IST707 Team Project: HR attrition prediction

Bomin Powers, Karen Kim

12/18/2021

# import the data

hrdata<- read.csv("D:/Documents/school/IST707/project/HRdata.csv")  
summary(hrdata)

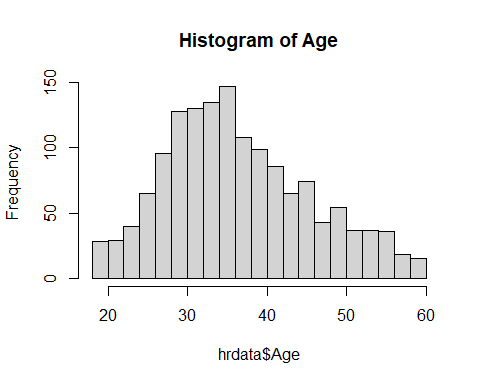
## Age Attrition BusinessTravel Department   
## Min. :18.00 Length:1470 Length:1470 Length:1470   
## 1st Qu.:30.00 Class :character Class :character Class :character   
## Median :36.00 Mode :character Mode :character Mode :character   
## Mean :36.92   
## 3rd Qu.:43.00   
## Max. :60.00   
## DistanceFromHome Gender JobInvolvement JobLevel   
## Min. : 1.000 Length:1470 Min. :1.00 Min. :1.000   
## 1st Qu.: 2.000 Class :character 1st Qu.:2.00 1st Qu.:1.000   
## Median : 7.000 Mode :character Median :3.00 Median :2.000   
## Mean : 9.193 Mean :2.73 Mean :2.064   
## 3rd Qu.:14.000 3rd Qu.:3.00 3rd Qu.:3.000   
## Max. :29.000 Max. :4.00 Max. :5.000   
## JobRole JobSatisfaction MaritalStatus MonthlyIncome   
## Length:1470 Min. :1.000 Length:1470 Min. : 1009   
## Class :character 1st Qu.:2.000 Class :character 1st Qu.: 2911   
## Mode :character Median :3.000 Mode :character Median : 4919   
## Mean :2.729 Mean : 6503   
## 3rd Qu.:4.000 3rd Qu.: 8379   
## Max. :4.000 Max. :19999   
## NumCompaniesWorked OverTime PercentSalaryHike PerformanceRating  
## Min. :0.000 Length:1470 Min. :11.00 Min. :3.000   
## 1st Qu.:1.000 Class :character 1st Qu.:12.00 1st Qu.:3.000   
## Median :2.000 Mode :character Median :14.00 Median :3.000   
## Mean :2.693 Mean :15.21 Mean :3.154   
## 3rd Qu.:4.000 3rd Qu.:18.00 3rd Qu.:3.000   
## Max. :9.000 Max. :25.00 Max. :4.000   
## StockOptionLevel TotalWorkingYears TrainingTimesLastYear YearsAtCompany   
## Min. :0.0000 Min. : 0.00 Min. :0.000 Min. : 0.000   
## 1st Qu.:0.0000 1st Qu.: 6.00 1st Qu.:2.000 1st Qu.: 3.000   
## Median :1.0000 Median :10.00 Median :3.000 Median : 5.000   
## Mean :0.7939 Mean :11.28 Mean :2.799 Mean : 7.008   
## 3rd Qu.:1.0000 3rd Qu.:15.00 3rd Qu.:3.000 3rd Qu.: 9.000   
## Max. :3.0000 Max. :40.00 Max. :6.000 Max. :40.000   
## YearsSinceLastPromotion YearsWithCurrManager Higher\_Education   
## Min. : 0.000 Min. : 0.000 Length:1470   
## 1st Qu.: 0.000 1st Qu.: 2.000 Class :character   
## Median : 1.000 Median : 3.000 Mode :character   
## Mean : 2.188 Mean : 4.123   
## 3rd Qu.: 3.000 3rd Qu.: 7.000   
## Max. :15.000 Max. :17.000   
## Date\_of\_Hire Date\_of\_termination Status\_of\_leaving Mode\_of\_work   
## Length:1470 Mode:logical Length:1470 Length:1470   
## Class :character NA's:1470 Class :character Class :character   
## Mode :character Mode :character Mode :character   
##   
##   
##   
## Leaves Absenteeism Work\_accident Source\_of\_Hire   
## Min. :0.000 Min. :0.000 Length:1470 Length:1470   
## 1st Qu.:1.000 1st Qu.:1.000 Class :character Class :character   
## Median :3.000 Median :2.000 Mode :character Mode :character   
## Mean :2.569 Mean :1.525   
## 3rd Qu.:4.000 3rd Qu.:2.000   
## Max. :5.000 Max. :3.000   
## Job\_mode X   
## Length:1470 Mode:logical   
## Class :character NA's:1470   
## Mode :character   
##   
##   
##

# data cleansing

# remove unnecessary columns  
hrdata<- subset(hrdata, select = -c(X,Date\_of\_termination))

# convert character data type to nominal types to conduct correlation analysis

# convert age column(int) to nominal type  
  
# check how age's data distributed  
hist(hrdata$Age, breaks = 15, main = paste("Histogram of Age"))



# split age variable into somewhat equal-sized groups  
hrdata$ageG <- as.factor(cut2(hrdata$Age, g=6))  
# check how each group range look like  
summary(hrdata$ageG)

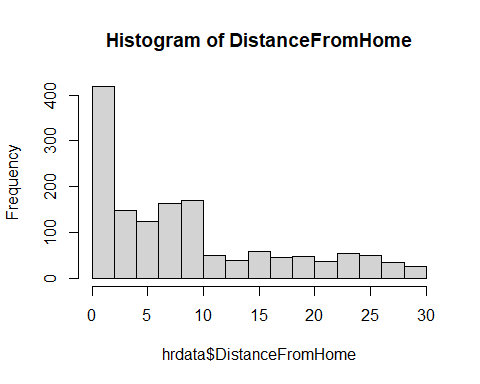
## [18,29) [29,33) [33,37) [37,41) [41,47) [47,60]   
## 258 258 282 207 225 240

## group1 = 18:28, group2 = 29:32, group3 = 33:36, group4 = 37:40, group5 = 41:46, group6 = 47:60  
# now, convert age column(int) to nominal type  
hrdata$ageG <- as.numeric(cut2(hrdata$Age, g=6))  
hrdata<- subset(hrdata, select = -c(Age))

# convert Attrition column(cha) to nominal type  
# convert Yes to 1 and No to 0  
hrdata$Attrition <- ifelse(hrdata$Attrition=="Yes",1,0)

# convert BusinessTravel column(cha) to nominal type  
hrdata$BusinessTravel = as.numeric(as.factor(hrdata$BusinessTravel))  
# Non-Travel == 1, Travel\_Frequently == 2,Travel\_Rarely ==3  
  
# convert Department column(cha) to nominal type  
hrdata$Department = as.numeric(as.factor(hrdata$Department))  
# Human Resources == 1, Sales == 3,Research & Development ==2  
  
# convert JobRole column(cha) to nominal type  
hrdata$JobRole = as.numeric(as.factor(hrdata$JobRole))  
# Healthcare Representative == 1, Human Resources == 2, Laboratory Technician ==3,   
# Manager=4, Manufacturing Director==5, Research Director ==6,  
# Research Scientist ==7, Sales Executive==8, Sales Representative==9  
  
# convert MaritalStatus column(cha) to nominal type  
hrdata$MaritalStatus = as.numeric(as.factor(hrdata$MaritalStatus))  
# Divorced == 1, Married == 2,Single == 3  
  
# convert Higher\_Education column(cha) to nominal type  
hrdata$Higher\_Education = as.numeric(as.factor(hrdata$Higher\_Education))  
# 12th == 1, Graduation == 2,PHD == 3, Post-Graduation == 4  
  
# convert Status\_of\_leaving column(cha) to nominal type  
hrdata$Status\_of\_leaving = as.numeric(as.factor(hrdata$Status\_of\_leaving))  
# Better Opportunity == 1,  
  
# convert Source\_of\_Hire column(cha) to nominal type  
hrdata$Source\_of\_Hire = as.numeric(as.factor(hrdata$Source\_of\_Hire))  
# Job Event == 1, Portal == 2,Recruiter == 3, Walk-in == 4  
  
# convert Job\_mode column(cha) to nominal type  
hrdata$Job\_mode = as.numeric(as.factor(hrdata$Job\_mode))  
# Contract == 1, FullTime == 2, Part Time == 3

# convert DistanceFromHome column(int) to nominal type  
  
# check how DistanceFromHome's data distributed  
hist(hrdata$DistanceFromHome, breaks = 15, main = paste("Histogram of DistanceFromHome"))



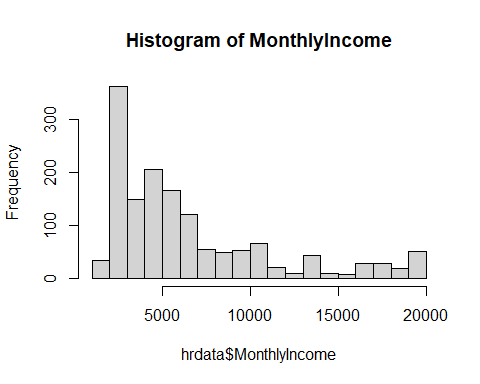
# split DistanceFromHome variable into somewhat equal-sized groups  
hrdata$DistanceFromHomeG <- as.factor(cut2(hrdata$DistanceFromHome, g=8))  
# check how each group range look like  
summary(hrdata$DistanceFromHomeG)

## 1 2 [ 3, 5) [ 5, 8) [ 8,10) [10,15) [15,23) [23,29]   
## 208 211 148 208 165 175 188 167

# group1 = 1, group2 = 2, group3 =3:4, group4 = 5:7, group5 = 8:9, group6 = 10:14, group7 = 15:22, group8=23:29  
# now, convert DistanceFromHome column(int) to nominal type  
hrdata$DistanceFromHomeG <- as.numeric(cut2(hrdata$DistanceFromHome, g=8))  
hrdata<- subset(hrdata, select = -c(DistanceFromHome))

# convert Gender column(cha) to binary type  
  
# convert Male to 1 and Female to 0  
hrdata$Gender <- ifelse(hrdata$Gender =="Male",1,0)

# convert MonthlyIncome column(int) to nominal type  
  
# check how MonthlyIncome's data distributed  
hist(hrdata$MonthlyIncome, breaks = 15, main = paste("Histogram of MonthlyIncome"))

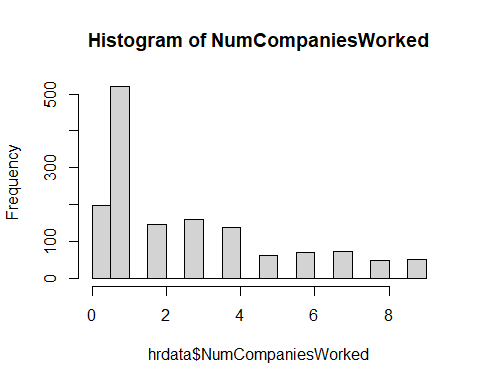


# split MonthlyIncome variable into somewhat equal-sized groups  
hrdata$MonthlyIncomeG <- as.factor(cut2(hrdata$MonthlyIncome, g=6))  
# check how each group range look like  
summary(hrdata$MonthlyIncomeG)

## [ 1009, 2561) [ 2561, 3633) [ 3633, 4930) [ 4930, 6538) [ 6538,10552)   
## 245 245 245 245 245   
## [10552,19999]   
## 245

# group1 = 1009:2560, group2 = 2561:3632, group3 =3633:4929, group4 = 4930:6537, group5 = 6538:10551, group6 = 10552:19999  
# now, convert age column(int) to nominal type  
hrdata$MonthlyIncomeG <- as.numeric(cut2(hrdata$MonthlyIncome, g=6))  
hrdata<- subset(hrdata, select = -c(MonthlyIncome))

# check how NumCompaniesWorked's data distributed  
hist(hrdata$NumCompaniesWorked, breaks = 15, main = paste("Histogram of NumCompaniesWorked"))

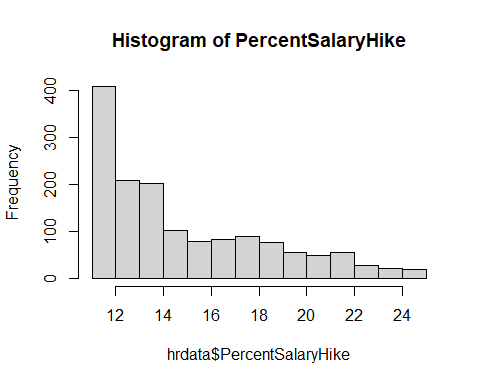


## this could be a problem since the values are clustered between 0 to 2 years but, for now keep this way.  
summary(as.factor(hrdata$NumCompaniesWorked))

## 0 1 2 3 4 5 6 7 8 9   
## 197 521 146 159 139 63 70 74 49 52

# convert OverTime column(cha) to binary type  
  
# convert Yes to 1 and No to 0  
hrdata$OverTime <- ifelse(hrdata$OverTime =="Yes",1,0)

# convert PercentSalaryHike column(int) to nominal type  
  
# check how PercentSalaryHike's data distributed  
hist(hrdata$PercentSalaryHike, breaks = 15, main = paste("Histogram of PercentSalaryHike"))

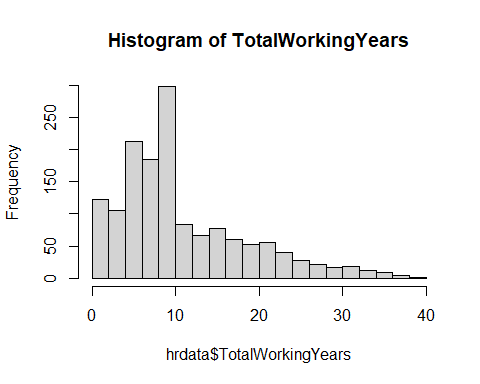


# split PercentSalaryHike variable into somewhat equal-sized groups  
hrdata$PercentSalaryHikeG <- as.factor(cut2(hrdata$PercentSalaryHike, g=9))  
# check how each group range look like  
summary(hrdata$PercentSalaryHikeG)

## 11 12 13 14 [15,17) [17,19) [19,22) [22,25]   
## 210 198 209 201 179 171 179 123

# group1 = 11, group2 = 12, group3 =13, group4 = 14, group5 = 15:16, group6 = 17:18, group7 = 19:21, group8=22:25  
# now, convert PercentSalaryHike column(int) to nominal type  
hrdata$PercentSalaryHikeG <- as.numeric(cut2(hrdata$PercentSalaryHike, g=9))  
hrdata<- subset(hrdata, select = -c(PercentSalaryHike))

# convert TotalWorkingYears column(int) to nominal type  
  
# check how TotalWorkingYears's data distributed  
hist(hrdata$TotalWorkingYears, breaks = 15, main = paste("Histogram of TotalWorkingYears"))



# split TotalWorkingYears variable into somewhat equal-sized groups  
hrdata$TotalWorkingYearsG <- as.factor(cut2(hrdata$TotalWorkingYears, g=5))  
# check how each group range look like  
summary(hrdata$TotalWorkingYearsG)

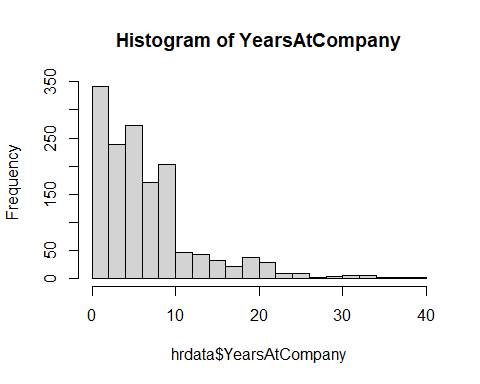
## [ 0, 6) [ 6, 9) [ 9,11) [11,18) [18,40]   
## 316 309 298 261 286

# group1 = 0:5, group2 = 6:8, group3 =9:10, group4 = 11:17, group5 = 18:40  
# now, convert TotalWorkingYears column(int) to nominal type  
hrdata$TotalWorkingYearsG <- as.numeric(cut2(hrdata$TotalWorkingYears, g=5))  
hrdata<- subset(hrdata, select = -c(TotalWorkingYears))

# check how TrainingTimesLastYear's data distributed  
## this could be a problem since the values are clustered between 2 to 3 but, for now keep this way.  
summary(as.factor(hrdata$TrainingTimesLastYear))

## 0 1 2 3 4 5 6   
## 54 71 547 491 123 119 65

# convert YearsAtCompany column(int) to nominal type  
  
# check how YearsAtCompany's data distributed  
hist(hrdata$YearsAtCompany, breaks = 15, main = paste("Histogram of YearsAtCompany"))

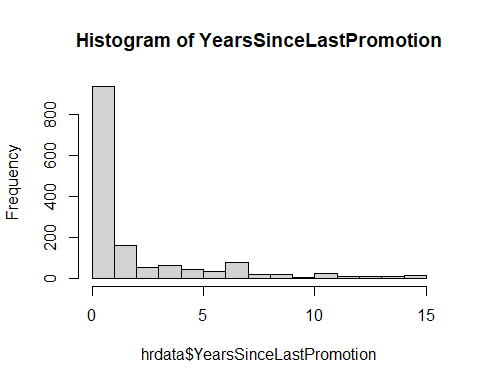


# split YearsAtCompany variable into somewhat equal-sized groups  
hrdata$YearsAtCompanyG <- as.factor(cut2(hrdata$YearsAtCompany, g=4))  
# check how each group range look like  
summary(hrdata$YearsAtCompanyG)

## [ 0, 4) [ 4, 6) [ 6,10) [10,40]   
## 470 306 328 366

# group1 = 0:3, group2 = 4:5, group3 =6:9, group4 = 10:40  
# now, convert YearsAtCompany column(int) to nominal type  
hrdata$YearsAtCompanyG <- as.numeric(cut2(hrdata$YearsAtCompany, g=4))  
hrdata<- subset(hrdata, select = -c(YearsAtCompany))

# convert YearsSinceLastPromotion column(int) to nominal type  
  
# check how YearsSinceLastPromotion's data distributed  
hist(hrdata$YearsSinceLastPromotion, breaks = 15, main = paste("Histogram of YearsSinceLastPromotion"))

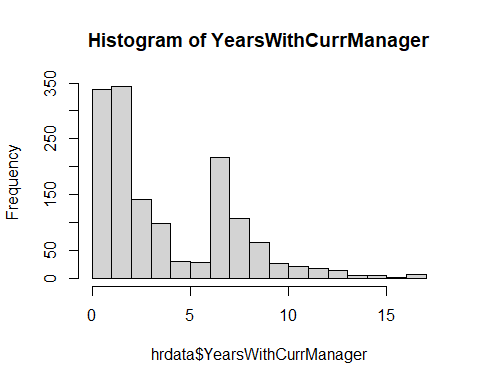


# split YearsSinceLastPromotion variable into somewhat equal-sized groups  
hrdata$YearsSinceLastPromotionG <- as.factor(cut2(hrdata$YearsSinceLastPromotion, g=4))  
# check how each group range look like  
summary(hrdata$YearsSinceLastPromotionG)

## 0 1 [2, 4) [4,15]   
## 581 357 211 321

# group1 = 0, group2 = 1, group3 =2:3, group4 = 4:15,  
# now, convert YearsSinceLastPromotion column(int) to nominal type  
hrdata$YearsSinceLastPromotionG <- as.numeric(cut2(hrdata$YearsSinceLastPromotion, g=4))  
hrdata<- subset(hrdata, select = -c(YearsSinceLastPromotion))

# convert YearsWithCurrManager column(int) to nominal type  
  
# check how YearsWithCurrManager's data distributed  
hist(hrdata$YearsWithCurrManager, breaks = 15, main = paste("Histogram of YearsWithCurrManager"))

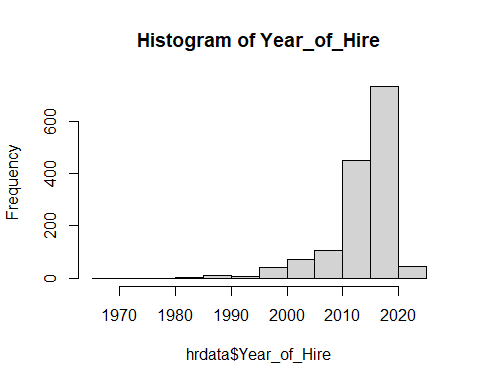


# split YearsWithCurrManager variable into somewhat equal-sized groups  
hrdata$YearsWithCurrManagerG <- as.factor(cut2(hrdata$YearsWithCurrManager, g=5))  
# check how each group range look like  
summary(hrdata$YearsWithCurrManagerG)

## [0, 2) 2 [3, 5) [5, 8) [8,17]   
## 339 344 240 276 271

# group1 = 0:1, group2 = 2, group3 =3:4, group4 = 5:7, group5 = 8:17  
# now, convert YearsWithCurrManager column(int) to nominal type  
hrdata$YearsWithCurrManagerG <- as.numeric(cut2(hrdata$YearsWithCurrManager, g=5))  
hrdata<- subset(hrdata, select = -c(YearsWithCurrManager))

# convert Date\_of\_Hire column(int) to nominal type  
  
# extract year and month from Date\_of\_Hire data  
hrdata$Date\_of\_Hire <- strptime(hrdata$Date\_of\_Hire, "%d-%m-%Y")  
hrdata$Year\_of\_Hire <- year(as.POSIXlt(hrdata$Date\_of\_Hire, format="%d/%m/%Y"))  
hrdata$Month\_of\_Hire <- month(as.POSIXlt(hrdata$Date\_of\_Hire, format="%d/%m/%Y"))  
  
# check how Year\_of\_Hire's data distributed  
hist(hrdata$Year\_of\_Hire, breaks = 15, main = paste("Histogram of Year\_of\_Hire"))



# split Year\_of\_Hire variable into somewhat equal-sized groups  
hrdata$Year\_of\_Hire\_G <- as.factor(cut2(hrdata$Year\_of\_Hire, g=5))  
# check how each group range look like  
summary(hrdata$Year\_of\_Hire\_G)

## [1969,2012) [2012,2015) [2015,2017) [2017,2020) [2020,2021]   
## 366 252 272 365 215

# group1 = 1969:2011, group2 = 2012:2014, group3 =2015:2016, group4 = 2017:2019, group5 = 2020:2021  
# now, convert Year\_of\_Hire column(int) to nominal type  
hrdata$Year\_of\_Hire\_G <- as.numeric(cut2(hrdata$Year\_of\_Hire, g=5))  
hrdata<- subset(hrdata, select = -c(Year\_of\_Hire))  
  
# check how each month contains data points  
summary(hrdata$Month\_of\_Hire)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 1.000 2.000 4.000 3.538 5.000 6.000

hrdata<- subset(hrdata, select = -c(Date\_of\_Hire))

# convert Mode\_of\_work column(cha) to binary type  
  
# convert OFFICE to 1 and WFH to 0  
hrdata$Mode\_of\_work <- ifelse(hrdata$Mode\_of\_work =="OFFICE",1,0)

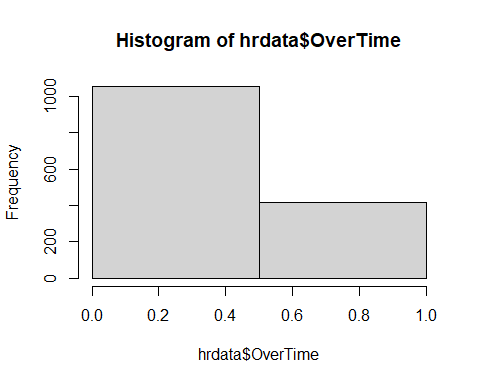
# convert Work\_accident column(cha) to binary type  
  
# convert Yes to 1 and No to 0  
hrdata$Work\_accident <- ifelse(hrdata$Work\_accident =="Yes",1,0)

# correlation analysis

hr\_cor <- cor(hrdata[ ,colnames(hrdata) != "Attrition"], hrdata$Attrition)  
hr\_cor

## [,1]  
## BusinessTravel 7.377695e-05  
## Department 6.399060e-02  
## Gender 2.945325e-02  
## JobInvolvement -1.300160e-01  
## JobLevel -1.691048e-01  
## JobRole 6.715150e-02  
## JobSatisfaction -1.034811e-01  
## MaritalStatus 1.620702e-01  
## NumCompaniesWorked 4.349374e-02  
## OverTime 2.461180e-01  
## PerformanceRating 2.888752e-03  
## StockOptionLevel -1.371449e-01  
## TrainingTimesLastYear -5.947780e-02  
## Higher\_Education 3.641604e-03  
## Status\_of\_leaving 2.075012e-02  
## Mode\_of\_work 6.741883e-03  
## Leaves -4.181966e-02  
## Absenteeism -3.786683e-02  
## Work\_accident 9.845918e-03  
## Source\_of\_Hire 4.462131e-03  
## Job\_mode -5.566298e-02  
## ageG -1.579734e-01  
## DistanceFromHomeG 7.894931e-02  
## MonthlyIncomeG -1.965973e-01  
## PercentSalaryHikeG -2.062394e-02  
## TotalWorkingYearsG -1.829648e-01  
## YearsAtCompanyG -1.641019e-01  
## YearsSinceLastPromotionG -5.334780e-02  
## YearsWithCurrManagerG -1.624467e-01  
## Month\_of\_Hire -1.184360e-02  
## Year\_of\_Hire\_G 1.871048e-01

hist(hrdata$OverTime, breaks=2)



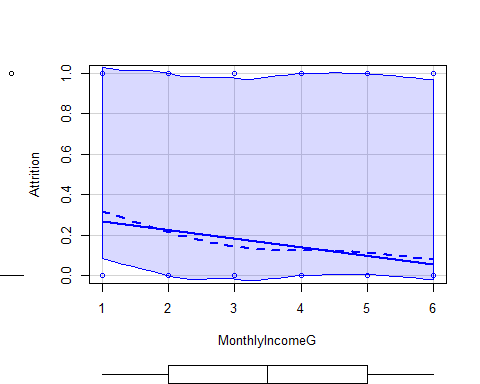
summary(as.factor(hrdata$OverTime))

## 0 1   
## 1054 416

# Correlation scatterplot between Attrition and Monthly Income Group variables   
summary(as.factor(hrdata$MonthlyIncomeG))

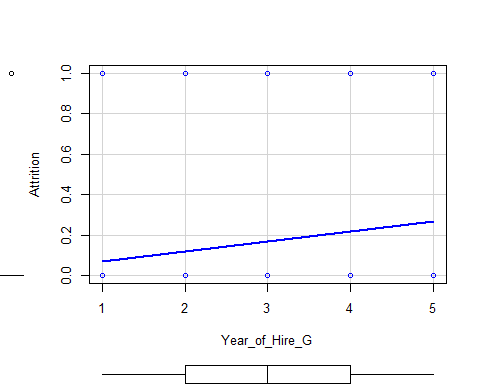
## 1 2 3 4 5 6   
## 245 245 245 245 245 245

scatterplot(Attrition~MonthlyIncomeG, smooth=TRUE, data=hrdata)

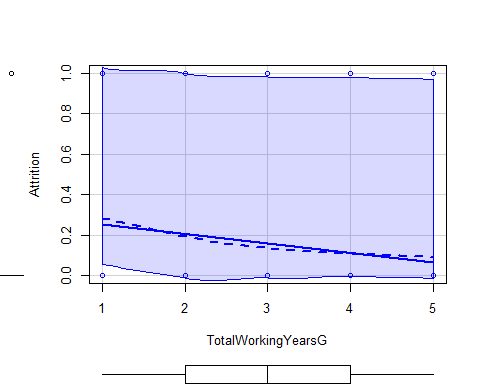


# Correlation scatterplot between Attrition and Year of Hire Group variables  
scatterplot(Attrition~Year\_of\_Hire\_G, smooth=TRUE, data=hrdata)

## Warning in smoother(.x, .y, col = col[1], log.x = logged("x"), log.y =  
## logged("y"), : could not fit smooth

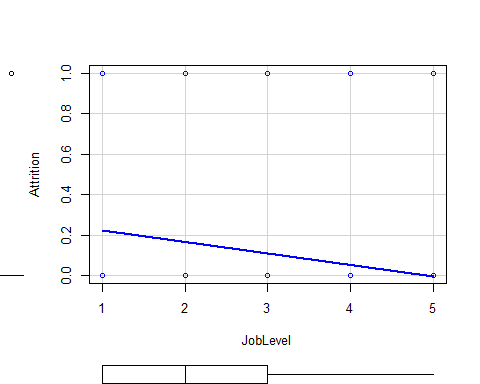


# Correlation scatterplot between Attrition and Total working years group variables  
scatterplot(Attrition~TotalWorkingYearsG, smooth=TRUE, data=hrdata)



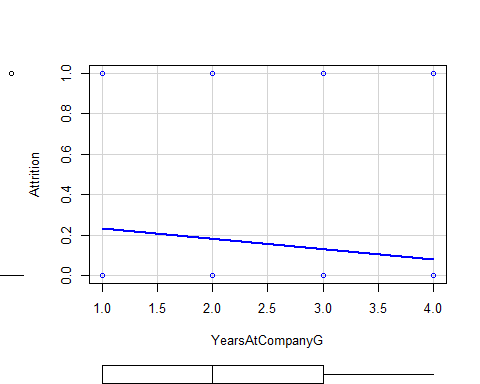
# Correlation scatterplot between Attrition and Job level variables  
scatterplot(Attrition~JobLevel, smooth=TRUE, data=hrdata)

## Warning in smoother(.x, .y, col = col[1], log.x = logged("x"), log.y =  
## logged("y"), : could not fit smooth



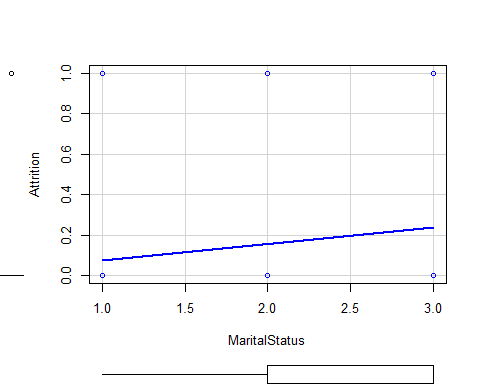
# Correlation scatterplot between Attrition and YearsAtCompanyG variables  
scatterplot(Attrition~YearsAtCompanyG, smooth=TRUE, data=hrdata)

## Warning in smoother(.x, .y, col = col[1], log.x = logged("x"), log.y =  
## logged("y"), : could not fit smooth

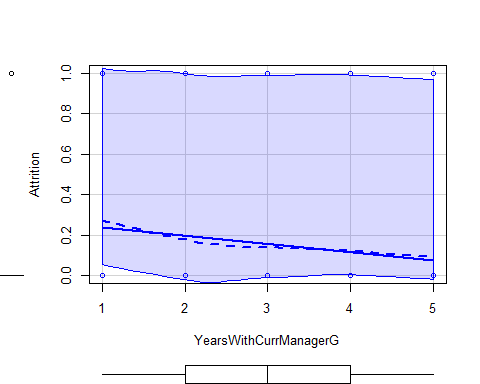


# Correlation scatterplot between Attrition and MaritalStatus variables  
scatterplot(Attrition~MaritalStatus, smooth=TRUE, data=hrdata)

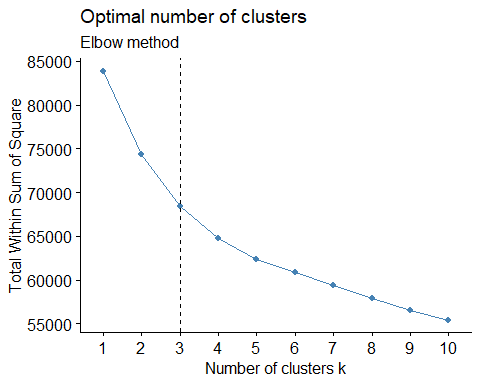
## Warning in smoother(.x, .y, col = col[1], log.x = logged("x"), log.y =  
## logged("y"), : could not fit smooth



# Correlation scatterplot between Attrition and YearsWithCurrManagerG variables  
scatterplot(Attrition~YearsWithCurrManagerG, smooth=TRUE, data=hrdata)

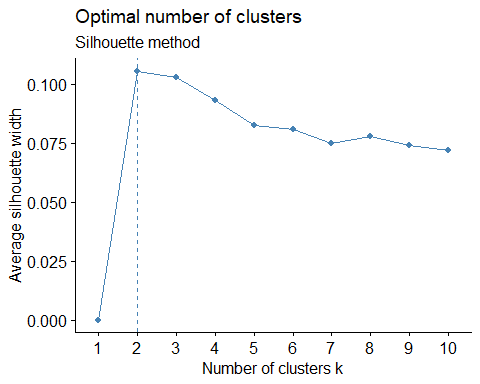


# the elbow method  
fviz\_nbclust(hrdata, kmeans, method = "wss") +  
 geom\_vline(xintercept = 3, linetype = 2) + # add line for better visualization  
 labs(subtitle = "Elbow method") # add subtitle



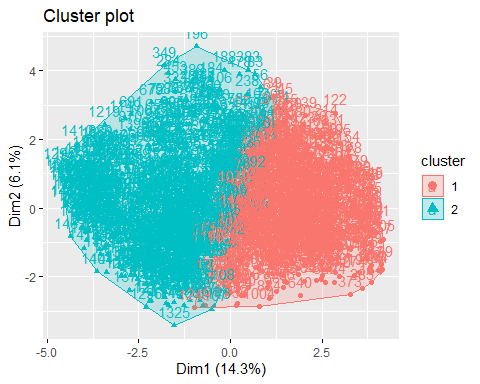
# The elbow method suggests 3 clusters

# the silhouette method  
# Silhouette method  
fviz\_nbclust(hrdata, kmeans, method = "silhouette") +  
 labs(subtitle = "Silhouette method")

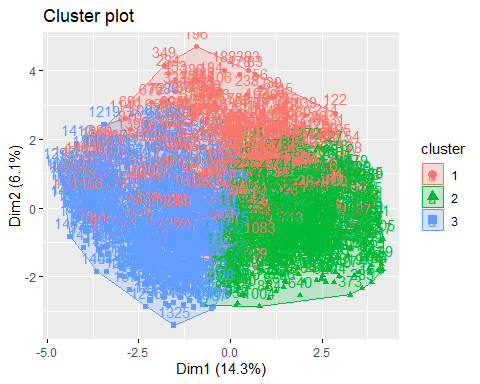


# The Silhouette method suggests 2 clusters.

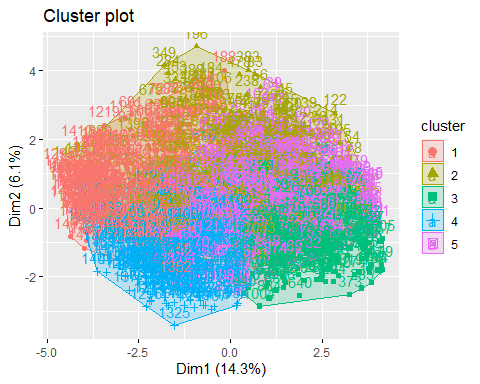
# generate 2 clusters  
hrdata\_km2 <- kmeans(hrdata, centers = 2, nstart =30)  
fviz\_cluster(hrdata\_km2, data=hrdata)



# generate 3 clusters  
hrdata\_km3 <- kmeans(hrdata, centers = 3, nstart =30)  
fviz\_cluster(hrdata\_km3, data=hrdata)



# generate 5 clusters  
hrdata\_km5 <- kmeans(hrdata, centers = 5, nstart =30)  
fviz\_cluster(hrdata\_km5, data=hrdata)



# create data frame including the 2 clusters results  
hrdata\_km2df <- data.frame(hrdata, hrdata\_km2$cluster)  
  
# subset of hrdata\_km2df where attrition is 1 or 0  
hrdata\_km2at1 <- subset(hrdata\_km2df, Attrition == 1)  
hrdata\_km2at0 <- subset(hrdata\_km2df, Attrition == 0)  
  
# randomly sampled attrition=1 or 0 data  
hrdata\_km2sam1 <- hrdata\_km2at1[sample(1:nrow(hrdata\_km2at1), size = 50, replace=FALSE), ]  
hrdata\_km2sam2 <- hrdata\_km2at0[sample(1:nrow(hrdata\_km2at0), size = 50, replace=FALSE), ]  
  
# adds sampled data together  
hrdata\_km2sam <- rbind(hrdata\_km2sam1,hrdata\_km2sam2)  
  
# confusion matrix to find the connection with attrition variable  
table(hrdata\_km2sam$Attrition,hrdata\_km2sam$hrdata\_km2.cluster)

##   
## 1 2  
## 0 24 26  
## 1 39 11

## there is not much clear connection between attrition and cluster group

# create data frame including the 3 clusters results  
hrdata\_km3df <- data.frame(hrdata, hrdata\_km3$cluster)  
  
# subset of hrdata\_km3df where attrition is 1 or 0  
hrdata\_km3at1 <- subset(hrdata\_km3df, Attrition == 1)  
hrdata\_km3at0 <- subset(hrdata\_km3df, Attrition == 0)  
  
# randomly sampled attrition=1 or 0 data  
hrdata\_km3sam1 <- hrdata\_km3at1[sample(1:nrow(hrdata\_km3at1), size = 50, replace=FALSE), ]  
hrdata\_km3sam2 <- hrdata\_km3at0[sample(1:nrow(hrdata\_km3at0), size = 50, replace=FALSE), ]  
  
# adds sampled data together  
hrdata\_km3sam <- rbind(hrdata\_km3sam1,hrdata\_km3sam2)  
  
# confusion matrix to find the connection with attrition variable  
table(hrdata\_km3sam$Attrition,hrdata\_km3sam$hrdata\_km3.cluster)

##   
## 1 2 3  
## 0 15 20 15  
## 1 16 24 10

# Classification - Decision Tree Models

# Experimental Design

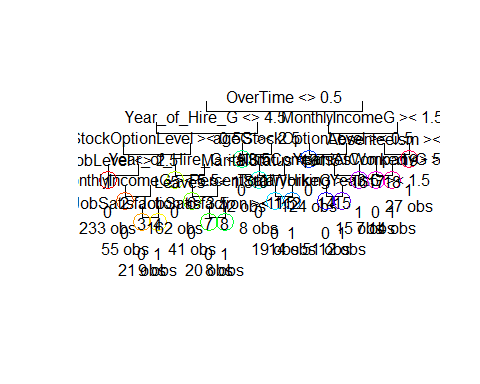
# Now that the data is labeled, its time to design an experiment.   
# Below we randomly select a train and test set for validation  
  
# Make Train and Test sets  
numTotalem = nrow(hrdata)  
trainRatio <- .60  
  
set.seed(11) # Set Seed so that same sample can be reproduced in future also  
sample <- sample.int(n = numTotalem, size = floor(trainRatio\*numTotalem), replace = F)  
  
train <- hrdata[sample, ]  
test <- hrdata[-sample, ]  
   
# train / test ratio  
length(sample)/nrow(hrdata)

## [1] 0.6

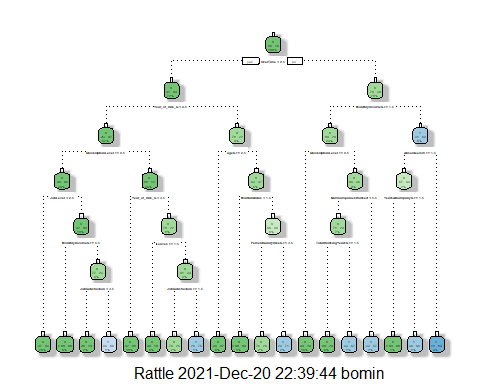
#Train Tree Model 1  
train\_tree1 <- rpart(Attrition ~ ., data = train, method="class", control=rpart.control(cp=0))  
summary(train\_tree1)

## Call:  
## rpart(formula = Attrition ~ ., data = train, method = "class",   
## control = rpart.control(cp = 0))  
## n= 882   
##   
## CP nsplit rel error xerror xstd  
## 1 0.067669173 0 1.0000000 1.0000000 0.07990626  
## 2 0.026315789 2 0.8646617 0.9548872 0.07839519  
## 3 0.018796992 4 0.8120301 1.0225564 0.08064045  
## 4 0.011278195 7 0.7443609 1.0000000 0.07990626  
## 5 0.007518797 11 0.6992481 1.0526316 0.08159810  
## 6 0.002506266 15 0.6691729 1.0751880 0.08230085  
## 7 0.000000000 18 0.6616541 1.1052632 0.08321783  
##   
## Variable importance  
## MonthlyIncomeG OverTime MaritalStatus   
## 18 12 7   
## StockOptionLevel TotalWorkingYearsG NumCompaniesWorked   
## 7 7 6   
## Year\_of\_Hire\_G JobSatisfaction Absenteeism   
## 5 5 4   
## JobRole YearsAtCompanyG YearsWithCurrManagerG   
## 4 4 4   
## ageG Leaves JobLevel   
## 4 3 3   
## PercentSalaryHikeG DistanceFromHomeG Higher\_Education   
## 3 2 2   
## JobInvolvement Department Job\_mode   
## 1 1 1   
##   
## Node number 1: 882 observations, complexity param=0.06766917  
## predicted class=0 expected loss=0.1507937 P(node) =1  
## class counts: 749 133  
## probabilities: 0.849 0.151   
## left son=2 (632 obs) right son=3 (250 obs)  
## Primary splits:  
## OverTime < 0.5 to the left, improve=14.712710, (0 missing)  
## Year\_of\_Hire\_G < 4.5 to the left, improve=10.873150, (0 missing)  
## YearsAtCompanyG < 1.5 to the right, improve=10.211240, (0 missing)  
## MonthlyIncomeG < 1.5 to the right, improve= 9.107462, (0 missing)  
## TotalWorkingYearsG < 1.5 to the right, improve= 8.896417, (0 missing)  
##   
## Node number 2: 632 observations, complexity param=0.0112782  
## predicted class=0 expected loss=0.09335443 P(node) =0.7165533  
## class counts: 573 59  
## probabilities: 0.907 0.093   
## left son=4 (549 obs) right son=5 (83 obs)  
## Primary splits:  
## Year\_of\_Hire\_G < 4.5 to the left, improve=2.915268, (0 missing)  
## JobRole < 8.5 to the left, improve=2.742014, (0 missing)  
## StockOptionLevel < 0.5 to the right, improve=2.679769, (0 missing)  
## YearsWithCurrManagerG < 1.5 to the right, improve=2.291517, (0 missing)  
## ageG < 2.5 to the right, improve=2.110042, (0 missing)  
## Surrogate splits:  
## YearsWithCurrManagerG < 1.5 to the right, agree=0.913, adj=0.337, (0 split)  
##   
## Node number 3: 250 observations, complexity param=0.06766917  
## predicted class=0 expected loss=0.296 P(node) =0.2834467  
## class counts: 176 74  
## probabilities: 0.704 0.296   
## left son=6 (202 obs) right son=7 (48 obs)  
## Primary splits:  
## MonthlyIncomeG < 1.5 to the right, improve=18.210560, (0 missing)  
## TotalWorkingYearsG < 1.5 to the right, improve=14.936110, (0 missing)  
## JobLevel < 1.5 to the right, improve=13.879940, (0 missing)  
## YearsAtCompanyG < 1.5 to the right, improve=12.559130, (0 missing)  
## MaritalStatus < 2.5 to the left, improve= 9.088552, (0 missing)  
## Surrogate splits:  
## JobRole < 8.5 to the left, agree=0.828, adj=0.104, (0 split)  
##   
## Node number 4: 549 observations, complexity param=0.007518797  
## predicted class=0 expected loss=0.07468124 P(node) =0.622449  
## class counts: 508 41  
## probabilities: 0.925 0.075   
## left son=8 (318 obs) right son=9 (231 obs)  
## Primary splits:  
## StockOptionLevel < 0.5 to the right, improve=1.7269100, (0 missing)  
## Leaves < 1.5 to the right, improve=1.1861610, (0 missing)  
## DistanceFromHomeG < 4.5 to the left, improve=0.8234564, (0 missing)  
## Year\_of\_Hire\_G < 3.5 to the left, improve=0.7831497, (0 missing)  
## JobRole < 7.5 to the left, improve=0.7220965, (0 missing)  
## Surrogate splits:  
## MaritalStatus < 2.5 to the left, agree=0.894, adj=0.749, (0 split)  
## TrainingTimesLastYear < 0.5 to the right, agree=0.581, adj=0.004, (0 split)  
## YearsWithCurrManagerG < 1.5 to the right, agree=0.581, adj=0.004, (0 split)  
## Month\_of\_Hire < 1.5 to the right, agree=0.581, adj=0.004, (0 split)  
##   
## Node number 5: 83 observations, complexity param=0.0112782  
## predicted class=0 expected loss=0.2168675 P(node) =0.09410431  
## class counts: 65 18  
## probabilities: 0.783 0.217   
## left son=10 (42 obs) right son=11 (41 obs)  
## Primary splits:  
## ageG < 2.5 to the right, improve=3.596952, (0 missing)  
## Month\_of\_Hire < 1.5 to the right, improve=2.284928, (0 missing)  
## PercentSalaryHikeG < 2.5 to the right, improve=1.929711, (0 missing)  
## JobRole < 8.5 to the left, improve=1.845518, (0 missing)  
## Higher\_Education < 2.5 to the left, improve=1.805562, (0 missing)  
## Surrogate splits:  
## TotalWorkingYearsG < 1.5 to the right, agree=0.819, adj=0.634, (0 split)  
## NumCompaniesWorked < 1.5 to the right, agree=0.807, adj=0.610, (0 split)  
## JobLevel < 1.5 to the right, agree=0.723, adj=0.439, (0 split)  
## MonthlyIncomeG < 2.5 to the right, agree=0.699, adj=0.390, (0 split)  
## JobRole < 6.5 to the left, agree=0.627, adj=0.244, (0 split)  
##   
## Node number 6: 202 observations, complexity param=0.01879699  
## predicted class=0 expected loss=0.2029703 P(node) =0.2290249  
## class counts: 161 41  
## probabilities: 0.797 0.203   
## left son=12 (124 obs) right son=13 (78 obs)  
## Primary splits:  
## StockOptionLevel < 0.5 to the right, improve=6.184806, (0 missing)  
## MaritalStatus < 2.5 to the left, improve=5.095174, (0 missing)  
## TotalWorkingYearsG < 1.5 to the right, improve=4.075192, (0 missing)  
## YearsAtCompanyG < 1.5 to the right, improve=3.489211, (0 missing)  
## Year\_of\_Hire\_G < 4.5 to the left, improve=3.205927, (0 missing)  
## Surrogate splits:  
## MaritalStatus < 2.5 to the left, agree=0.881, adj=0.692, (0 split)  
## Year\_of\_Hire\_G < 4.5 to the left, agree=0.629, adj=0.038, (0 split)  
## JobRole < 8.5 to the left, agree=0.619, adj=0.013, (0 split)  
##   
## Node number 7: 48 observations, complexity param=0.02631579  
## predicted class=1 expected loss=0.3125 P(node) =0.05442177  
## class counts: 15 33  
## probabilities: 0.312 0.687   
## left son=14 (21 obs) right son=15 (27 obs)  
## Primary splits:  
## Absenteeism < 1.5 to the right, improve=5.005952, (0 missing)  
## ageG < 2.5 to the right, improve=4.465290, (0 missing)  
## YearsAtCompanyG < 1.5 to the right, improve=3.402778, (0 missing)  
## DistanceFromHomeG < 6.5 to the left, improve=3.125000, (0 missing)  
## Higher\_Education < 3.5 to the right, improve=2.778846, (0 missing)  
## Surrogate splits:  
## Leaves < 3.5 to the right, agree=0.688, adj=0.286, (0 split)  
## DistanceFromHomeG < 2.5 to the left, agree=0.688, adj=0.286, (0 split)  
## YearsWithCurrManagerG < 1.5 to the right, agree=0.646, adj=0.190, (0 split)  
## JobInvolvement < 2.5 to the right, agree=0.625, adj=0.143, (0 split)  
## JobSatisfaction < 3.5 to the right, agree=0.625, adj=0.143, (0 split)  
##   
## Node number 8: 318 observations, complexity param=0.002506266  
## predicted class=0 expected loss=0.0408805 P(node) =0.3605442  
## class counts: 305 13  
## probabilities: 0.959 0.041   
## left son=16 (233 obs) right son=17 (85 obs)  
## Primary splits:  
## JobLevel < 2.5 to the left, improve=0.3990610, (0 missing)  
## MonthlyIncomeG < 4.5 to the left, improve=0.3908963, (0 missing)  
## Higher\_Education < 2.5 to the right, improve=0.3693947, (0 missing)  
## JobRole < 8.5 to the left, improve=0.3459342, (0 missing)  
## DistanceFromHomeG < 6.5 to the left, improve=0.2685172, (0 missing)  
## Surrogate splits:  
## MonthlyIncomeG < 4.5 to the left, agree=0.937, adj=0.765, (0 split)  
## TotalWorkingYearsG < 4.5 to the left, agree=0.858, adj=0.471, (0 split)  
## ageG < 5.5 to the left, agree=0.761, adj=0.106, (0 split)  
## YearsAtCompanyG < 3.5 to the left, agree=0.761, adj=0.106, (0 split)  
## Year\_of\_Hire\_G < 1.5 to the right, agree=0.761, adj=0.106, (0 split)  
##   
## Node number 9: 231 observations, complexity param=0.007518797  
## predicted class=0 expected loss=0.1212121 P(node) =0.2619048  
## class counts: 203 28  
## probabilities: 0.879 0.121   
## left son=18 (162 obs) right son=19 (69 obs)  
## Primary splits:  
## Year\_of\_Hire\_G < 3.5 to the left, improve=1.820280, (0 missing)  
## Absenteeism < 1.5 to the left, improve=1.266236, (0 missing)  
## Leaves < 1.5 to the right, improve=1.208253, (0 missing)  
## NumCompaniesWorked < 4.5 to the left, improve=1.062650, (0 missing)  
## JobSatisfaction < 3.5 to the right, improve=1.019165, (0 missing)  
## Surrogate splits:  
## YearsAtCompanyG < 1.5 to the right, agree=0.892, adj=0.638, (0 split)  
## YearsWithCurrManagerG < 2.5 to the right, agree=0.844, adj=0.478, (0 split)  
## TotalWorkingYearsG < 1.5 to the right, agree=0.784, adj=0.275, (0 split)  
## JobRole < 8.5 to the left, agree=0.732, adj=0.101, (0 split)  
## MonthlyIncomeG < 1.5 to the right, agree=0.727, adj=0.087, (0 split)  
##   
## Node number 10: 42 observations  
## predicted class=0 expected loss=0.07142857 P(node) =0.04761905  
## class counts: 39 3  
## probabilities: 0.929 0.071   
##   
## Node number 11: 41 observations, complexity param=0.0112782  
## predicted class=0 expected loss=0.3658537 P(node) =0.04648526  
## class counts: 26 15  
## probabilities: 0.634 0.366   
## left son=22 (8 obs) right son=23 (33 obs)  
## Primary splits:  
## MaritalStatus < 1.5 to the left, improve=2.660754, (0 missing)  
## PercentSalaryHikeG < 3.5 to the right, improve=2.309414, (0 missing)  
## Higher\_Education < 2.5 to the left, improve=2.148200, (0 missing)  
## Month\_of\_Hire < 1.5 to the right, improve=2.086890, (0 missing)  
## DistanceFromHomeG < 4.5 to the left, improve=1.405343, (0 missing)  
## Surrogate splits:  
## DistanceFromHomeG < 7.5 to the right, agree=0.829, adj=0.125, (0 split)  
##   
## Node number 12: 124 observations  
## predicted class=0 expected loss=0.1048387 P(node) =0.1405896  
## class counts: 111 13  
## probabilities: 0.895 0.105   
##   
## Node number 13: 78 observations, complexity param=0.01879699  
## predicted class=0 expected loss=0.3589744 P(node) =0.08843537  
## class counts: 50 28  
## probabilities: 0.641 0.359   
## left son=26 (63 obs) right son=27 (15 obs)  
## Primary splits:  
## NumCompaniesWorked < 5.5 to the left, improve=3.516484, (0 missing)  
## JobSatisfaction < 3.5 to the right, improve=2.843150, (0 missing)  
## Year\_of\_Hire\_G < 4.5 to the left, improve=2.750114, (0 missing)  
## TotalWorkingYearsG < 1.5 to the right, improve=2.685315, (0 missing)  
## YearsAtCompanyG < 1.5 to the right, improve=1.908002, (0 missing)  
##   
## Node number 14: 21 observations, complexity param=0.02631579  
## predicted class=0 expected loss=0.4285714 P(node) =0.02380952  
## class counts: 12 9  
## probabilities: 0.571 0.429   
## left son=28 (7 obs) right son=29 (14 obs)  
## Primary splits:  
## YearsAtCompanyG < 1.5 to the right, improve=3.8571430, (0 missing)  
## ageG < 2.5 to the right, improve=3.1746030, (0 missing)  
## Higher\_Education < 2.5 to the right, improve=1.3412700, (0 missing)  
## JobSatisfaction < 2.5 to the right, improve=0.9972527, (0 missing)  
## TotalWorkingYearsG < 1.5 to the right, improve=0.6311688, (0 missing)  
## Surrogate splits:  
## YearsWithCurrManagerG < 2.5 to the right, agree=0.857, adj=0.571, (0 split)  
## NumCompaniesWorked < 0.5 to the left, agree=0.810, adj=0.429, (0 split)  
## Year\_of\_Hire\_G < 3.5 to the left, agree=0.810, adj=0.429, (0 split)  
## JobInvolvement < 3.5 to the right, agree=0.762, adj=0.286, (0 split)  
## Higher\_Education < 3.5 to the right, agree=0.762, adj=0.286, (0 split)  
##   
## Node number 15: 27 observations  
## predicted class=1 expected loss=0.1111111 P(node) =0.03061224  
## class counts: 3 24  
## probabilities: 0.111 0.889   
##   
## Node number 16: 233 observations  
## predicted class=0 expected loss=0.02575107 P(node) =0.2641723  
## class counts: 227 6  
## probabilities: 0.974 0.026   
##   
## Node number 17: 85 observations, complexity param=0.002506266  
## predicted class=0 expected loss=0.08235294 P(node) =0.09637188  
## class counts: 78 7  
## probabilities: 0.918 0.082   
## left son=34 (55 obs) right son=35 (30 obs)  
## Primary splits:  
## MonthlyIncomeG < 5.5 to the right, improve=2.113725, (0 missing)  
## JobSatisfaction < 3.5 to the left, improve=1.150004, (0 missing)  
## Higher\_Education < 1.5 to the right, improve=1.107928, (0 missing)  
## JobRole < 2.5 to the right, improve=1.073725, (0 missing)  
## DistanceFromHomeG < 7.5 to the left, improve=1.073725, (0 missing)  
## Surrogate splits:  
## TotalWorkingYearsG < 3.5 to the right, agree=0.824, adj=0.500, (0 split)  
## JobLevel < 3.5 to the right, agree=0.800, adj=0.433, (0 split)  
## ageG < 3.5 to the right, agree=0.776, adj=0.367, (0 split)  
## JobRole < 7 to the left, agree=0.753, adj=0.300, (0 split)  
## Month\_of\_Hire < 2.5 to the right, agree=0.718, adj=0.200, (0 split)  
##   
## Node number 18: 162 observations  
## predicted class=0 expected loss=0.08024691 P(node) =0.1836735  
## class counts: 149 13  
## probabilities: 0.920 0.080   
##   
## Node number 19: 69 observations, complexity param=0.007518797  
## predicted class=0 expected loss=0.2173913 P(node) =0.07823129  
## class counts: 54 15  
## probabilities: 0.783 0.217   
## left son=38 (41 obs) right son=39 (28 obs)  
## Primary splits:  
## Leaves < 1.5 to the right, improve=2.901606, (0 missing)  
## JobSatisfaction < 1.5 to the right, improve=2.805534, (0 missing)  
## DistanceFromHomeG < 4.5 to the left, improve=1.700483, (0 missing)  
## JobRole < 7.5 to the left, improve=1.432509, (0 missing)  
## Department < 2.5 to the left, improve=1.196156, (0 missing)  
## Surrogate splits:  
## MaritalStatus < 2.5 to the right, agree=0.638, adj=0.107, (0 split)  
## TrainingTimesLastYear < 1.5 to the right, agree=0.638, adj=0.107, (0 split)  
## Higher\_Education < 1.5 to the right, agree=0.638, adj=0.107, (0 split)  
## JobSatisfaction < 1.5 to the right, agree=0.623, adj=0.071, (0 split)  
## Department < 1.5 to the right, agree=0.609, adj=0.036, (0 split)  
##   
## Node number 22: 8 observations  
## predicted class=0 expected loss=0 P(node) =0.009070295  
## class counts: 8 0  
## probabilities: 1.000 0.000   
##   
## Node number 23: 33 observations, complexity param=0.0112782  
## predicted class=0 expected loss=0.4545455 P(node) =0.03741497  
## class counts: 18 15  
## probabilities: 0.545 0.455   
## left son=46 (19 obs) right son=47 (14 obs)  
## Primary splits:  
## PercentSalaryHikeG < 3.5 to the right, improve=3.280930, (0 missing)  
## TrainingTimesLastYear < 3.5 to the right, improve=1.859289, (0 missing)  
## DistanceFromHomeG < 6.5 to the left, improve=1.843636, (0 missing)  
## JobRole < 4 to the right, improve=1.696970, (0 missing)  
## Higher\_Education < 2.5 to the left, improve=1.386193, (0 missing)  
## Surrogate splits:  
## JobRole < 2.5 to the right, agree=0.727, adj=0.357, (0 split)  
## Department < 1.5 to the right, agree=0.697, adj=0.286, (0 split)  
## Job\_mode < 1.5 to the left, agree=0.667, adj=0.214, (0 split)  
## JobLevel < 2.5 to the left, agree=0.636, adj=0.143, (0 split)  
## StockOptionLevel < 2.5 to the left, agree=0.636, adj=0.143, (0 split)  
##   
## Node number 26: 63 observations, complexity param=0.01879699  
## predicted class=0 expected loss=0.2857143 P(node) =0.07142857  
## class counts: 45 18  
## probabilities: 0.714 0.286   
## left son=52 (51 obs) right son=53 (12 obs)  
## Primary splits:  
## TotalWorkingYearsG < 1.5 to the right, improve=4.302521, (0 missing)  
## Year\_of\_Hire\_G < 4.5 to the left, improve=3.277223, (0 missing)  
## Leaves < 1.5 to the right, improve=3.104530, (0 missing)  
## ageG < 3.5 to the right, improve=2.568756, (0 missing)  
## YearsWithCurrManagerG < 1.5 to the right, improve=2.092747, (0 missing)  
## Surrogate splits:  
## JobRole < 8.5 to the left, agree=0.825, adj=0.083, (0 split)  
## ageG < 1.5 to the right, agree=0.825, adj=0.083, (0 split)  
## MonthlyIncomeG < 2.5 to the right, agree=0.825, adj=0.083, (0 split)  
##   
## Node number 27: 15 observations  
## predicted class=1 expected loss=0.3333333 P(node) =0.0170068  
## class counts: 5 10  
## probabilities: 0.333 0.667   
##   
## Node number 28: 7 observations  
## predicted class=0 expected loss=0 P(node) =0.007936508  
## class counts: 7 0  
## probabilities: 1.000 0.000   
##   
## Node number 29: 14 observations  
## predicted class=1 expected loss=0.3571429 P(node) =0.01587302  
## class counts: 5 9  
## probabilities: 0.357 0.643   
##   
## Node number 34: 55 observations  
## predicted class=0 expected loss=0 P(node) =0.06235828  
## class counts: 55 0  
## probabilities: 1.000 0.000   
##   
## Node number 35: 30 observations, complexity param=0.002506266  
## predicted class=0 expected loss=0.2333333 P(node) =0.03401361  
## class counts: 23 7  
## probabilities: 0.767 0.233   
## left son=70 (21 obs) right son=71 (9 obs)  
## Primary splits:  
## JobSatisfaction < 3.5 to the left, improve=2.669841, (0 missing)  
## DistanceFromHomeG < 6.5 to the left, improve=2.087371, (0 missing)  
## ageG < 4.5 to the left, improve=1.699841, (0 missing)  
## Higher\_Education < 1.5 to the right, improve=1.551515, (0 missing)  
## Status\_of\_leaving < 1.5 to the left, improve=1.400000, (0 missing)  
## Surrogate splits:  
## Higher\_Education < 1.5 to the right, agree=0.767, adj=0.222, (0 split)  
## Absenteeism < 0.5 to the right, agree=0.733, adj=0.111, (0 split)  
## DistanceFromHomeG < 7.5 to the left, agree=0.733, adj=0.111, (0 split)  
##   
## Node number 38: 41 observations  
## predicted class=0 expected loss=0.09756098 P(node) =0.04648526  
## class counts: 37 4  
## probabilities: 0.902 0.098   
##   
## Node number 39: 28 observations, complexity param=0.007518797  
## predicted class=0 expected loss=0.3928571 P(node) =0.03174603  
## class counts: 17 11  
## probabilities: 0.607 0.393   
## left son=78 (20 obs) right son=79 (8 obs)  
## Primary splits:  
## JobSatisfaction < 1.5 to the right, improve=2.857143, (0 missing)  
## DistanceFromHomeG < 3.5 to the left, improve=2.772527, (0 missing)  
## Gender < 0.5 to the left, improve=1.613827, (0 missing)  
## Higher\_Education < 1.5 to the left, improve=1.613827, (0 missing)  
## NumCompaniesWorked < 1.5 to the left, improve=1.157143, (0 missing)  
##   
## Node number 46: 19 observations  
## predicted class=0 expected loss=0.2631579 P(node) =0.02154195  
## class counts: 14 5  
## probabilities: 0.737 0.263   
##   
## Node number 47: 14 observations  
## predicted class=1 expected loss=0.2857143 P(node) =0.01587302  
## class counts: 4 10  
## probabilities: 0.286 0.714   
##   
## Node number 52: 51 observations  
## predicted class=0 expected loss=0.1960784 P(node) =0.05782313  
## class counts: 41 10  
## probabilities: 0.804 0.196   
##   
## Node number 53: 12 observations  
## predicted class=1 expected loss=0.3333333 P(node) =0.01360544  
## class counts: 4 8  
## probabilities: 0.333 0.667   
##   
## Node number 70: 21 observations  
## predicted class=0 expected loss=0.0952381 P(node) =0.02380952  
## class counts: 19 2  
## probabilities: 0.905 0.095   
##   
## Node number 71: 9 observations  
## predicted class=1 expected loss=0.4444444 P(node) =0.01020408  
## class counts: 4 5  
## probabilities: 0.444 0.556   
##   
## Node number 78: 20 observations  
## predicted class=0 expected loss=0.25 P(node) =0.02267574  
## class counts: 15 5  
## probabilities: 0.750 0.250   
##   
## Node number 79: 8 observations  
## predicted class=1 expected loss=0.25 P(node) =0.009070295  
## class counts: 2 6  
## probabilities: 0.250 0.750

#plot the decision tree  
draw.tree(train\_tree1)



#plot the decision tree  
fancyRpartPlot(train\_tree1)



#predict the test dataset using the model for train tree No. 1  
predicted1<- predict(train\_tree1, test, type="class")  
#confusion matrix to find correct and incorrect predictions  
table(Attrition=predicted1, true=test$Attrition)

## true  
## Attrition 0 1  
## 0 449 75  
## 1 35 29

confusionMatrix(data = predicted1, as.factor(test$Attrition))

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction 0 1  
## 0 449 75  
## 1 35 29  
##   
## Accuracy : 0.8129   
## 95% CI : (0.779, 0.8437)  
## No Information Rate : 0.8231   
## P-Value [Acc > NIR] : 0.7606539   
##   
## Kappa : 0.2433   
##   
## Mcnemar's Test P-Value : 0.0002004   
##   
## Sensitivity : 0.9277   
## Specificity : 0.2788   
## Pos Pred Value : 0.8569   
## Neg Pred Value : 0.4531   
## Prevalence : 0.8231   
## Detection Rate : 0.7636   
## Detection Prevalence : 0.8912   
## Balanced Accuracy : 0.6033   
##   
## 'Positive' Class : 0   
##

# Naive Bayes Train model

#g generate the model  
train\_naibayes <- naiveBayes(train$Attrition~., data=train, na.action = na.pass)  
summary(train\_naibayes)

## Length Class Mode   
## apriori 2 table numeric   
## tables 31 -none- list   
## levels 2 -none- character  
## isnumeric 31 -none- logical   
## call 4 -none- call

# Naive Bayes model Prediction  
nb\_Pred <- predict(train\_naibayes, test)  
# confusion matrix to find correct and incorrect predictions  
table(Attrition=nb\_Pred, true=test$Attrition)

## true  
## Attrition 0 1  
## 0 405 46  
## 1 79 58

# Association Rule Mining

# data preparation

hrdata2 <- read.csv("D:/Documents/school/IST707/project/HRdata.csv")

# remove unnecessary columns  
hrdata2<- subset(hrdata2, select = -c(X,Date\_of\_termination))

# convert character data type to nominal types to conduct Association Rule Mining

# convert age column(int) to nominal type  
# split age variable into somewhat equal-sized groups  
hrdata2$ageG <- as.factor(cut2(hrdata2$Age, g=6))  
# check how each group range look like  
summary(hrdata2$ageG)

## [18,29) [29,33) [33,37) [37,41) [41,47) [47,60]   
## 258 258 282 207 225 240

## group1 = 18:28, group2 = 29:32, group3 = 33:36, group4 = 37:40, group5 = 41:46, group6 = 47:60  
hrdata2<- subset(hrdata2, select = -c(Age))

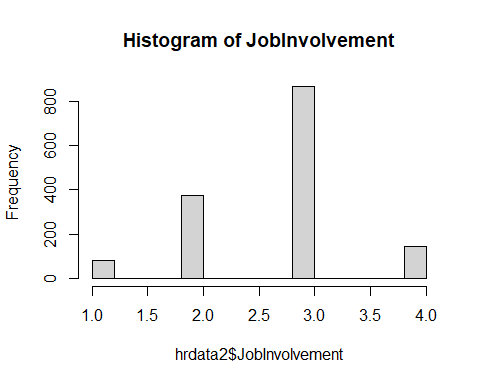
# columns that are character type  
chacols <- c('Attrition', 'BusinessTravel', 'Department', 'Gender', 'JobRole', 'MaritalStatus','OverTime', 'Higher\_Education','Date\_of\_Hire', 'Status\_of\_leaving','Mode\_of\_work','Work\_accident', 'Source\_of\_Hire', 'Job\_mode')  
# convert character type columns to nominal type  
hrdata2[chacols] <- lapply(hrdata2[chacols], factor)

# convert DistanceFromHome column(int) to nominal type  
# split DistanceFromHome variable into somewhat equal-sized groups  
hrdata2$DistanceFromHomeG <- as.factor(cut2(hrdata2$DistanceFromHome, g=8))  
# check how each group range look like  
summary(hrdata2$DistanceFromHomeG)

## 1 2 [ 3, 5) [ 5, 8) [ 8,10) [10,15) [15,23) [23,29]   
## 208 211 148 208 165 175 188 167

# group1 = 1, group2 = 2, group3 =3:4, group4 = 5:7, group5 = 8:9, group6 = 10:14, group7 = 15:22, group8=23:29  
hrdata2 <- subset(hrdata2, select = -c(DistanceFromHome))

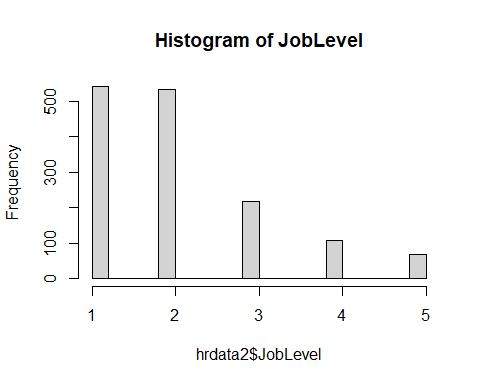
# check how JobInvolvement's data distributed  
hist(hrdata2$JobInvolvement, breaks = 15, main = paste("Histogram of JobInvolvement"))



# convert JobInvolvement column(int) to nominal type  
# split JobInvolvement variable into somewhat equal-sized groups  
hrdata2$JobInvolvement <- as.factor(cut2(hrdata2$JobInvolvement, g=4))  
# check how each group range look like  
summary(hrdata2$JobInvolvement)

## [1,3) 3 4   
## 458 868 144

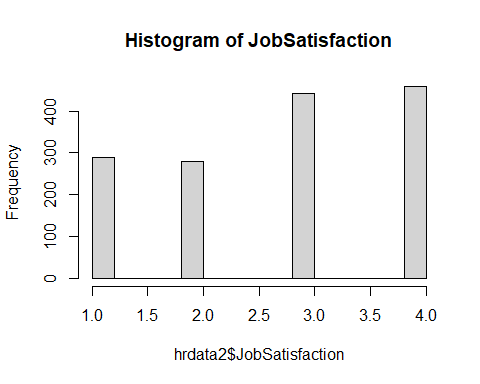
# check how JobLevel's data distributed  
hist(hrdata2$JobLevel, breaks = 15, main = paste("Histogram of JobLevel"))



# convert JobLevel column(int) to nominal type  
# split JobLevel variable into somewhat equal-sized groups  
hrdata2$JobLevel <- as.factor(cut2(hrdata2$JobLevel, g=3))  
# check how each group range look like  
summary(hrdata2$JobLevel)

## 1 2 [3,5]   
## 543 534 393

# check how JobSatisfaction's data distributed  
hist(hrdata2$JobSatisfaction, breaks = 15, main = paste("Histogram of JobSatisfaction"))



# convert JobSatisfaction column(int) to nominal type  
# split JobSatisfaction variable into somewhat equal-sized groups  
hrdata2$JobSatisfaction <- as.factor(cut2(hrdata2$JobSatisfaction, g=3))  
# check how each group range look like  
summary(hrdata2$JobSatisfaction)

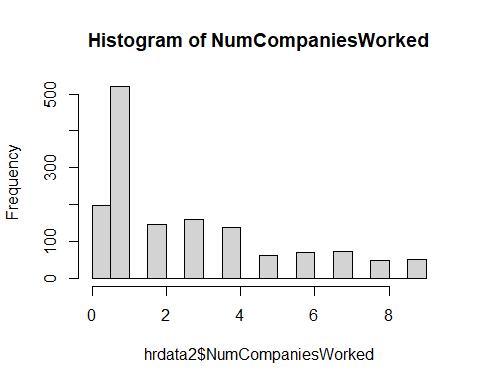
## [1,3) 3 4   
## 569 442 459

# convert MonthlyIncome column(int) to nominal type  
  
# split MonthlyIncome variable into somewhat equal-sized groups  
hrdata2$MonthlyIncomeG <- as.factor(cut2(hrdata2$MonthlyIncome, g=6))  
# check how each group range look like  
summary(hrdata2$MonthlyIncomeG)

## [ 1009, 2561) [ 2561, 3633) [ 3633, 4930) [ 4930, 6538) [ 6538,10552)   
## 245 245 245 245 245   
## [10552,19999]   
## 245

# group1 = 1009:2560, group2 = 2561:3632, group3 =3633:4929, group4 = 4930:6537, group5 = 6538:10551, group6 = 10552:19999  
hrdata2<- subset(hrdata2, select = -c(MonthlyIncome))

# check how NumCompaniesWorked's data distributed  
hist(hrdata2$NumCompaniesWorked, breaks = 15, main = paste("Histogram of NumCompaniesWorked"))



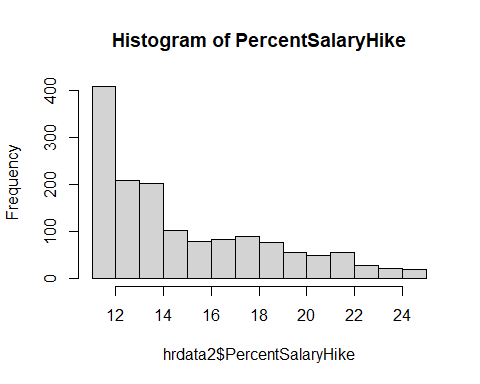
# split MonthlyIncome variable into somewhat equal-sized groups  
hrdata2$NumCompaniesWorked <- as.factor(cut2(hrdata2$NumCompaniesWorked, g=8))  
# check how each group range look like  
summary(hrdata2$NumCompaniesWorked)

## 0 1 2 3 4 [5,7) [7,9]   
## 197 521 146 159 139 133 175

# check the OverTime variable value  
summary(hrdata2$OverTime)

## No Yes   
## 1054 416

# convert PercentSalaryHike column(int) to nominal type  
  
# check how PercentSalaryHike's data distributed  
hist(hrdata2$PercentSalaryHike, breaks = 15, main = paste("Histogram of PercentSalaryHike"))



# split PercentSalaryHike variable into somewhat equal-sized groups  
hrdata2$PercentSalaryHike <- as.factor(cut2(hrdata2$PercentSalaryHike, g=9))  
# check how each group range look like  
summary(hrdata2$PercentSalaryHike)

## 11 12 13 14 [15,17) [17,19) [19,22) [22,25]   
## 210 198 209 201 179 171 179 123

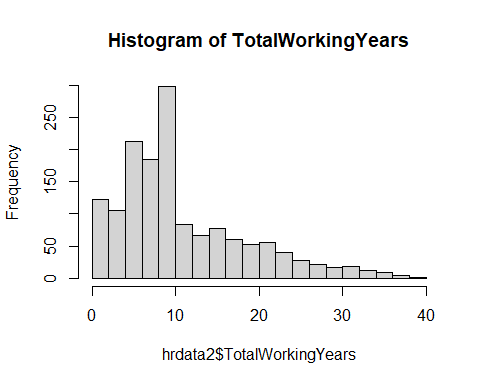
# convert PerformanceRating column(int) to nominal type  
# split PercentSalaryHike variable into somewhat equal-sized groups  
hrdata2$PerformanceRating <- as.factor(hrdata2$PerformanceRating)  
# check the OverTime variable value  
summary(hrdata2$PerformanceRating)

## 3 4   
## 1244 226

# convert StockOptionLevel column(int) to nominal type  
# split StockOptionLevel variable into somewhat equal-sized groups  
hrdata2$StockOptionLevel <- as.factor(cut2(hrdata2$StockOptionLevel, g=3))  
# check how each group range look like  
summary(hrdata2$StockOptionLevel)

## 0 1 [2,3]   
## 631 596 243

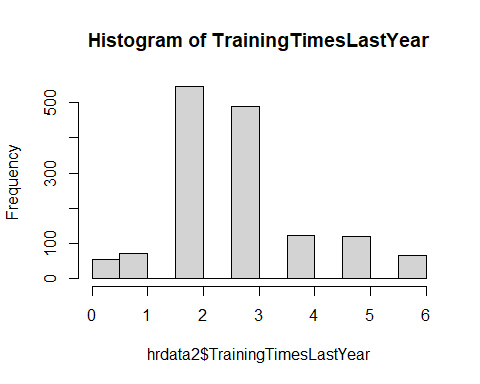
# convert TotalWorkingYears column(int) to nominal type  
  
# check how TotalWorkingYears's data distributed  
hist(hrdata2$TotalWorkingYears, breaks = 15, main = paste("Histogram of TotalWorkingYears"))



# split TotalWorkingYears variable into somewhat equal-sized groups  
hrdata2$TotalWorkingYears <- as.factor(cut2(hrdata2$TotalWorkingYears, g=5))  
# check how each group range look like  
summary(hrdata2$TotalWorkingYears)

## [ 0, 6) [ 6, 9) [ 9,11) [11,18) [18,40]   
## 316 309 298 261 286

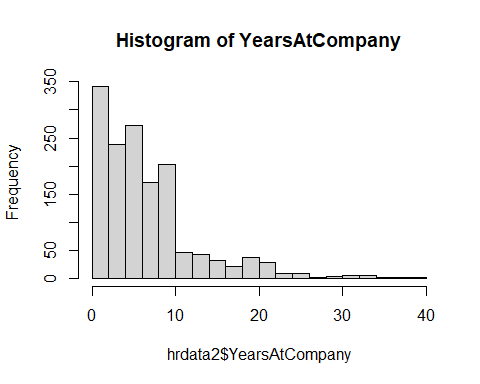
# check how TrainingTimesLastYear's data distributed  
# check how TrainingTimesLastYear's data distributed  
hist(hrdata2$TrainingTimesLastYear, breaks = 15, main = paste("Histogram of TrainingTimesLastYear"))



# split TrainingTimesLastYear variable into somewhat equal-sized groups  
hrdata2$TrainingTimesLastYear <- as.factor(cut2(hrdata2$TrainingTimesLastYear, g=4))  
# check how each group range look like  
summary(hrdata2$TrainingTimesLastYear)

## [0,3) 3 [4,6]   
## 672 491 307

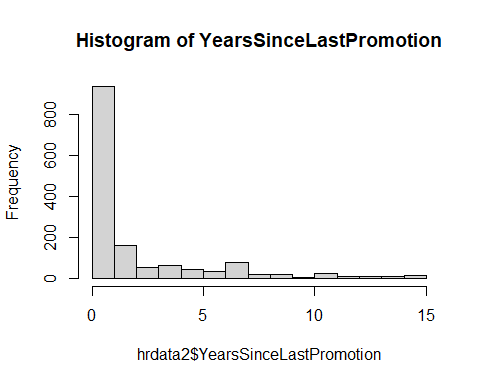
# convert YearsAtCompany column(int) to nominal type  
  
# check how YearsAtCompany's data distributed  
hist(hrdata2$YearsAtCompany, breaks = 15, main = paste("Histogram of YearsAtCompany"))



# split YearsAtCompany variable into somewhat equal-sized groups  
hrdata2$YearsAtCompany <- as.factor(cut2(hrdata2$YearsAtCompany, g=4))  
# check how each group range look like  
summary(hrdata2$YearsAtCompany)

## [ 0, 4) [ 4, 6) [ 6,10) [10,40]   
## 470 306 328 366

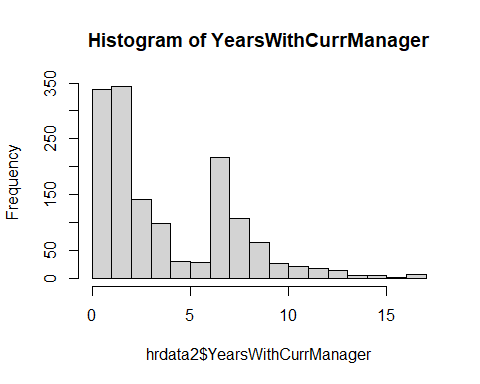
# convert YearsSinceLastPromotion column(int) to nominal type  
  
# check how YearsSinceLastPromotion's data distributed  
hist(hrdata2$YearsSinceLastPromotion, breaks = 15, main = paste("Histogram of YearsSinceLastPromotion"))



# split YearsSinceLastPromotion variable into somewhat equal-sized groups  
hrdata2$YearsSinceLastPromotion <- as.factor(cut2(hrdata2$YearsSinceLastPromotion, g=4))  
# check how each group range look like  
summary(hrdata2$YearsSinceLastPromotion)

## 0 1 [2, 4) [4,15]   
## 581 357 211 321

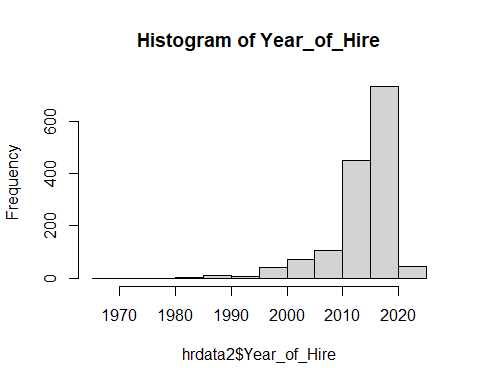
# convert YearsWithCurrManager column(int) to nominal type  
  
# check how YearsWithCurrManager's data distributed  
hist(hrdata2$YearsWithCurrManager, breaks = 15, main = paste("Histogram of YearsWithCurrManager"))



# split YearsWithCurrManager variable into somewhat equal-sized groups  
hrdata2$YearsWithCurrManager <- as.factor(cut2(hrdata2$YearsWithCurrManager, g=5))  
# check how each group range look like  
summary(hrdata2$YearsWithCurrManager)

## [0, 2) 2 [3, 5) [5, 8) [8,17]   
## 339 344 240 276 271

# convert Date\_of\_Hire column(int) to nominal type  
  
# extract year and month from Date\_of\_Hire data  
hrdata2$Date\_of\_Hire <- strptime(hrdata2$Date\_of\_Hire, "%d-%m-%Y")  
hrdata2$Year\_of\_Hire <- year(as.POSIXlt(hrdata2$Date\_of\_Hire, format="%d/%m/%Y"))  
hrdata2$Month\_of\_Hire <- month(as.POSIXlt(hrdata2$Date\_of\_Hire, format="%d/%m/%Y"))  
  
# check how Year\_of\_Hire's data distributed  
hist(hrdata2$Year\_of\_Hire, breaks = 15, main = paste("Histogram of Year\_of\_Hire"))



# split Year\_of\_Hire variable into somewhat equal-sized groups  
hrdata2$Year\_of\_Hire\_G <- as.factor(cut2(hrdata2$Year\_of\_Hire, g=5))  
# check how each group range look like  
summary(hrdata2$Year\_of\_Hire\_G)

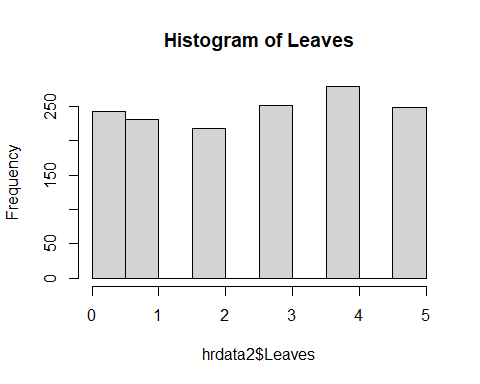
## [1969,2012) [2012,2015) [2015,2017) [2017,2020) [2020,2021]   
## 366 252 272 365 215

hrdata2<- subset(hrdata2, select = -c(Year\_of\_Hire))  
  
# check how each month contains data points  
hrdata2$Month\_of\_Hire <- as.factor(hrdata2$Month\_of\_Hire)  
summary(hrdata2$Month\_of\_Hire)

## 1 2 3 4 5 6   
## 220 218 288 261 261 222

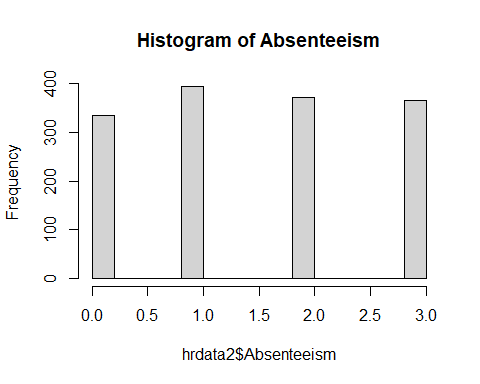
hrdata2<- subset(hrdata2, select = -c(Date\_of\_Hire))

# convert Leaves column(int) to nominal type  
  
# check how Leaves's data distributed  
hist(hrdata2$Leaves, breaks = 15, main = paste("Histogram of Leaves"))



# evenly distributed, thus dose need not be split  
hrdata2$Leaves <- as.factor(hrdata2$Leaves)

# convert Absenteeism column(int) to nominal type  
  
# check how Absenteeism's data distributed  
hist(hrdata2$Absenteeism, breaks = 15, main = paste("Histogram of Absenteeism"))



# evenly distributed, thus dose need not be split  
hrdata2$Absenteeism <- as.factor(hrdata2$Absenteeism)

# the most unevenly distributed variables  
# OverTime and PerformanceRating  
summary(hrdata2$OverTime)

## No Yes   
## 1054 416

summary(hrdata2$PerformanceRating)

## 3 4   
## 1244 226

# subset the data frame  
# majority(OverTime=No and PerformanceRating=3) group  
hrdatat2ot0pf3 <- hrdata2[(hrdata2$OverTime == "No") & (hrdata2$PerformanceRating == 3),]  
# minority(OverTime=Yes and PerformanceRating=4) group  
hrdatat2ot1pf4 <- hrdata2[(hrdata2$OverTime == "Yes") & (hrdata2$PerformanceRating == 4),]  
  
set.seed(1234)  
# randomly sampled majority and minority group dataset  
hrdata\_maj <- hrdatat2ot0pf3[sample(1:nrow(hrdatat2ot0pf3), size = 50, replace=FALSE), ]  
hrdata\_min <- hrdatat2ot1pf4[sample(1:nrow(hrdatat2ot1pf4), size = 50, replace=FALSE), ]  
  
# combined sampled data together  
hrdata\_com <- rbind(hrdata\_maj,hrdata\_min)  
  
# OverTime and PerformanceRating  
summary(hrdata\_com$OverTime)

## No Yes   
## 50 50

summary(hrdata\_com$PerformanceRating)

## 3 4   
## 50 50

# conducted Association Rule Mining

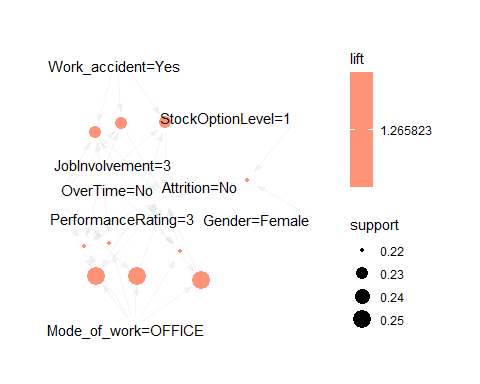
# test rules: targeting employees who stayed  
hr\_rules1<- apriori(hrdata\_com, parameter = list(support = 0.07, confidence = 1), appearance = list(default="lhs", rhs="Attrition=No"), control=list(verbose=F))  
summary(hr\_rules1)

## set of 2778 rules  
##   
## rule length distribution (lhs + rhs):sizes  
## 2 3 4 5 6 7 8   
## 6 249 1022 1062 386 52 1   
##   
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 2.000 4.000 5.000 4.624 5.000 8.000   
##   
## summary of quality measures:  
## support confidence coverage lift   
## Min. :0.08000 Min. :1 Min. :0.08000 Min. :1.266   
## 1st Qu.:0.08000 1st Qu.:1 1st Qu.:0.08000 1st Qu.:1.266   
## Median :0.09000 Median :1 Median :0.09000 Median :1.266   
## Mean :0.09462 Mean :1 Mean :0.09462 Mean :1.266   
## 3rd Qu.:0.10000 3rd Qu.:1 3rd Qu.:0.10000 3rd Qu.:1.266   
## Max. :0.25000 Max. :1 Max. :0.25000 Max. :1.266   
## count   
## Min. : 8.000   
## 1st Qu.: 8.000   
## Median : 9.000   
## Mean : 9.462   
## 3rd Qu.:10.000   
## Max. :25.000   
##   
## mining info:  
## data ntransactions support confidence  
## hrdata\_com 100 0.07 1  
## call  
## apriori(data = hrdata\_com, parameter = list(support = 0.07, confidence = 1), appearance = list(default = "lhs", rhs = "Attrition=No"), control = list(verbose = F))

# sort the same set of rules to view based on support  
hr\_rulessupport<- sort(hr\_rules1, by="support", decreasing=TRUE)  
inspect(hr\_rulessupport[1:20])

## lhs rhs support confidence coverage lift count  
## [1] {OverTime=No,   
## Mode\_of\_work=OFFICE} => {Attrition=No} 0.25 1 0.25 1.265823 25  
## [2] {PerformanceRating=3,   
## Mode\_of\_work=OFFICE} => {Attrition=No} 0.25 1 0.25 1.265823 25  
## [3] {OverTime=No,   
## PerformanceRating=3,   
## Mode\_of\_work=OFFICE} => {Attrition=No} 0.25 1 0.25 1.265823 25  
## [4] {JobInvolvement=3,   
## OverTime=No,   
## Work\_accident=Yes} => {Attrition=No} 0.23 1 0.23 1.265823 23  
## [5] {JobInvolvement=3,   
## PerformanceRating=3,   
## Work\_accident=Yes} => {Attrition=No} 0.23 1 0.23 1.265823 23  
## [6] {JobInvolvement=3,   
## OverTime=No,   
## PerformanceRating=3,   
## Work\_accident=Yes} => {Attrition=No} 0.23 1 0.23 1.265823 23  
## [7] {Gender=Female,   
## StockOptionLevel=1} => {Attrition=No} 0.22 1 0.22 1.265823 22  
## [8] {JobInvolvement=3,   
## OverTime=No,   
## Mode\_of\_work=OFFICE} => {Attrition=No} 0.22 1 0.22 1.265823 22  
## [9] {JobInvolvement=3,   
## PerformanceRating=3,   
## Mode\_of\_work=OFFICE} => {Attrition=No} 0.22 1 0.22 1.265823 22  
## [10] {JobInvolvement=3,   
## OverTime=No,   
## PerformanceRating=3,   
## Mode\_of\_work=OFFICE} => {Attrition=No} 0.22 1 0.22 1.265823 22  
## [11] {Gender=Female,   
## MaritalStatus=Married} => {Attrition=No} 0.21 1 0.21 1.265823 21  
## [12] {JobInvolvement=3,   
## MaritalStatus=Married,   
## StockOptionLevel=1} => {Attrition=No} 0.21 1 0.21 1.265823 21  
## [13] {OverTime=No,   
## Higher\_Education=Graduation} => {Attrition=No} 0.19 1 0.19 1.265823 19  
## [14] {PerformanceRating=3,   
## Higher\_Education=Graduation} => {Attrition=No} 0.19 1 0.19 1.265823 19  
## [15] {OverTime=No,   
## PerformanceRating=3,   
## Higher\_Education=Graduation} => {Attrition=No} 0.19 1 0.19 1.265823 19  
## [16] {Department=Research & Development,   
## OverTime=No,   
## Work\_accident=Yes} => {Attrition=No} 0.19 1 0.19 1.265823 19  
## [17] {Department=Research & Development,   
## PerformanceRating=3,   
## Work\_accident=Yes} => {Attrition=No} 0.19 1 0.19 1.265823 19  
## [18] {JobInvolvement=3,   
## StockOptionLevel=1,   
## Work\_accident=Yes} => {Attrition=No} 0.19 1 0.19 1.265823 19  
## [19] {Department=Research & Development,   
## OverTime=No,   
## PerformanceRating=3,   
## Work\_accident=Yes} => {Attrition=No} 0.19 1 0.19 1.265823 19  
## [20] {JobInvolvement=3,   
## TrainingTimesLastYear=3} => {Attrition=No} 0.18 1 0.18 1.265823 18

#plot the rules  
hr\_rulessupport <- hr\_rulessupport[1:10]  
plot(hr\_rulessupport, method="graph")



# sort the same set of rules to view based on lift  
hr\_ruleslift<- sort(hr\_rules1, by="lift", decreasing=TRUE)  
inspect(hr\_ruleslift[1:20])

## lhs rhs support confidence coverage lift count  
## [1] {Year\_of\_Hire\_G=[2015,2017)} => {Attrition=No} 0.08 1 0.08 1.265823 8  
## [2] {ageG=[47,60]} => {Attrition=No} 0.10 1 0.10 1.265823 10  
## [3] {YearsWithCurrManager=[3, 5)} => {Attrition=No} 0.10 1 0.10 1.265823 10  
## [4] {JobRole=Manufacturing Director} => {Attrition=No} 0.11 1 0.11 1.265823 11  
## [5] {MonthlyIncomeG=[10552,19999]} => {Attrition=No} 0.13 1 0.13 1.265823 13  
## [6] {PercentSalaryHike=14} => {Attrition=No} 0.14 1 0.14 1.265823 14  
## [7] {JobInvolvement=3,   
## Year\_of\_Hire\_G=[2015,2017)} => {Attrition=No} 0.08 1 0.08 1.265823 8  
## [8] {Department=Research & Development,   
## ageG=[47,60]} => {Attrition=No} 0.08 1 0.08 1.265823 8  
## [9] {BusinessTravel=Travel\_Rarely,   
## ageG=[47,60]} => {Attrition=No} 0.08 1 0.08 1.265823 8  
## [10] {YearsAtCompany=[ 4, 6),   
## YearsWithCurrManager=[3, 5)} => {Attrition=No} 0.09 1 0.09 1.265823 9  
## [11] {YearsWithCurrManager=[3, 5),   
## Work\_accident=Yes} => {Attrition=No} 0.08 1 0.08 1.265823 8  
## [12] {BusinessTravel=Travel\_Rarely,   
## YearsWithCurrManager=[3, 5)} => {Attrition=No} 0.08 1 0.08 1.265823 8  
## [13] {Department=Research & Development,   
## JobRole=Manufacturing Director} => {Attrition=No} 0.11 1 0.11 1.265823 11  
## [14] {BusinessTravel=Travel\_Rarely,   
## JobRole=Manufacturing Director} => {Attrition=No} 0.08 1 0.08 1.265823 8  
## [15] {BusinessTravel=Non-Travel,   
## OverTime=No} => {Attrition=No} 0.08 1 0.08 1.265823 8  
## [16] {BusinessTravel=Non-Travel,   
## PerformanceRating=3} => {Attrition=No} 0.08 1 0.08 1.265823 8  
## [17] {Mode\_of\_work=OFFICE,   
## DistanceFromHomeG= 1} => {Attrition=No} 0.08 1 0.08 1.265823 8  
## [18] {MaritalStatus=Married,   
## DistanceFromHomeG= 2} => {Attrition=No} 0.08 1 0.08 1.265823 8  
## [19] {TotalWorkingYears=[18,40],   
## MonthlyIncomeG=[10552,19999]} => {Attrition=No} 0.09 1 0.09 1.265823 9  
## [20] {YearsWithCurrManager=[8,17],   
## MonthlyIncomeG=[10552,19999]} => {Attrition=No} 0.08 1 0.08 1.265823 8

summary(hr\_ruleslift)

## set of 2778 rules  
##   
## rule length distribution (lhs + rhs):sizes  
## 2 3 4 5 6 7 8   
## 6 249 1022 1062 386 52 1   
##   
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 2.000 4.000 5.000 4.624 5.000 8.000   
##   
## summary of quality measures:  
## support confidence coverage lift   
## Min. :0.08000 Min. :1 Min. :0.08000 Min. :1.266   
## 1st Qu.:0.08000 1st Qu.:1 1st Qu.:0.08000 1st Qu.:1.266   
## Median :0.09000 Median :1 Median :0.09000 Median :1.266   
## Mean :0.09462 Mean :1 Mean :0.09462 Mean :1.266   
## 3rd Qu.:0.10000 3rd Qu.:1 3rd Qu.:0.10000 3rd Qu.:1.266   
## Max. :0.25000 Max. :1 Max. :0.25000 Max. :1.266   
## count   
## Min. : 8.000   
## 1st Qu.: 8.000   
## Median : 9.000   
## Mean : 9.462   
## 3rd Qu.:10.000   
## Max. :25.000   
##   
## mining info:  
## data ntransactions support confidence  
## hrdata\_com 100 0.07 1  
## call  
## apriori(data = hrdata\_com, parameter = list(support = 0.07, confidence = 1), appearance = list(default = "lhs", rhs = "Attrition=No"), control = list(verbose = F))

#plot the rules  
hr\_rulesliftbest <- hr\_ruleslift[1:10]  
plot(hr\_rulesliftbest, method="graph")

