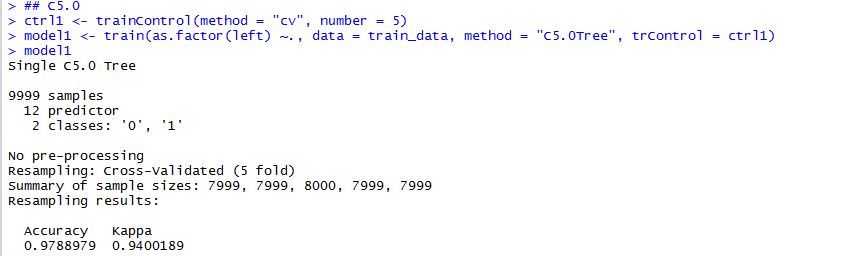
**5.2.a**



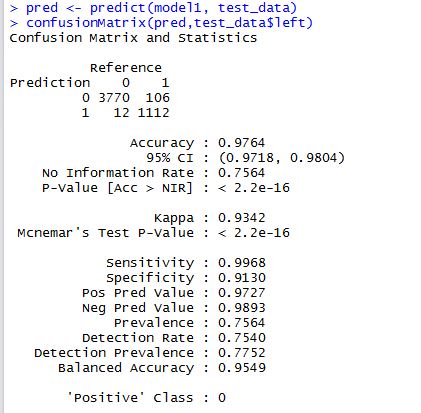
0 1

7646 2353

**5.2.b**



**5.2.c**



**5.2.d**

*model2<- rpart(factor(left) ~.,data=train\_data)*

*model2*

*pred1 <- predict(model2, test\_data)*

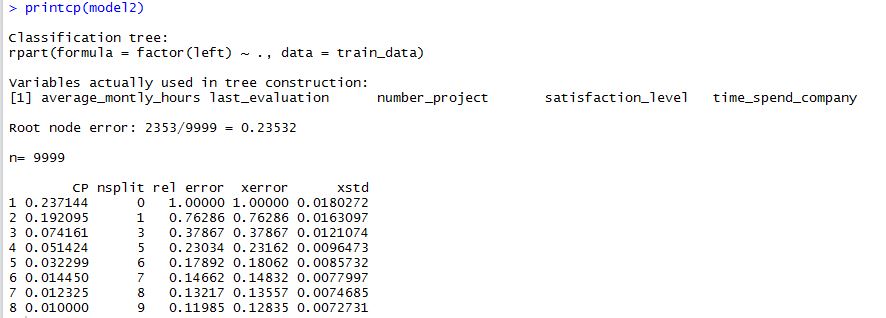
*auc(as.numeric(test\_data$left) - 1, pred1[, 2])*

*rpart.plot(model2, type = 2, cex = 1)*

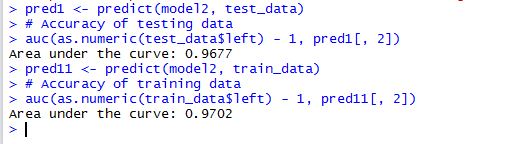
*printcp(model2)*

*fancyRpartPlot(model2)*

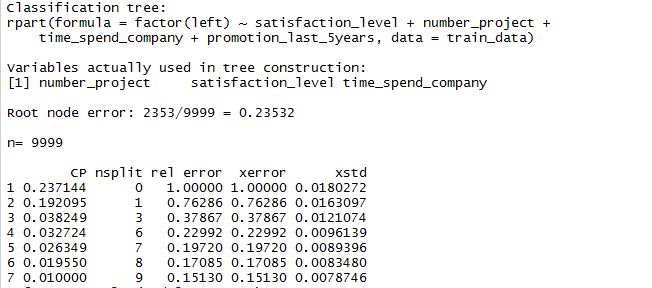
**5.2.e**



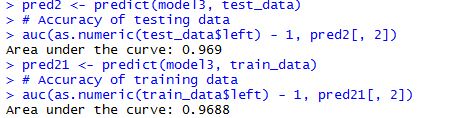
**5.2.f**



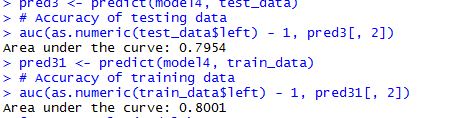
**5.2.g**



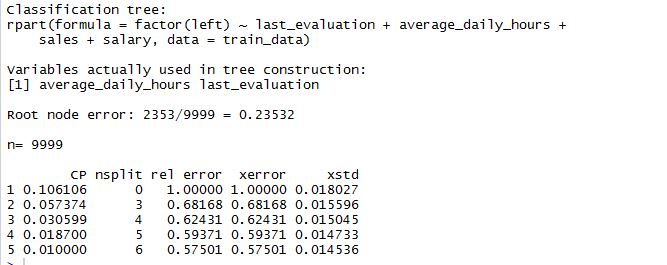
**5.2.h**



**5.2.i**



**5.2.j**



**5.2.k**

# Model 1- Random Forest tree creation with the pre processed data(Train set)

> rf\_mod <- randomForest(as.factor(left) ~. , data = trainSplit)

> importance(rf\_mod)

MeanDecreaseGini

satisfaction\_level 1298.7574102

last\_evaluation 454.0990003

number\_project 755.2870679

average\_montly\_hours 442.3187248

time\_spend\_company 750.0397972

Work\_accident 26.3129983

promotion\_last\_5years 5.2517508

sales 65.9388845

salary 32.4111914

ImproperEvaluation 38.2799883

Over\_Rated 0.1221354

average\_daily\_hours 448.5607197

**5.2.l**

# Prediction of the pre processed dataset's test data

> pred\_rf <- predict(rf\_mod,testSplit)

> summary(pred\_rf)

> confusionMatrix(pred\_rf,testSplit$left)

Confusion Matrix and Statistics

Reference

Prediction 0 1

0 2301 19

1 7 672

Accuracy : 0.9913

95% CI : (0.9873, 0.9943)

No Information Rate : 0.7696

P-Value [Acc > NIR] : < 2e-16

Kappa : 0.9754

Mcnemar's Test P-Value : 0.03098

Sensitivity : 0.9970

Specificity : 0.9725

Pos Pred Value : 0.9918

Neg Pred Value : 0.9897

Prevalence : 0.7696

Detection Rate : 0.7673

Detection Prevalence : 0.7736

Balanced Accuracy : 0.9847

'Positive' Class : 0

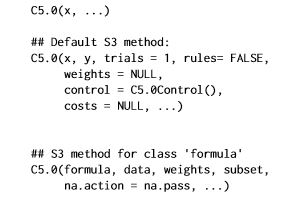
**5.2.m**

# Random Forest tree creation and prediction using the train set and test set of the modified data

|  |
| --- |
| > rf1\_mod <- randomForest(as.factor(left) ~. , data = n\_trainSplit)  > importance(rf1\_mod)  MeanDecreaseGini  satisfaction\_level 8.695780e+01  last\_evaluation 2.389231e+01  number\_project 1.576233e+02  average\_montly\_hours 7.733056e+01  time\_spend\_company 3.138239e+01  Work\_accident 1.096531e+00  promotion\_last\_5years 2.720205e-02  sales 9.156992e-01  salary 2.169073e+00  ImproperEvaluation 0.000000e+00  Over\_Rated 7.664682e-03  average\_daily\_hours 7.488100e+01  > pred\_rf1 <- predict(rf1\_mod,n\_testSplit)  > summary(pred\_rf1)  > confusionMatrix(pred\_rf1,n\_testSplit$left)  Confusion Matrix and Statistics  Reference  Prediction 0 1  0 92 1  1 1 177    Accuracy : 0.9926  95% CI : (0.9736, 0.9991)  No Information Rate : 0.6568  P-Value [Acc > NIR] : <2e-16    Kappa : 0.9836  Mcnemar's Test P-Value : 1    Sensitivity : 0.9892  Specificity : 0.9944  Pos Pred Value : 0.9892  Neg Pred Value : 0.9944  Prevalence : 0.3432  Detection Rate : 0.3395  Detection Prevalence : 0.3432  Balanced Accuracy : 0.9918    'Positive' Class : 0      **5.2.n**  > new\_HR1 <- HR[HR$satisfaction\_level < 0.6 & !salh & HR$last\_evaluation < 0.6 & prom\_wrk & (HR$sales == "sales" | HR$sales == "technical" | HR$sales == "support") ,] |

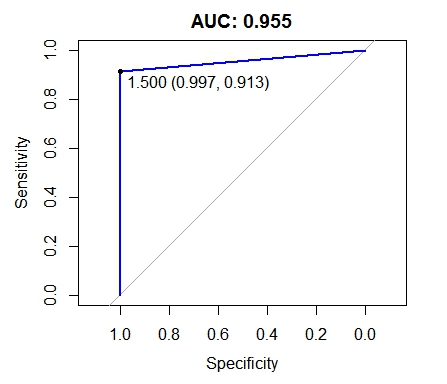
***Figures***

***5.2.a***



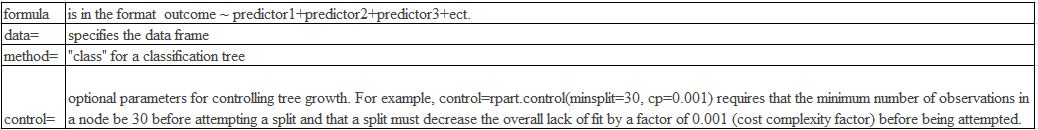
|  |
| --- |
| **Fig. 5.2.a** |

**5.2.b**



|  |
| --- |
| **Fig. 5.2.b** |

**5.2.c**



|  |
| --- |
| **Fig. 5.2.c** |

**5.2.d**

|  |
| --- |
|  |
| **Fig. 5.2.d** |

|  |
| --- |
|  |
| **Fig. 5.2.e** |

**5.2.f**

|  |
| --- |
|  |
| **Fig.4.2.f** |

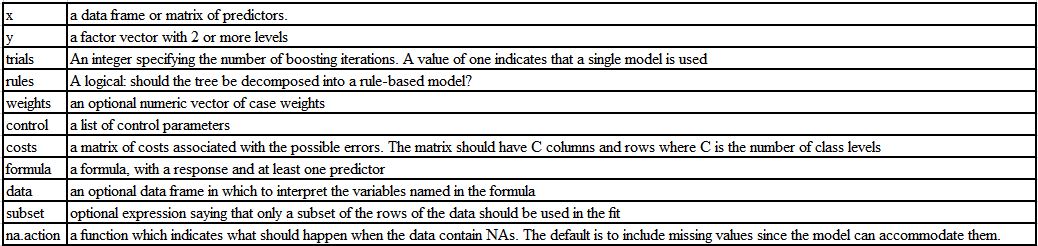
**5.2.g**

# ROC curves for given data and modified data respectively

|  |
| --- |
|  |
|  |
|  |
|  |

***Table***

***5.2.a***



|  |
| --- |
| **Table. 5.2.a** |

***5.3 Naïve Bayes***

***codes***

***5.3.a***

#Install packages e1071, tm for naive bayes, ggplot2 for graphs and caret for confusion matrix

>> install.packages("e1071")

>> install.packages("ggplot2")

>> install.packages("caret")

>> install.packages("tm")

#Load those libraries

>> library(e1071)

>> library(tm)

>> library(caret)

>> library(ggplot2)

# read the dataset

>> hra <-read.csv("C:/Users/maksh/Desktop/foo1.csv", stringsAsFactors <- TRUE)

#Remove the first column

>> hra[,1]<- NULL

# Display the contents of the dataset

>> str(hra)

>> summary(hra)

# Display the number of people who left the company

>> table(hra$left)

#split the dataset to 80% training and 20% test

>> split.hra <- floor(0.8 \* nrow(hra))

>> set.seed(1000)

>> train <- sample(seq\_len(nrow(hra)), size = split.hra)

>> train\_hra <- hra[train, ]

>> test\_hra <- hra[-train, ]

# Display the rows of training dataset and Test Dataset

>> nrow(train\_hra)

>> nrow(test\_hra)

#Display the no. of people left the company separately in training and test set

>> table(test\_hra$left)

>> table(train\_hra$left)

#Train the dataset using naive bayes classifier

>> model.bayes <- naiveBayes(as.factor(left) ~ ., data = train\_hra)

>> model.bayes

#Show the probability of salary w.r.t people left

>> prop.table(table(train\_hra$salary, train\_hra$left),2

# Plot the graph comparing various attributes with people who left the company

>> ggplot(train\_hra,aes(x=salary,fill=as.factor(left)))+geom\_bar(position = "fill")+coord\_flip()+scale\_fill\_brewer(palette = "PiYG")

>> ggplot(train\_hra,aes(x=sales,fill=as.factor(left)))+geom\_bar(position = "fill")+coord\_flip()+scale\_fill\_brewer(palette = "PiYG")

>> ggplot(train\_hra,aes(x=satisfaction\_level,fill=as.factor(left)))+ geom\_bar(position = "fill")+coord\_flip()+scale\_fill\_brewer(palette = "PiYG")

>> ggplot(train\_hra,aes(x=number\_project,fill=as.factor(left)))+ geom\_bar(position = "fill")+coord\_flip()+scale\_fill\_brewer(palette = "PiYG")

>> ggplot(train\_hra,aes(x=promotion\_last\_5years,fill=as.factor(left)))+ geom\_bar(position = "fill")+coord\_flip()+scale\_fill\_brewer(palette = "PiYG")

# predict the naive bayes model stats for training data set

>> res\_nb=predict(model.bayes,train\_hra)

# Confusion Matrix for Train dataset

>> confusionMatrix(res\_nb,train\_hra$left)

#plot the accuracy for train dataset

>> auc <- roc(as.numeric(train\_hra$left)-1, as.numeric(res\_nb)-1)

>> plot(auc, ylim=c(0,1), print.thres=TRUE, main=paste('AUC:',round(auc$auc[[1]],3)),col = 'green')

# predict the naive bayes model stats for test data set

>> test\_resnb=predict(model.bayes,test\_hra)

#Confusion Matrix for Test dataset

>> confusionMatrix(test\_resnb,test\_hra$left)

# plot the accuracy for test dataset

>> auc2 <- roc(as.numeric(test\_hra$left)-1, as.numeric(test\_resnb)-1)

>> plot(auc, ylim=c(0,1), print.thres=TRUE, main=paste('AUC:',round(auc2$auc[[1]],3)),col = 'green')

***Figures***

**Fig.5.3.a** The Plot Between Number of People who left the company according to salary levels

|  |
| --- |
|  |
| **Fig.5.3.a** |

**Fig.5.3.b** Number of People who left the company according to the division they belong

|  |
| --- |
|  |
| **Fig.5.3.b** |

**Fig.5.3.c** Number of People who left the company based on the satisfaction level

|  |
| --- |
|  |
| **Fig.5.3.c** |

**Fig.5.3.d** Number of People who left the company according to the number of projects they took.

|  |
| --- |
|  |
| **Fig.5.3.d** |

**Fig.5.3.e** Number of People who left the company based on the promotion for last 5 years.

|  |
| --- |
|  |
| **Fig.5.3.e** |

**Fig.5.3.f** Confusion Matrix and other statistics of training dataset

|  |
| --- |
|  |
| **Fig.5.3.f** |

**Fig.5.3.g** ROC of Test dataset

|  |
| --- |
|  |
| **Fig.5.3.g** |

**Fig.5.3.h** Confusion Matrix and other statistics of test dataset

|  |
| --- |
|  |
| **Fig.5.3.h** |

**Fig.5.3.i** ROC of Test dataset

|  |
| --- |
|  |
| **Fig.5.3.i** |