Table of Contents

[Introduction 1](#_Toc491363017)

[1.1 DATA ANALYSIS 1](#_Toc491363018)

[1.2 DATA SCIENCE 1](#_Toc491363019)

[1.3 DATA SCIENCE VS DATA ANALYSIS 1](#_Toc491363020)

[Tools & Technologies 3](#_Toc491363021)

[2.1 Technology 3](#_Toc491363022)

[2.2 Software 3](#_Toc491363023)

[2.3 Hardware 3](#_Toc491363024)

[2.4 R 3](#_Toc491363025)

[2.5 R Studio 3](#_Toc491363026)

[Modus Operandi of Data Science 5](#_Toc491363027)

[3.1 Data Acquisition 5](#_Toc491363028)

[3.2 Data Preparation 5](#_Toc491363029)

[3.3 EDA: Exploratory Data Analysis 7](#_Toc491363030)

[3.4 Modeling 16](#_Toc491363031)

[Analysis of HR Data 18](#_Toc491363032)

[4.1 Problem statement 18](#_Toc491363033)

[4.2 Data Acquisition 18](#_Toc491363034)

[4.3 Data Preparation 18](#_Toc491363035)

[4.4 EDA 18](#_Toc491363036)

[4.5 Modeling 32](#_Toc491363037)

[Conclusion 46](#_Toc491363038)

[5.1 Learning 46](#_Toc491363039)

[5.2 Conclusion 46](#_Toc491363040)

[Future Scope 48](#_Toc491363041)

[REFERENCES 49](#_Toc491363042)

**CHAPTER 1**

# Introduction

## 1.1 DATA ANALYSIS

Data analysis, is a process of inspecting, cleansing, transforming and modelling data. It is used to discover useful information, suggest conclusions, and support decision-making. It has numerous faces and approaches which results in variety of methods under diverse names, in variant domains such as science, social science and businesses.

In it’s statistical applications it can be divided into:

* Descriptive statistics
* Exploratory data analysis (discovering new features in the data)
* Confirmatory data analysis (confirming or falsifying existing hypotheses)[1]

## 1.2 DATA SCIENCE

It is an interdisciplinary field about scientific methods, processes and systems. It is used either on structured or unstructured data to get insights.

It unifies statistics, data analysis and the methods related to them with the aim to understand and analyze actual facts. The theories and techniques used in it are harassed from numerous fields of statistics, mathematics, computer science and information science.

When given a challenging question, it initiates with exploration, they try to understand the pattern or characteristics within the data by investigating leads.

In order  to get a level deeper with the intent to frame together a forensic view of what data is saying, quantitative techniques can be applied by data scientists. E.g. inferential models, segmentation analysis, time series forecasting, synthetic control experiments, etc.

Data scientist guides business stakeholders on how to takes decisions i.e. they act as consultants as this data-driven insight is central to providing strategic guidance.[2]

## 1.3 DATA SCIENCE VS DATA ANALYSIS

|  |  |
| --- | --- |
| **Data science** | **Data analytics** |
| It is a field that comprises of everything that is related to data cleansing, preparation and analysis. | It involves automating into a certain dataset and as well as supposes the usage of queries and data aggregation procedure. |
| Data science algorithms are used in industries such as internet searches, digital advertisements, search recommenders etc. | Data analytics is used in industries such as healthcare, gaming, energy management, travel etc. |
| Skills required:   * In depth knowledge in SAS and/or R. * Python coding * Hadoop platform * SQL database * Working with unstructured data | Skills required:   * Programming skills * Statistical skills * Mathematics * Machine learning skills * Data wrangling * Communication and data visualisation  skills * Data intitution |

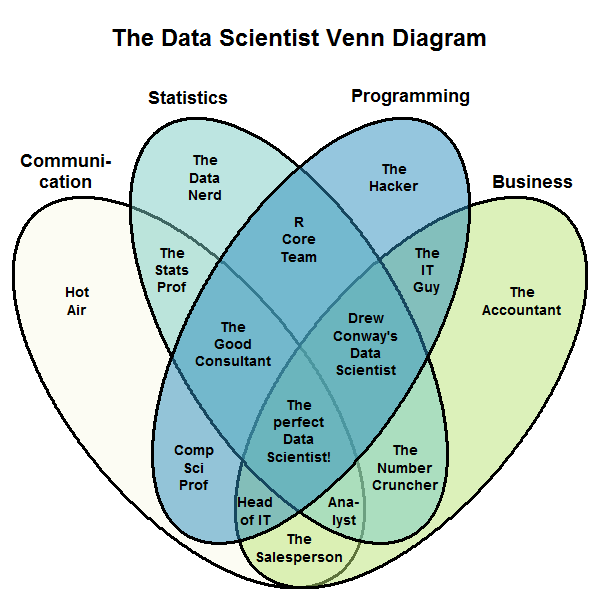


Fig 1: *Venn diagram showing skills required to be indulged in a specific role.*

**CHAPTER 2**

# Tools & Technologies

## 2.1 Technology

* R

## 2.2 Software

* R Studio

## 2.3 Hardware

* Intel(R) Core(TM) i7-7700HQ [CPU @ 2.80GHz](mailto:CPU@2.00GHz) 2.81 GHz
* 8 GB ram
* hdd 1 TB

## 2.4 R

* The R statistical programming language is a free open source package based on the S language developed by Bell Labs.
* It is an interpreted language.
* It is very powerful for writing programs.
* It has many statistical functions built in.
* It is free.[3]

## 2.5 R Studio

It is an IDE which allows the user to run R in more user  friendly manner. It was founded by JJ Allaire. IT is available in two editions: RStudio Desktop and RStudio Server. It is written in C++ and for graphical framework it uses Qt framework. Development of it was started at the end of 2010. It’s first beta version (v0.92) was released in February, 2011. On 1 November, 2016 Version 1.0 was released.

It has four components:

* File editor
* Console
* Workspace
* Help/Package

It aids users by providing features such as Code completion (it predicts the possible arguments, functions, braces etc.), Command history search(it  gives you the liberty to look for previous commands), Command history to R script/file, Function extraction from Rscript etc.[8]

**CHAPTER 3**

# Modus Operandi of Data Science

## 3.1 Data Acquisition

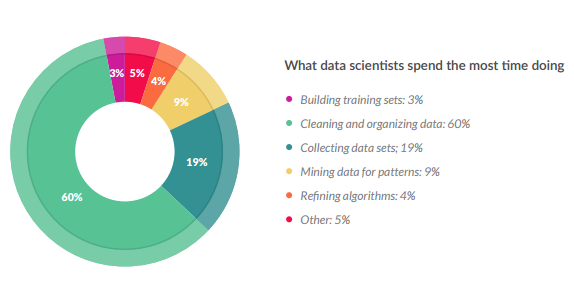
It is the first and the foremost step of any Data Science project. Usually people are not able to find the data at one place because more often than not it is distributed across the line of business. There are numerous ways to acquire data e.g. entering it manually in a spread sheet, downloading data files, streaming data on demand from online sources via APIs, generation of data by computer softwares.

## 3.2 Data Preparation

Data scientists roughly spend 50-80% of their time in gathering and cleaning data i.e trying to get it in the form on which they can perform desired operations upon. What makes it so important to devote such amount of time to this activity is the diverse nature of data that they encounter from their size to formats to ordering.

Data cleaning provides the direction to our research. It is extent to which we cleaned the data which decides how fruitful our algorithms will be. This quite clearly tells that cleaning the data is as significant to any firm as making good algorithms is. More often than not this turns out to be the point of differentiation between great enterprise data scientist and moderate enterprise data scientist.[4]

According to the survey which was performed on 80 data scientists San Francisco based crwodsourcing and data mining company ‘CrowdFlower’

****Fig2: Time distribution of Data Scientists

Even though it is the one of the least glamorous tasks but most of the time a data scientist is spent in data cleaning which we also call data munging and data wrangling. The amount of time needed to prepare data (clean it) is dependent on the fact that how healthy data is i.e how complete it is, how many inconsistencies it has, how many missing values are there and many such things.[5]

Data cleaning is essential because given data can have discrepancies in the names or codes, it may also have outliers or errors, to summarize the initial dataset is not qualitative rather it is quantitative.

Steps involved in data preparation are:

* **Data cleaning:**

This step includes dealing with missing and inconsistent values and thereby smoothing out noisy data. The missing values can either be treated by removing them or by filling them with appropriate values depending upon the situation. Noisy data is either is tackled manually or using various regression and clustering techniques.

* **Data integration:**

It involves things like data conflict resolving, handling of redundancies in data if there are any & schema integration

* **Data transformation:**

In this step noise is removed from the data if there is any  and it also does normalization, aggregation and generalization.

* **Data reduction:**

This step is done with the motive to reduce the size of data keeping but at the same time maintaining it’s utility i.e. making sure that it has the same statistical inference qualities as it used to have. Depending the requirement we can use either of the three approaches: dimensionality reduction, data cube aggregation and numerosity reduction.

* **Data discretization:**

This step is mainly used for the algorithms which only accept categorical attributes. This step helps data scientists to divide continuous data into intervals and thereby also assisting in reducing the size of the data in hand.

## 3.3 EDA: Exploratory Data Analysis

It is the approach to analyze data with the motive to summarize the main characteristics of the data and get vital insights. Mostly graphical methods like scatter plots, histograms, bar graphs, line plots etc. are used but it does have certain quantitative techniques as well. EDA’s main motive is to help us:

* Select the appropriate tool and technique that should be used.
* Design hypothesis tests.
* Evaluate the assumptions which will be the basis of our statistical inferences.
* Assess the requirement of data collection that has to be done in future in order to make proper conclusions or may even predict things.

**Example 1**

I tried to perform EDA on numerous datasets but to begin with I will try to elaborate the process with the ‘iris’ dataset which is available in ‘datasets’ library. This famous (Fisher's or Anderson's) iris data set gives the measurements in centimeters of the variables sepal length and width and petal length and width, respectively, for 50 flowers from each of 3 species of iris. The species are Iris setosa, versicolor, and virginica. iris is a data frame with 150 cases (rows) and 5 variables (columns) named Sepal.Length, Sepal.Width, Petal.Length, Petal.Width, and Species.[6]

**Report generated with R Markdown**

**analysis\_on\_iris.R**

Mrinal Jhamb

Wed Jun 21 13:17:12 2017

**library**('datasets')  
**data**()  
**data**(iris)  
**View**(iris)  
**head**(iris)

## Sepal.LengthSepal.WidthPetal.LengthPetal.Width Species  
## 1 5.1 3.5 1.4 0.2 setosa  
## 2 4.9 3.0 1.4 0.2 setosa  
## 3 4.7 3.2 1.3 0.2 setosa  
## 4 4.6 3.1 1.5 0.2 setosa  
## 5 5.0 3.6 1.4 0.2 setosa  
## 6 5.4 3.9 1.7 0.4 setosa

**summary**(iris)

## Sepal.LengthSepal.WidthPetal.LengthPetal.Width  
## Min. :4.300 Min. :2.000 Min. :1.000 Min. :0.100  
## 1st Qu.:5.100 1st Qu.:2.800 1st Qu.:1.600 1st Qu.:0.300   
## Median :5.800 Median :3.000 Median :4.350 Median :1.300   
## Mean :5.843 Mean :3.057 Mean :3.758 Mean :1.199   
## 3rd Qu.:6.400 3rd Qu.:3.300 3rd Qu.:5.100 3rd Qu.:1.800   
## Max. :7.900 Max. :4.400 Max. :6.900 Max. :2.500   
## Species   
## setosa :50   
## versicolor:50   
## virginica :50   
##   
##   
##

*# What are the names of different columns in IRIS data ?*  
**names**(iris)

## [1] "Sepal.Length" "Sepal.Width" "Petal.Length" "Petal.Width"   
## [5] "Species"

*#Identify the means of relevant columns in IRIS data ?*  
lst=iris[,1:4]  
**View**(lst)  
?sapply

## startinghttpd help server ...

## done

**sapply**(lst,mean)

## Sepal.LengthSepal.WidthPetal.LengthPetal.Width  
## 5.843333 3.057333 3.758000 1.199333

*#Identify the means and standard deviation (sd) of different columns for each specie separately.*  
specie1=iris[iris$Species=="setosa",1:4]  
**sapply**(specie1,mean)

## Sepal.LengthSepal.WidthPetal.LengthPetal.Width  
## 5.006 3.428 1.462 0.246

**sapply**(specie1,sd)

## Sepal.LengthSepal.WidthPetal.LengthPetal.Width  
## 0.3524897 0.3790644 0.1736640 0.1053856

specie2=iris[iris$Species=="virginica",1:4]  
**sapply**(specie2,mean)

## Sepal.LengthSepal.WidthPetal.LengthPetal.Width  
## 6.588 2.974 5.552 2.026

**sapply**(specie2,sd)

## Sepal.LengthSepal.WidthPetal.LengthPetal.Width  
## 0.6358796 0.3224966 0.5518947 0.2746501

specie3=iris[iris$Species=="versicolor",1:4]  
**sapply**(specie3,mean)

## Sepal.LengthSepal.WidthPetal.LengthPetal.Width  
## 5.936 2.770 4.260 1.326

**sapply**(specie3,sd)

## Sepal.LengthSepal.WidthPetal.LengthPetal.Width  
## 0.5161711 0.3137983 0.4699110 0.1977527

**summary**(specie1)

## Sepal.LengthSepal.WidthPetal.LengthPetal.Width  
## Min. :4.300 Min. :2.300 Min. :1.000 Min. :0.100  
## 1st Qu.:4.800 1st Qu.:3.200 1st Qu.:1.400 1st Qu.:0.200   
## Median :5.000 Median :3.400 Median :1.500 Median :0.200   
## Mean :5.006 Mean :3.428 Mean :1.462 Mean :0.246   
## 3rd Qu.:5.200 3rd Qu.:3.675 3rd Qu.:1.575 3rd Qu.:0.300   
## Max. :5.800 Max. :4.400 Max. :1.900 Max. :0.600

**summary**(specie2)

## Sepal.LengthSepal.WidthPetal.LengthPetal.Width  
## Min. :4.900 Min. :2.200 Min. :4.500 Min. :1.400   
## 1st Qu.:6.225 1st Qu.:2.800 1st Qu.:5.100 1st Qu.:1.800   
## Median :6.500 Median :3.000 Median :5.550 Median :2.000   
## Mean :6.588 Mean :2.974 Mean :5.552 Mean :2.026   
## 3rd Qu.:6.900 3rd Qu.:3.175 3rd Qu.:5.875 3rd Qu.:2.300   
## Max. :7.900 Max. :3.800 Max. :6.900 Max. :2.500

**summary**(specie3)

## Sepal.LengthSepal.WidthPetal.LengthPetal.Width  
## Min. :4.900 Min. :2.000 Min. :3.00 Min. :1.000   
## 1st Qu.:5.600 1st Qu.:2.525 1st Qu.:4.00 1st Qu.:1.200   
## Median :5.900 Median :2.800 Median :4.35 Median :1.300   
## Mean :5.936 Mean :2.770 Mean :4.26 Mean :1.326   
## 3rd Qu.:6.300 3rd Qu.:3.000 3rd Qu.:4.60 3rd Qu.:1.500   
## Max. :7.000 Max. :3.400 Max. :5.10 Max. :1.800

**Conclusions drawn**

* On an avg. mean size of petals is the greatest in virginica and it is the least in setosa.
* Size of the petals are most diverse in virginica.
* On an avg. sepals of setosa are expected to be the most circular or the least elliptical(because of the least difference between the mean length and width of setosa).
* Versicolor has the sepal with the least width whereas width is the largest in setosa.
* Setosa has the sepal with the least length whereas width is the least in versicolor.
* Virginica has the petal with the largest width and it also has the one with the largest length.
* Setosa has the petal with the least width and it also has the one with the least length.
* From the exploratory data analysis we found that the shape &size of flowers are different.

**Example 2**

Second EDA is on the ‘women’ dataset which is available in ‘datasets’ library. This data set gives the average heights and weights for American women aged 30–39. Data has been taken from the ‘American Society of Actuaries Build and Blood Pressure Study’. This is a data frame with 15 observations on 2 variables: height (in) & weight (lbs).[7]

**Report generated with R Markdown**

**analysis\_on\_women.R**

Mrinal Jhamb

Wed Jun 21 13:16:13 2017

**library**(datasets)  
**data**()  
**data**(women)  
**View**(women)  
**sapply**(women,mean)

## height weight   
## 65.0000 136.7333

**sapply**(women,min)

## height weight   
## 58 115

**sapply**(women,max)

## height weight   
## 72 164

**sapply**(women,sd)

## height weight   
## 4.472136 15.498694

**sapply**(women,median)

## height weight   
## 65 135

136.7333-15.48694

## [1] 121.2464

136.7333+15.48694

## [1] 152.2202

65+4.472136

## [1] 69.47214

65-4.472136

## [1] 60.52786

lst1=women[women$height>"69.47214",]  
**length**(lst1$height)

## [1] 3

lst2=women[women$height<"60.52786",]  
**length**(lst2$height)

## [1] 3

lst3=women[women$height>"60.52786"&women$height<"69.47214",]  
**length**(lst3$height)

## [1] 9

lst4=women[women$wieght>"152.2202",]  
**length**(lst1$height)

## [1] 3

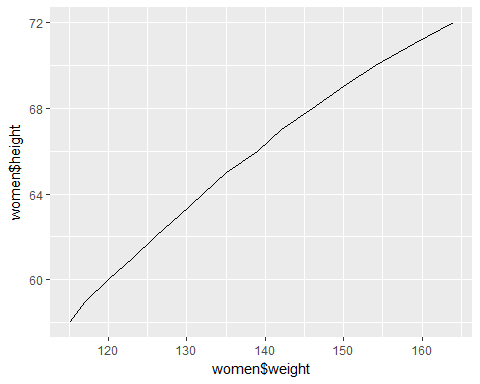
lst5=women[women$weight<"121.2464",]  
**length**(lst2$height)

## [1] 3

lst6=women[women$weight>"121.2464"&women$height<"152.2202",]  
**length**(lst3$height)

## [1] 9

**library**(ggplot2)  
**qplot**(women$weight,women$height,data=women,geom="line")



bmi=**data.frame**(703\*women[[2]]/(women[[1]]\*women[[1]]))  
**colnames**(bmi)=**c**("bmi")  
**View**(bmi)  
**length**(bmi[bmi<18.5])

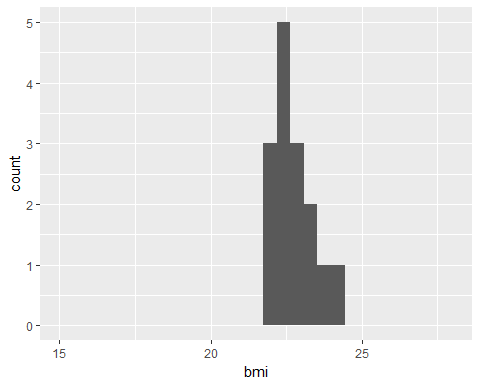
## [1] 0

**length**(bmi[bmi>18.5&bmi<24.9])

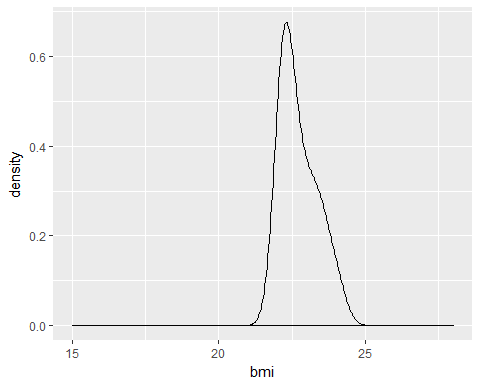
## [1] 15

**qplot**(bmi,data=bmi,geom="histogram",bins=30,xlim=**c**(15,28))

## Warning: Removed 1 rows containing missing values (geom\_bar).



**qplot**(bmi,data=bmi,geom="density",xlim=**c**(15,28))



**Conclusions drawn**

* On the basis of standard deviation the women can be categorised on the basis of their weight and height.
* Categorisation:
  + Weight:
    - Women with weight between 121.2464 & 152.2202: Normal Count:9
    - Women with weight less than 121.2464: Under Weight Count:3
    - Women with weight greater than 152.2202: Over Weight Count:3
  + Height:
    - Women with height between 60.52786 & 69.47214: Normal Heighted Count:9
    - Women with height less than 60.52786: Short Heighted Count:3
    - Women with height greater than 69.47214: Tall Heighted Count:3
* The trend quite clearly leads us to conclusion that women who are 'Tall Heighted' also lie in the category of 'Over Weight' & those who are 'Short Heighted' lie in the 'Under Weight'.
* So by looking at the trend over judgement of people being 'Over Weight' & 'Under Weight' is wrong. Since height also has the role to play. That's why we have BMI.
* BMI:
  + underweight (BMI less than 18.5) count:0
  + normal weight (BMI between 18.5 & 24.9) count:15
  + overweight (BMI between 25.0 & 29.9) count:0
  + obese (BMI 30.0 and above) count:0

These turns out to be the excellent example of how EDA can extremely useful to add new dimensions to our analysis. In this case it did assist us with the fact that only by looking at the weight one can’t judge someone to be underweight or overweight as height also has the role to play which got to know by looking at the line plot between the height and the weight.

**Example 3**

Third EDA is on the ‘chickweight’ dataset which is available in ‘datasets’ library. The ChickWeight data frame has 578 rows and 4 columns from an experiment on the effect of diet on early growth of chicks. The body weights of the chicks were measured at birth and every second day thereafter until day 20. They were also measured on day 21. There were four groups on chicks on different protein diets.[8]

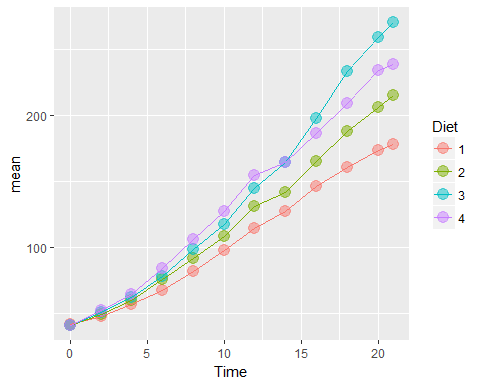
**Report generated with R Markdown**

**analysis\_on\_chick.R**

Mrinal Jhamb

Wed Jun 21 13:12:20 2017

**library**(datasets)  
**data**("ChickWeight")  
**View**(ChickWeight)  
  
cw=ChickWeight  
count=**data.frame**(**table**(cw$Chick))  
**View**(count)  
defective=count[**which**(count$Freq<12),]  
**View**(defective)  
clean=cw[-**which**(cw$Chick %in%defective$Var1),]  
**View**(clean)  
**library**(plyr)  
x=**ddply**(clean[,**c**(1,2,4)],**c**('Time','Diet'),function(df) **mean**(df$weight))  
**colnames**(x)[3]='mean'  
**View**(x)  
**library**(ggplot2)  
p=**ggplot**(x, **aes**(Time,mean))  
p+**geom\_point**(**aes**(color=Diet),size=4,alpha=1/2)+**geom\_line**(**aes**(color=Diet))



**Conclusions Drawn**

* For initial growth Diet 4 is the best but after a particualr point Diet 3 is the best for fast growth. Diet 1 is the worst one whent its come to growth.

## 3.4 Modeling

In machine learning we make computers learn from experiences. For this purpose we training data to train our models which is called modelling and then use the learning to estimate or classify results. There are various models which can be used depending upon the problem. Few of them are mentioned below:

* **Regression**: Process to find relationship between dependent and independent variable with the aim to predict values for dependent variables in future is known as regression. In linear regression we find a straight line that best fits the given scatter plot. In it to avoid the difficulty of dealing negative distances when points are on the both sides of the line we prefer to use square of distances of points from the line to find error. So, this method is also known as 'least square'. The process used too make the line fit the best is known as gradient descent.
* **Naive Bayes Algo**: It uses the probability of occurrence of an attribute while considering each attribute to be independent to predict future events e.g. detecting spam emails can take into account the attributes such as spelling mistakes, no title & use of word 'cheap'. Then their combination can be used to predict if the recent mail is spam or not.
* **Decision tree**: Each node of the tree reduces the possible options by looking at the trend followed in historic data. Hence, decision tree. E.g. recommendation of clothes on e-commerce site can be made on the basis of the info of the current user and this can pretty well be aided by the 'decision tree'.
* **Logistic regression**: It is used to separate kinds of data by using linear decision boundary. Number of errors in it can be reduced by gradient descent. But here the focus is on the reduction of log loss function which gives higher values to misclassified points as compared to that of the adequately classified ones. This is further explained in depth in next chapter.
* **Neural networks**: It is combination of lines of regression which are used to specify the section on the graph to split and classify the mappings on the scatter plots.
* **Support Vector Machines**: Out of the two possible regression lines the best is the one which is optimistically away from boundary point of each section. For this now we don't use 'Gradient Descent' rather we use 'linear optimization'. And on the whole this method is called 'Support Vector Machines'. Kernel Trick is used in Support Vector Machines when line is not enough to split. We either use different geometric curves or we can either use different planes for different sets (both of them are same).

**CHAPTER 4**

# Analysis of HR Data

## 4.1 Problem statement

A company (undisclosed) has been in industry since a long time. Their business had been increasing quite well over past, however in recent years, there has been a slowdown in terms of growth because their best and most experienced employees leaving prematurely. The VP of the firm is not very happy with the company’s best and most experienced employees leaving prematurely. The VP of the firm has employed you to find out insights in the company employee data and find out an answer as to know why best and most experienced employees are leaving prematurely.

## 4.2 Data Acquisition

Data was given in the spreadsheet format.

## 4.3 Data Preparation

Data was already in the useable format and it did not require cleaning.

## 4.4 EDA

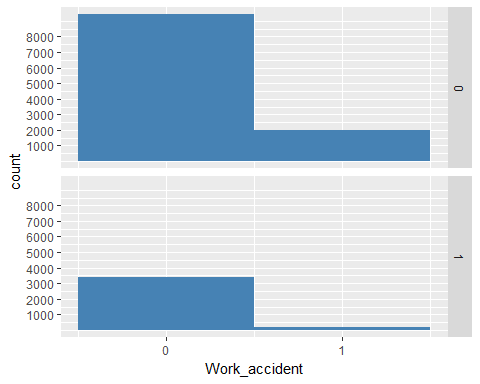
It is the approach to analyze data with the motive to summarize the main characteristics of the data and get vital insights. Mostly graphical methods like scatter plots, histograms, bar graphs, line plots etc. are used but it does have certain quantitative techniques as well. Below is the report knitted by using R Markdown:

**eda.R**

Mrinal

Mon Jul 24 19:23:17 2017

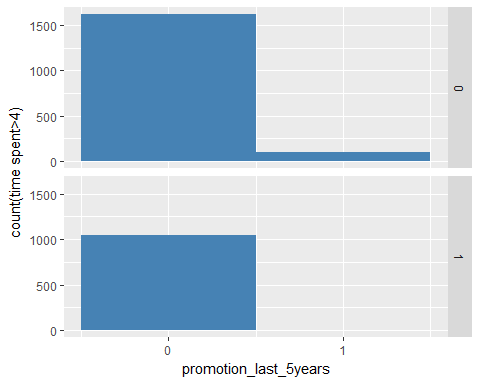
HR\_Data=**read.csv**("E:\\R\_Lectures\\dr. mohit\\PROJECT\\HR\_Data.csv")  
  
**library**(ggplot2)  
  
*#Impact of work accident on people leaving their jobs.*  
**ggplot**(HR\_Data,**aes**(Work\_accident))+**geom\_histogram**(fill="steelblue",binwidth = 1)+**facet\_grid**(left~.)+**scale\_x\_continuous**(breaks = **c**(0,1))+**scale\_y\_continuous**(breaks = **c**(1000,2000,3000,4000,5000,6000,7000,8000))



*#From the second histogram we can quite clearly see that majority of people leaving didn't have a work accident. To take a decisive call of its impact on leaving we will find out the proportion of people who had work accident in subcategories made on the basis of whether they have left or not.*  
*#Proportion of people who had work accident in subcategories made on the basis of whether they have left or not.*  
**table**(HR\_Data$Work\_accident,HR\_Data$left,dnn=**c**("Work\_accident","left"))

## left  
## Work\_accident 0 1  
## 0 9428 3402  
## 1 2000 169

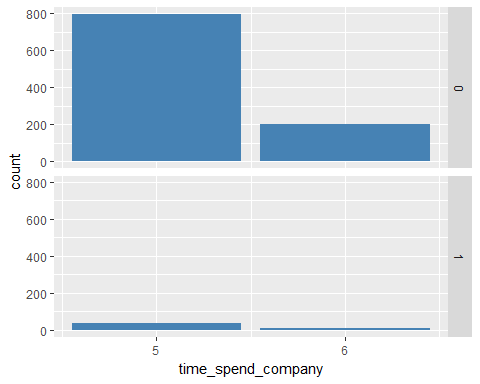
*#From this table we can quite clearly see that proportion of people having work accident is significantly low in people who have left as compared to those who haven't. So it is quite clear that it is not as considerable a factor.*  
  
*#Impact of not getting promotion on people leaving their jobs. Now this has to be for those who have worked above significant time and according to the data provided this is maintained for whether someone has been promoted in last five yaers or not.*  
**ggplot**(HR\_Data[**which**(HR\_Data$time\_spend\_company>=5),],**aes**(promotion\_last\_5years))+**geom\_histogram**(fill="steelblue",binwidth = 1)+**facet\_grid**(left~.)+**scale\_x\_continuous**(breaks = **c**(0,1))+**ylab**("count(time spent>4)")



*#From the second hisogram it is quite clear that almost all of the senior employes who have left are those who weren't promoted in last 5 years.So now we will try to look at the proportion of people who were promoted in the subcategories made on the basis of whether they have left or not.*  
*#proportion of people who were promoted in the subcategories made on the basis of whether they have left or not*  
data=HR\_Data[**which**(HR\_Data$time\_spend\_company>4),]  
**table**(data$promotion\_last\_5years,data$left,dnn=**c**("promotion","left"))

## left  
## promotion 0 1  
## 0 1618 1041  
## 1 95 1

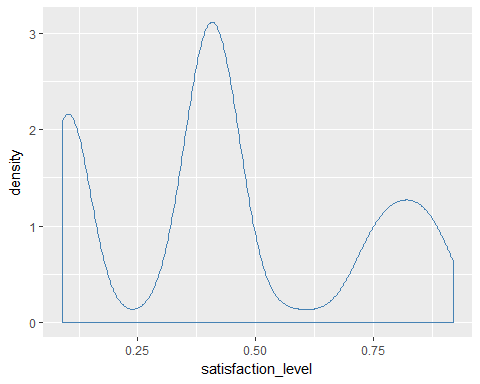
*#We can quite clearly see that proportion of people who have been promoted is significantly low in people who have left as compared to who haven't. This shows that there is quite a possibility of it being a significant factor.*  
  
*#Combined impact of getting promotion and time spent in the company for those who have worked for more than 4 years.*  
**ggplot**(HR\_Data[**which**(HR\_Data$left==1 & HR\_Data$time\_spend\_company>=5), ],**aes**(time\_spend\_company))+**geom\_bar**(fill="steelblue")+**facet\_grid**(Work\_accident~.)+**scale\_x\_continuous**(breaks = **c**(5,6,7,8,9,10))



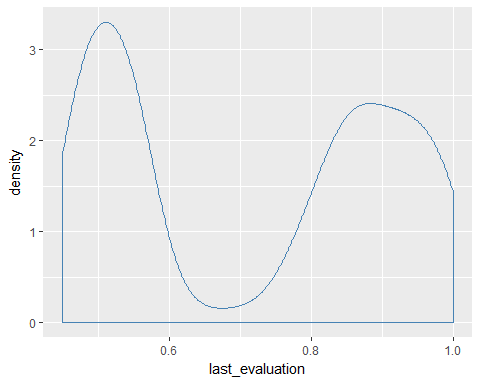
*#We can see that the count of people who were not promoted and have left is less for those who have spent 7 years in company as compared to those who have spent 6 years and there after that is for those who have spent more than 7 years this count is zero. But to be more certain about this we will find out the proportion of people left without promotion for the categorise made on the basis of their time spent in the company.*  
*#proportion of people left without promotion for the categories made on the basis of their time spent in the company.*  
data=HR\_Data[**which**(HR\_Data$left==1 & HR\_Data$time\_spend\_company>=5), ]  
**table**(data$promotion\_last\_5years, data$time\_spend\_compan,dnn=**c**("promotion","time\_spent"))

## time\_spent  
## promotion 5 6  
## 0 832 209  
## 1 1 0

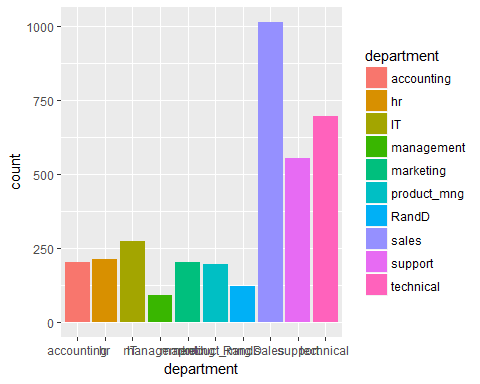
*#we can pretty clearly see that almost all of the people who have have left after working 5 years or more are those which have not been promoted except one but still ther are people who have not been promoted and still working without promotion. So there combined effect can be significant while modelling.*  
  
*#Seprate density plots of people who have left and those who haven't left on the basis of their respective satisfaction level.*  
**ggplot**(HR\_Data[**which**(HR\_Data$left==1),],**aes**(satisfaction\_level))+**geom\_density**(color="steelblue")



*#Seprate density plots of people who have left and those who haven't left on the basis of how they have been evaluated last time.*  
**ggplot**(HR\_Data[**which**(HR\_Data$left==1),],**aes**(last\_evaluation))+**geom\_density**(color="steelblue")



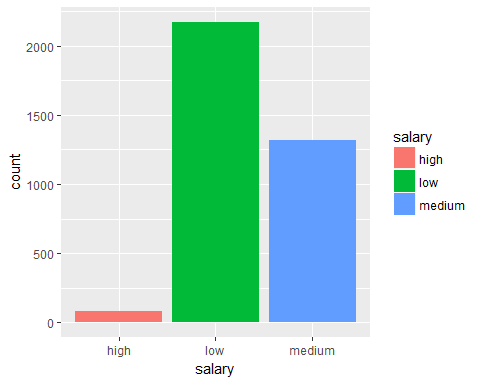
*#Count of people leaving of different departments.*  
**ggplot**(HR\_Data[**which**(HR\_Data$left==1),],**aes**(department,fill=department))+**geom\_bar**()



*#By this we can quite clearly see that amount of people leaving from sales department is max but we also have to look at the proportion of people leaving from each department.*  
*#proportion of people leaving.*  
**table**(HR\_Data$department,HR\_Data$left,dnn=**c**('department','left'))

## left  
## department 0 1  
## accounting 563 204  
## hr 524 215  
## IT 954 273  
## management 539 91  
## marketing 655 203  
## product\_mng 704 198  
## RandD 666 121  
## sales 3126 1014  
## support 1674 555  
## technical 2023 697

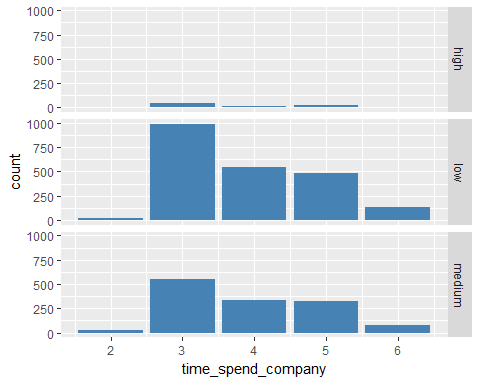
*#This pretty well tells that department has the role to play as we can see that ratio of people left from each deparment to the people who didn't is varying.*   
  
*#count of people leaving with high, medium and low salary.*  
**ggplot**(HR\_Data[**which**(HR\_Data$left==1),],**aes**(salary,fill=salary))+**geom\_bar**()



*#Clearly the salary category has the role to play since count of people leaving is highest in low salary category and is lowest in high salary category.*  
*#proportion of people leaving from each salary category.*  
**table**(HR\_Data$salary,HR\_Data$left,dnn=**c**('salary','left'))

## left  
## salary 0 1  
## high 1155 82  
## low 5144 2172  
## medium 5129 1317

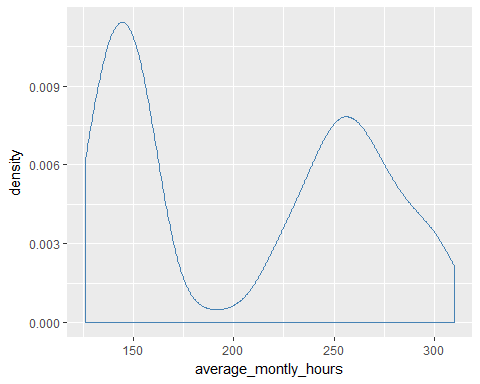
*#Table pretty clearly tells that salary is significant as the proportion of people who have left with low salary is the hisghest and those left with the high salary is the lowest.*  
  
*#Impact of time spent in company and salary on the decision of people leaving the job.*  
**ggplot**(HR\_Data[**which**(HR\_Data$left==1),],**aes**(time\_spend\_company))+**geom\_bar**(fill="steelblue")+**facet\_grid**(salary~.)+**scale\_x\_continuous**(breaks = **c**(2,3,4,5,6,7,8,9,10))



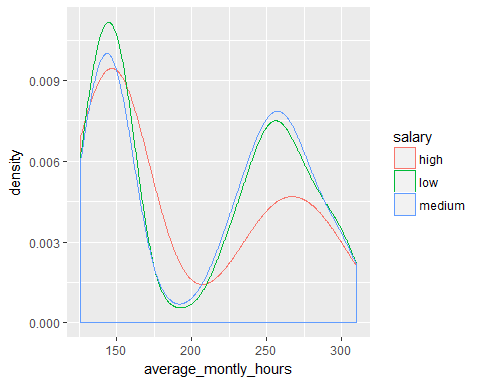
*#proportion of people leaving those who have spent the same amount of time in the company.*  
**table**(HR\_Data$salary,HR\_Data$left,HR\_Data$time\_spend\_company, dnn=**c**('salary','left','time spent'))

## , , time spent = 2  
##   
## left  
## salary 0 1  
## high 303 0  
## low 1505 22  
## medium 1383 31  
##   
## , , time spent = 3  
##   
## left  
## salary 0 1  
## high 474 46  
## low 2219 986  
## medium 2164 554  
##   
## , , time spent = 4  
##   
## left  
## salary 0 1  
## high 157 16  
## low 759 541  
## medium 751 333  
##   
## , , time spent = 5  
##   
## left  
## salary 0 1  
## high 46 20  
## low 311 488  
## medium 283 325  
##   
## , , time spent = 6  
##   
## left  
## salary 0 1  
## high 55 0  
## low 198 135  
## medium 256 74  
##   
## , , time spent = 7  
##   
## left  
## salary 0 1  
## high 38 0  
## low 36 0  
## medium 114 0  
##   
## , , time spent = 8  
##   
## left  
## salary 0 1  
## high 18 0  
## low 60 0  
## medium 84 0  
##   
## , , time spent = 10  
##   
## left  
## salary 0 1  
## high 64 0  
## low 56 0  
## medium 94 0

*#These two factors have the combined effect as we can quite clearly see that the ratio of people leaving to that of who have stayed for almost all categories of salary is increasing with inccrease in the number of years spend in the company and there after the count of people leaving is 0 as no one has left the company after working for 6 years.*  
  
*#Density plot of people leaving on the basis of their average monthly working hours.*  
**ggplot**(HR\_Data[**which**(HR\_Data$left==1),],**aes**(average\_montly\_hours))+**geom\_density**(color="steelblue")



*#Seprate density plots of people of different salary category leaving on the basis of their average monthly working hours.*  
**ggplot**(HR\_Data[**which**(HR\_Data$left==1),],**aes**(average\_montly\_hours,color=salary))+**geom\_density**()



*#The count of people leaving in each salary category follows the similar trend so we will check the proportion of people leaving in each of the subcategories made on the basis of average monthly hours:*  
 *#average\_monthly\_hours<175,175<average\_monthly\_hours<225,225<average\_monthly\_hours*  
*#proportion of people leaving in each of the subcategories.*  
data1=HR\_Data[**which**(HR\_Data$average\_montly\_hours<=175),]  
**table**(data1$left)

##   
## 0 1   
## 3962 1614

data2=HR\_Data[**which**(HR\_Data$average\_montly\_hours>175 & HR\_Data$average\_montly\_hours>225),]  
**table**(data2$left)

##   
## 0 1   
## 3735 1767

data3=HR\_Data[**which**(HR\_Data$average\_montly\_hours>=225),]  
**table**(data3$left)

##   
## 0 1   
## 3817 1780

*#We can roughly say that the proportion of people leaving is continuously decreasing with increse in average monthly working hours. So this can be significant factor in modelling.*  
*#combined effect of salary and average monthly working hours.*  
data1=HR\_Data[**which**(HR\_Data$average\_montly\_hours<=175),]  
**table**(data1$left,data1$salary)

##   
## high low medium  
## 0 385 1784 1793  
## 1 50 1003 561

data2=HR\_Data[**which**(HR\_Data$average\_montly\_hours>175 & HR\_Data$average\_montly\_hours>225),]  
**table**(data2$left,data2$salary)

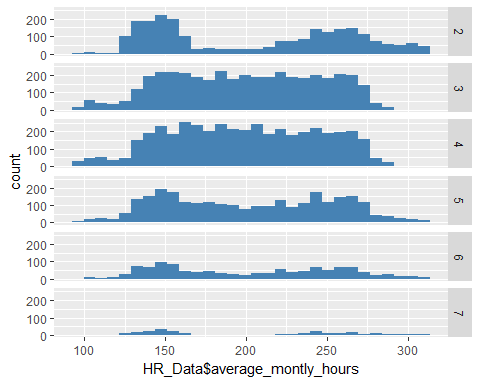
##   
## high low medium  
## 0 393 1632 1710  
## 1 31 1058 678

data3=HR\_Data[**which**(HR\_Data$average\_montly\_hours>=225),]  
**table**(data3$left,data3$salary)

##   
## high low medium  
## 0 406 1668 1743  
## 1 31 1069 680

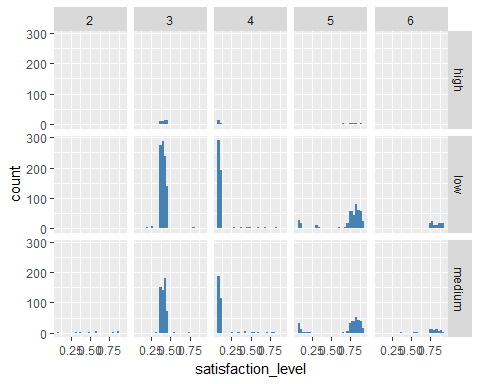
*#We can quite clearly see that proportion of people leaving in each category of average monthly working hours is varying in different salary categories. So these can have the combined effect in modelling.*  
  
*#Seprate histogrmas for people with different number of projects on the basis of their average monthly working hours.*  
**ggplot**(HR\_Data,**aes**(HR\_Data$average\_montly\_hours))+**geom\_histogram**(fill="steelblue")+**facet\_grid**(number\_project~.)

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



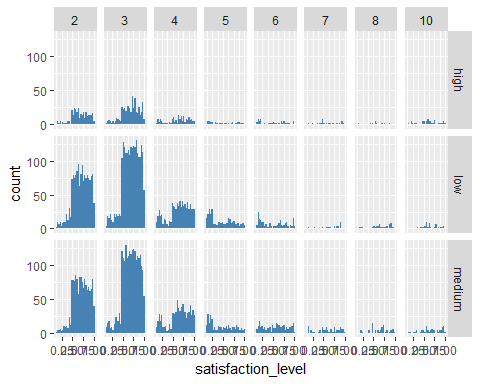
*#Histograms quite clearly show that there is no corelation between average monthly working hours and number of projects.*  
  
*#Plotting seprate histograms on the basis of satisfaction level for each salary category, time spent in the company & whtehter they have left or not.*  
**ggplot**(HR\_Data[**which**(HR\_Data$left==1),],**aes**(satisfaction\_level))+**geom\_histogram**(fill="steelblue")+**facet\_grid**(salary~time\_spend\_company)

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



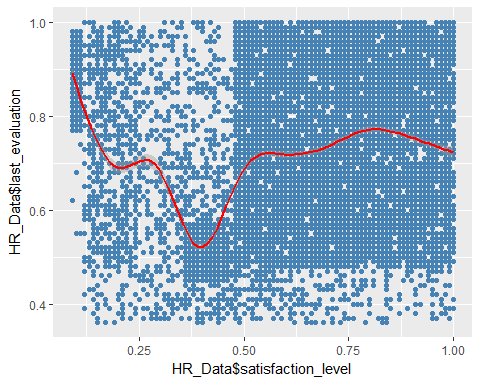
**ggplot**(HR\_Data[**which**(HR\_Data$left==0),],**aes**(satisfaction\_level))+**geom\_histogram**(fill="steelblue")+**facet\_grid**(salary~time\_spend\_company)

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



*#We can quite clearly see that initially for first 2 years satisfacation level is not that significant in each of the salary category. But for those who have spend 3 & 4 years we can quite clearly see that count of poeple leving with satisfaction level less than 0.5 is more than those who have left this difference in count is comparitively less in high salary category. And for those who have spent 5 & 6 years count of people leaving with high satisfaction level is more in medium and low salary category as compared to those who have stayed. No one has left the company after working more than 6 years and in high salary category no one has left after working more than 5 years. This shows that there can be interaction between these three terms which can be significant while modelling.*   
  
*#Scatter plot between last evaluation and satisfaction level.*  
**ggplot**(HR\_Data,**aes**(HR\_Data$satisfaction\_level,HR\_Data$last\_evaluation))+**geom\_point**(color="steelblue")+**geom\_smooth**(color="red")

## `geom\_smooth()` using method = 'gam'



*#Plot shows no clear cut trend.Hencce, no corelation between the used two attributes.*

We here tried to figure out the impact of:

* Work\_accident
* promotion\_last\_5years
* interaction of promotion\_last\_5years & time\_spend\_company
* satisfaction\_level
* last\_evaluation
* department
* salary
* interaction of salary & time\_spend\_company
* average\_monthly\_hours
* interaction of salary & average\_monthly\_hours
* interaction of time\_spend\_company & average\_monthly\_hours
* interaction of salary & time\_spend\_company &satisfaction\_level

We even looked for the correlation between satisfaction\_level & last\_evaluation.

## 4.5 Modeling

**modelling\_HR\_Data.R**

Mrinal

Mon Jul 24 18:22:27 2017

HR\_Data=**read.csv**("E:\\R\_Lectures\\dr. mohit\\PROJECT\\HR\_Data.csv")  
**View**(HR\_Data)  
?ChickWeight

## starting httpd help server ...

## done

#################Creation of training and testing dataset. So that we can test the created model on both the sets to look at the training and testing accuracy in order to judge that how effective over modelled prediction structure will be on unseen data.  
l=**sample**(1:**length**(HR\_Data$left),**floor**(0.7\***length**(HR\_Data$left)))  
train\_data=HR\_Data[l,]  
test\_data=HR\_Data[-l,]  
  
  
###################Modelling on train data.  
lm\_train=**glm**(left~.,data=train\_data,family=binomial)  
**summary**(lm\_train)

##   
## Call:  
## glm(formula = left ~ ., family = binomial, data = train\_data)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.2120 -0.6696 -0.4085 -0.1215 3.0187   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.6842351 0.2290539 -7.353 1.94e-13 \*\*\*  
## satisfaction\_level -4.0174466 0.1159510 -34.648 < 2e-16 \*\*\*  
## last\_evaluation 0.8406383 0.1777654 4.729 2.26e-06 \*\*\*  
## number\_project -0.3046936 0.0253106 -12.038 < 2e-16 \*\*\*  
## average\_montly\_hours 0.0049391 0.0006134 8.052 8.14e-16 \*\*\*  
## time\_spend\_company 0.2620968 0.0181052 14.476 < 2e-16 \*\*\*  
## Work\_accident -1.5225358 0.1068665 -14.247 < 2e-16 \*\*\*  
## promotion\_last\_5years -1.4666370 0.3089152 -4.748 2.06e-06 \*\*\*  
## departmenthr 0.3309520 0.1541206 2.147 0.03176 \*   
## departmentIT -0.1714985 0.1448775 -1.184 0.23651   
## departmentmanagement -0.3772352 0.1873155 -2.014 0.04402 \*   
## departmentmarketing -0.0475648 0.1561968 -0.305 0.76073   
## departmentproduct\_mng -0.1653244 0.1552326 -1.065 0.28687   
## departmentRandD -0.5540770 0.1706342 -3.247 0.00117 \*\*   
## departmentsales -0.0204986 0.1204263 -0.170 0.86484   
## departmentsupport -0.0032883 0.1291388 -0.025 0.97969   
## departmenttechnical 0.0210904 0.1254225 0.168 0.86646   
## salarylow 1.9066361 0.1526267 12.492 < 2e-16 \*\*\*  
## salarymedium 1.3751480 0.1534199 8.963 < 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: 11544.4 on 10498 degrees of freedom  
## Residual deviance: 9081.9 on 10480 degrees of freedom  
## AIC: 9119.9  
##   
## Number of Fisher Scoring iterations: 5

*#Asteriks against every entry in the table genrated by the summary signifies the importance of coresponding variables on the result.*  
  
*#Training accuracy.*  
lm\_train.prob =**predict** (lm\_train,type ="response")  
lm\_train.pred=**rep**("0",**length**(train\_data$left))  
lm\_train.pred[lm\_train.prob>0.5]="1"  
**table**(lm\_train.pred,train\_data$left,dnn = **c**('predicted','true'))

## true  
## predicted 0 1  
## 0 7407 1742  
## 1 584 766

accu\_train=(**sum**((lm\_train.pred==train\_data$left)/**length**(train\_data$left)))\*100  
accu\_train

## [1] 77.84551

*#Testing accuracy.*  
lm\_test.prob =**predict**(lm\_train,test\_data,type ="response")  
lm\_test.pred=**rep**("0",**length**(test\_data$left))  
lm\_test.pred[lm\_test.prob>0.5]="1"  
**table**(lm\_test.pred,test\_data$left,dnn = **c**('predicted','true'))

## true  
## predicted 0 1  
## 0 3205 715  
## 1 232 348

accu\_test=(**sum**((lm\_test.pred==test\_data$left)/**length**(test\_data$left)))\*100  
accu\_test

## [1] 78.95556

*#*  
**print**(**paste**("tarining accuracy:",accu\_train,"% testing accuracy:",accu\_test,"%" ))

## [1] "tarining accuracy: 77.8455090961044 % testing accuracy: 78.9555555555556 %"

##################Modelling data only on the basis of variables with p values less than '2e-16'.  
lm\_train1=**glm**(left~.-department-promotion\_last\_5years-average\_montly\_hours-last\_evaluation ,data=train\_data,family=binomial)  
**summary**(lm\_train1)

##   
## Call:  
## glm(formula = left ~ . - department - promotion\_last\_5years -   
## average\_montly\_hours - last\_evaluation, family = binomial,   
## data = train\_data)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.0624 -0.6748 -0.4320 -0.1419 2.8700   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -0.93693 0.18519 -5.059 4.21e-07 \*\*\*  
## satisfaction\_level -3.79888 0.10966 -34.642 < 2e-16 \*\*\*  
## number\_project -0.14856 0.01977 -7.514 5.75e-14 \*\*\*  
## time\_spend\_company 0.26143 0.01750 14.943 < 2e-16 \*\*\*  
## Work\_accident -1.53568 0.10578 -14.518 < 2e-16 \*\*\*  
## salarylow 1.99637 0.14964 13.341 < 2e-16 \*\*\*  
## salarymedium 1.44859 0.15041 9.631 < 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: 11544.4 on 10498 degrees of freedom  
## Residual deviance: 9272.2 on 10492 degrees of freedom  
## AIC: 9286.2  
##   
## Number of Fisher Scoring iterations: 5

*#We can quite clearly see that the p value of every variable has moved more close to 0.*  
  
*#Training accuracy*  
lm\_train1.probs =**predict** (lm\_train1,type ="response")  
lm\_train1.pred=**rep**("0",**length**(train\_data$left))  
lm\_train1.pred[lm\_train1.probs>0.5]="1"  
**table**(lm\_train1.pred,train\_data$left,dnn = **c**('predicted','true'))

## true  
## predicted 0 1  
## 0 7372 1573  
## 1 619 935

accu\_train1=(**sum**((lm\_train1.pred==train\_data$left)/**length**(train\_data$left)))\*100  
accu\_train1

## [1] 79.12182

*#Testing accuracy.*  
lm\_test1.prob =**predict**(lm\_train1,test\_data,type ="response")  
lm\_test1.pred=**rep**("0",**length**(test\_data$left))  
lm\_test1.pred[lm\_test1.prob>0.5]="1"  
**table**(lm\_test1.pred,test\_data$left,dnn = **c**('predicted','true'))

## true  
## predicted 0 1  
## 0 3201 639  
## 1 236 424

accu\_test1=(**sum**((lm\_test1.pred==test\_data$left)/**length**(test\_data$left)))\*100  
accu\_test1

## [1] 80.55556

*#*  
**print**(**paste**("tarining accuracy:",accu\_train1,"% testing accuracy:",accu\_test1,"%" ))

## [1] "tarining accuracy: 79.1218211258215 % testing accuracy: 80.5555555555556 %"

*#We have quite clearly seen the improvement in train and test accuracy than what it was before.*  
  
  
  
###############Modelling data after adding the interaction term(time\_spend\_company:promotion\_last\_5years) obtained on the basis of EDA.  
lm\_train3=**glm**(left~.-department-promotion\_last\_5years-average\_montly\_hours-last\_evaluation+time\_spend\_company:promotion\_last\_5years,data=train\_data,family=binomial)  
**summary**(lm\_train3)

##   
## Call:  
## glm(formula = left ~ . - department - promotion\_last\_5years -   
## average\_montly\_hours - last\_evaluation + time\_spend\_company:promotion\_last\_5years,   
## family = binomial, data = train\_data)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.1145 -0.6739 -0.4245 -0.1342 2.8626   
##   
## Coefficients:  
## Estimate Std. Error z value  
## (Intercept) -0.88387 0.18405 -4.802  
## satisfaction\_level -3.80190 0.10984 -34.613  
## number\_project -0.15404 0.01985 -7.762  
## time\_spend\_company 0.27694 0.01776 15.594  
## Work\_accident -1.53577 0.10622 -14.458  
## salarylow 1.92025 0.14928 12.864  
## salarymedium 1.38530 0.15018 9.224  
## time\_spend\_company:promotion\_last\_5years -0.39473 0.07778 -5.075  
## Pr(>|z|)   
## (Intercept) 1.57e-06 \*\*\*  
## satisfaction\_level < 2e-16 \*\*\*  
## number\_project 8.37e-15 \*\*\*  
## time\_spend\_company < 2e-16 \*\*\*  
## Work\_accident < 2e-16 \*\*\*  
## salarylow < 2e-16 \*\*\*  
## salarymedium < 2e-16 \*\*\*  
## time\_spend\_company:promotion\_last\_5years 3.88e-07 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 11544.4 on 10498 degrees of freedom  
## Residual deviance: 9222.6 on 10491 degrees of freedom  
## AIC: 9238.6  
##   
## Number of Fisher Scoring iterations: 6

*#Training accuracy*  
lm\_train3.probs =**predict** (lm\_train3,type ="response")  
lm\_train3.pred=**rep**("0",**length**(train\_data$left))  
lm\_train3.pred[lm\_train3.probs>0.5]="1"  
**table**(lm\_train3.pred,train\_data$left,dnn = **c**('predicted','true'))

## true  
## predicted 0 1  
## 0 7374 1515  
## 1 617 993

accu\_train3=(**sum**((lm\_train3.pred==train\_data$left)/**length**(train\_data$left)))\*100  
accu\_train3

## [1] 79.6933

*#Testing accuracy.*  
lm\_test3.prob =**predict**(lm\_train3,test\_data,type ="response")  
lm\_test3.pred=**rep**("0",**length**(test\_data$left))  
lm\_test3.pred[lm\_test3.prob>0.5]="1"  
**table**(lm\_test3.pred,test\_data$left,dnn = **c**('predicted','true'))

## true  
## predicted 0 1  
## 0 3203 614  
## 1 234 449

accu\_test3=(**sum**((lm\_test3.pred==test\_data$left)/**length**(test\_data$left)))\*100  
accu\_test3

## [1] 81.15556

*#*  
**print**(**paste**("tarining accuracy:",accu\_train3,"% testing accuracy:",accu\_test3,"%" ))

## [1] "tarining accuracy: 79.6933041242023 % testing accuracy: 81.1555555555556 %"

*#Both train and test accuracy has improved as compared to lm\_train1.*  
  
###########Modelling data after adding the interaction term(average\_montly\_hours:salary) obtained on the basis of EDA.  
lm\_train4=**glm**(left~.-department-promotion\_last\_5years-average\_montly\_hours-last\_evaluation+average\_montly\_hours:salary,data=train\_data,family=binomial)  
**summary**(lm\_train4)

##   
## Call:  
## glm(formula = left ~ . - department - promotion\_last\_5years -   
## average\_montly\_hours - last\_evaluation + average\_montly\_hours:salary,   
## family = binomial, data = train\_data)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.2472 -0.6740 -0.4190 -0.1332 3.0851   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) -4.152e-01 5.656e-01 -0.734 0.463  
## satisfaction\_level -3.907e+00 1.121e-01 -34.851 < 2e-16  
## number\_project -2.680e-01 2.372e-02 -11.300 < 2e-16  
## time\_spend\_company 2.540e-01 1.762e-02 14.420 < 2e-16  
## Work\_accident -1.536e+00 1.063e-01 -14.453 < 2e-16  
## salarylow 8.867e-01 5.739e-01 1.545 0.122  
## salarymedium 1.644e-01 5.805e-01 0.283 0.777  
## average\_montly\_hours:salaryhigh 1.155e-05 2.740e-03 0.004 0.997  
## average\_montly\_hours:salarylow 5.564e-03 7.420e-04 7.498 6.46e-14  
## average\_montly\_hours:salarymedium 6.440e-03 8.408e-04 7.660 1.86e-14  
##   
## (Intercept)   
## satisfaction\_level \*\*\*  
## number\_project \*\*\*  
## time\_spend\_company \*\*\*  
## Work\_accident \*\*\*  
## salarylow   
## salarymedium   
## average\_montly\_hours:salaryhigh   
## average\_montly\_hours:salarylow \*\*\*  
## average\_montly\_hours:salarymedium \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 11544.4 on 10498 degrees of freedom  
## Residual deviance: 9171.3 on 10489 degrees of freedom  
## AIC: 9191.3  
##   
## Number of Fisher Scoring iterations: 5

*#Training accuracy*  
lm\_train4.probs =**predict** (lm\_train4,type ="response")  
lm\_train4.pred=**rep**("0",**length**(train\_data$left))  
lm\_train4.pred[lm\_train4.probs>0.5]="1"  
**table**(lm\_train4.pred,train\_data$left,dnn = **c**('predicted','true'))

## true  
## predicted 0 1  
## 0 7414 1728  
## 1 577 780

accu\_train4=(**sum**((lm\_train4.pred==train\_data$left)/**length**(train\_data$left)))\*100  
accu\_train4

## [1] 78.04553

*#Testing accuracy.*  
lm\_test4.prob =**predict**(lm\_train4,test\_data,type ="response")  
lm\_test4.pred=**rep**("0",**length**(test\_data$left))  
lm\_test4.pred[lm\_test4.prob>0.5]="1"  
**table**(lm\_test4.pred,test\_data$left,dnn = **c**('predicted','true'))

## true  
## predicted 0 1  
## 0 3202 702  
## 1 235 361

accu\_test4=(**sum**((lm\_test4.pred==test\_data$left)/**length**(test\_data$left)))\*100  
accu\_test4

## [1] 79.17778

*#*  
**print**(**paste**("tarining accuracy:",accu\_train4,"% testing accuracy:",accu\_test4,"%" ))

## [1] "tarining accuracy: 78.0455281455377 % testing accuracy: 79.1777777777778 %"

*#Both train and test accuracy has decreased as compared to lm\_train1.*  
  
  
  
  
##########Modelling data after adding the interaction term(time\_spend\_company:salary) obtained on the basis of EDA.  
lm\_train5=**glm**(left~.-department-promotion\_last\_5years-average\_montly\_hours-last\_evaluation+time\_spend\_company:salary,data=train\_data,family=binomial)  
**summary**(lm\_train5)

##   
## Call:  
## glm(formula = left ~ . - department - promotion\_last\_5years -   
## average\_montly\_hours - last\_evaluation + time\_spend\_company:salary,   
## family = binomial, data = train\_data)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.4594 -0.6659 -0.4216 -0.1482 2.9067   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 0.49743 0.32222 1.544 0.12265   
## satisfaction\_level -3.87034 0.11101 -34.865 < 2e-16 \*\*\*  
## number\_project -0.16475 0.02000 -8.239 < 2e-16 \*\*\*  
## time\_spend\_company -0.05296 0.07266 -0.729 0.46612   
## Work\_accident -1.58020 0.10799 -14.633 < 2e-16 \*\*\*  
## salarylow 0.05283 0.32270 0.164 0.86994   
## salarymedium 0.47720 0.32234 1.480 0.13876   
## time\_spend\_company:salarylow 0.48415 0.07772 6.229 4.68e-10 \*\*\*  
## time\_spend\_company:salarymedium 0.21902 0.07697 2.846 0.00443 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 11544.4 on 10498 degrees of freedom  
## Residual deviance: 9193.1 on 10490 degrees of freedom  
## AIC: 9211.1  
##   
## Number of Fisher Scoring iterations: 5

*#Training accuracy*  
lm\_train5.probs =**predict** (lm\_train5,type ="response")  
lm\_train5.pred=**rep**("0",**length**(train\_data$left))  
lm\_train5.pred[lm\_train5.probs>0.5]="1"  
**table**(lm\_train5.pred,train\_data$left,dnn = **c**('predicted','true'))

## true  
## predicted 0 1  
## 0 7372 1646  
## 1 619 862

accu\_train5=(**sum**((lm\_train5.pred==train\_data$left)/**length**(train\_data$left)))\*100  
accu\_train5

## [1] 78.42652

*#Testing accuracy.*  
lm\_test5.prob =**predict**(lm\_train5,test\_data,type ="response")  
lm\_test5.pred=**rep**("0",**length**(test\_data$left))  
lm\_test5.pred[lm\_test5.prob>0.5]="1"  
**table**(lm\_test5.pred,test\_data$left,dnn = **c**('predicted','true'))

## true  
## predicted 0 1  
## 0 3197 665  
## 1 240 398

accu\_test5=(**sum**((lm\_test5.pred==test\_data$left)/**length**(test\_data$left)))\*100  
accu\_test5

## [1] 79.88889

*#*  
**print**(**paste**("tarining accuracy:",accu\_train5,"% testing accuracy:",accu\_test5,"%" ))

## [1] "tarining accuracy: 78.4265168111249 % testing accuracy: 79.8888888888889 %"

*#Both train and test accuracy has decreased as compared to lm\_train1.*  
  
  
################Modelling after adding interaction term(time\_spend\_company:satisfaction\_level:salary).  
lm\_train2=**glm**(left~.-department-promotion\_last\_5years-average\_montly\_hours-last\_evaluation+time\_spend\_company:satisfaction\_level:salary ,data=train\_data,family=binomial)  
**summary**(lm\_train2)

##   
## Call:  
## glm(formula = left ~ . - department - promotion\_last\_5years -   
## average\_montly\_hours - last\_evaluation + time\_spend\_company:satisfaction\_level:salary,   
## family = binomial, data = train\_data)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -3.2605 -0.6513 -0.3179 -0.0643 3.0348   
##   
## Coefficients:  
## Estimate Std. Error  
## (Intercept) 5.32296 0.32439  
## satisfaction\_level -13.65712 0.42598  
## number\_project -0.08535 0.02067  
## time\_spend\_company -1.14687 0.06109  
## Work\_accident -1.59973 0.11113  
## salarylow 0.93903 0.21430  
## salarymedium 1.02868 0.21606  
## satisfaction\_level:time\_spend\_company:salaryhigh 1.96059 0.11272  
## satisfaction\_level:time\_spend\_company:salarylow 2.50873 0.10002  
## satisfaction\_level:time\_spend\_company:salarymedium 2.19723 0.09589  
## z value Pr(>|z|)   
## (Intercept) 16.409 < 2e-16 \*\*\*  
## satisfaction\_level -32.061 < 2e-16 \*\*\*  
## number\_project -4.130 3.63e-05 \*\*\*  
## time\_spend\_company -18.774 < 2e-16 \*\*\*  
## Work\_accident -14.395 < 2e-16 \*\*\*  
## salarylow 4.382 1.18e-05 \*\*\*  
## salarymedium 4.761 1.92e-06 \*\*\*  
## satisfaction\_level:time\_spend\_company:salaryhigh 17.394 < 2e-16 \*\*\*  
## satisfaction\_level:time\_spend\_company:salarylow 25.083 < 2e-16 \*\*\*  
## satisfaction\_level:time\_spend\_company:salarymedium 22.914 < 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: 11544.4 on 10498 degrees of freedom  
## Residual deviance: 8461.2 on 10489 degrees of freedom  
## AIC: 8481.2  
##   
## Number of Fisher Scoring iterations: 6

*#We can quite clearly see that the p value of every variable has moved more close to 0.*  
  
*#Training accuracy*  
lm\_train2.probs =**predict** (lm\_train2,type ="response")  
lm\_train2.pred=**rep**("0",**length**(train\_data$left))  
lm\_train2.pred[lm\_train2.probs>0.5]="1"  
**table**(lm\_train2.pred,train\_data$left,dnn = **c**('predicted','true'))

## true  
## predicted 0 1  
## 0 7372 1318  
## 1 619 1190

accu\_train2=(**sum**((lm\_train2.pred==train\_data$left)/**length**(train\_data$left)))\*100  
accu\_train2

## [1] 81.55062

*#Testing accuracy.*  
lm\_test2.prob =**predict**(lm\_train2,test\_data,type ="response")  
lm\_test2.pred=**rep**("0",**length**(test\_data$left))  
lm\_test2.pred[lm\_test2.prob>0.5]="1"  
**table**(lm\_test2.pred,test\_data$left,dnn = **c**('predicted','true'))

## true  
## predicted 0 1  
## 0 3209 513  
## 1 228 550

accu\_test2=(**sum**((lm\_test2.pred==test\_data$left)/**length**(test\_data$left)))\*100  
accu\_test2

## [1] 83.53333

*#*  
**print**(**paste**("tarining accuracy:",accu\_train2,"% testing accuracy:",accu\_test2,"%" ))

## [1] "tarining accuracy: 81.5506238689399 % testing accuracy: 83.5333333333333 %"

*#We have quite clearly seen the improvement in train and test accuracy as compared to lm\_train5.*  
  
  
#############Modelling after adding interaction term(time\_spend\_company:satisfaction\_level:salary & #Modelling after adding interaction term(time\_spend\_company:satisfaction\_level:salary).  
   
  
lm\_train6=**glm**(left~.-department-average\_montly\_hours-last\_evaluation+time\_spend\_company:promotion\_last\_5years+time\_spend\_company:satisfaction\_level:salary ,data=train\_data,family=binomial)  
**summary**(lm\_train6)

##   
## Call:  
## glm(formula = left ~ . - department - average\_montly\_hours -   
## last\_evaluation + time\_spend\_company:promotion\_last\_5years +   
## time\_spend\_company:satisfaction\_level:salary, family = binomial,   
## data = train\_data)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -3.2752 -0.6479 -0.3123 -0.0593 3.0240   
##   
## Coefficients:  
## Estimate Std. Error  
## (Intercept) 5.28565 0.32588  
## satisfaction\_level -13.64019 0.42814  
## number\_project -0.08924 0.02072  
## time\_spend\_company -1.13601 0.06149  
## Work\_accident -1.59157 0.11145  
## promotion\_last\_5years 0.73242 0.89176  
## salarylow 0.96087 0.21523  
## salarymedium 1.05085 0.21706  
## time\_spend\_company:promotion\_last\_5years -0.55741 0.25349  
## satisfaction\_level:time\_spend\_company:salaryhigh 1.98221 0.11469  
## satisfaction\_level:time\_spend\_company:salarylow 2.50394 0.10055  
## satisfaction\_level:time\_spend\_company:salarymedium 2.19752 0.09667  
## z value Pr(>|z|)   
## (Intercept) 16.219 < 2e-16 \*\*\*  
## satisfaction\_level -31.859 < 2e-16 \*\*\*  
## number\_project -4.308 1.65e-05 \*\*\*  
## time\_spend\_company -18.475 < 2e-16 \*\*\*  
## Work\_accident -14.280 < 2e-16 \*\*\*  
## promotion\_last\_5years 0.821 0.4115   
## salarylow 4.464 8.03e-06 \*\*\*  
## salarymedium 4.841 1.29e-06 \*\*\*  
## time\_spend\_company:promotion\_last\_5years -2.199 0.0279 \*   
## satisfaction\_level:time\_spend\_company:salaryhigh 17.284 < 2e-16 \*\*\*  
## satisfaction\_level:time\_spend\_company:salarylow 24.903 < 2e-16 \*\*\*  
## satisfaction\_level:time\_spend\_company:salarymedium 22.732 < 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: 11544.4 on 10498 degrees of freedom  
## Residual deviance: 8428.7 on 10487 degrees of freedom  
## AIC: 8452.7  
##   
## Number of Fisher Scoring iterations: 6

*#We can quite clearly see that the p value of every variable has moved more close to 0.*  
  
*#Training accuracy*  
lm\_train6.probs =**predict** (lm\_train6,type ="response")  
lm\_train6.pred=**rep**("0",**length**(train\_data$left))  
lm\_train6.pred[lm\_train6.probs>0.5]="1"  
**table**(lm\_train6.pred,train\_data$left,dnn = **c**('predicted','true'))

## true  
## predicted 0 1  
## 0 7364 1282  
## 1 627 1226

accu\_train6=(**sum**((lm\_train6.pred==train\_data$left)/**length**(train\_data$left)))\*100  
accu\_train6

## [1] 81.81732

*#Testing accuracy.*  
lm\_test6.prob =**predict**(lm\_train6,test\_data,type ="response")  
lm\_test6.pred=**rep**("0",**length**(test\_data$left))  
lm\_test6.pred[lm\_test6.prob>0.5]="1"  
**table**(lm\_test6.pred,test\_data$left,dnn = **c**('predicted','true'))

## true  
## predicted 0 1  
## 0 3203 499  
## 1 234 564

accu\_test6=(**sum**((lm\_test6.pred==test\_data$left)/**length**(test\_data$left)))\*100  
accu\_test6

## [1] 83.71111

*#*  
**print**(**paste**("tarining accuracy:",accu\_train6,"% testing accuracy:",accu\_test6,"%" ))

## [1] "tarining accuracy: 81.8173159348509 % testing accuracy: 83.7111111111111 %"

*#Test and train accuracy both have increased as compared to lm\_train2.*  
  
###########Checking corelation between used variables.  
**cor**(train\_data[,**c**("satisfaction\_level","number\_project","time\_spend\_company","Work\_accident")])

## satisfaction\_level number\_project time\_spend\_company  
## satisfaction\_level 1.00000000 -0.147984098 -0.09166199  
## number\_project -0.14798410 1.000000000 0.19803773  
## time\_spend\_company -0.09166199 0.198037725 1.00000000  
## Work\_accident 0.06361790 -0.007311875 -0.00731463  
## Work\_accident  
## satisfaction\_level 0.063617895  
## number\_project -0.007311875  
## time\_spend\_company -0.007314630  
## Work\_accident 1.000000000

*#There is no significant amount of corelation among checked vairables.*

Even though the last two models namely ‘lm6’ and ‘lm2’ perform almost almost identically but ‘lm6’ edge the ‘lm2’ slighlty with better train and test accuracies 81.8173159348509 % and 83.7111111111111 % respectively.

* ‘lm6’ incorporates the impact of:
* satisfaction\_level
* number\_project
* time\_spend\_company
* Work\_accident
* promotion\_last\_5years
* salary
* Interaction of time\_spend\_company & promotion\_last\_5years
* Interaction of satisfaction\_level & time\_spend\_company & salary
* lm2’ lacked the interaction of time\_spend\_company & promotion\_last\_5years.
* Even though we can see the ‘p value’ of ‘promotion\_last\_5years’ in ‘lm6’ is high but for interaction of time\_spend\_company & promotion\_last\_5years it is low enough to make the impact of interaction term significant.
* The interactions which had the role to play were also explored during EDA.

**CHAPTER 5**

# Conclusion

## 5.1 Learning

During the course of the internship I got to learn the following things:

* I got to know the nature of data science and data analysis and managed to figure out the difference in the nature of job of a Data Analyst and Data Scientist.
* I learned the flow of a Data Science project and then was able to implement it to make a model to predict the characteristics which are leading to churn out of employs.
* I got proper hold on the EDA (Exploratory Data Analysis).
* I learned the implementation of ‘linear regression’ and ‘logistic regression’ model with proper understanding of the concepts.
* I learned the usage of following packages:
  + ggplot2
  + datasets
  + plyr
  + reshape2
* I explored through following datasets during the course of internship:
  + Chickweight
  + iris
  + women
  + and obviously the HR\_Data which is the dataset of the main project

## 5.2 Conclusion

After performing EDA and then modelling data using logistic regression we have come to know that whether the employ will leave the firm or stay depends on the below mentioned factors:

* **satisfaction\_level:**

Coefficient associated with summary signifies that with increse in satisfaction level chance of person leaving the job decreases. Which sounds quite true to a lay man as well because probability of staying at a place is obviously directly proportioanl to how satisfied you are at that place?

* **number\_project:**

Estimated negative coefficient which is not that large in magnitude suggests that with increase in number of projects for a given person his/her chance of staying increases.

* **time\_spend\_company:**

Even though the magnitude of coefficient here is not that high but negative sign implies that with every increasing year the person has stayed at the company his/her chance of leaving the firm has decreased.

* **Work\_accident:**

Having a work accident has decreased the chance of person leaving the job.

* **promotion\_last\_5years:**

Even though the ‘p value’ for this parameter is not less to make it significant but we are including it as it has its importance in one of the interaction terms. So considering only the additive effect and concluding its impact on the chances of leaving is not appropriate.

* **Salary:**

‘p value’ suggests that salary is quite a significant factor. Estimates of coefficients for the dummy variables suggest that chance of people leaving with low salary and medium salary are 0.91734 & 0.95293 more than those who have left with high salary respectively.

* **Interaction of time\_spend\_company & promotion\_last\_5years:**

From the tables made during the analysis of combined effect of the above mentioned terms we can quite clearly see that the ratio of people leaving to that of who have stayed for almost all categories of salary is increasing with increase in the number of years spend in the company and there after the count of people leaving is 0 as no one has left the company after working for 6 years.

* **Interaction of satisfaction\_level & time\_spend\_company & salary :**

From the graphs plotted during EDA we can quite clearly see that initially for first 2 years satisfacation level is not that significant in each of the salary category. But for those who have spend 3 & 4 years we can quite clearly see that count of poeple leving with satisfaction level less than 0.5 is more than those who have left this difference in count is comparatively less in high salary category. And for those who have spent 5 & 6 years count of people leaving with high satisfaction level is more in medium and low salary category as compared to those who have stayed. No one has left the company after working more than 6 years and in high salary category no one has left after working more than 5 years. ‘p value’ during modelling shows that this term is quite important.

**CHAPTER 6**

# Future Scope

These were just certain models which I was able to get hold of in the available time. There is still an ocean of techniques and methodologies available to be explored and learnt. The area that can be explored in this field is always quite wide from applying cross validation to make results of each logistic model better and even more trustworthy to applying models like:

* **Decision tree**: Each node of the tree reduces the possible options by looking at the trend followed in historic data. Hence, decision tree. e.g. recommendation of clothes on e-commerce site can be made on the basis of the info of the current user and this can pretty well be aided by the 'decision tree'.
* **Neural networks**: It is combination of lines of regression which are used to specify the section on the graph to split and classify the mappings on the scatter plots.
* **Support Vector Machines**: Out of the two possible regression lines the best is the one which is optimistically away from boundary point of each section. For this now we don't use 'Gradient Descent' rather we use 'linear optimization'. And on the whole this method is called 'Support Vector Machines'. Kernel Trick is used in Support Vector Machines when line is not enough to split. We either use different geometric curves or we can either use different planes for different sets (both of them are same).

These algorithms are comparatively complex to understand but once you get hold of them they provide quite effective and useful models for data where linear regression and logistic regression don’t turn out to be that significant.

# REFERENCES

[1] <https://en.wikipedia.org/wiki/Data_science>

[2] <https://en.wikipedia.org/wiki/Data_analysis>

[3] <https://en.wikipedia.org/wiki/R_(programming_language)>

[4] <https://www.dezyre.com/article/why-data-preparation-is-an-important-part-of-data-science/242>

[5] <http://visit.crowdflower.com/rs/416-ZBE-142/images/CrowdFlower_DataScienceReport_2016.pdf>

[6] Becker, R. A., Chambers, J. M. and Wilks, A. R. (1988) The New S Language. Wadsworth & Brooks/Cole. (has iris3 as iris.)

[7] Crowder, M. and Hand, D. (1990), *Analysis of Repeated Measures*, Chapman and Hall (example 5.3)

[8] McNeil, D. R. (1977) Interactive Data Analysis. Wiley.