Human Resource Analytics

Table of Contents

1. Introduction and Motivations

- It has been observed that a company loses way more money by hiring new people and training them for a period of time than giving their well-deserved employee incentives.
- Our motivation lies on determining what the company is not doing or needs to do for their employees, so that they can retain their employees.
- Our project revolves around analyzing the best model for a company to do HR analysis for figuring out reasons to retain talent.
- Our model will be created considering the most impactful parameters that can help determine factors that impact employees' decision to stay or leave.

2. Data Description

- The dataset belongs to Human Resource which is also our domain.
- The data was obtained from: https://www.kaggle.com/ludobenistant/hr-analytics
- Our dataset has 14999 rows and 10 columns.
- The fields of the data set can be divided into categorical data and numerical data.

Field Name	Data Type
satisfaction_level	Numerical Data
last_evaluation	Numerical Data
number_project	Numerical Data
average_montly_hours	Numerical Data
time_spend_company	Numerical Data
Work_accident	Numerical Data
left	Numerical Data
promotion_last_5years	Numerical Data
sales	Categorical data
salary	Categorical data

3. Research Problems and Solutions

Research problems

- The primary goal of our research is to find a solution for the problem of the best and most experienced employees leaving the company prematurely. These are some of the researched problems which we want to explore:

  1) Exploring employee's level of satisfaction, their evaluation and check if any department is under staffed leading to high load on the existing employees and having no balance in work and personal life.

2) Considering different parameters like number_project, average_monthly_hours, time_spend_company, work_accident and promotion_last_5years against people who have left company in past, we can determine which of these reasons impact the most in losing talent.

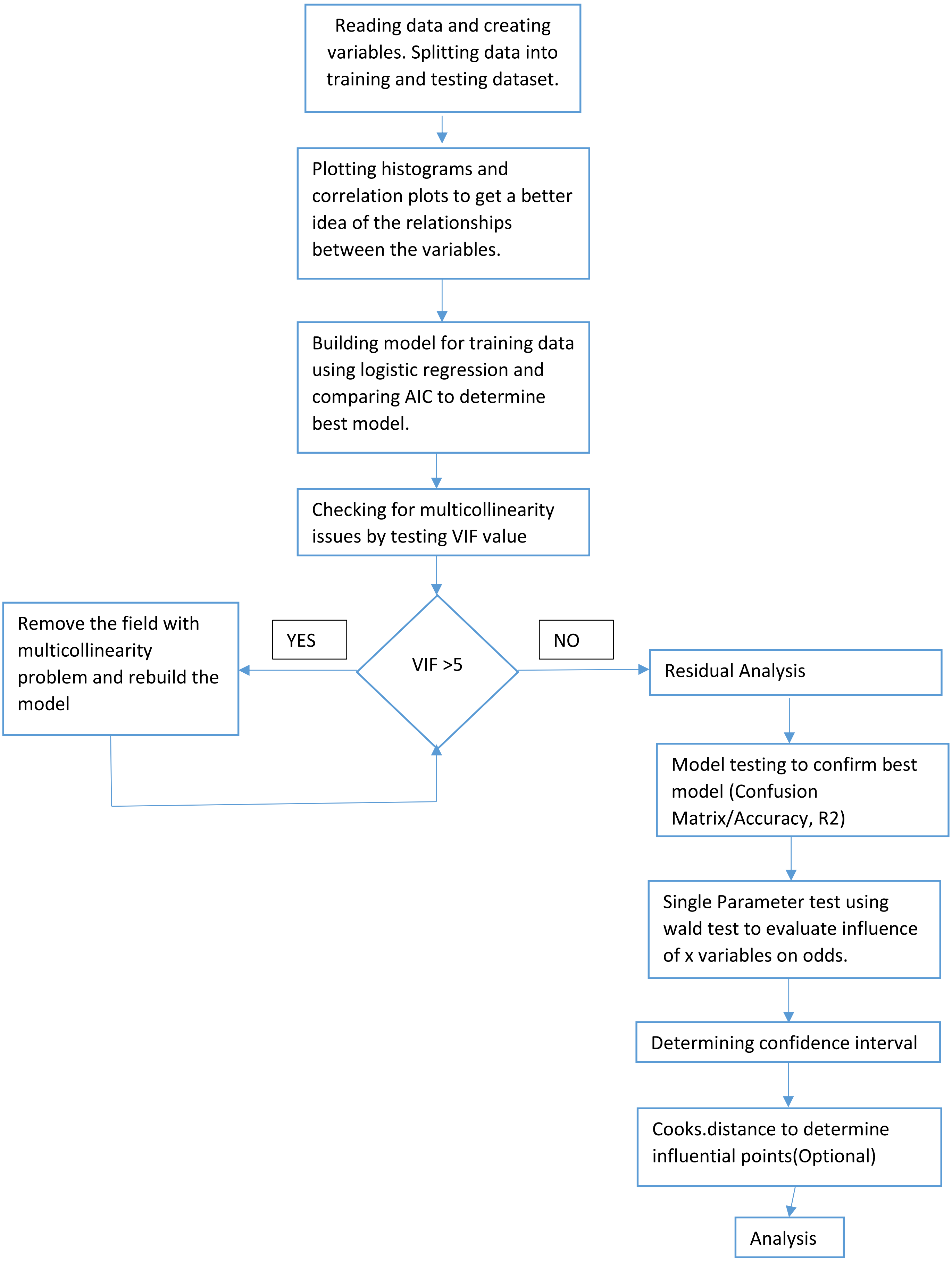
3) We will explore employee details and analyze the ones that need to be retained. A company needs to know which employees were valuable and have left or might leave.

Potential Solutions

Our model should be able to determine employees probable on leaving the companies and this will help the company to reach out to them, in order to retain the talent. This can be done department wise or taking into consideration other factors like time_spend_company for retaining experienced employees.

4. Model Learning

4.1. Data Processing



Reading the Dataset and splitting it into train(80%) for model selection and test(20%) data for model evaluation

```
> HrData=HrData[sample(nrow(HrData)),]
> select.data= sample (1:nrow(HrData), 0.8*nrow(HrData))
> train.data= HrData[select.data,]
> test.data= HrData[-select.data,]
>
>
> satisfaction_level=HrData$satisfaction_level
> last_evaluation=HrData$last_evaluation
> number_project=HrData$number_project
> average_montly_hours=HrData$average_montly_hours
> time_spend_company=HrData$time_spend_company
> Work_accident=HrData$Work_accident
> left=HrData$left
> promotion_last_5years=HrData$promotion_last_5years
> sales=factor(HrData$sales)
> salary=factor(HrData$salary)
>
```

To display the number of rows for training and testing data.

```
>
> nrow(test.data)
[1] 3000
> nrow(train.data)
[1] 11999
>
>
```

4.2. Data Analytics Tasks and Processes

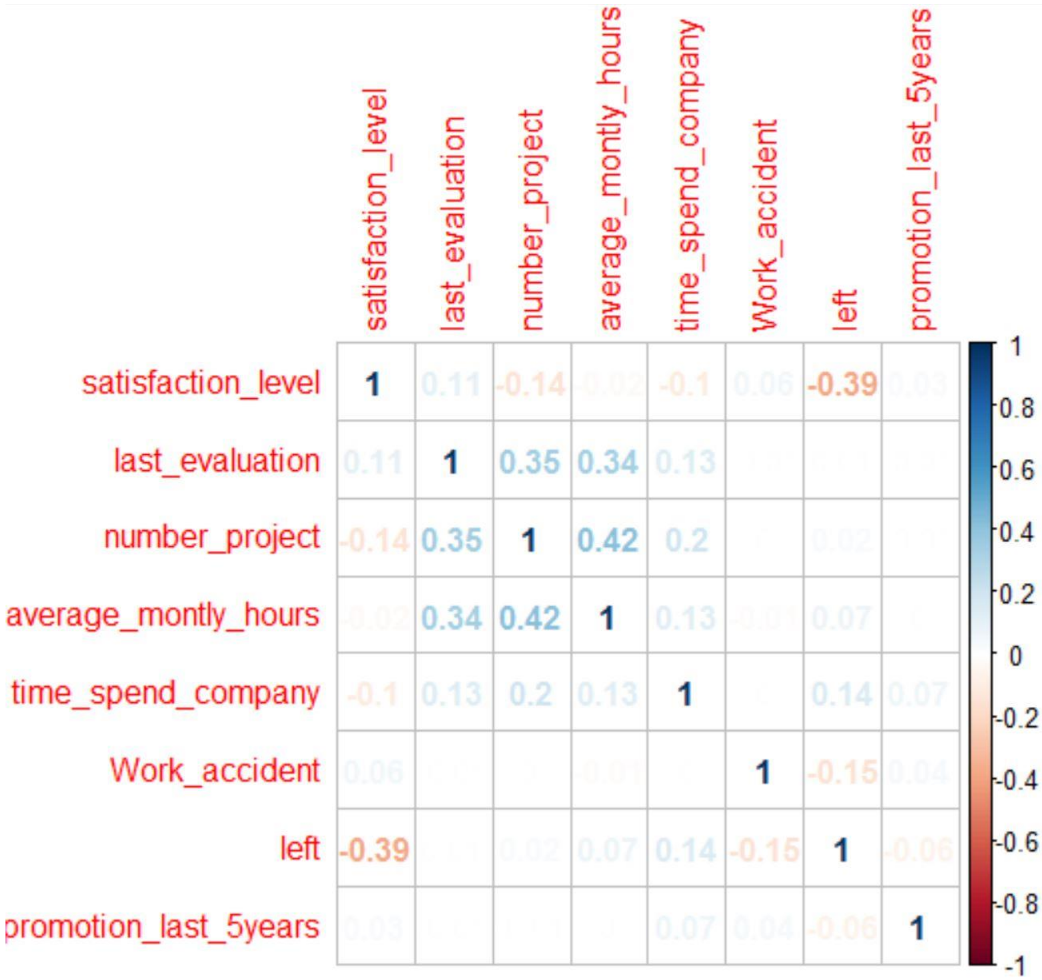
Statistical description of the dataset.

This table describes the characteristics of each parameters. For eg. We can see that satisfaction level is equal to 62%, performance average is around 72%, people mostly work on 3 to 4 projects etc.

```
> summary(HrData)
 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         promotion_last_5years
 Min.   : 2.000     Min.   :0.0000   Min.   :0.0000   Min.   :0.00000
 1st Qu.: 3.000     1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.00000
 Median : 3.000     Median :0.0000   Median :0.0000   Median :0.00000
 Mean   : 3.498     Mean   :0.1446   Mean   :0.2381   Mean   :0.02127
 3rd Qu.: 4.000     3rd Qu.:0.0000   3rd Qu.:0.0000   3rd Qu.:0.00000
 Max.   :10.000     Max.   :1.0000   Max.   :1.0000   Max.   :1.00000

         sales         salary
 sales      :4140   high  :1237
 technical  :2720   low   :7316
 support    :2229   medium:6446
 IT         :1227
 product_mng: 902
 marketing  : 858
 (Other)    :2923
>
```

Displaying correlations between variables by correlation plot.
Load library "corrplot" for the same after installing the package.

```
>
> library(corrplot)
Warning message:
package 'corrplot' was built under R version 3.3.3
> m=cor(HrData[,-c(9,10)])
> corrplot(m,method="number")
>
>
```

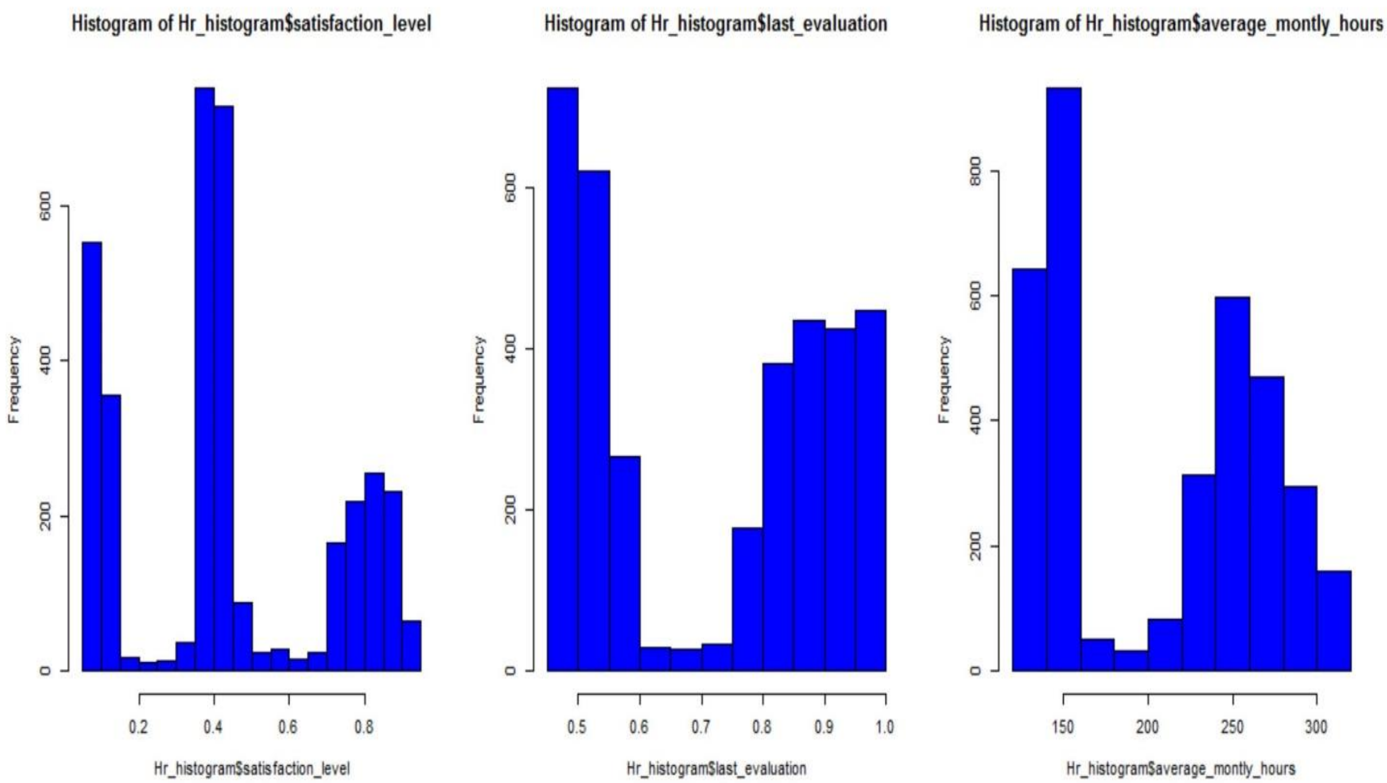


Here the number represents the significance of the correlation while the color presents the direction i.e. positive or negative

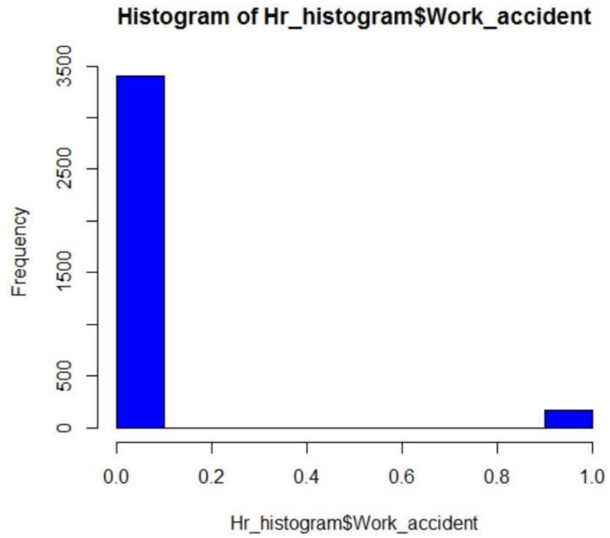
Histogram representation of different parameters(satisfaction_level,last_evaluation and average_montly_hours,work_accident and salary) for scenario of left==1.

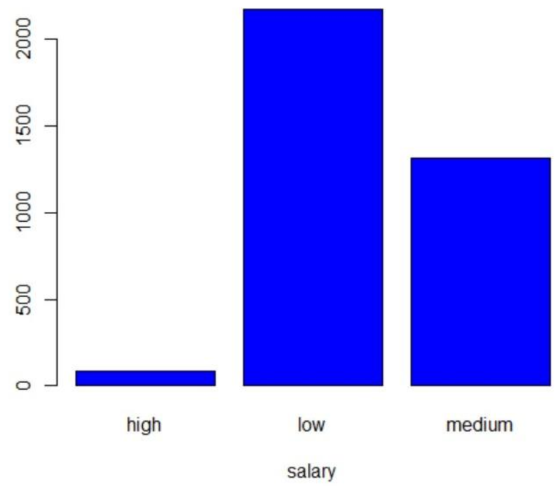
Load library "dplyr" for the same after installing the package.

```
> 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

Warning message:
package 'dplyr' was built under R version 3.3.3
> Hr_histogram=filter(HrData,left==1)
> par(mfrow=c(1,3))
> hist(Hr_histogram$satisfaction_level,col="blue")
> hist(Hr_histogram$last_evaluation,col="blue")
> hist(Hr_histogram$average_montly_hours,col="blue")
>
```




```
>
> par(mfrow=c(1,2))
> hist(Hr_histogram$Work_accident,col="blue")
> plot(Hr_histogram$salary,col="blue",xlab="salary")
>
```





Model Selection

We will perform model selection based on train data set

- Building model without categorical data - m1

```
>
> m1=glm(left ~ satisfaction_level+last_evaluation+number_project+average_montly_hours+time_spend_company+Work_accident+
+ promotion_last_5years,data=train.data,family=binomial())
> summary(m1)

Call:
glm(formula = left ~ satisfaction_level + last_evaluation + number_project +
    average_montly_hours + time_spend_company + Work_accident +
    promotion_last_5years, family = binomial(), data = train.data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.3319  -0.6823  -0.4354  -0.1510   3.1608

Coefficients:
                       Estimate Std. Error z value Pr(>|z|)
(Intercept)            0.207015   0.130856   1.582    0.114
satisfaction_level    -4.129106   0.108236 -38.149  < 2e-16 ***
last_evaluation        0.804429   0.163378   4.924 8.49e-07 ***
number_project        -0.305632   0.023314 -13.109  < 2e-16 ***
average_montly_hours   0.004362   0.000562   7.761 8.39e-15 ***
time_spend_company     0.220496   0.016586  13.294  < 2e-16 ***
Work_accident         -1.464746   0.097627 -15.003  < 2e-16 ***
promotion_last_5years -1.948149   0.296582  -6.569 5.08e-11 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 13193  on 11998  degrees of freedom
Residual deviance: 10688  on 11991  degrees of freedom
AIC: 10704
```

- Building model with categorical data - m2

```
> m2=glm(left ~ satisfaction_level+last_evaluation+number_project+average_montly_hours+time_spend_company+Work_accident+
+ promotion_last_5years+salary+sales,data=train.data,family=binomial())
> summary(m2)

Call:
glm(formula = left ~ satisfaction_level + last_evaluation + number_project +
    average_montly_hours + time_spend_company + Work_accident +
    promotion_last_5years + salary + sales, family = binomial(),
    data = train.data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.2105  -0.6613  -0.4015  -0.1137   3.0861

Coefficients:
                        Estimate Std. Error z value Pr(>|z|)
(Intercept)           -1.5589519  0.2167582  -7.192 6.38e-13 ***
satisfaction_level    -4.1420325  0.1100679 -37.632  < 2e-16 ***
last_evaluation        0.7474836  0.1673503   4.467 7.95e-06 ***
number_project        -0.3105897  0.0238781 -13.007  < 2e-16 ***
average_montly_hours   0.0045034  0.0005766   7.811 5.68e-15 ***
time_spend_company     0.2616390  0.0174520  14.992  < 2e-16 ***
Work_accident         -1.4968097  0.0990446 -15.112  < 2e-16 ***
promotion_last_5years -1.5143858  0.2975207  -5.090 3.58e-07 ***
salarylow              2.0086379  0.1439724  13.952  < 2e-16 ***
salarymedium           1.4796288  0.1448026  10.218  < 2e-16 ***
saleshr                0.2992939  0.1462292   2.047   0.0407 *
salesIT               -0.1663918  0.1372996  -1.212   0.2256
salesmanagement       -0.4186147  0.1778950  -2.353   0.0186 *
salesmarketing        -0.0071032  0.1484049  -0.048   0.9618
salesproduct_mng      -0.0796495  0.1469683  -0.542   0.5879
salesRandD            -0.5925952  0.1638949  -3.616   0.0003 ***
salessales            -0.0345029  0.1153554  -0.299   0.7649

salessupport           0.0392502  0.1226361   0.320   0.7489
salestechnical         0.0796572  0.1196290   0.666   0.5055
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 13193  on 11998  degrees of freedom
Residual deviance: 10284  on 11980  degrees of freedom
AIC: 10322

Number of Fisher Scoring iterations: 5

>
```

Trying to build model with step function

- Using step function and building model with backward elimination

```
> backward=step(m2,direction="backward",trace=F)
> summary(backward)

Call:
glm(formula = left ~ satisfaction_level + last_evaluation + number_project +
    average_montly_hours + time_spend_company + Work_accident +
    promotion_last_5years + salary + sales, family = binomial(),
    data = train.data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.2105  -0.6613  -0.4015  -0.1137   3.0861

Coefficients:
                       Estimate Std. Error z value Pr(>|z|)
(Intercept)          -1.5589519  0.2167582  -7.192 6.38e-13 ***
satisfaction_level   -4.1420325  0.1100679 -37.632  < 2e-16 ***
last_evaluation       0.7474836  0.1673503   4.467 7.95e-06 ***
number_project       -0.3105897  0.0238781 -13.007  < 2e-16 ***
average_montly_hours  0.0045034  0.0005766   7.811 5.68e-15 ***
time_spend_company    0.2616390  0.0174520  14.992  < 2e-16 ***
Work_accident        -1.4968097  0.0990446 -15.112  < 2e-16 ***
promotion_last_5years -1.5143858  0.2975207  -5.090 3.58e-07 ***
salarylow             2.0086379  0.1439724  13.952  < 2e-16 ***
salarymedium          1.4796288  0.1448026  10.218  < 2e-16 ***
saleshr               0.2992939  0.1462292   2.047   0.0407 *
salesIT              -0.1663918  0.1372996  -1.212   0.2256
salesmanagement      -0.4186147  0.1778950  -2.353   0.0186 *
salesmarketing       -0.0071032  0.1484049  -0.048   0.9618
salesproduct_mng     -0.0796495  0.1469683  -0.542   0.5879
salesRandD           -0.5925952  0.1638949  -3.616   0.0003 ***
salessales           -0.0345029  0.1153554  -0.299   0.7649
salessupport          0.0392502  0.1226361   0.320   0.7489

salestechnical        0.0796572  0.1196290   0.666   0.5055
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 13193  on 11998  degrees of freedom
Residual deviance: 10284  on 11980  degrees of freedom
AIC: 10322

Number of Fisher Scoring iterations: 5

>
```

- Using Step function and building model with forward selection

```
> base=glm(left~satisfaction_level,data=train.data,family=binomial)
> forward=step(base, scope=list(upper=m2, lower=~1), direction="forward", trace=F)
> summary(forward)

Call:
glm(formula = left ~ satisfaction_level + salary + Work_accident +
    time_spend_company + number_project + average_montly_hours +
    promotion_last_5years + sales + last_evaluation, family = binomial,
    data = train.data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.2105  -0.6613  -0.4015  -0.1137   3.0861

Coefficients:
                       Estimate Std. Error z value Pr(>|z|)
(Intercept)          -1.5589519  0.2167582  -7.192 6.38e-13 ***
satisfaction_level   -4.1420325  0.1100679 -37.632  < 2e-16 ***
salarylow             2.0086379  0.1439724  13.952  < 2e-16 ***
salarymedium          1.4796288  0.1448026  10.218  < 2e-16 ***
Work_accident        -1.4968097  0.0990446 -15.112  < 2e-16 ***
time_spend_company    0.2616390  0.0174520  14.992  < 2e-16 ***
number_project       -0.3105897  0.0238781 -13.007  < 2e-16 ***
average_montly_hours  0.0045034  0.0005766   7.811 5.68e-15 ***
promotion_last_5years -1.5143858  0.2975207  -5.090 3.58e-07 ***
saleshr               0.2992939  0.1462292   2.047   0.0407 *
salesIT              -0.1663918  0.1372996  -1.212   0.2256
salesmanagement      -0.4186147  0.1778950  -2.353   0.0186 *
salesmarketing       -0.0071032  0.1484049  -0.048   0.9618
salesproduct_mng     -0.0796495  0.1469683  -0.542   0.5879
salesRandD           -0.5925952  0.1638949  -3.616   0.0003 ***
salessales           -0.0345029  0.1153554  -0.299   0.7649
salessupport          0.0392502  0.1226361   0.320   0.7489

salestechnical        0.0796572  0.1196290   0.666   0.5055
last_evaluation       0.7474836  0.1673503   4.467 7.95e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 13193  on 11998  degrees of freedom
Residual deviance: 10284  on 11980  degrees of freedom
AIC: 10322

Number of Fisher Scoring iterations: 5
```

- Using Step function and keeping direction=both

```
> both=step(base, scope=list(upper=m2, lower=~1), direction="both", trace=F)
> summary(both)

Call:
glm(formula = left ~ satisfaction_level + salary + Work_accident +
    time_spend_company + number_project + average_montly_hours +
    promotion_last_5years + sales + last_evaluation, family = binomial,
    data = train.data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.2105  -0.6613  -0.4015  -0.1137   3.0861

Coefficients:
                       Estimate Std. Error z value Pr(>|z|)
(Intercept)          -1.5589519  0.2167582  -7.192 6.38e-13 ***
satisfaction_level   -4.1420325  0.1100679 -37.632  < 2e-16 ***
salarylow             2.0086379  0.1439724  13.952  < 2e-16 ***
salarymedium          1.4796288  0.1448026  10.218  < 2e-16 ***
Work_accident        -1.4968097  0.0990446 -15.112  < 2e-16 ***
time_spend_company    0.2616390  0.0174520  14.992  < 2e-16 ***
number_project       -0.3105897  0.0238781 -13.007  < 2e-16 ***
average_montly_hours  0.0045034  0.0005766   7.811 5.68e-15 ***
promotion_last_5years -1.5143858  0.2975207  -5.090 3.58e-07 ***
saleshr               0.2992939  0.1462292   2.047   0.0407 *
salesIT              -0.1663918  0.1372996  -1.212   0.2256
salesmanagement      -0.4186147  0.1778950  -2.353   0.0186 *
salesmarketing       -0.0071032  0.1484049  -0.048   0.9618
salesproduct_mng     -0.0796495  0.1469683  -0.542   0.5879
salesRandD           -0.5925952  0.1638949  -3.616   0.0003 ***
salessales           -0.0345029  0.1153554  -0.299   0.7649
salessupport          0.0392502  0.1226361   0.320   0.7489
salestechnical        0.0796572  0.1196290   0.666   0.5055

last_evaluation        0.7474836  0.1673503   4.467 7.95e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 13193  on 11998  degrees of freedom
Residual deviance: 10284  on 11980  degrees of freedom
AIC: 10322

Number of Fisher Scoring iterations: 5

>
```

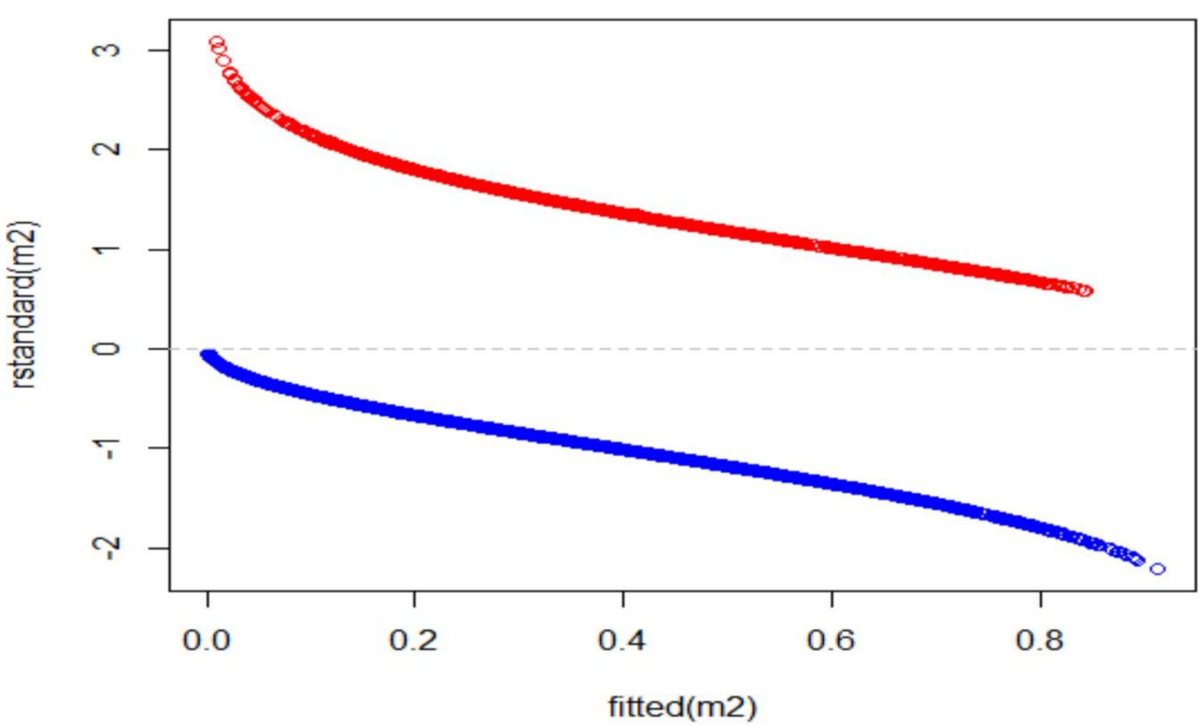
Checking multicollinearity

Multicollinearity problems arise when 2 x-variable are strongly correlated to each other. If such scenarios exist, we don't need to add both variables in the model. Hence we consider the more influence variable out of 2 and build the model again
We will use VIF to check for multicollinearity. If VIF >5 multicollinearity problem exists. Load library "car".

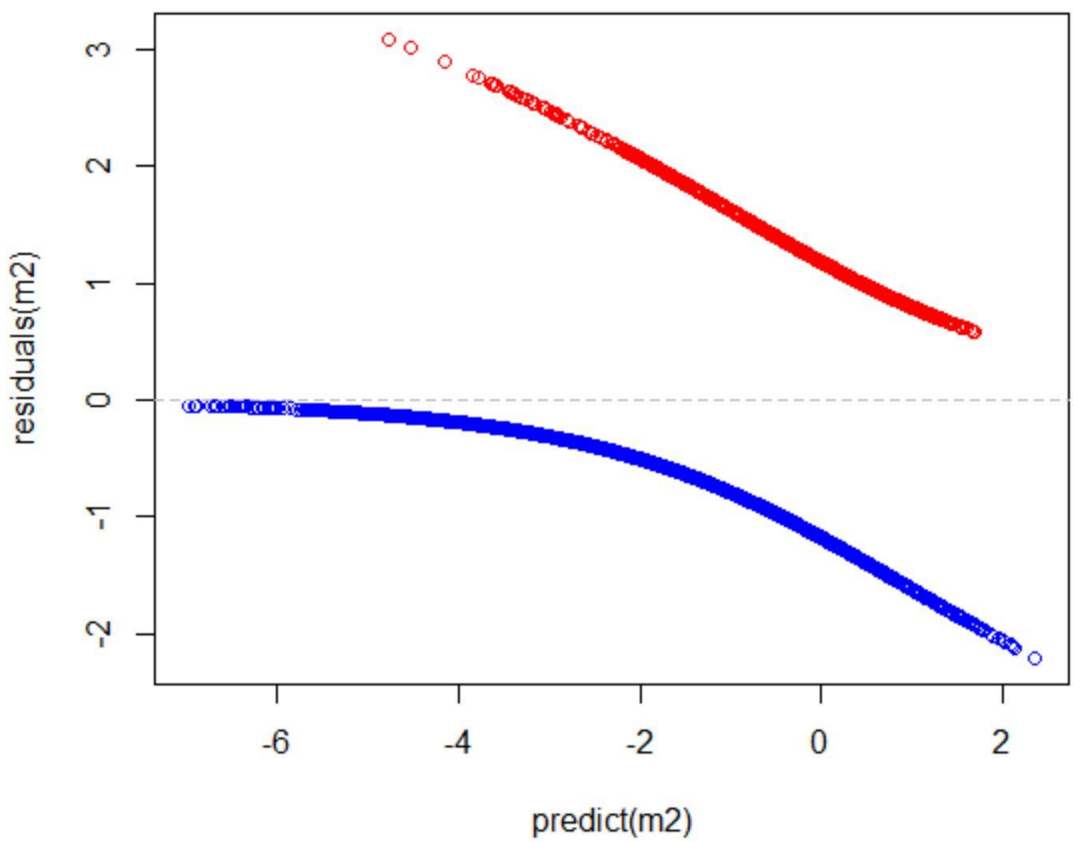
```
> library(car)

Attaching package: 'car'

The following object is masked from 'package:dplyr':

    recode

Warning message:
package 'car' was built under R version 3.3.3
> vif(m2)
                          GVIF Df GVIF^(1/(2*Df))
satisfaction_level    1.165232  1        1.079459
last_evaluation       1.457659  1        1.207336
number_project        1.792464  1        1.338829
average_montly_hours  1.525758  1        1.235216
time_spend_company    1.113813  1        1.055373
Work_accident         1.011139  1        1.005554
promotion_last_5years 1.017331  1        1.008628
salary                1.049204  2        1.012080
sales                 1.055526  9        1.003007
>
```

Residual analysis

```
>
> plot(fitted(m2),rstandard(m2),col=c("blue","red")[1+train.data$left])
> abline(h=0,lty=2,col="grey")
>
```

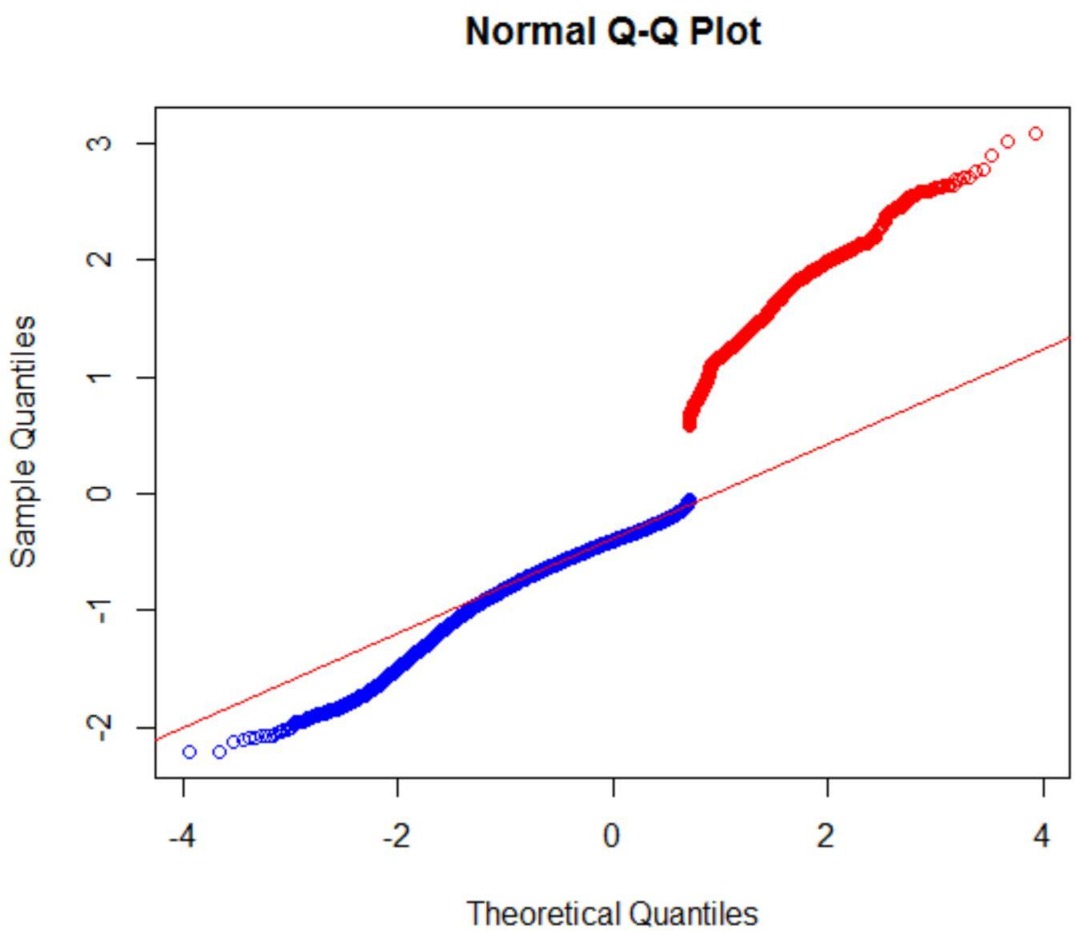


```
>
> plot(predict(m2),residuals(m2),col=c("blue","red")[1+train.data$left])
> abline(h=0,lty=2,col="grey")
>
```



The Residuals vs Fitted plots looks like there are problems with the model, but we know there aren't any. These plots, intended for linear models, are simply often misleading when used with a logistic regression model. It is more difficult to analyze than in linear regression.

```
>
> qqnorm(residuals(m2),col=c("blue","red")[1+train.data$left])
> qqline(residuals(m2),col=2)
>
```



The Normal Q-Q plot shows if residuals are normally distributed. But the residuals in logistic model don't have to be normally distributed for the model to be valid, so the normality / non-normality of the residuals doesn't necessarily tell us anything and can be ignored.

5. Evaluations and Results

5.1. Evaluation Methods

For model evaluation we split data into train and test data set initially

```
>
> HrData=read.table("HR_comma_sep.csv",header=T,sep=",")
> HrData=HrData[sample(nrow(HrData)),]
> select.data= sample (1:nrow(HrData), 0.8*nrow(HrData))
> train.data= HrData[select.data,]
> test.data= HrData[-select.data,]
>
```

Calculating classfication error/accuracy for both models with confusion matrix

Model 1: keeping cutoff value = 0.5

```
> pred1=predict(m1, test.data, type="response")
> model_pred_left=rep("0",3000)
> model_pred_left[pred1>0.5]="1"
> tab=table(model_pred_left,test.data$left)
> tab

model_pred_left    0    1
              0 2129  527
              1  166  178
> 1-sum(diag(tab))/sum(tab)
[1] 0.231
>
```

Model 1: keeping cutoff value = 0.4

```
>
> pred2=predict(m1, test.data, type="response")
> model_pred_left2=rep("0",3000)
> model_pred_left2[pred2>0.4]="1"
> tab2=table(model_pred_left2,test.data$left)
> tab2

model_pred_left2    0    1
               0 2039  363
               1  256  342
> 1-sum(diag(tab2))/sum(tab2)
[1] 0.2063333
>
```

Model 1: keeping cutoff value = 0.6

```
> pred3=predict(m1, test.data, type="response")
> model_pred_left3=rep("0",3000)
> model_pred_left3[pred3>0.6]="1"
> tab3=table(model_pred_left3,test.data$left)
> tab3

model_pred_left3    0    1
               0 2196  543
               1   99  162
> 1-sum(diag(tab3))/sum(tab3)
[1] 0.214
>
```

Model 2: keeping cutoff value = 0.5

```
> pred4=predict(m2, test.data, type="response")
> model_pred_left4=rep("0",3000)
> model_pred_left4[pred4>0.5]="1"
> tab4=table(model_pred_left4,test.data$left)
> tab4

model_pred_left4    0    1
               0 2137  447
               1  158  258
> 1-sum(diag(tab4))/sum(tab4)
[1] 0.2016667
>
```

Model 2: keeping cutoff value = 0.4

```
>
> pred5=predict(m2, test.data, type="response")
> model_pred_left5=rep("0",3000)
> model_pred_left5[pred5>0.4]="1"
> tab5=table(model_pred_left5,test.data$left)
> tab5

model_pred_left5    0    1
               0 2029  325
               1  266  380
> 1-sum(diag(tab5))/sum(tab5)
[1] 0.197
>
```

Model 2: keeping cutoff value = 0.6

```
> pred6=predict(m2, test.data, type="response")
> model_pred_left6=rep("0",3000)
> model_pred_left6[pred6>0.6]="1"
> tab6=table(model_pred_left6,test.data$left)
> tab6

model_pred_left6    0    1
               0 2198  555
               1   97  150
> 1-sum(diag(tab6))/sum(tab6)
[1] 0.2173333
>
```

Accuracy

Model 1 with cutoff value 0.4

```
> sum(diag(tab2))/sum(tab2)
[1] 0.7936667
>
```

Model 2 with cutoff value 0.4

```
>
> sum(diag(tab5))/sum(tab5)
[1] 0.803
>
```

Best model based on confusion matrix and accuracy

After trying different cutoff values for both the models we can see that Model 2 with cutoff value =0.4 has lowest classification error = 19.7% and highest accuracy with 80%
Hence according to confusion matrix and accuracy we consider model 2 with cutoff value = 0.4 as the best model

McFadden's R^2

In logistic regression, there is no concept of Adjusted R^2 hence we use McFadden's R^2 to determine best model
Calculating the same for both the models

```
>
> nullmodel=glm(left~1,family="binomial")
> 1-logLik(m1)/logLik(nullmodel)
'log Lik.' 0.3508656 (df=8)
> 1-logLik(m2)/logLik(nullmodel)
'log Lik.' 0.375413 (df=19)
>
```

McFadden's R^2 value for model 1: 35.08%
McFaddens's R^2 value for model 2: 37.54%

Higher the value of McFadden's R-square, better the model.

According to confusion matrix depending upon the accuracy and McFadden R^2, we can see that Model 2 is the best model and the model equation for the same is as follows:

Log(odd) = -1.5589519 – 4.1420325*satisfaction_level +0.7474836*last_evaluation - 0.3105897 *number_project + 0.0045034*average_montly_hours + 0.2616390*time_spend_company - 1.4968097*work_accident - 1.5143858*promotion_last_5years +2.0086379*salarylow + 1.4796288*salarymedium + 0.2992939*saleshr - 0.1663918*salesIT - 0.4186147*salesmanagement - 0.0071032*salesmarketing - 0.0796495*salesproduct_mng - 0.5925952*salesRandD - 0.0345029*salessales + 0.0392502*salesupport + 0.0796572*salestechnical

Where, values in red are : $\beta0$, $\beta1$, $\beta2$, $\beta3$, $\beta4$, $\beta5$…, $\beta19$

5.2. Results and Findings

Now that we have built our model, let us discuss some findings that can be made from the model.

Wald Test Hypothesis

Wald test helps in evaluation the significance of x-variable on p(odds).
In R Install package and load library “aod”.
Considering 2 scenarios for hypothesis:

1) Significance of sales departments on the odds of a person leaving.

H0 (Null Hypothesis): No sales department have significance on pr(Y).
Ha (Alternate Hypothesis): One or more department have significance on pr(Y).

```
> library(aod)
Warning message:
package 'aod' was built under R version 3.3.3
>

> wald.test(b=coef(m2), Sigma=vcov(m2), Terms=11:19)
Wald test:
----------

Chi-squared test:
X2 = 45.6, df = 9, P(> X2) = 7.1e-07
>
```

The chi-squared test statistics of 45.6 with 9 degree of freedom is associated with a p-value of 7.1e-07 which is less than 0.05. Which means that we can reject null hypothesis. This indicates that the overall effect of all departments on odds of left is significant.

```
>
> wald.test(b=coef(m2), Sigma=vcov(m2), Terms=9)
Wald test:
----------

Chi-squared test:
X2 = 194.6, df = 1, P(> X2) = 0.0
>
```

Similarly calculating for low salary, we get the chi-squared test statistics of 194.6 with 1 degree of freedom is associated with a p-value of 0.0 indicating that the overall effect of low salary on odds of left is significant.

Calculating confidence Interval

Confidence Interval gives us a range of factor or percentage where in which the value of a certain parameter must fall in.

```
>
> confint(m2)
Waiting for profiling to be done...
                             2.5 %       97.5 %
(Intercept)            -1.988467739 -1.138298347
satisfaction_level     -4.359175150 -3.927667673
last_evaluation         0.419876014  1.075963513
number_project         -0.357579068 -0.263966929
average_montly_hours    0.003375425  0.005635792
time_spend_company      0.227452533  0.295881039
Work_accident          -1.694681847 -1.306173945
promotion_last_5years -2.139829815 -0.965266795
salarylow               1.734060325  2.299262994
salarymedium            1.203221256  1.771689949
saleshr                 0.012980437  0.586410644
salesIT                -0.435032768  0.103385631
salesmanagement        -0.770348481 -0.072442634
salesmarketing         -0.298101186  0.283884816
salesproduct_mng       -0.367879386  0.208481110
salesRandD             -0.916089335 -0.273189235
salessales             -0.258951785  0.193440918
salessupport           -0.199775740  0.281145382
salestechnical         -0.153330621  0.315805224
>
```

```
>
> coef(m2)
          (Intercept)    satisfaction_level       last_evaluation        number_project  average_montly_hours
         -1.558951933          -4.142032529           0.747483622          -0.310589700           0.004503382
   time_spend_company         Work_accident promotion_last_5years             salarylow          salarymedium
          0.261639023          -1.496809674          -1.514385815           2.008637940           1.479628776
              saleshr               salesIT       salesmanagement        salesmarketing      salesproduct_mng
          0.299293886          -0.166391825          -0.418614678          -0.007103192          -0.079649494
           salesRandD            salessales          salessupport        salestechnical
         -0.592595234          -0.034502876           0.039250215           0.079657163
> exp(coef(m2))
          (Intercept)    satisfaction_level       last_evaluation        number_project  average_montly_hours
           0.21035642            0.01589052            2.11167954            0.73301457            1.00451354
   time_spend_company         Work_accident promotion_last_5years             salarylow          salarymedium
           1.29905753            0.22384315            0.21994323            7.45315878            4.39131522
              saleshr               salesIT       salesmanagement        salesmarketing      salesproduct_mng
           1.34890599            0.84671441            0.65795767            0.99292198            0.92343996
           salesRandD            salessales          salessupport        salestechnical
           0.55289054            0.96608556            1.04003068            1.08291574
>
```

The above values of coef(m2) and exp(coef(m2) give us the coefficients for all variables and their exponential. Now for logistic regression, comparing the exponential values is better. We have determined the confidence interval for salarylow and time_spend_company to show their impact on person leaving.

- Determining confidence interval for salarylow.

```
>
> ci=confint(m2,parm="salarylow")
Waiting for profiling to be done...
> ci
   2.5 %   97.5 %
1.734060 2.299263
> exp(ci)
   2.5 %   97.5 %
5.663603 9.966834
>
```

From the output it can be said the value of salarylow must lie within a 95% CI of(1.734060, 2.299263). Thus the corresponding 95% correspondence limits for the odds ratio are (exp(1.734060), exp(2.299263)) = (5.663603,9.966834). Thus, the odds of left/leaving increases between 46.6% to 89.6% for a person getting a low salary.

- Determining confidence interval for time spend in company

```
>
> ci2=confint(m2,parm="time_spend_company")
Waiting for profiling to be done...
> ci2
    2.5 %    97.5 %
0.2274525 0.2958810
> exp(ci2)
   2.5 %   97.5 %
1.255398 1.344310
>
```

From the output it can be said the value of time_spend_company must lie within a 95% CI of(0.2274525, 0.2958810). Thus the corresponding 95% correspondence limits for the odds ratio are (exp(0.2274525), exp(0.2958810)) = (1.255398,1.344310). Thus, the odds of left/leaving increases between 26% to 34% for a person spending more time in a company.

Interpreting odds

As we have our model, let us interpret the odds, to check for possibility of a person leaving or staying. For this we will use the equation of our best model and compare the values of left to check if the model is giving us output as expected or not.

Log(odd) = -1.5589519 – 4.1420325*satisfaction_level +0.7474836*last_evaluation - 0.3105897 *number_project + 0.0045034*average_montly_hours + 0.2616390*time_spend_company - 1.4968097*work_accident - 1.5143858*promotion_last_5years +2.0086379*salarylow + 1.4796288*salarymedium + 0.2992939*saleshr - 0.1663918*salesIT - 0.4186147*salesmanagement - 0.0071032*salesmarketing - 0.0796495*salesproduct_mng - 0.5925952*salesRandD - 0.0345029*salessales + 0.0392502*salesupport + 0.0796572*salestechnical+e

Let us consider p=Pr(Y=1) as probability of “left”.

We will set 0.5 as a threshold value and interpret the odds of left as:

- If odd>1 then pr(Y=1) **>** Pr(Y=0) -> Pr(Y=1) > 0.5
- If odd=1 then Pr(Y=1) **=** Pr(Y=0) -> Pr(Y=1)=0.5
- If odd<1 then p=pr(Y=1) **<** Pr(Y=0) -> Pr(Y=1) < 0.5

Calculating log(odd) for 1st row of test.data:

```
> head(test.data)
      satisfaction_level last_evaluation number_project average_montly_hours time_spend_company Work_accident left promotion_last_5years      sales salary
10425               0.75            0.62              5                  144                  3             0    0                     0  technical    low
14624               0.45            0.53              2                  138                  3             0    1                     0 accounting medium
14475               0.11            0.78              6                  260                  4             0    1                     0         hr medium
3210                0.93            0.66              4                  242                  4             0    0                     0    support    low
10240               0.95            0.81              5                  210                  4             0    0                     0      sales medium
6913                0.92            0.67              4                  241                  3             0    0                     0  technical   high
> 
```

Log(odd)= -1.559-4.142*0.75 +0.747*0.62 -0.311*5 +0.005*144 +0.262*3 -1.497*0 -1.514*0 + 2.009*1 +0.079*1 (Considering all other fields as 0 as salary= ”low” and sales=”technical”).

Log(odd) = -2.243

Hence odd= exp(-2.243)/1+exp(-2.243) = 0.2122792

```
>
> exp(-2.243)/1+exp(-2.243)
[1] 0.2122792
>
>
```

Therefore odd<1, Hence probability of not leaving is more than left. And as we see in test.data output, value of left=0.

Similarly, calculating and comparing for 3rd row:

Log(odd)= -1.559-4.142*0.11 +0.747*0.78 -0.311*6 +0.005*260 +0.262*4 -1.497*0 -1.514*0 +1.480*1+0.299*1 (Considering all other fields as 0 as salary=”medium” and sales=”hr”).

Log(odd)= 0.829

Hence odd= exp(0.829)/1+exp(0.829) = 4.582053

```
>
> exp(0.829)/1+exp(0.829)
[1] 4.582053
>
```

Therefore, odd>1, Hence probability of left is more than not left. And as we see in test.data output, value of left=1.

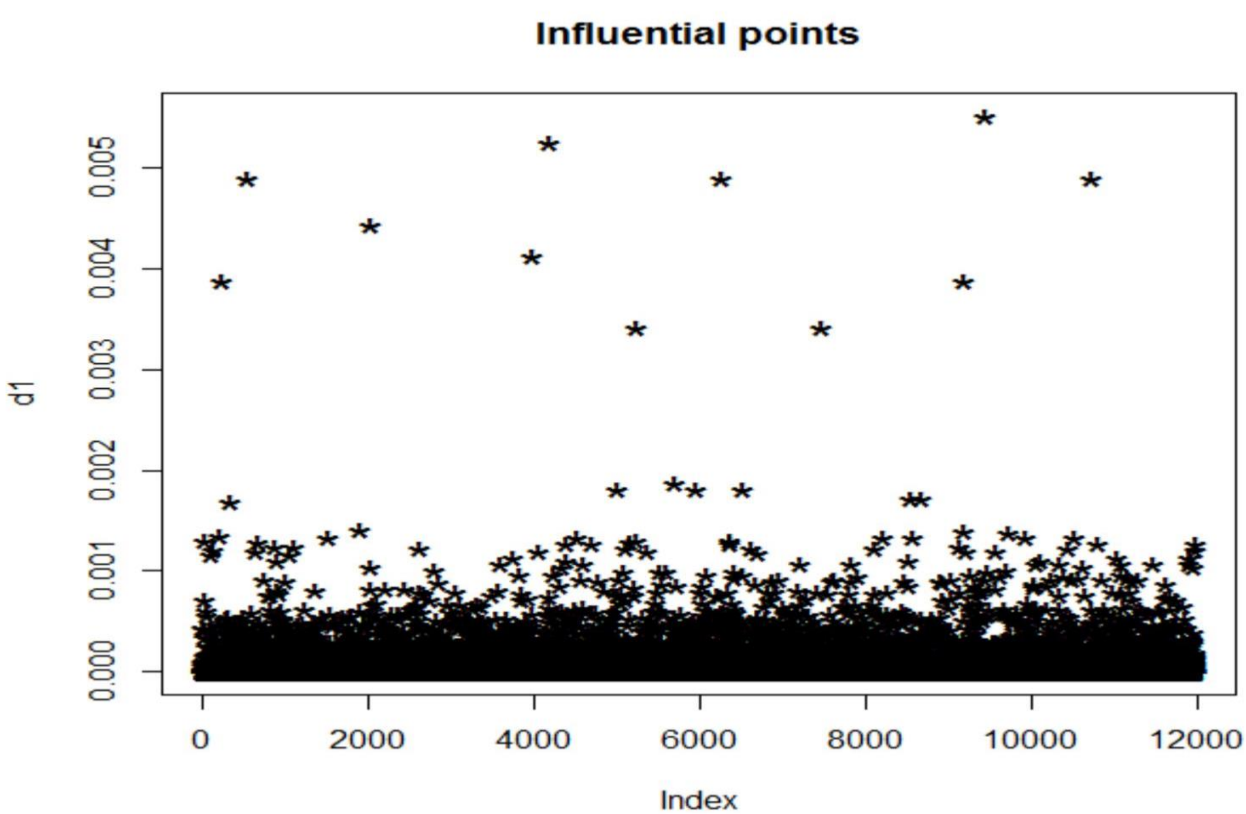
Determining the influential Points using cooks.distance

Influential points are outliers that have an influence on the model. Calculating them using cooks.distance:

```
>
> d1=cooks.distance(m2)
> a=cbind(train.data,d1)
> q=a[d1>4/11999,]
> head(q)
      satisfaction_level last_evaluation number_project average_montly_hours time_spend_company Work_accident left promotion_last_5years       sales salary
14503               0.87            0.84              5                  231                  5             0    1                     0          IT    low
1533                0.75            0.98              4                  245                  5             0    1                     0  management    low
1672                0.42            0.54              2                  143                  3             0    1                     0 product_mng   high
1374                0.37            0.53              2                  147                  3             0    1                     0       RandD    low
837                 0.43            0.56              2                  133                  3             0    1                     0       RandD    low
12068               0.90            0.98              4                  264                  6             0    1                     0 product_mng medium
                d1
14503 0.0004372810
1533  0.0007166975
1672  0.0013080791
1374  0.0003548076
837   0.0004343594
12068 0.0005733791
>
> nrow(q)
[1] 736
>
```

As seen above the number of outliers are 736. However, model is not getting impacted by them as from all the findings we can see the model is giving optimum result. The influential points can also be plotted as follows:

```
>
> plot(d1, pch="*", cex=2, main="Influential points")
>
>
```



Analysis

Field with positive coefficients		Field with negative coefficients	
Field Name	**Values**	**Field Name**	**Values**
Last_evaluation	0.747	Satisfaction_level	-4.142
Average_monthly_hours	0.005	Number_project	-0.310
Time_spend_company	0.261	Work_accident	-1.497
salarylow	2.009	Promotion_last_5years	-1.514
salarymedium	1.479	salesIT	-0.166
saleshr	0.299	salesmanagement	-0.419
salessupport	0.039	salesmarketing	-0.007
salestechnical	0.079	Salesproduct_mng	-0.079
		salesRandD	-0.593
		salessales	-0.035

- Considering the field with positive coefficients certain conclusions can be made like;

  1) People who have stayed in company for more years or spent more hours in company, lead to higher possibility of the person leaving.
  2) People getting low to medium salary are also at higher probability of leaving the company.
  3) It can also be seen that people in HR, support and technical department in sales have higher chance of leaving, probably because these jobs are similar in every company and it's easy to switch between companies.

- Considering the fields with negative coefficients some solutions to retain talent can be obtained. For Instance;

  1) If the satisfaction_level of the employee is higher, it's lesser probability for the employee to leave the company.
  2) If the person has more number_project, the person will be get more experience and learning and hence lesser chance of them leaving.
  3) If the person is from the management team in sales, there is less chance for person to leave as managers have high payment and better treatment in the company and it's difficult for other companies to match up the existing high salary.

6. Conclusions and Future Work

6.1. Conclusions

The model built considers 19 parameters in total that possibly have significant effect on the model. Some conclusions that can be made about the model and the outcomes are as follows.

1) People who have stayed in company for more years or spent more hours in company, lead to higher possibility of the person leaving. For one unit year more stayed by a person, there is a chance of 0.261 factor for the person to leave. This could be probably because the person has no more scope to learn in the company and decide on moving to a new company to learn more.
2) People getting low to medium salary are also at higher probability of leaving the company. For every person getting a low salary the probability of them leaving increases by a factor of 2.009, which is obvious because if they receive a better payment for the job they are doing from a different company, they will obviously leave.
3) It can also be seen that people in HR, support and technical department in sales have higher chance of leaving, probably because these jobs are similar in every company and it's easy to switch between companies.
4) It is essential for the company to keep a check on the satisfaction level of the employee. A quarterly report or survey can be taken to check for employee satisfaction and talents which show an inclination of poor satisfaction level can be given incentives or better work depending on their reason of having a poor satisfaction.
5) The company should see that it sees that deserved candidate get a promotion every 5 years. Promotions are like a trophy which tells the employees that their hard work is appreciated and recognized. This is essential in boosting their confidence and interest in the work they are doing leading them to stay back in the company.

6.2. Limitations

Every model needs some extrapolation to figure out reasons that might pop up in future. There could be some reasons which are out of scope of our model and cannot be taken into account while considering retaining talent. Some of these conditions could be:

1) If any employee has to leave the company because of personal commitment. Person reasons are something a company can never take into account as they come and go adhoc.
2) If any employee wanted to pursue higher education. Many fresher employees who have promising skillset might leave the company to pursue Masters or MBA for increasing their skillset or attain degree. These choices of employees is something a company cannot do anything about hence is out of scope from the model or company's hand.
3) Basically this means reasons(fields) other than the ones present in the model arise.

6.3. Potential Improvements or Future Work

1) Further analysis can be done to find out number of people leaving for each condition or combination of conditions. For instance, How many people left for low satisfaction level in the IT department or How many left the company who had work_accident and have high time_spend_company.