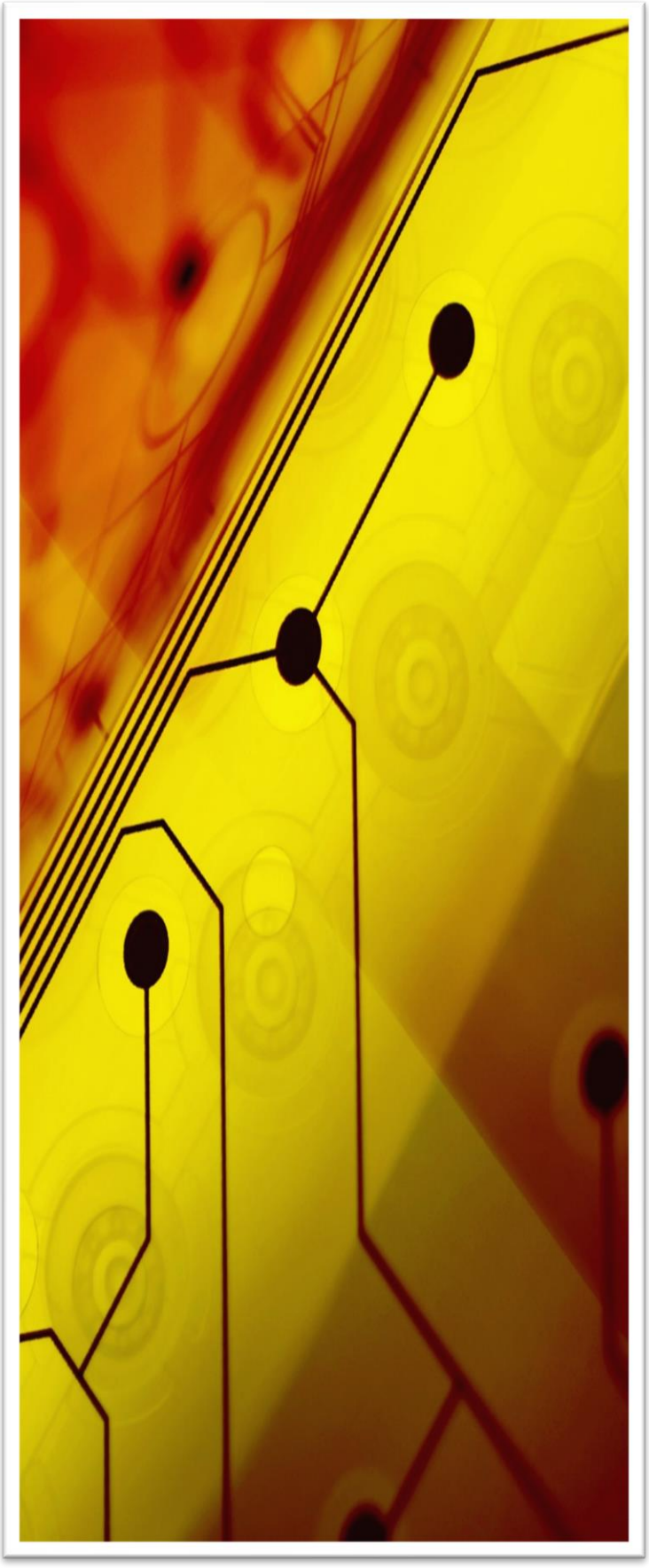


Attrition in an organization

AKANKSHA PORWAL &

DIPSHI JAIN

MSC - DATA SCIENCE

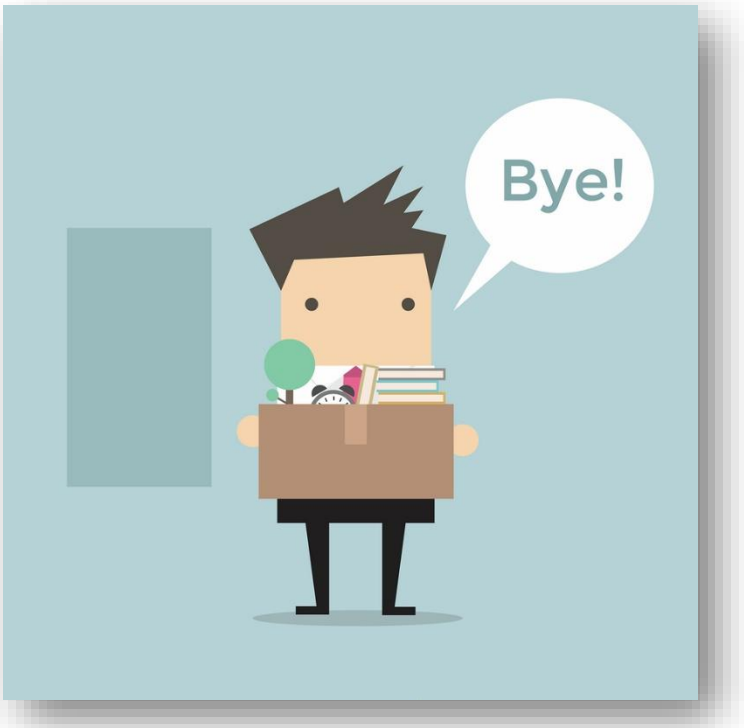
SEM-I

202118017 & 202118018

Instructor– Mr. Pritam Anand Sir & Ms. Swati Madam

19-12-2021

Attrition in an Organization || Why Workers Quit? Employee Attritions



An Interesting Quote I Found:

" Managers tend to blame their turnover problems on everything under the sun, while ignoring the crux of the matter: people don't leave jobs; they leave managers." by Travis Bradberry

What is my aim with this project?

- **Recommendations:** What recommendations will I give the organization based on the analysis made with this data. How can the organization reduce the rate of attrition inside the company? In my opinion, this is the most important part of the analysis because it gives us a better understanding of what the organization could do to avoid the negative effect of attrition.

So, what is Attrition and what determines it?
Attrition: It is basically the turnover rate of employees inside an organization.

This can happen for many reasons:

- Employees looking for better opportunities.
- A negative working environment.
- Bad management
- Sickness of an employee (or even death)
- Excessive working hours

Structure of the Project:

This project will be structured in the following way:

- **Questions:** Questions will be asked previous to the visualization to make sure the visualizations shown in this project are insightful.
- **Summary:** I will provide a summary to understand what we got from the visualizations.
- **Recommendations:** What recommendations could be given to the organization to reduce the **attrition rate**.

Table of Contents:

Dataset Source: https://www.kaggle.com

>#Reading the dataset into df variable

> df<-read.csv("C:\\Akanksha Porwal\\R Programming\\HR-Employee-Attrition.csv")

> head(df)*#for seeing first 5 rows of the data*

> tail(df) *#for seeing first 5 rows of the data*

For viewing our dataset

> View(df)

Filter

	ï..Age	Attrition	BusinessTravel	DailyRate	Department	DistanceFromHome	Education	EducationField	EmployeeCount	EmployeeNumber
2	18	Yes	Travel_Rarely	230	Research & Development	3	3	Life Sciences	1	405
302	18	No	Travel_Rarely	812	Sales	10	3	Medical	1	411
458	18	Yes	Travel_Frequently	1306	Sales	5	3	Marketing	1	614
728	18	No	Non-Travel	287	Research & Development	5	2	Life Sciences	1	1012
829	18	Yes	Non-Travel	247	Research & Development	8	1	Medical	1	1156
973	18	No	Non-Travel	1124	Research & Development	1	3	Life Sciences	1	1368
1154	18	Yes	Travel_Frequently	544	Sales	3	2	Medical	1	1624
1312	18	No	Non-Travel	1431	Research & Development	14	3	Medical	1	1839
128	19	Yes	Travel_Rarely	528	Sales	22	1	Marketing	1	167
150	19	No	Travel_Rarely	1181	Research & Development	3	1	Medical	1	201
172	19	Yes	Travel_Frequently	602	Sales	1	1	Technical Degree	1	235
178	19	Yes	Travel_Rarely	303	Research & Development	2	3	Life Sciences	1	243
423	19	Yes	Travel_Rarely	489	Human Resources	2	2	Technical Degree	1	566
689	19	Yes	Travel_Rarely	419	Sales	21	3	Other	1	959
854	19	No	Travel_Rarely	645	Research & Development	9	2	Life Sciences	1	1193

 # Variable's type:

- **Numeric variables**:

- Related to personal information: age, distance_from_home, employee_number (id variable)

- Related to income: hourly_rate, daily_rate, monthly_rate, monthly_income, percent_salary_hike

- Related to time in company: years_at_company, years_in_current_role, years_since_last_promotion, years_with_curr_manager, total_working_years

- other: num_companies_worked, standard_hours(to delete), training_times_last_year, employee_count (to delete)

- **Categorical variables**:

- **Binary variables**: attrition(target variable),

gender, over18 (to delete),

over_time

- **Nominal variables**: department, education_field, job_role, marital_status

- **Ordinal variables**:

- Ordinal regarding satisfaction and performance : environment_satisfaction, job_satisfaction, relationship_satisfaction, work_life_balance,job_involvement,performance_rating

- Other ordinal: business_travel, education, job_level, stock_option_level

Summary of our Data

Before we get into the deep visualizations, we want to make sure how our data looks like right? This will better help us have a better grasp as to how we should work with our data later throughout the project.

Questions we could Ask Ourselves:

- **Columns and Observations:** How many columns and observations is there in our dataset?
- **Missing data:** Are there any missing data in our dataset?
- **Data Type:** The different datatypes we are dealing in this dataset.
- **Distribution of our Data:** Is it right-skewed, left-skewed or symmetric? This might be useful especially if we are implementing any type of statistical analysis or even for modelling.
- **Structure of our Data:** Some datasets are a bit complex to work with however, the tidy-verse, ggplot2 etc packages are really useful to deal with complex datasets.
- **Meaning of our Data:** What does our data mean? Most features in this dataset are **ordinal variables** which are similar to categorical variables however, ordering of those variables matter. A lot of the variables in this dataset have a range from 1-4 or 1-5, **The lower the ordinal variable, the worse it is in this case.** For instance, Job Satisfaction 1 = "Low" while 4 = "Very High".
- **Label:** What is our label in the dataset or in other words the output?

Summary:

- **Dataset Structure:** 1470 observations (rows), 35 features (variables)
- **Missing Data:** Luckily for us, there is no missing data! this will make it easier to work with the dataset.
- **Data Type:** We only have two datatypes in this dataset: factors and integers
- **Label"** Attrition is the label in our dataset and we would like to find out why employees are leaving the organization!
- **Imbalanced dataset:** 1237 (84% of cases) employees did not leave the organization while 237 (16% of cases) did leave the organization making our dataset to be considered **imbalanced** since more people stay in the organization than they actually leave.

How many employees did not leave the organization?

```
> df$Attrition <- ifelse(df$Attrition=="Yes", 0, 1)
> sum(df$Attrition)
[1] 1470
```

To get summary of our data frame

summary statistic is computed using summary () function in R. **summary** () function is automatically applied to each column. The format of the result depends on the data type of the column.

- If the column is a numeric variable, mean, median, min, max and quartiles are returned.
- If the column is a factor variable, the number of observations in each group is returned.

Descriptive statistics in R with simple summary function calculates

- minimum value of each column
- maximum value of each column
- mean value of each column
- median value of each column
- 1st quartile of each column (25th percentile)
- 3rd quartile of each column (75th percentile)

>summary(df)

```
> summary(df)
     ï..Age       Attrition         BusinessTravel       DailyRate       Department        DistanceFromHome
 Min.   :18.00   Length:1470        Length:1470        Min.   : 102.0   Length:1470        Min.   : 1.000
 1st Qu.:30.00   Class :character   Class :character   1st Qu.: 465.0   Class :character   1st Qu.: 2.000
 Median :36.00   Mode  :character   Mode  :character   Median : 802.0   Mode  :character   Median : 7.000
 Mean   :36.92                                         Mean   : 802.5                      Mean   : 9.193
 3rd Qu.:43.00                                         3rd Qu.:1157.0                      3rd Qu.:14.000
 Max.   :60.00                                         Max.   :1499.0                      Max.   :29.000
   Education     EducationField     EmployeeCount EmployeeNumber   EnvironmentSatisfaction    Gender
 Min.   :1.000   Length:1470        Min.   :1     Min.   :   1.0   Min.   :1.000           Length:1470
 1st Qu.:2.000   Class :character   1st Qu.:1     1st Qu.: 491.2   1st Qu.:2.000           Class :character
 Median :3.000   Mode  :character   Median :1     Median :1020.5   Median :3.000           Mode  :character
 Mean   :2.913                     Mean   :1     Mean   :1024.9   Mean   :2.722
 3rd Qu.:4.000                     3rd Qu.:1     3rd Qu.:1555.8   3rd Qu.:4.000
 Max.   :5.000                     Max.   :1     Max.   :2068.0   Max.   :4.000
   HourlyRate     JobInvolvement    JobLevel       JobRole          JobSatisfaction MaritalStatus
 Min.   : 30.00   Min.   :1.00   Min.   :1.000   Length:1470        Min.   :1.000   Length:1470
 1st Qu.: 48.00   1st Qu.:2.00   1st Qu.:1.000   Class :character   1st Qu.:2.000   Class :character
```

The glimpse method can be used to see the columns of data and display some portion of the data for each variable that can be fit on a single line.

Using an insightful summary

>glimpse (df)

```
> glimpse(df)
Rows: 1,470
Columns: 35
$ ï..Age                  <int> 41, 49, 37, 33, 27, 32, 59, 30, 38, 36, 35, 29, 31, 34, 28, 29, 32, 22, 53, 38, 24,~
$ Attrition               <chr> "Yes", "No", "Yes", "No", "No", "No", "No", "No", "No", "No", "No", "No", "No", "No~
$ BusinessTravel          <chr> "Travel_Rarely", "Travel_Frequently", "Travel_Rarely", "Travel_Frequently", "Travel~
$ DailyRate               <int> 1102, 279, 1373, 1392, 591, 1005, 1324, 1358, 216, 1299, 809, 153, 670, 1346, 103, ~
$ Department              <chr> "Sales", "Research & Development", "Research & Development", "Research & Developmen~
$ DistanceFromHome        <int> 1, 8, 2, 3, 2, 2, 3, 24, 23, 27, 16, 15, 26, 19, 24, 21, 5, 16, 2, 2, 11, 9, 7, 15,~
$ Education               <int> 2, 1, 2, 4, 1, 2, 3, 1, 3, 3, 3, 2, 1, 2, 3, 4, 2, 2, 4, 3, 2, 4, 4, 2, 1, 3, 1, 4,~
$ EducationField          <chr> "Life Sciences", "Life Sciences", "Other", "Life Sciences", "Medical", "Life Scienc~
$ EmployeeCount           <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,~
$ EmployeeNumber          <int> 1, 2, 4, 5, 7, 8, 10, 11, 12, 13, 14, 15, 16, 18, 19, 20, 21, 22, 23, 24, 26, 27, 2~
$ EnvironmentSatisfaction <int> 2, 3, 4, 4, 1, 4, 3, 4, 4, 3, 1, 4, 1, 2, 3, 2, 1, 4, 1, 4, 1, 3, 1, 3, 2, 3, 2, 3,~
$ Gender                  <chr> "Female", "Male", "Male", "Female", "Male", "Male", "Female", "Male", "Male", "Male~
```

Some inferences:

- **Age by Gender:** The average age of females is 37.3 and for males is 36.7 and both distributions are **similar**.

```
> labels<-summarize_at(group_by(df,Gender),vars(ï..Age),funs(mean(.,na.rm=FALSE)))
> labels
# A tibble: 2 x 2
  Gender ï..Age
   <dbl>  <dbl>
1      0   36.7
2      1   37.3
```

- **Attrition by environment satisfaction:** For individuals who didn't leave the organization, Environment satisfaction levels are higher than those who left the organization.
- However, for people who **left the organization,** had a lower Environment satisfaction level as opposed to previous.

```
> labels<-summarize_at(group_by(df,Attrition),vars(EnvironmentSatisfaction),funs(mean(.,na.rm=TRUE)))
> labels
# A tibble: 2 x 2
  Attrition EnvironmentSatisfaction
      <dbl>                   <dbl>
1         0                    2.77
2         1                    2.46
>
```

`

- **Salaries:** The average salaries for both genders are practically the same with **males** having an average of 6381.0 and **females** 6687.0.

```
> labels<-summarize_at(group_by(df,Gender),vars(MonthlyIncome),funs(mean(.,na.rm=FALSE)))
> labels
# A tibble: 2 x 2
  Gender MonthlyIncome
   <dbl>         <dbl>
1      0         6381.
2      1         6687.
>
```

Summarise

summarise_at, summarise_if, summarise_all in R – Summary of the dataset (Mean, Median and Mode) in R can be done using Dplyr.

#Dplyr package in R is provided with **group_by ()** function which groups the dataframe by multiple columns with mean, sum and other functions like count, maximum and minimum.

#Usually, we can use the argument **na.rm = TRUE** to **exclude missing values** when calculating descriptive statistics in R.

>summarize_at(group_by(df,Gender),vars(ï..Age),funs(mean(.,na.rm=TRUE)))

```
> summarize_at(group_by(df,Gender),vars(ï..Age),funs(mean(.,na.rm=TRUE)))
# A tibble: 2 x 2
  Gender ï..Age
  <chr>   <dbl>
1 Female   37.3
2 Male     36.7
```

>summarize_at(group_by(df,Gender),vars(ï..Age),funs(median(.,na.rm=TRUE)))

```
> summarize_at(group_by(df,Gender),vars(ï..Age),funs(median(.,na.rm=TRUE)))
# A tibble: 2 x 2
  Gender ï..Age
  <chr>   <dbl>
1 Female     36
2 Male       35
```

>summarize_at(group_by(df,Gender),vars(ï..Age),funs(mfv(.,na.rm=TRUE)))

```
# A tibble: 2 x 2
  Gender ï..Age
  <chr>   <int>
1 Female     34
2 Male       35
>
```

Check minimum and maximum salary by gender

#Now we will see the minimum and maximum salary given to employees to check if there is a discrimination over gender in the company.

```
> summarize_at(group_by(df,Gender),vars(MonthlyIncome),funs(min(.,na.rm=TRUE)))
# A tibble: 2 x 2
  Gender MonthlyIncome
  <chr>          <int>
1 Female          1129
2 Male            1009
```

> summarize_at (group_by (df, Gender), vars (MonthlyIncome), funs (max (., na.rm=TRUE)))

```
# A tibble: 2 x 2
  Gender MonthlyIncome
  <chr>          <int>
1 Female         19973
2 Male           19999
>
```

Here, we can see the Minimum salary for female is greater than man and of maximum it's more for male so it's clear that there is no salary wise discrimination on gender.

Boxplots

They are a measure of how well distributed is the data in a data set. It divides the data set into three quartiles. This graph represents the minimum, maximum, median, first quartile and third quartile in the data set. It is also useful in comparing the distribution of data across data sets by drawing boxplots for each of them.

Job Satisfaction

 # Boxplot with attrition in the X-axis and Job Satisfaction in the y-Axis

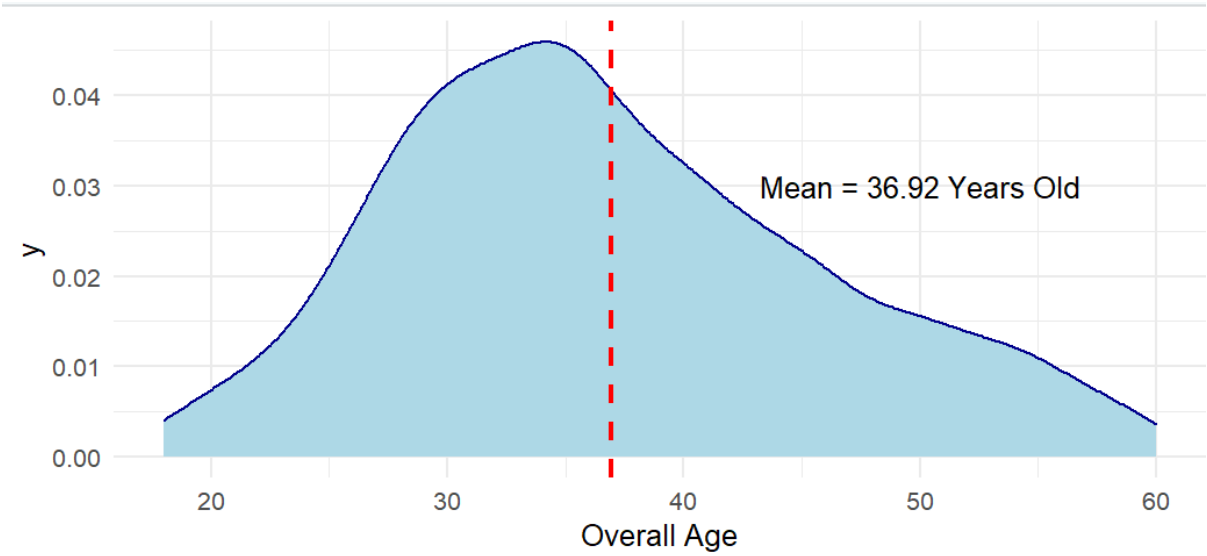
> ggplot(data=df,aes(x=Attrition, y=JobSatisfaction, fill=Attrition)) + geom_boxplot(color="black") + theme_minimal() + facet_wrap(~Gender) +
+ scale_fill_manual(values=c("#FA5858", "#9FF781"))

 # Distribution of Job Satisfaction

```
> ggplot(data=df,aes(x=JobSatisfaction)) +
geom_density(color="#013ADF", fill="#81BEF7", trim=TRUE) +
xlim(range(c(1,4)))
```

 # Distribution of age

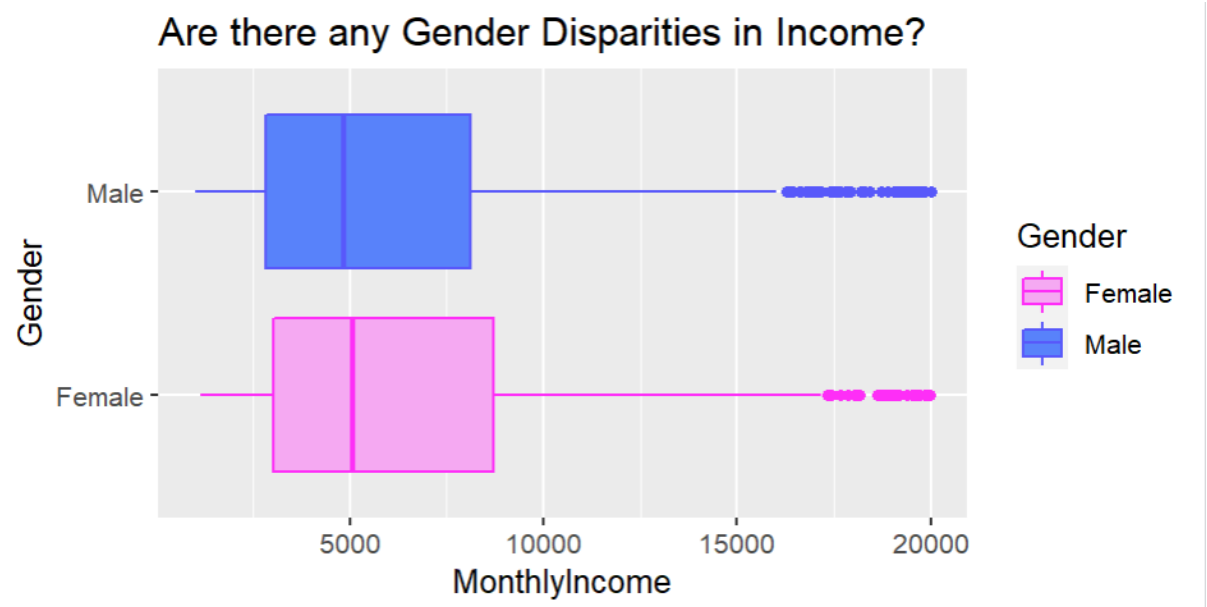
```
> ggplot(data=df, mapping=aes(x=ï..Age)) +
geom_density(color="darkblue", fill="lightblue") +
+     geom_vline(aes(xintercept=mean(ï..Age)),
+                color="red", linetype="dashed", size=1) +  theme_minimal() +
labs(x="Overall Age") +
+     annotate("text", label = "Mean = 36.92 Years Old", x = 50, y = 0.03,
color = "black")
```



#Monthly Income by Gender

```
p <- ggplot(df, aes(x=Gender, y=MonthlyIncome, color=Gender,
fill=Gender)) + geom_boxplot() +
scale_fill_manual(values=c("#F5A9F2", "#5882FA")) +
scale_color_manual(values=c("#FE2EF7", "#5858FA")) +
coord_flip() + labs(title="Are there any Gender Disparities in Income?")
```

```
p
```



#Variance

```
> var(df$ï..Age)
[1] 83.45505
```

#Standard deviation

```
> sd(df$ï..Age)
[1] 9.135373
```

#Quantiles

Quantiles are values that divide a ranked dataset into equal groups.

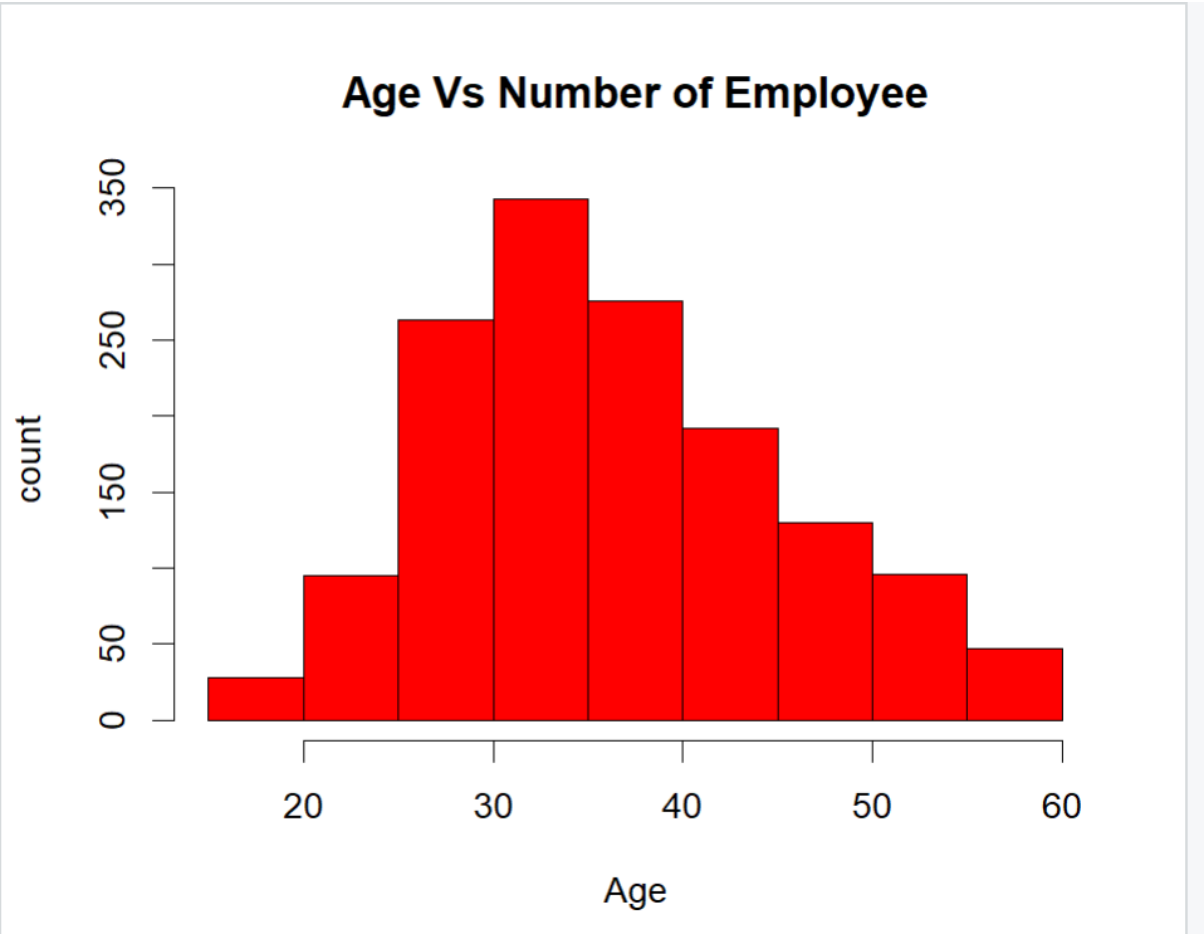
The **quantile()** function in R can be used to calculate sample quantiles of a dataset.

```
> quantile (df$ï.. Age, probs = seq(0, 1, 0.25), na.rm = FALSE)
  0%  25%  50%  75% 100%
  18   30   36   43   60
```

#Histogram

>The histogram consists of an x-axis, a y-axis and various bars of different heights. The y-axis shows how frequently the values on the x-axis occur in the data, while the bars group ranges of values or continuous categories on the x-axis.

```
> hist(main="Age Vs Number of
Employee",xlab="Age",ylab="count",df$ï..Age,col="red")
```



 # Correlation coefficient

A **correlation coefficient** quite close to 0, but either positive or negative, implies little or no relationship between the two variables. A correlation coefficient close to plus 1 means a positive relationship between the two variables, with increases in one of the variables being associated with increases in the other variable.

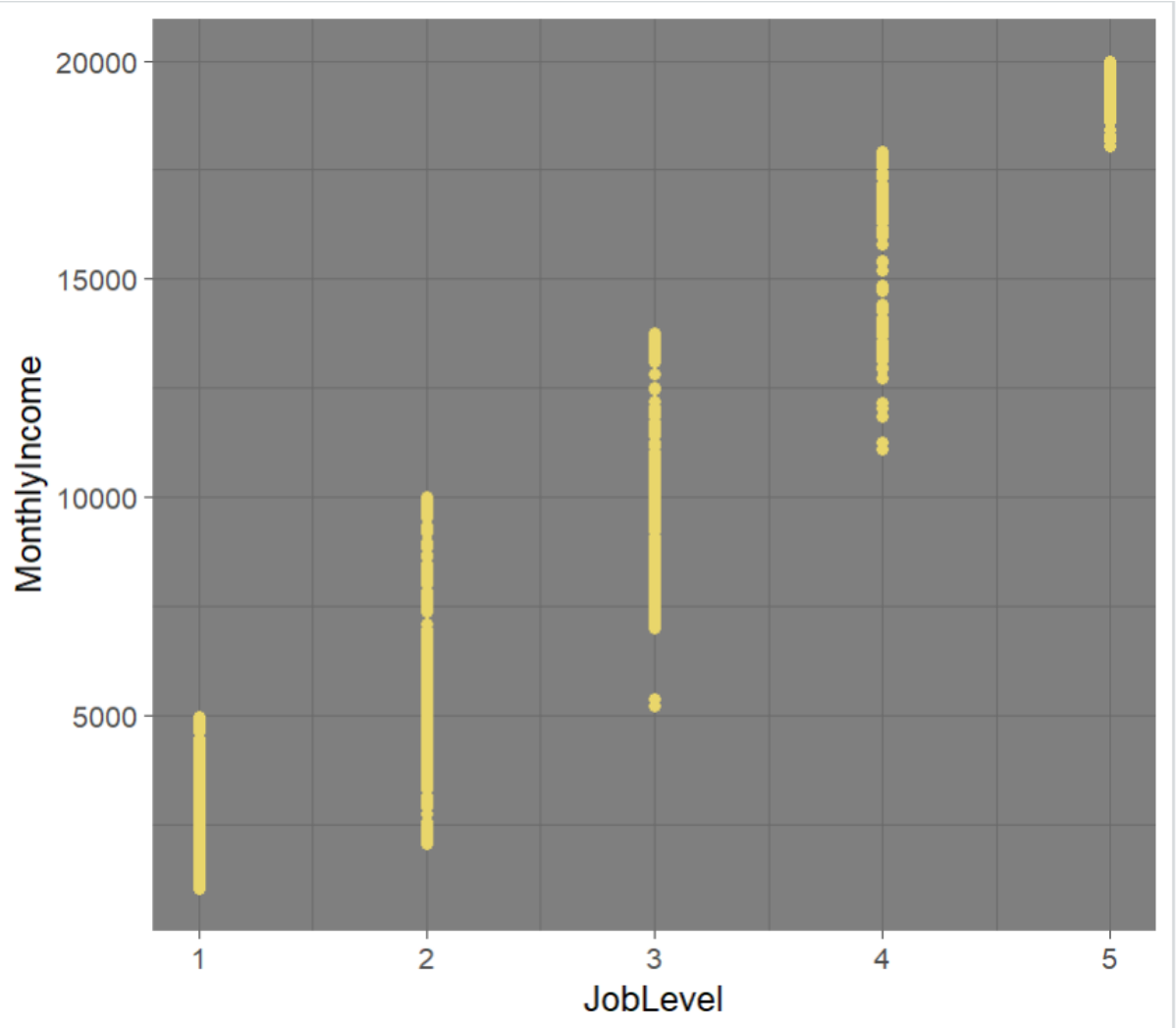
A correlation coefficient close to -1 indicates a negative relationship between two variables, with an increase in one of the variables being associated with a decrease in the other variable. A correlation coefficient can be produced for ordinal, interval or ratio level variables, but has little meaning for variables which are measured on a scale which is no more than nominal.

In statistics, Correlation studies and measures the direction and extent of relationship among variables, so the correlation measures co-variation, not causation. Therefore, we should never interpret correlation as implying cause and effect relation. For example, there exists a correlation between two variables X and Y, which means the value of one variable is found to change in one direction, the value of the other variable is found to change either in the same direction (i.e. positive change) or in the opposite direction (i.e. negative change). Furthermore, if the correlation exists, it is linear, i.e., we can represent the relative movement of the two variables by drawing a straight line on graph paper.

```
> cor(df$ï..Age,df$DistanceFromHome,method="spearman")
[1] -0.01929091
> cor(df$ï..Age,df$DailyRate,method="spearman")
[1] 0.007289728
> cor(df$ï..Age,df$Education,method="spearman")
[1] 0.2049367
> cor(df$YearsAtCompany,df$JobSatisfaction,method="spearman")
[1] 0.01228041
> cor(df$YearsAtCompany,df$MonthlyIncome,method="spearman")
[1] 0.4643152
> cor(df$JobLevel,df$MonthlyIncome,method="spearman")
[1] 0.9204287
> cor(df$JobLevel,df$MonthlyIncome,method="pearson")
[1] 0.9502999
> cor(df$PerformanceRating,df$PercentSalaryHike,method="pearson")
[1] 0.77355
```

#Scatter plot

```
> library(ggplot2)
> df<-df
> ggplot(df)+
+     aes(x=JobLevel,y=MonthlyIncome)+
+     geom_point(color="#e9d66b")+
+     theme_dark()
```



#Regression Analysis

- Simple linear regression is **the most straight forward case having a single scalar predictor variable x and a single scalar response variable y.** The equation for this regression is given as

y=a+bx The expansion to multiple and vector-valued predictor variables is known as multiple linear regression. It is also known as multivariable linear regression. The equation for this regression is given as Y = a+bX.

- *The following example shows how to use this function in R to do the following:*

- Fit a regression model
- View the summary of the regression model fit
- View the diagnostic plots for the model
- Plot the fitted regression model
- Make predictions using the regression model

```
> Model = lm(JobLevel~MonthlyIncome,data=df) #Create the linear
 regression
> print(Model)

Call:
lm(formula = JobLevel ~ MonthlyIncome, data = df)

Coefficients:
  (Intercept)  MonthlyIncome
    0.6109596      0.0002234

> linear equation: 0.6109596+0.0002234*x
```

The ***lm()*** function in R is used to fit linear regression models.

This function uses the following basic syntax:

lm(formula, data, …)

where:

- **formula:** The formula for the linear model (e.g. y ~ x1 + x2)
- **data:** The name of the data frame that contains the data

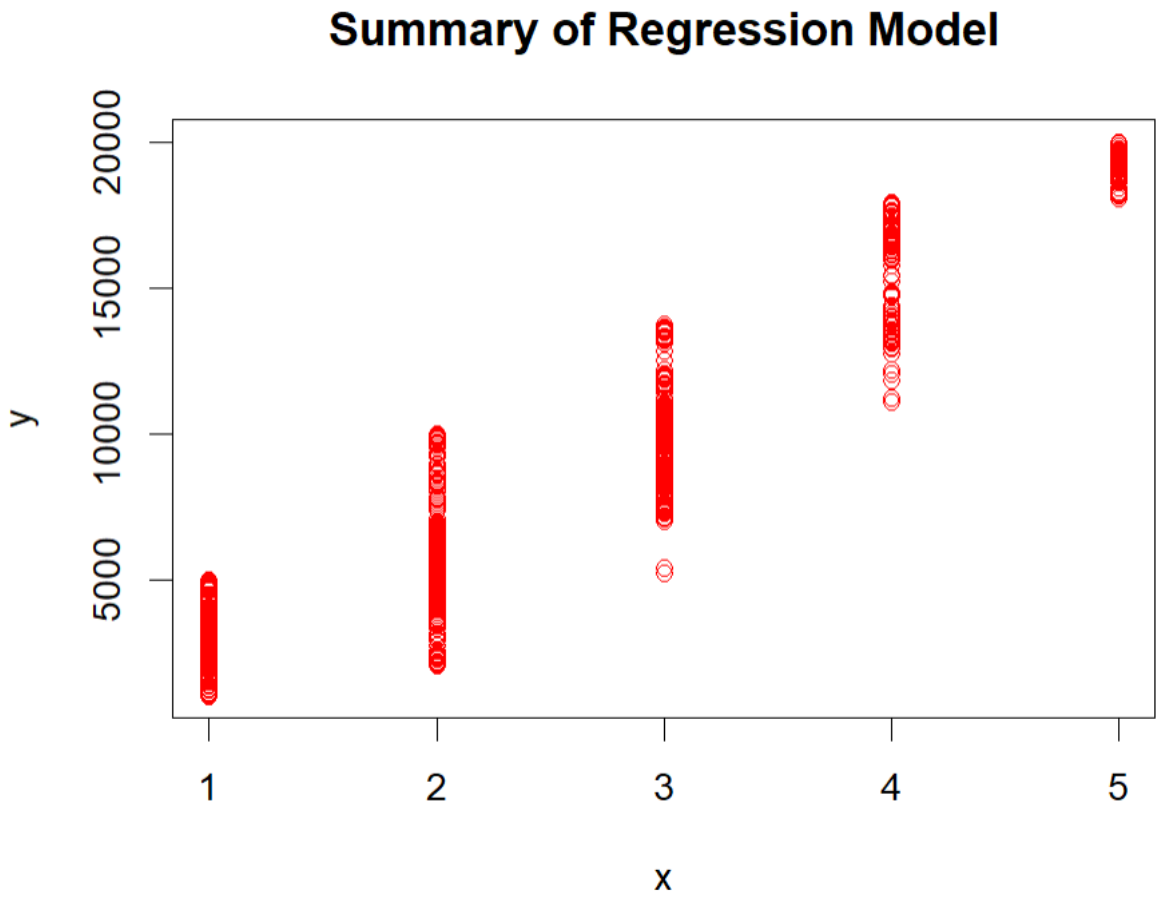
We can then use the **summary ()** function to view the summary of the regression model fit:

```
> summary(Model)

Call:
lm(formula = JobLevel ~ MonthlyIncome, data = df)

Residuals:
     Min       1Q   Median       3Q      Max
-0.84487 -0.22602 -0.04835  0.23796  1.22494

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)   6.110e-01  1.534e-02   39.84   <2e-16 ***
MonthlyIncome 2.234e-04  1.911e-06  116.95   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.3447 on 1468 degrees of freedom
Multiple R-squared:  0.9031,    Adjusted R-squared:  0.903
F-statistic: 1.368e+04 on 1 and 1468 DF,  p-value: < 2.2e-16
```

#Plot the Fitted Regression Model

We can use the **abline()** function to plot the fitted regression model:

```
> #create scatterplot of raw data
> plot(df$JobLevel, df$MonthlyIncome, col='red', main='Summary o
f Regression Model', xlab='x', ylab='y')
>
> #add fitted regression line
> abline(Model)
>
```



#Use the Regression Model to Make Predictions

We can use the **predict()** function to predict the response value for a new observation:

```
> #define new observation
> new <- data.frame(x=c(5))
>
> #use the fitted model to predict the value for the new observa
tion
> predict(Model, newdata = df)
        1         2         3         4         5         6
1.9500088 1.7571839 1.0779399 1.2609336 1.3858341 1.2964598
        7         8         9        10        11        12
1.2075325 1.2126715 2.7394066 1.7810915 1.1530142 1.5478248
       13        14        15        16        17        18
1.2613805 1.2055216 1.0640869 2.8408463 1.3478500 1.2667429
       19        20        21        22        23        24
4.0578997 1.4921894 1.5071596 1.3722045 3.2908455 0.8862322
       25        26        27        28        29        30
1.2723288 4.8772379 1.4866035 2.1359072 2.9007270 4.8443929
       31        32        33        34        35        36
1.1686547 2.0554704 1.1038584 1.0770461 1.1232973 1.2019466
       37        38        39        40        41        42
1.2104371 1.0609588 1.3748857 1.8121491 1.0468823 1.1340222
       43        44        45        46        47        48
1.1232973 2.5606581 1.5071596 4.9780073 1.6316132 1.2861818
```

3D PLOT

Performance rating and PercentSalaryHike

```
> data_frame<-data.frame(df$PerformanceRating,df$PercentSalaryHike)
> mean(df$PerformanceRating)
[1] 3.153741
> mean(df$PercentSalaryHike)
[1] 15.20952
```

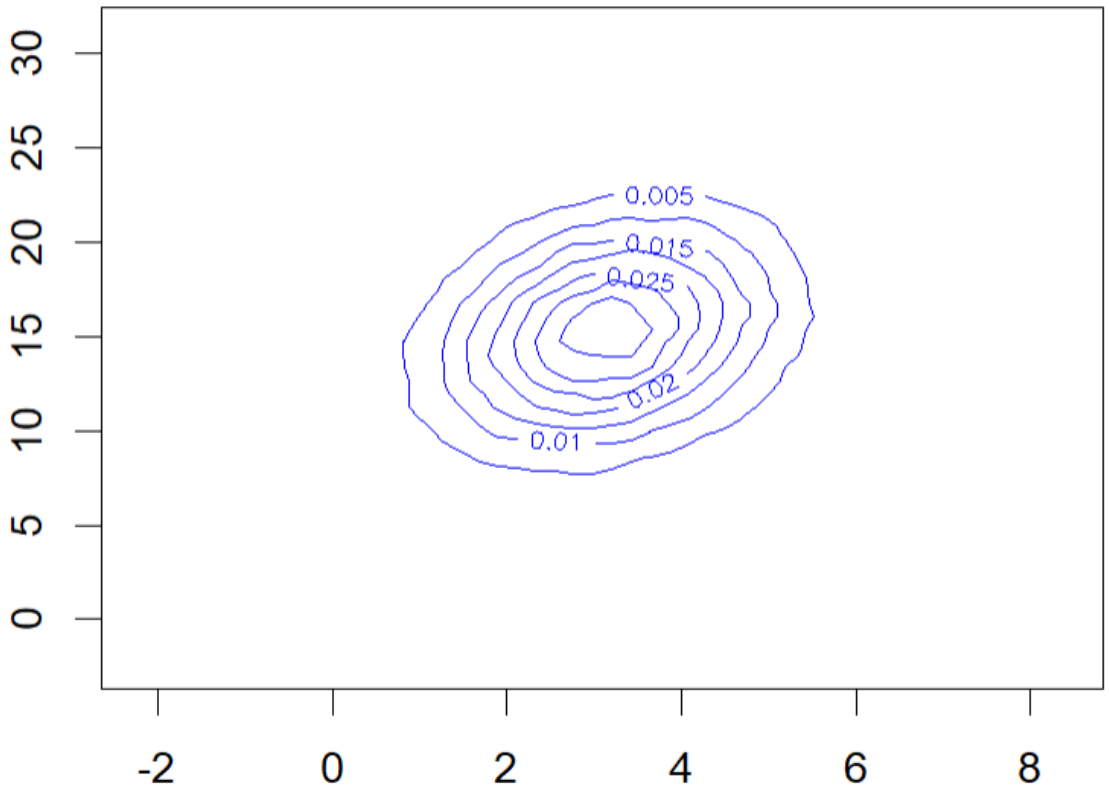
```
> cov(data_frame)
                        df.PerformanceRating df.PercentSalaryHike
df.PerformanceRating               0.1301936              1.021544
df.PercentSalaryHike               1.0215436             13.395144
>
```

> library (MASS)

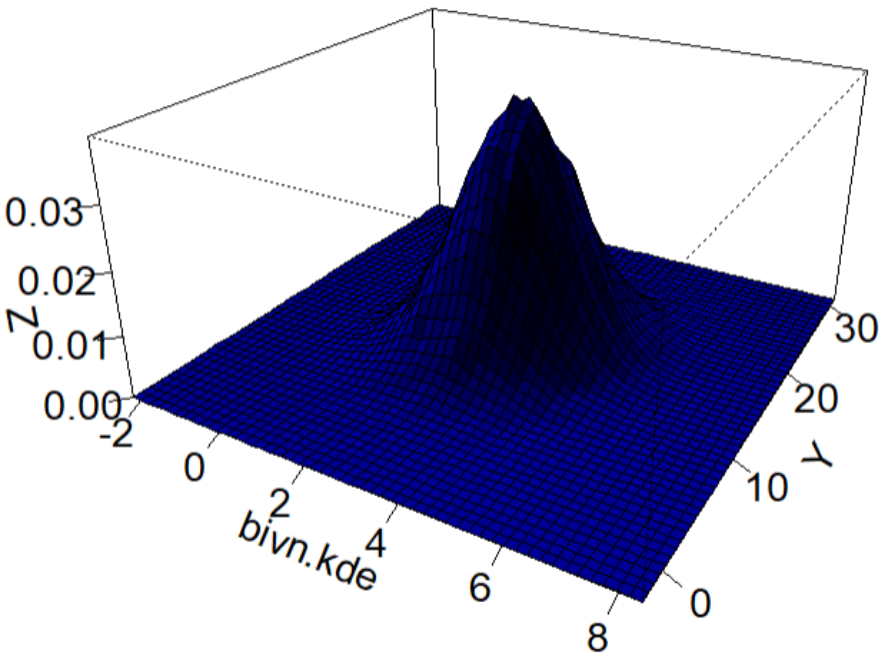
> mu1<-c (3.153,15.209)

> sigma1<-matrix(c(1.301936,1.021544,1.021544,13.395144),ncol=2)

```
> bivn<-mvrnorm(100000,mu=mu1,Sigma = sigma1)
> head(bivn)
         [,1]     [,2]
[1,] 1.317991 18.24967
[2,] 3.990962 16.13695
[3,] 3.107126 14.64036
[4,] 3.159632 12.17732
[5,] 3.460476 16.63745
[6,] 2.900196 10.93986
>
> bivn.kde<-kde2d(bivn[,1],bivn[,2],n=50)
> contour(bivn.kde,col="blue")
```




```
> persp(bivn.kde,theta=30,phi=25,
+       shade=0.75,col="blue",expand=0.5,r=2,
+       ticktype="detailed")
```



Confidence Intervals-

Known variance-

Estimate the two-sided confidence interval of the mean of Normal distribution with known variance from a given sample and α.

> shapiro.test(df$ï..Age)

```
> shapiro.test(df$ï..Age)

        Shapiro-Wilk normality test

data:  df$ï..Age
W = 0.97745, p-value = 2.037e-14
```

Ho: Data is normally distributed

H1: Data is not normally distributed

p>alpha

so we accept the null hypothesis that data is normally distributed.

The lower bound on W is actually determined by the size of the sample. Normally distributed samples will result in a high value of W and samples deviating away from a normal distribution will have a lower value of W.

Now we will find CI:

First we will find standard deviation:-

```
> sd(df$MonthlyIncome)
[1] 4707.957
```

```
normCI_TS = function(n,sd,alpha){

mu=mean(n)

len1=length(n)

z1=qnorm(1-(alpha/2),mean=0,sd=1)

f1=(mu-(z1*(sd/sqrt(len1))))

f2=(mu+(z1*(sd/sqrt(len1))))

f=c(f1,f2)

print("Two Sided Confidence Interval is:")

return(f)

}
```

```
> n=df$ï..Age
```

```
> sd(df$ï..Age)
[1] 9.135373
> normCI_TS = function(n,sd,alpha){
+     mu=mean(n)
+     len1=length(n)
+     z1=qnorm(1-(alpha/2),mean=0,sd=1)
+     f1=(mu-(z1*(sd/sqrt(len1))))
+     f2=(mu+(z1*(sd/sqrt(len1))))
+     f=c(f1,f2)
+     print("Two Sided Confidence Interval is:")
+               return(f)
+ }
> normCI_TS(n,9.135373,0.05)
[1] "Two Sided Confidence Interval is:"
[1] 36.45681 37.39081
>
```

#Upper confidence interval of the mean of Normal distribution with known variance from a given sample and α

```
> normCI_UpperCI = function(n,sd,alpha){
+     mu=mean(n)
+     len1=length(n)
+     z1=qnorm(1-(alpha),mean=0,sd=1)
+     f1=(mu-(z1*(sd/sqrt(len1))))
+     f2=qnorm(1,0,1)
+     f=c(f1,f2)
+     print("Upper Confidence Interval is:")
+               return(f)
+ }
> normCI_UpperCI(n,9.135373,0.05)
[1] "Upper Confidence Interval is:"
[1] 36.53189      Inf
>
```

#Estimate the lower confidence interval of the mean of

Normal distribution with known variance from a given sample and α.

```
> normCI_LowerCI = function(n,sd,alpha){
+     mu=mean(n)
+     len1=length(n)
+     z1=qnorm((1-alpha),mean=0,sd=1)
+     f1=qnorm(0,0,1)
+     f2=(mu+(z1*(sd/sqrt(len1))))
+     f=c(f1,f2)
+     print("Lower Confidence Interval is:")
+               return(f)
+ }
> normCI_LowerCI(n,9.135373,0.05)
[1] "Lower Confidence Interval is:"
[1]     -Inf 37.31573
>
```

Coming to Hypothesis testing…

#To test a null hypothesis μ = m and alternative hypothesis μ ≠ m.

```
> null_hypothesis<-function(samp,s,mu,alpha)
+ {
+     xbar<-mean(samp)
+     len1<-length(samp)
+     z_stat<-qnorm(1-(alpha/2),mean=0,sd=1)
+     z1<-(xbar-mu)*sqrt(len1)/s
+     if(abs(z1)<=z_stat){
+         print("Hyptothesis is in acceptance region")
+     }else{
+         print("Hypothesis is in rejectance region")
+     }
+ }
>
> null_hypothesis(samp,9.135373,45,0.05)
[1] "Hypothesis is in rejectance region"
>
```

Inference:

Actual mean was around 36.01 and so we have rejected the hypothesis at 95% ci that mean i.e. u=45.

#Pearsons's correlation

By Pearson's correlation test we can see they are highly positive correlation.

Ho=There is no relationship between the two variables.

H1=There is a relationship between the two variables.

```
> cor.test(df$Education,df$MonthlyIncome)

        Pearson's product-moment correlation

data:  df$Education and df$MonthlyIncome
t = 3.6549, df = 1468, p-value = 0.0002664
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.04404707 0.14538229
sample estimates:
       cor
0.09496068

>
```

t-statistic and p-value?

t-statistic = 3.6549

p- value= 0.0002664

Inference->

p-value<alpha(0.05) so we reject the null hypothesis at 95% confidence interval and thus we say that there is a relationship between those two variables.

Top Reasons why Employees leave the Organization:

- **No Overtime** This was a surprise, employees who don't have overtime are most likely to leave the organization. This could be that employees would like to have a higher amount of income or employees could feel that they are underused.
- **Monthly Income:** As expected, Income is a huge factor as why employees leave the organization in search for a better salary.
- **Age:**
  This could also be expected, since people who are aiming to retire will leave the organization.

Knowing the most likely reasons why employees leave the organization, can help the organization take action and reduce the level of Attrition inside the organization.

THANK YOU!