Workforce_Analytics.R

Aditya

Mon Feb 25 11:24:06 2019

```
#Load the data
library(ggplot2)
mydata = read.csv("C:/Users/Aditya/Downloads/Projects/HR_Analytics.csv")
names(mydata)
```

```
##  [1] "satisfaction_level"    "last_evaluation"
##  [3] "number_project"        "average_montly_hours"
##  [5] "time_spend_company"    "Work_accident"
##  [7] "left"                  "promotion_last_5years"
##  [9] "sales"                 "salary"
```

```
#********************DATA MANIPULATION AND DATA PREPARATION********************

#Adding a new column called 'salaryOrder'
mydata$salaryOrder[which(mydata$salary == "low")] = 1
mydata$salaryOrder[which(mydata$salary == "medium")] = 2
mydata$salaryOrder[which(mydata$salary == "high")] = 3

#Adding a new column called 'employee_satisfaction'
mydata$employee_satisfaction[mydata$satisfaction_level >= 0.9] = '1.Maximum'
mydata$employee_satisfaction[mydata$satisfaction_level >= 0.8 & mydata$satisfaction_level < 0.9
 ] = '2.High'
mydata$employee_satisfaction[mydata$satisfaction_level >= 0.6 & mydata$satisfaction_level < 0.8
 ] = '3.Good'
mydata$employee_satisfaction[mydata$satisfaction_level >= 0.4 & mydata$satisfaction_level < 0.6
 ] = '4.Average'
mydata$employee_satisfaction[mydata$satisfaction_level >= 0.2 & mydata$satisfaction_level < 0.4
 ] = '5.Low'
mydata$employee_satisfaction[mydata$satisfaction_level <  0.2] = '6.Minimum'

#Converting the employee_satisfaction column as a factor
mydata$employee_satisfaction = as.factor(mydata$employee_satisfaction)

#One more new variable for 'left' for string representation.
mydata$leftFlag[mydata$left ==  1] = 'Left'
mydata$leftFlag[mydata$left ==  0] = 'Not Left'

#********************EDA********************

#********************SUMMARY********************

#Data Summary
dim(mydata)
```

```
## [1] 14999    13
```

```
str(mydata)
```

```
## 'data.frame':    14999 obs. of  13 variables:
##  $ satisfaction_level   : num  0.38 0.8 0.11 0.72 0.37 0.41 0.1 0.92 0.89 0.42 ...
##  $ last_evaluation      : num  0.53 0.86 0.88 0.87 0.52 0.5 0.77 0.85 1 0.53 ...
##  $ number_project       : int  2 5 7 5 2 2 6 5 5 2 ...
##  $ average_montly_hours : int  157 262 272 223 159 153 247 259 224 142 ...
##  $ time_spend_company   : int  3 6 4 5 3 3 4 5 5 3 ...
##  $ Work_accident        : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ left                 : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ promotion_last_5years: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ sales                : Factor w/ 10 levels "accounting","hr",..: 8 8 8 8 8 8 8 8 8 8 ...
##  $ salary               : Factor w/ 3 levels "high","low","medium": 2 3 3 2 2 2 2 2 2 2 ...
##  $ salaryOrder          : num  1 2 2 1 1 1 1 1 1 1 ...
##  $ employee_satisfaction: Factor w/ 6 levels "1.Maximum","2.High",..: 5 2 6 3 5 4 6 1 2 4 ...
##  $ leftFlag             : chr  "Left" "Left" "Left" "Left" ...
```

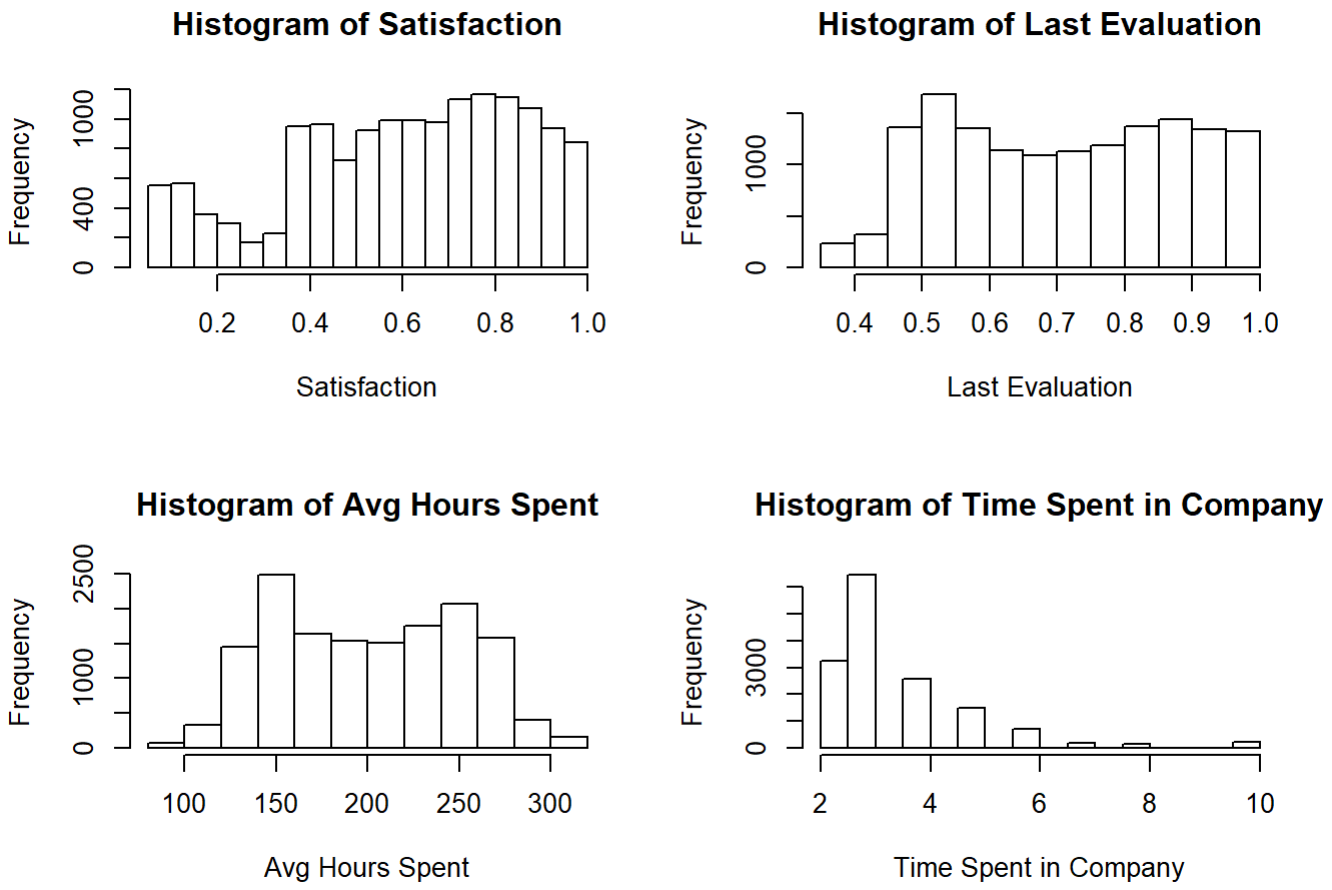
```
summary(mydata)
```

```
##  satisfaction_level last_evaluation  number_project  average_montly_hours
##  Min.   :0.0900     Min.   :0.3600   Min.   :2.000   Min.   : 96.0
##  1st Qu.:0.4400     1st Qu.:0.5600   1st Qu.:3.000   1st Qu.:156.0
##  Median :0.6400     Median :0.7200   Median :4.000   Median :200.0
##  Mean   :0.6128     Mean   :0.7161   Mean   :3.803   Mean   :201.1
##  3rd Qu.:0.8200     3rd Qu.:0.8700   3rd Qu.:5.000   3rd Qu.:245.0
##  Max.   :1.0000     Max.   :1.0000   Max.   :7.000   Max.   :310.0
##
##  time_spend_company Work_accident         left
##  Min.   : 2.000     Min.   :0.0000   Min.   :0.0000
##  1st Qu.: 3.000     1st Qu.:0.0000   1st Qu.:0.0000
##  Median : 3.000     Median :0.0000   Median :0.0000
##  Mean   : 3.498     Mean   :0.1446   Mean   :0.2381
##  3rd Qu.: 4.000     3rd Qu.:0.0000   3rd Qu.:0.0000
##  Max.   :10.000     Max.   :1.0000   Max.   :1.0000
##
##  promotion_last_5years         sales         salary      salaryOrder
##  Min.   :0.00000       sales      :4140   high  :1237   Min.   :1.000
##  1st Qu.:0.00000       technical  :2720   low   :7316   1st Qu.:1.000
##  Median :0.00000       support    :2229   medium:6446   Median :2.000
##  Mean   :0.02127       IT         :1227                 Mean   :1.595
##  3rd Qu.:0.00000       product_mng: 902                 3rd Qu.:2.000
##  Max.   :1.00000       marketing  : 858                 Max.   :3.000
##                        (Other)    :2923
##  employee_satisfaction   leftFlag
##  1.Maximum:2004        Length:14999
##  2.High   :2220        Class :character
##  3.Good   :4239        Mode  :character
##  4.Average:3621
##  5.Low    :1506
##  6.Minimum:1409
##
```

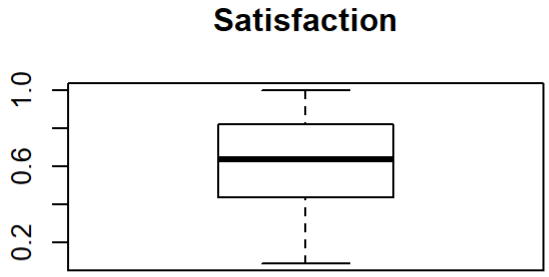
```
#Get the class
sapply(mydata,class)
```

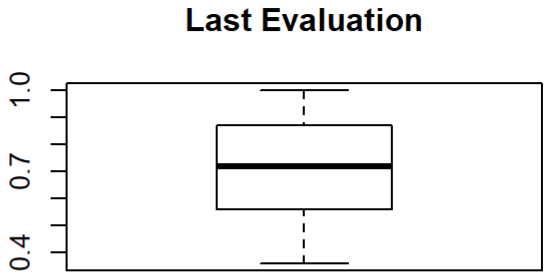
```
##    satisfaction_level       last_evaluation        number_project
##             "numeric"             "numeric"             "integer"
##  average_montly_hours    time_spend_company         Work_accident
##             "integer"             "integer"             "integer"
##                  left promotion_last_5years                 sales
##             "integer"             "integer"              "factor"
##                salary           salaryOrder employee_satisfaction
##              "factor"             "numeric"              "factor"
##              leftFlag
##           "character"
```

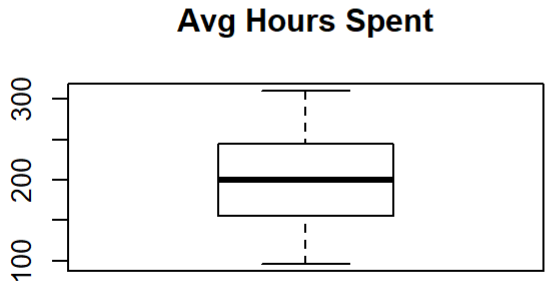
```
#par(mfrow=c(1,1))
#attach(mydata)

#********************HISTOGRAMS********************

par(mfrow=c(2,2))
hist(mydata$satisfaction_level, main = "Histogram of Satisfaction", xlab = "Satisfaction")
hist(mydata$last_evaluation, main = "Histogram of Last Evaluation", xlab = "Last Evaluation")
hist(mydata$average_montly_hours, main = "Histogram of Avg Hours Spent", xlab = "Avg Hours Spen
t")
hist(mydata$time_spend_company, main = "Histogram of Time Spent in Company", xlab = "Time Spent
 in Company")
```

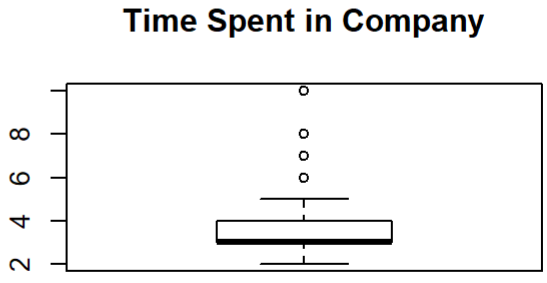



```
#********************BOXPLOTS********************

par(mfrow=c(2,2))
boxplot(mydata$satisfaction_level, main = "Satisfaction")
boxplot(mydata$last_evaluation, main = "Last Evaluation")
boxplot(mydata$average_montly_hours, main = "Avg Hours Spent")
boxplot(mydata$time_spend_company, main = "Time Spent in Company")
```

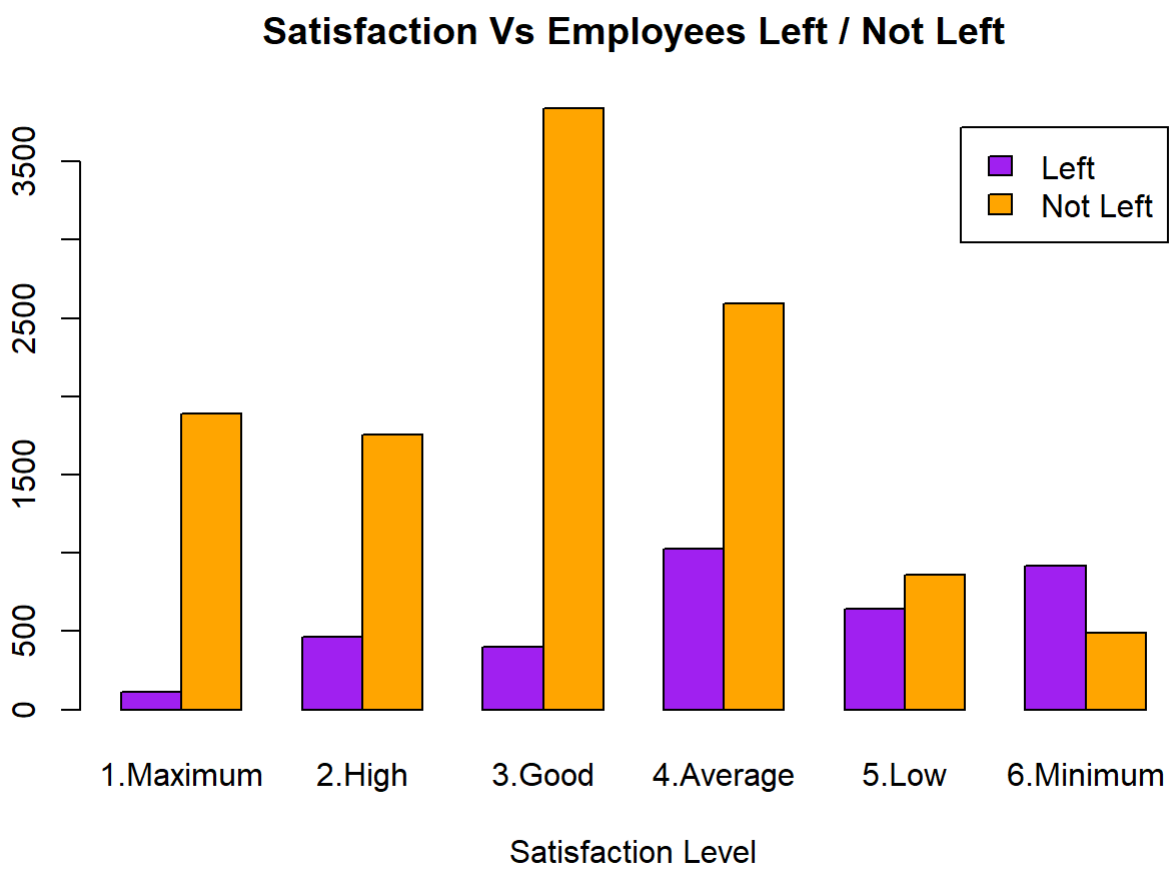




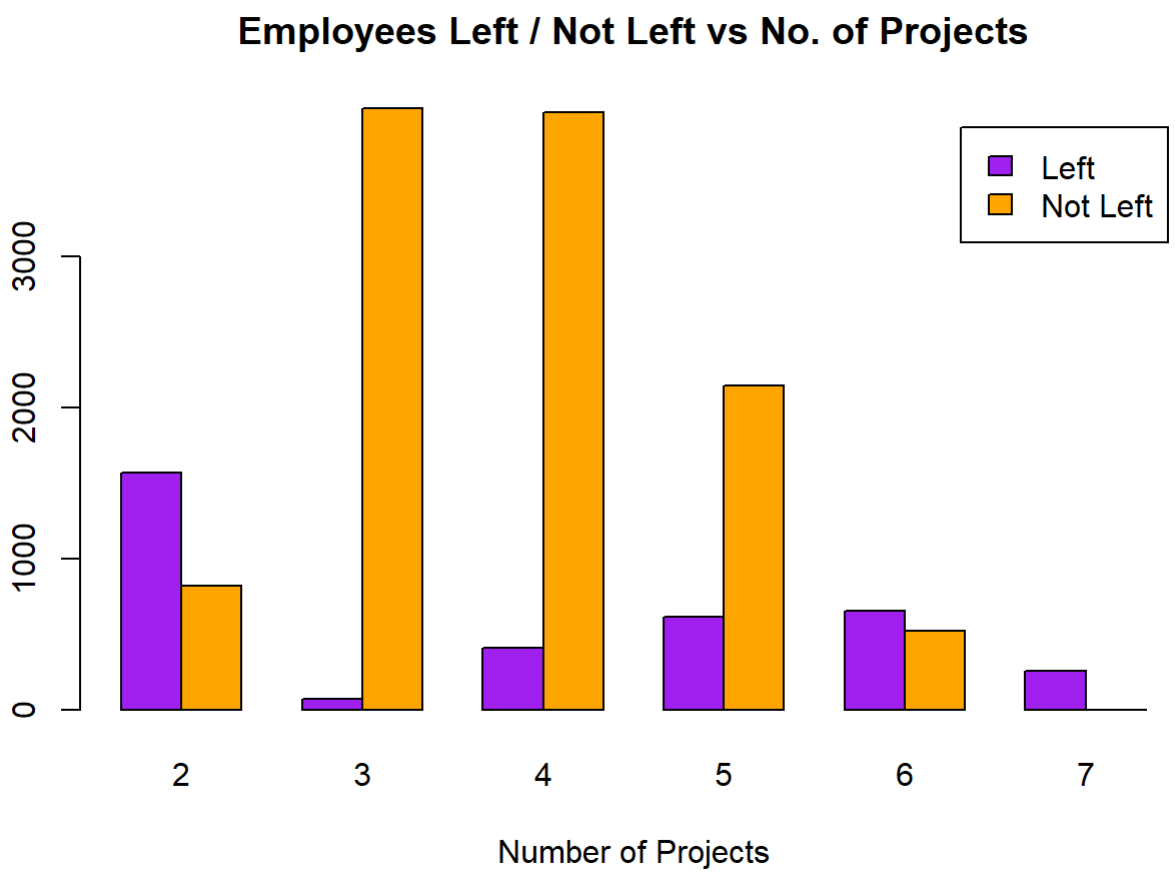




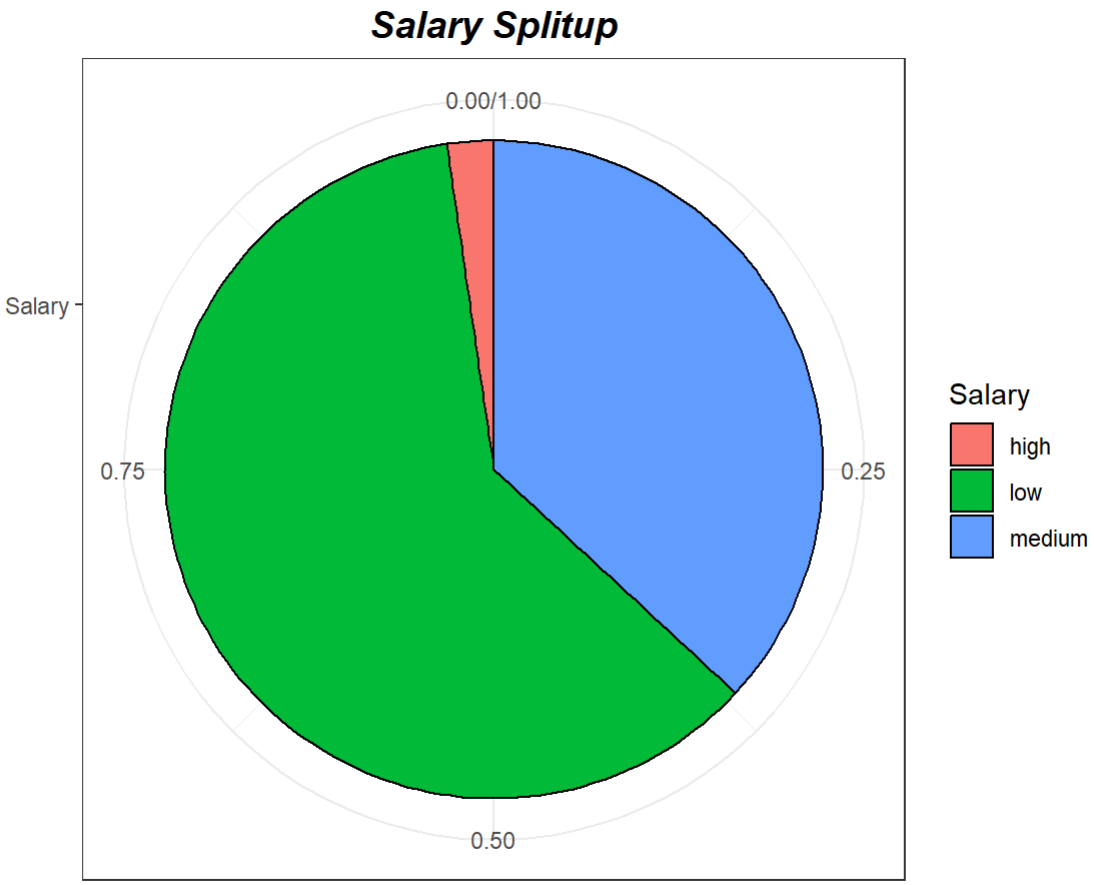

```
#*******************Satisfaction Vs Employees Left / Not Left*********************

par(mfrow=c(1,1))
#Create a barplot 'Employees left vs Satisfaction'
SatisfactionAndLeftTable <- table(mydata$leftFlag, mydata$employee_satisfaction)
barplot(SatisfactionAndLeftTable, main="Satisfaction Vs Employees Left / Not Left",
        xlab="Satisfaction Level", col=c("purple","orange"),
        legend = rownames(SatisfactionAndLeftTable), beside=TRUE)
```




```
#********************Employees Left / Not Left vs No. of Projects*********************

projectsPlotData <- table(mydata$leftFlag, mydata$number_project)
barplot(projectsPlotData, main="Employees Left / Not Left vs No. of Projects",
        xlab="Number of Projects", col=c("purple","orange"),
        legend = rownames(projectsPlotData), beside=TRUE)
```



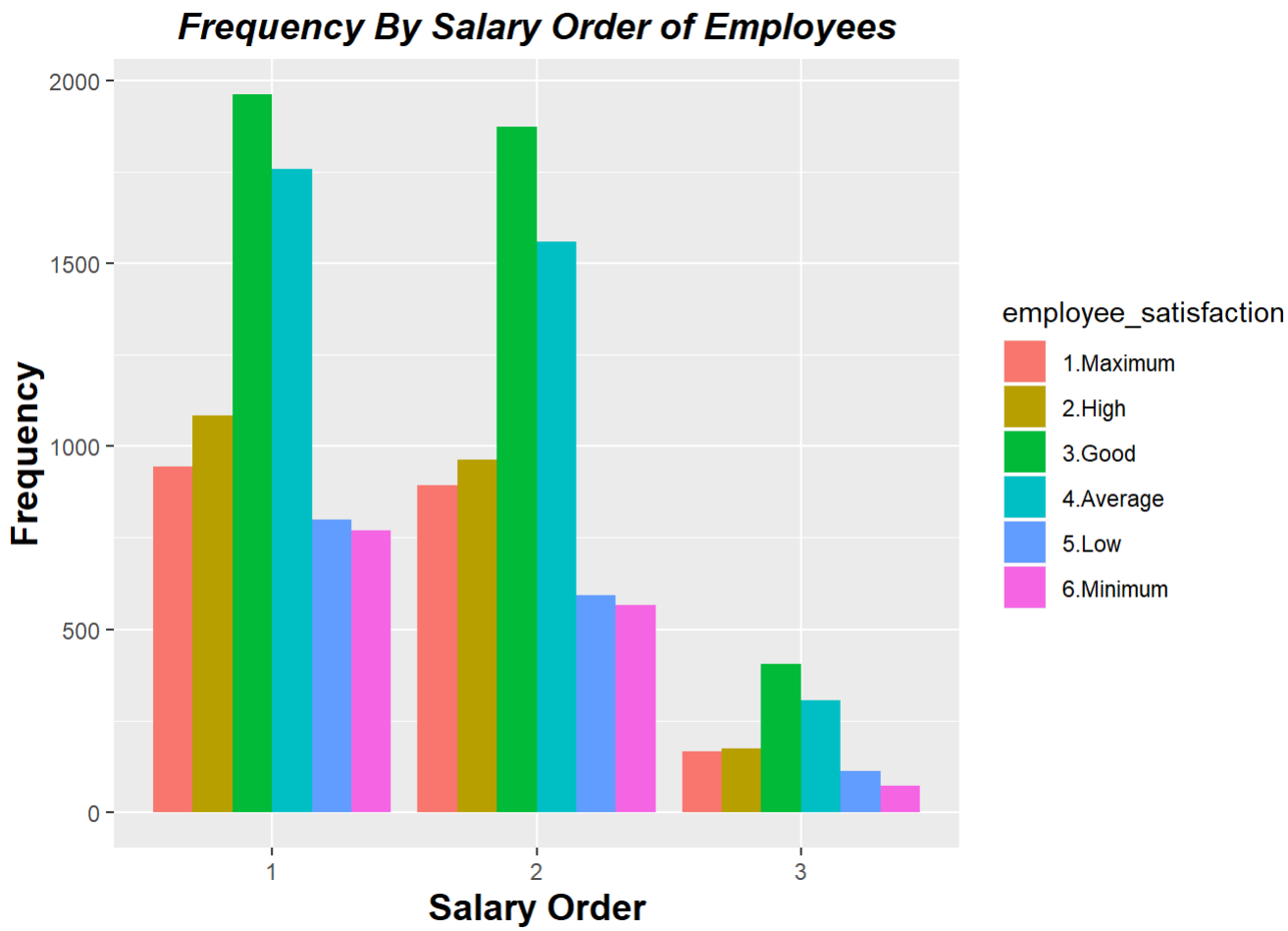

```
#********************PIE CHART********************

p = ggplot(subset(mydata,left==1), aes(x = factor('Salary'), fill = factor(salary))) +
  geom_bar(width = 1, position = "fill", color = "black") + coord_polar(theta = "y")+theme_bw()+
  ggtitle("Salary Splitup") +xlab("")+ylab("") + scale_fill_discrete(name="Salary")

p = p + theme(
  plot.title = element_text(color="Black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="Black", size=14, face="bold"),
  axis.title.y = element_text(color="Black", size=14, face="bold")
)
print(p)
```




```
#********************Frequency By Salary Order of Employees********************

table1<-table(mydata$salaryOrder,(mydata$employee_satisfaction))
#print(table1)
table1<-as.data.frame(table1)
table1$salaryOrder = table1$Var1
table1$employee_satisfaction = table1$Var2
table1$Var1= NULL
table1$Var2= NULL

print(table1)
```

```
##    Freq salaryOrder employee_satisfaction
## 1   944           1             1.Maximum
## 2   893           2             1.Maximum
## 3   167           3             1.Maximum
## 4  1084           1                2.High
## 5   963           2                2.High
## 6   173           3                2.High
## 7  1961           1                3.Good
## 8  1873           2                3.Good
## 9   405           3                3.Good
## 10 1757           1             4.Average
## 11 1558           2             4.Average
## 12  306           3             4.Average
## 13  800           1                 5.Low
## 14  593           2                 5.Low
## 15  113           3                 5.Low
## 16  770           1             6.Minimum
## 17  566           2             6.Minimum
## 18   73           3             6.Minimum
```

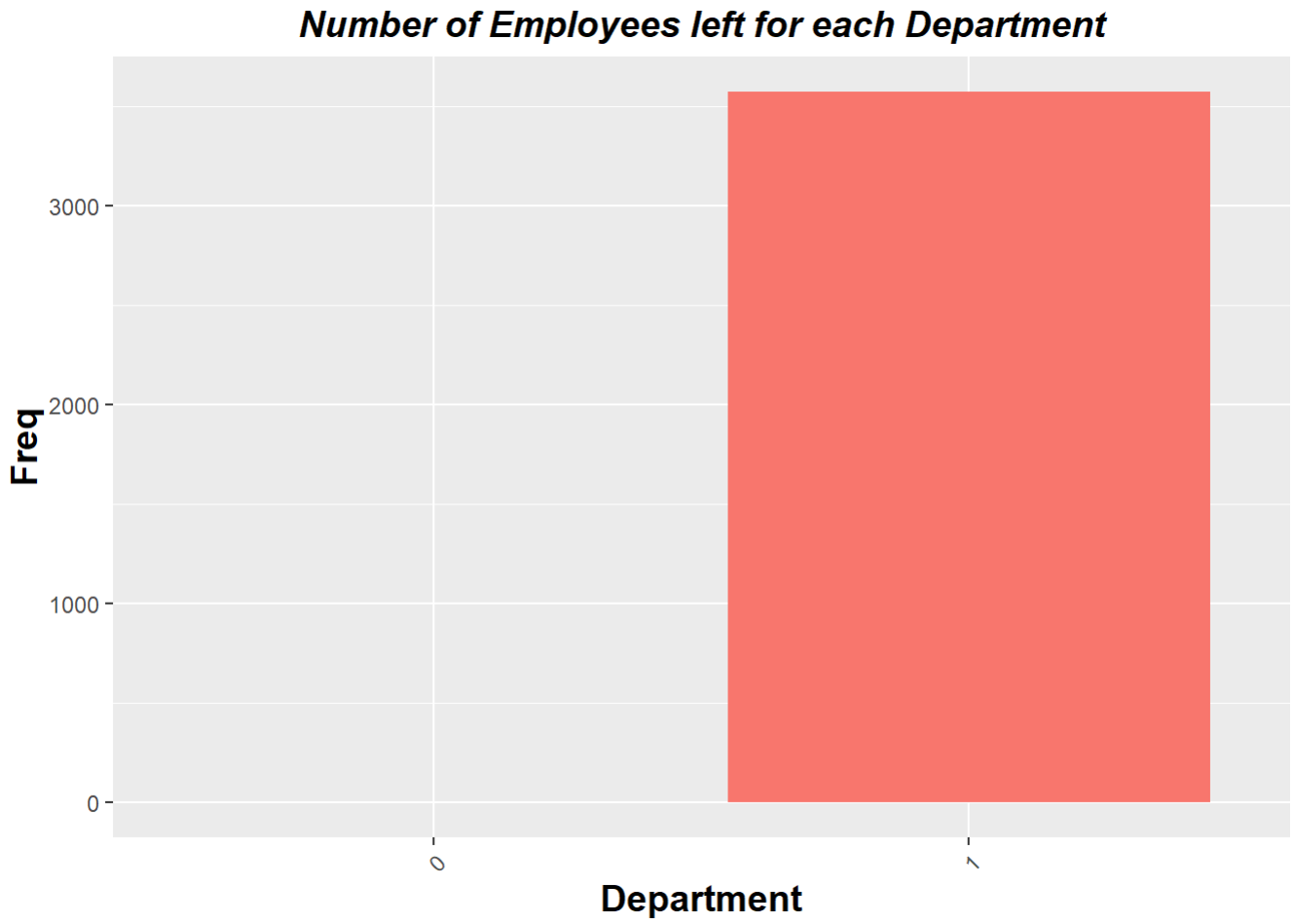
```
library(ggplot2)

p<-ggplot(table1, aes(x=salaryOrder,y=Freq,fill=employee_satisfaction)) +
  geom_bar(position="dodge",stat='identity') +
  ggtitle("Frequency By Salary Order of Employees") +xlab("Salary Order") +ylab("Frequency")

p = p + theme(
  plot.title = element_text(color="Black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="Black", size=14, face="bold"),
  axis.title.y = element_text(color="Black", size=14, face="bold")
)
print(p)
```




```
#********************Number of Employees left for each Department*********************
roleTable<-table(mydata$left,mydata$left)
roledf<-as.data.frame(roleTable)
roledf$role = roledf$Var1
roledf$leftFlag = roledf$Var2
roledf$Var1= NULL
roledf$Var2= NULL

roledfLeft<-subset(roledf,leftFlag==1)
print(roledfLeft)
```

```
##   Freq role leftFlag
## 3    0    0        1
## 4 3571    1        1
```

```
#Employees Left By Department
roledfLeft$left <- factor(roledfLeft$leftFlag, levels = roledfLeft$role[order(-roledfLeft$Fre
q)])
e<-ggplot(roledfLeft, aes(x=role,y=Freq,fill="Orange")) +
  geom_bar(stat='identity') +theme(axis.text.x = element_text(angle = 45, hjust = 1))+ guides(fi
ll=FALSE) +
  ggtitle("Number of Employees left for each Department") +xlab("Department")

e = e + theme(
  plot.title = element_text(color="Black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="Black", size=14, face="bold"),
  axis.title.y = element_text(color="Black", size=14, face="bold")
)
print(e)
```



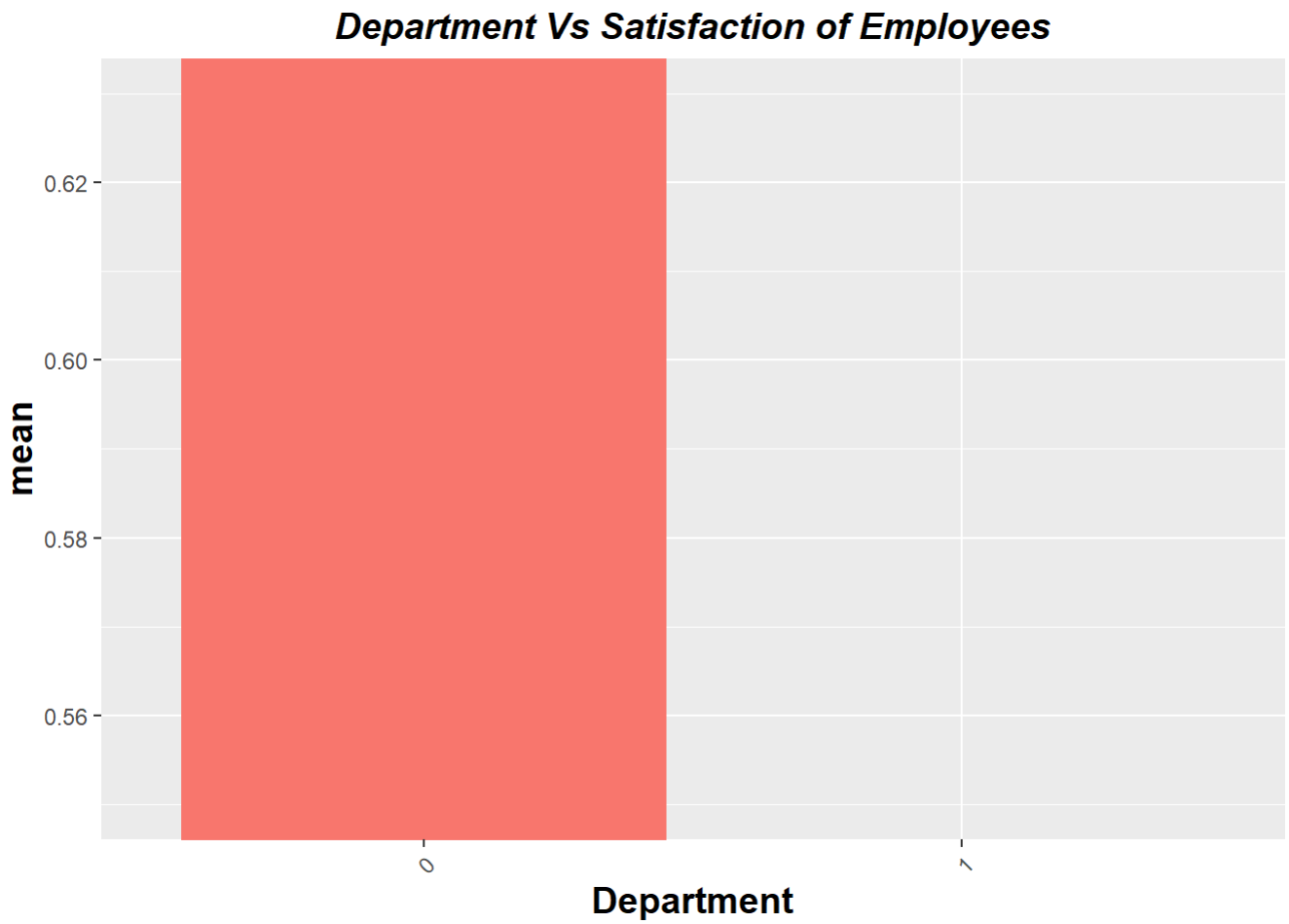

```
#*******************Department Vs Satisfaction of Employees*******************
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

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

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

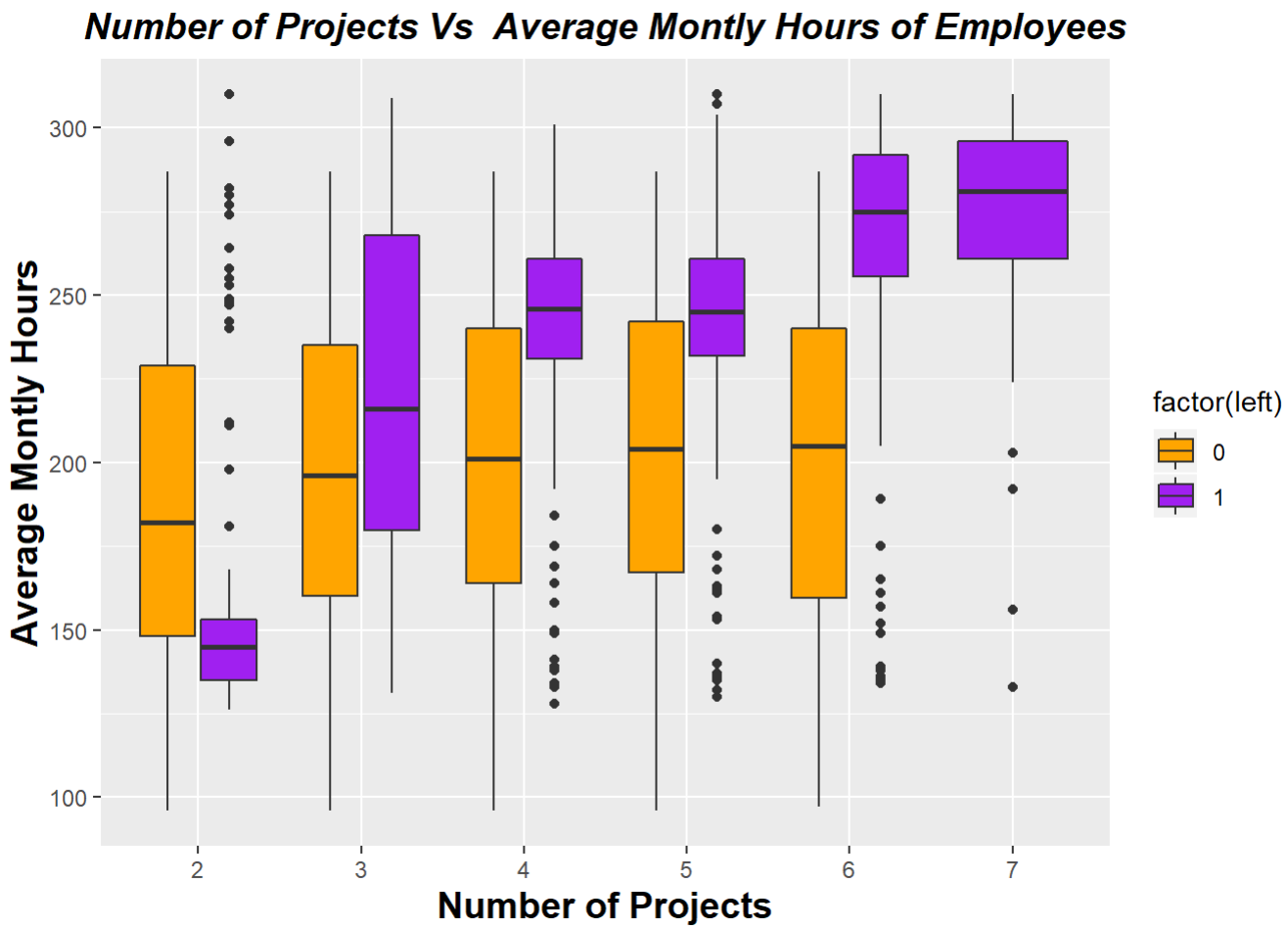
```
groupedByleft = mydata %>%
  group_by(left) %>%
  summarise(mean=mean(satisfaction_level), sd=sd(satisfaction_level), count=n())

groupedByleft = data.frame(groupedByleft)
groupedByleft = groupedByleft[order(groupedByleft$mean),]
p<-ggplot(groupedByleft, aes(x=reorder(left, -mean),y=mean,fill="Orange")) +
  geom_bar(stat='identity') +theme(axis.text.x = element_text(angle = 45, hjust = 1))+ guides(fi
ll=FALSE) +coord_cartesian(ylim = c(0.55, 0.63)) +
  ggtitle("Department Vs Satisfaction of Employees") +xlab("Department")

p = p + theme(
  plot.title = element_text(color="Black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="Black", size=14, face="bold"),
  axis.title.y = element_text(color="Black", size=14, face="bold")
)
print(p)
```




```
#********************Number of Projects Vs Satisfaction of Employees********************

p<-ggplot(mydata, aes(x = factor(number_project), y = satisfaction_level, fill=factor(left))) +
  geom_boxplot() + scale_fill_manual(values = c("orange", "purple"))+
  ggtitle("Number of Projects Vs Satisfaction of Employees") +xlab("Number of Projects") +ylab(
"Satisfaction Level")

p = p + theme(
  plot.title = element_text(color="Black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="Black", size=14, face="bold"),
  axis.title.y = element_text(color="Black", size=14, face="bold")
)
print(p)
```


```
#********************Number of Projects Vs  Last Evaluation Score of Employees**********************

p<-ggplot(mydata, aes(x = factor(number_project), y = last_evaluation, fill=factor(left))) +
  geom_boxplot() + scale_fill_manual(values = c("orange", "purple"))+
  ggtitle("Number of Projects Vs  Last Evaluation Score of Employees") +xlab("Number of Projects") +ylab("Last Evaluation")

p = p + theme(
  plot.title = element_text(color="Black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="Black", size=14, face="bold"),
  axis.title.y = element_text(color="Black", size=14, face="bold")
)
print(p)
```


```
#********************Number of Projects Vs  Average Montly Hours of Employees*******************
*

p<-ggplot(mydata, aes(x = factor(number_project), y = average_montly_hours, fill=factor(left)))
 +
  geom_boxplot() + scale_fill_manual(values = c("orange", "purple"))+
  ggtitle("Number of Projects Vs  Average Montly Hours of Employees") +xlab("Number of Projects"
) +ylab("Average Montly Hours")

p = p + theme(
  plot.title = element_text(color="Black", size=14, face="bold.italic", hjust = 0.5),
  axis.title.x = element_text(color="Black", size=14, face="bold"),
  axis.title.y = element_text(color="Black", size=14, face="bold")
)
print(p)
```

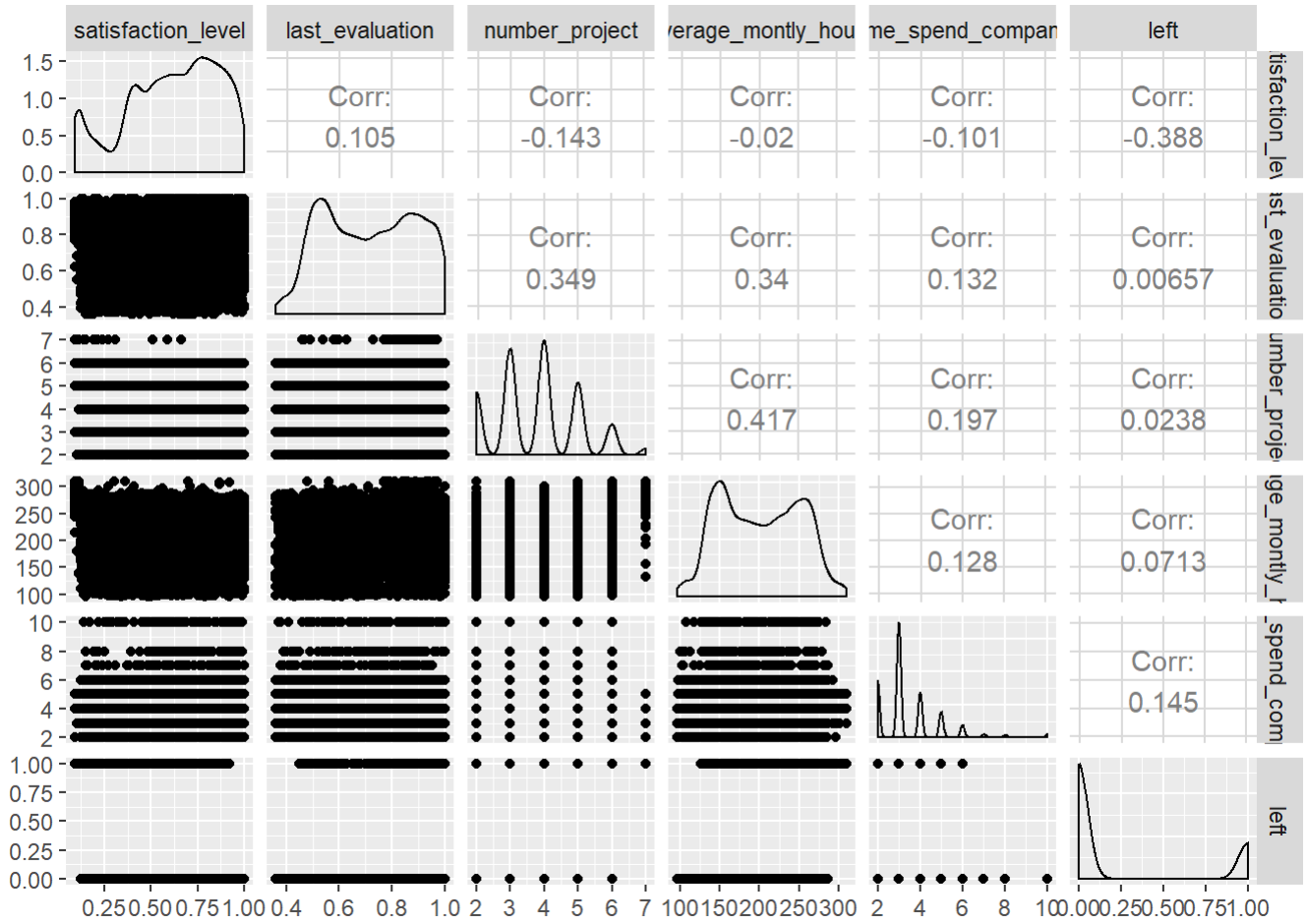



```
#********************CORRELATIONS********************
library("GGally")
```

```
##
## Attaching package: 'GGally'
```

```
## The following object is masked from 'package:dplyr':
##
##     nasa
```

```
ggpairs(mydata, columns=c("satisfaction_level","last_evaluation","number_project","average_montl
y_hours","time_spend_company","left"))
```




```
#The correlations can be interpreted as -
#Employees who worked on more projects spent more average monthly hours and their
#last evaluation was good.
#Also work accidents and whether or not an employee got promoted in last five years could
#be reasons for leaving the company.
#But most importantly, when an employee was dis-satisified they left the company.

summary(mydata)
```

```
##  satisfaction_level last_evaluation  number_project  average_montly_hours
##  Min.   :0.0900     Min.   :0.3600   Min.   :2.000   Min.   : 96.0
##  1st Qu.:0.4400     1st Qu.:0.5600   1st Qu.:3.000   1st Qu.:156.0
##  Median :0.6400     Median :0.7200   Median :4.000   Median :200.0
##  Mean   :0.6128     Mean   :0.7161   Mean   :3.803   Mean   :201.1
##  3rd Qu.:0.8200     3rd Qu.:0.8700   3rd Qu.:5.000   3rd Qu.:245.0
##  Max.   :1.0000     Max.   :1.0000   Max.   :7.000   Max.   :310.0
##
##  time_spend_company Work_accident         left
##  Min.   : 2.000     Min.   :0.0000   Min.   :0.0000
##  1st Qu.: 3.000     1st Qu.:0.0000   1st Qu.:0.0000
##  Median : 3.000     Median :0.0000   Median :0.0000
##  Mean   : 3.498     Mean   :0.1446   Mean   :0.2381
##  3rd Qu.: 4.000     3rd Qu.:0.0000   3rd Qu.:0.0000
##  Max.   :10.000     Max.   :1.0000   Max.   :1.0000
##
##  promotion_last_5years         sales         salary      salaryOrder
##  Min.   :0.00000       sales      :4140   high  :1237   Min.   :1.000
##  1st Qu.:0.00000       technical  :2720   low   :7316   1st Qu.:1.000
##  Median :0.00000       support    :2229   medium:6446   Median :2.000
##  Mean   :0.02127       IT         :1227                 Mean   :1.595
##  3rd Qu.:0.00000       product_mng: 902                 3rd Qu.:2.000
##  Max.   :1.00000       marketing  : 858                 Max.   :3.000
##                        (Other)    :2923
##  employee_satisfaction   leftFlag
##  1.Maximum:2004        Length:14999
##  2.High   :2220        Class :character
##  3.Good   :4239        Mode  :character
##  4.Average:3621
##  5.Low    :1506
##  6.Minimum:1409
##
```

```
names(mydata)
```

```
##  [1] "satisfaction_level"    "last_evaluation"
##  [3] "number_project"        "average_montly_hours"
##  [5] "time_spend_company"    "Work_accident"
##  [7] "left"                  "promotion_last_5years"
##  [9] "sales"                 "salary"
## [11] "salaryOrder"           "employee_satisfaction"
## [13] "leftFlag"
```

```
mydata$left <- factor(mydata$left)

#It can be seen that 11428 employees stayed and 3571 employees left.

#*******************LOGISTIC REGRESSION FOR 'LEFT' VARIABLE*******************
#Logistic regression is a method for fitting a regression curve, y = f(x), when y is a categoric
al variable.
#Since the model we are trying to build will be predicting whether an employee will stay (0) or
 leave (1) the company logistic regression model suits the best

#*************Split the data set to train and test data sets**********************
train <- mydata[1:12000,]
test <- mydata[12001:14999,]
dim(test)
```

```
## [1] 2999   13
```

```
dim(train)
```

```
## [1] 12000    13
```

```
library(pscl)
```

```
## Classes and Methods for R developed in the
## Political Science Computational Laboratory
## Department of Political Science
## Stanford University
## Simon Jackman
## hurdle and zeroinfl functions by Achim Zeileis
```

```
model<-glm(left~satisfaction_level+last_evaluation+average_montly_hours+salary+number_project,da
ta=train,binomial())
pR2(model)
```

```
##           llh       llhNull            G2      McFadden          r2ML
## -4533.0284865 -5406.7345064  1747.4120397     0.1615959     0.1355118
##          r2CU
##     0.2281780
```

```
summary(model)
```

```
##
## Call:
## glm(formula = left ~ satisfaction_level + last_evaluation + average_montly_hours +
##     salary + number_project, family = binomial(), data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6323  -0.5917  -0.4021  -0.2531   2.9808
##
## Coefficients:
##                         Estimate Std. Error z value Pr(>|z|)
## (Intercept)          -1.1996638  0.2106006  -5.696 1.22e-08 ***
## satisfaction_level   -4.2071384  0.1197671 -35.128  < 2e-16 ***
## last_evaluation       0.7210864  0.1745992   4.130 3.63e-05 ***
## average_montly_hours  0.0043830  0.0006041   7.255 4.01e-13 ***
## salarylow             1.6809334  0.1569462  10.710  < 2e-16 ***
## salarymedium          1.2759340  0.1584214   8.054 8.01e-16 ***
## number_project       -0.2355862  0.0249433  -9.445  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 10813.5  on 11999  degrees of freedom
## Residual deviance:  9066.1  on 11993  degrees of freedom
## AIC: 9080.1
##
## Number of Fisher Scoring iterations: 5
```

```
fitted.results <- predict(model,newdata=test,type='response')
fitted.results <- ifelse(fitted.results > 0.95,1,0)
misClasificError <- mean(fitted.results != test$left, na.rm = T)
print(paste('Accuracy',1-misClasificError))
```

```
## [1] "Accuracy 0.476158719573191"
```

```
#******************T Test to confirm the hypothesis:************************

#Let's conduct a t-test at 95% confidence level and see if it correctly rejects the null hypothe
sis that the sample comes from the same distribution as the employee population. To conduct a on
e sample t-test, we can use the stats.ttest_1samp() function:
overallSatisfaction <-mean(mydata$satisfaction_level)
left_pop<-subset(mydata,left==1)

emp_turnover_satisfaction <-mean(left_pop$satisfaction_level)
emp_turnover_satisfaction
```

```
## [1] 0.440098
```

```
#One Sample T Test
t.test(left_pop$satisfaction_level,mu=overallSatisfaction)
```

```
##
##  One Sample t-test
##
## data:  left_pop$satisfaction_level
## t = -39.109, df = 3570, p-value < 2.2e-16
## alternative hypothesis: true mean is not equal to 0.6128335
## 95 percent confidence interval:
##  0.4314385 0.4487576
## sample estimates:
## mean of x
##  0.440098
```

```
#p<0.05 - keep alt hypo; reject NULL hypo
#Reject the null hypothesis because:
#P-value is lower than confidence level of 5%

#Two Sample T Test
t.test(left_pop$satisfaction_level, mydata$satisfaction_level)
```

```
##
##  Welch Two Sample t-test
##
## data:  left_pop$satisfaction_level and mydata$satisfaction_level
## t = -35.535, df = 5182.8, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.182265 -0.163206
## sample estimates:
## mean of x mean of y
## 0.4400980 0.6128335
```

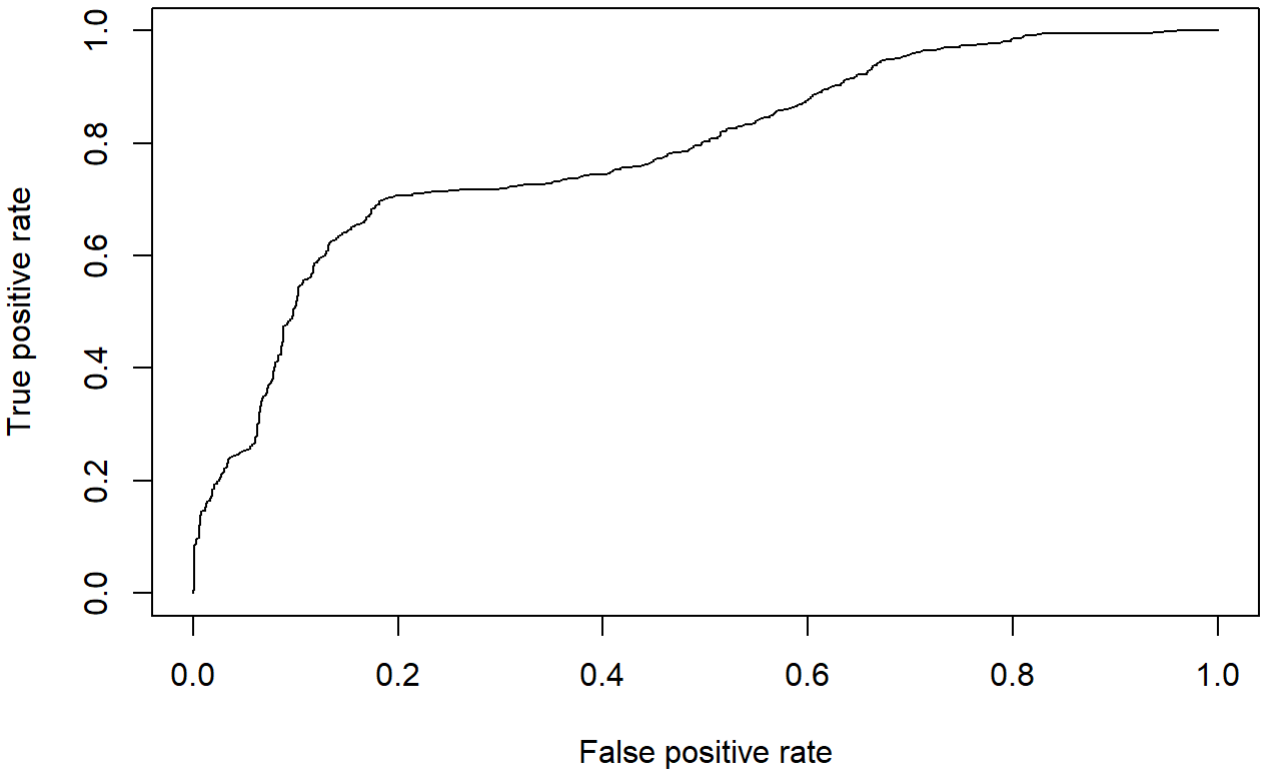
```
#p>0.05 - reject alt hypo
#p<0.05 - keep alt hypo; reject NULL hypo

library(ROCR)
```

```
## Loading required package: gplots
```

```
##
## Attaching package: 'gplots'
```

```
## The following object is masked from 'package:stats':
##
##     lowess
```

```
p <- predict(model, newdata=test, type="response")
ROCRpr <- prediction(p, test$left)
prf <- performance(ROCRpr, measure = "tpr", x.measure = "fpr")
plot(prf)
```




```
auc <- performance(ROCRpr, measure = "auc")
auc <- auc@y.values[[1]]
auc
```

```
## [1] 0.7862782
```

```
#We see that the model accuracy is not good so we try to improve our model
#One of the best ways of improving the model is by using floor and ceiling
#rounding of numbers in our data set and splitting it 80% to train and 20% to test.

#Floor method
data_size <- floor(0.8 * nrow(mydata))

#Set the seed to make your partition reproductible
set.seed(100)
train_data <- sample(seq_len(nrow(mydata)), size = data_size)
train1 <- mydata[train_data, ]
test1 <- mydata[-train_data, ]
dim(test1)
```

```
## [1] 3000   13
```

```
dim(train1)
```

```
## [1] 11999    13
```

```
model1<-glm(left ~ satisfaction_level + last_evaluation + number_project
            + average_montly_hours + time_spend_company + Work_accident
            + promotion_last_5years + salaryOrder,data=train1,binomial())

summary(model1)
```

```
##
## Call:
## glm(formula = left ~ satisfaction_level + last_evaluation + number_project +
##     average_montly_hours + time_spend_company + Work_accident +
##     promotion_last_5years + salaryOrder, family = binomial(),
##     data = train1)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1153  -0.6609  -0.4104  -0.1340   3.1374
##
## Coefficients:
##                         Estimate Std. Error z value Pr(>|z|)
## (Intercept)             1.195607   0.144686   8.263  < 2e-16 ***
## satisfaction_level     -4.123689   0.108802 -37.901  < 2e-16 ***
## last_evaluation         0.726453   0.165733   4.383 1.17e-05 ***
## number_project         -0.297526   0.023585 -12.615  < 2e-16 ***
## average_montly_hours    0.004200   0.000573   7.329 2.32e-13 ***
## time_spend_company      0.247627   0.017021  14.549  < 2e-16 ***
## Work_accident          -1.510462   0.099341 -15.205  < 2e-16 ***
## promotion_last_5years -1.495025   0.280668  -5.327 1.00e-07 ***
## salaryOrder            -0.687346   0.042118 -16.320  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 13179  on 11998  degrees of freedom
## Residual deviance: 10370  on 11990  degrees of freedom
## AIC: 10388
##
## Number of Fisher Scoring iterations: 5
```

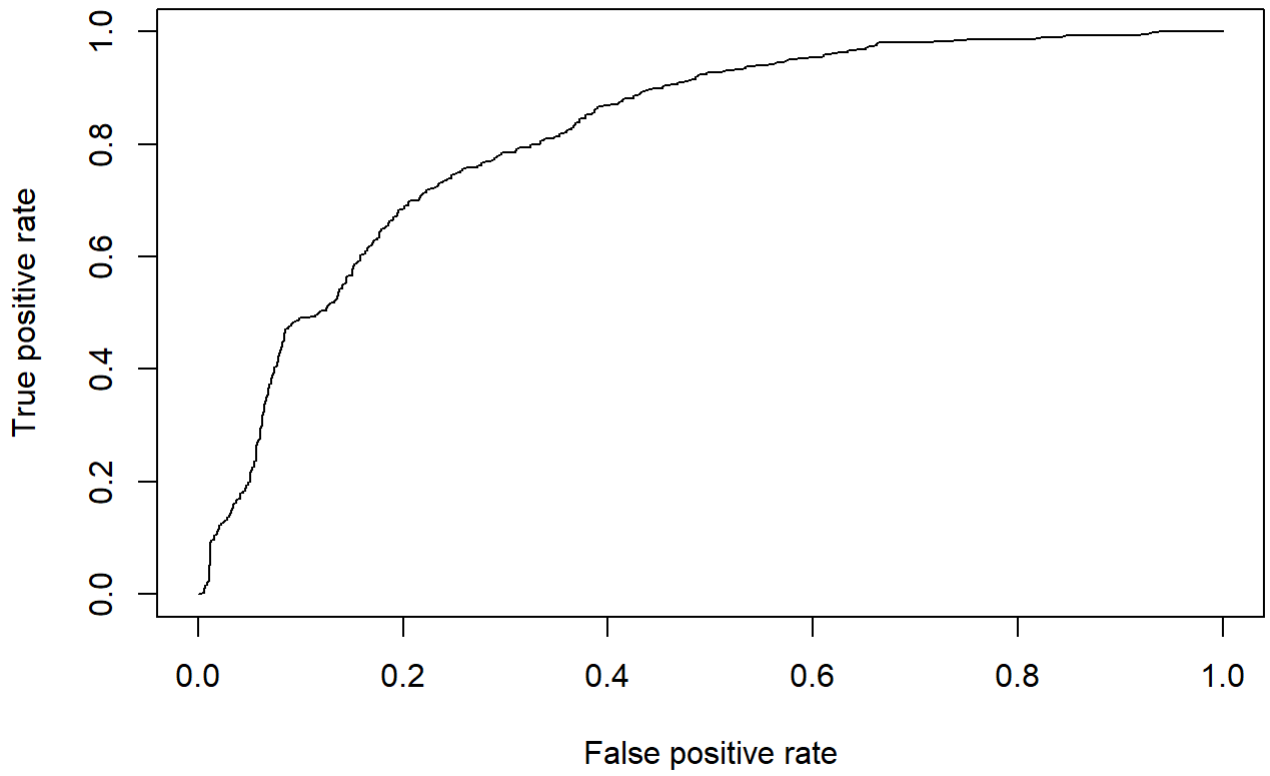
```
anova(model1, test="Chisq")
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: left
##
## Terms added sequentially (first to last)
##
##
##                       Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                                  11998      13179
## satisfaction_level     1  1857.06     11997      11322 < 2.2e-16 ***
## last_evaluation        1    13.39     11996      11309 0.0002525 ***
## number_project         1    87.68     11995      11221 < 2.2e-16 ***
## average_montly_hours   1    59.76     11994      11161 1.069e-14 ***
## time_spend_company     1   138.69     11993      11022 < 2.2e-16 ***
## Work_accident          1   307.46     11992      10715 < 2.2e-16 ***
## promotion_last_5years  1    57.67     11991      10657 3.099e-14 ***
## salaryOrder            1   287.17     11990      10370 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#pR2(model1)

fitted.results <- predict(model1,newdata=test1,type='response')
fitted.results <- ifelse(fitted.results > 0.95,1,0)
misClasificError <- mean(fitted.results != test1$left, na.rm = T)
print(paste('Accuracy',1-misClasificError))
```

```
## [1] "Accuracy 0.763"
```

```
p <- predict(model1, newdata=test1, type="response")
pr <- prediction(p, test1$left)

prf <- performance(pr, measure = "tpr", x.measure = "fpr")
plot(prf)
```




```
auc <- performance(pr, measure = "auc")
auc <- auc@y.values[[1]]
auc
```

```
## [1] 0.8173433
```

```
#We see a lot of improvement in our model through floor effect

#Ceiling effect
data_size1 <- ceiling(0.8 * nrow(mydata))

#Set the seed to make your partition reproductible
set.seed(100)
train_data1 <- sample(seq_len(nrow(mydata)), size = data_size1)
train2 <- mydata[train_data1, ]
test2 <- mydata[-train_data1, ]
dim(test2)
```

```
## [1] 2999   13
```

```
dim(train2)
```

```
## [1] 12000    13
```

```
model2<-glm(left ~ satisfaction_level + last_evaluation + number_project
            + average_montly_hours + time_spend_company + Work_accident
            + promotion_last_5years + salaryOrder,data=train2,binomial())

summary(model2)
```

```
##
## Call:
## glm(formula = left ~ satisfaction_level + last_evaluation + number_project +
##     average_montly_hours + time_spend_company + Work_accident +
##     promotion_last_5years + salaryOrder, family = binomial(),
##     data = train2)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1154  -0.6609  -0.4103  -0.1341   3.1374
##
## Coefficients:
##                         Estimate Std. Error z value Pr(>|z|)
## (Intercept)             1.195708   0.144687   8.264  < 2e-16 ***
## satisfaction_level     -4.123951   0.108801 -37.904  < 2e-16 ***
## last_evaluation         0.726877   0.165730   4.386 1.16e-05 ***
## number_project         -0.297586   0.023585 -12.618  < 2e-16 ***
## average_montly_hours    0.004199   0.000573   7.328 2.33e-13 ***
## time_spend_company      0.247667   0.017020  14.551  < 2e-16 ***
## Work_accident          -1.510452   0.099342 -15.205  < 2e-16 ***
## promotion_last_5years -1.495029   0.280672  -5.327 1.00e-07 ***
## salaryOrder            -0.687422   0.042118 -16.321  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 13180  on 11999  degrees of freedom
## Residual deviance: 10370  on 11991  degrees of freedom
## AIC: 10388
##
## Number of Fisher Scoring iterations: 5
```

```
anova(model2, test="Chisq")
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: left
##
## Terms added sequentially (first to last)
##
##
##                       Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                                  11999      13180
## satisfaction_level     1  1857.29     11998      11322 < 2.2e-16 ***
## last_evaluation        1    13.41     11997      11309 0.0002502 ***
## number_project         1    87.77     11996      11221 < 2.2e-16 ***
## average_montly_hours   1    59.74     11995      11161 1.084e-14 ***
## time_spend_company     1   138.76     11994      11023 < 2.2e-16 ***
## Work_accident          1   307.44     11993      10715 < 2.2e-16 ***
## promotion_last_5years  1    57.67     11992      10658 3.099e-14 ***
## salaryOrder            1   287.23     11991      10370 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#pR2(model2)
fitted.results <- predict(model2,newdata=test2,type='response')
fitted.results <- ifelse(fitted.results > 0.95,1,0)
misClasificError <- mean(fitted.results != test2$left, na.rm = T)
print(paste('Accuracy',1-misClasificError))
```

```
## [1] "Accuracy 0.762920973657886"
```

```
auc <- performance(pr, measure = "auc")
auc <- auc@y.values[[1]]
auc
```

```
## [1] 0.8173433
```

```
#We see similar accuracy in ceiling as in floor, both gave better accuracy than manual split mod
el

#Find the mean of employees population who stayed and mean of employees who left
data_set_satisfaction <-mean(mydata$satisfaction_level)
left_satisfaction <-subset(mydata,left==1)
stay_satisfaction <-subset(mydata,left==0)
data_set_left_satisfaction <-mean(left_satisfaction$satisfaction_level)
data_set_stay_satisfaction <-mean(stay_satisfaction$satisfaction_level)
print( c(data_set_stay_satisfaction,data_set_left_satisfaction ) )
```

```
## [1] 0.6668096 0.4400980
```

```
#Welch Two Sample t-test
t.test(left_satisfaction$satisfaction_level,mu=data_set_stay_satisfaction)
```

```
##
##  One Sample t-test
##
## data:  left_satisfaction$satisfaction_level
## t = -51.33, df = 3570, p-value < 2.2e-16
## alternative hypothesis: true mean is not equal to 0.6668096
## 95 percent confidence interval:
##  0.4314385 0.4487576
## sample estimates:
## mean of x
##  0.440098
```

```
# Employee Population mean ssatisfaction

#Convert the variable left to numeric and find the confidence interval
left_new <- sum(as.numeric(mydata$left))
LWR <-qt(0.025,left_new)  # Low Quartile
UPR <-qt(0.95,left_new)  # High Quartile
print (c(LWR, UPR))
```

```
## [1] -1.960092  1.644936
```

```
#To summarize our analysis on why employees leave the company, out of all the contributing facto
rs
#the strongest predictor is Employee satisfaction.
#Employees generally leave when they are overworked(more than 250 average_monthly_hours) or unde
rworked
#(less than 150 average_monthly_hours)
#Employees with low or really high evaluations are probably leaving the company
#Employees with low or medium salaries left the company
#Employees who had less(less than 3 number of projects) or more (6 or above) project count are l
eaving the company
```