Data Visualisation 1

DATA SCIENCE
INSTITUTE

What will we learn

- Grammar of Graphics
- Plotting Systems in R
- What is ggplot2?
- Bar Plot
- Pie Chart
- Box-Whisker Plot
- Histogram

About Data Visualization

What is Data Visualisation?

It is the visual representation of data generally in the form of graphs and plots

Why is it important?

It enables us to

- See the data and get insights in one glance
- Allows us to grasp difficult/ complex data in an easy manner
- Helps us to identify patterns or trends easily

Principles of Visualization

- Show distribution (overall and by groups)
- Show correlation and causality
- Show multivariate data; real world is complex
- Integration of evidence
- Describe and document evidence with appropriate labels, sources, scales, etc.
- Content is King

What is ggplot2?

- An implementation of Grammar of Graphics by Leland Wilkinson
- Written by Hadley Wickham (while he was a graduate student of Iowa State)
- A “third” graphics system for R (along with Base and Lattice)
- Available from CRAN via
  - install.packages("ggplot2")
  - library(ggplot2)
- Website: http://ggplot2.org (better documentation)
- # R base package and lattice package also provide rich graphics

What is ggplot2?

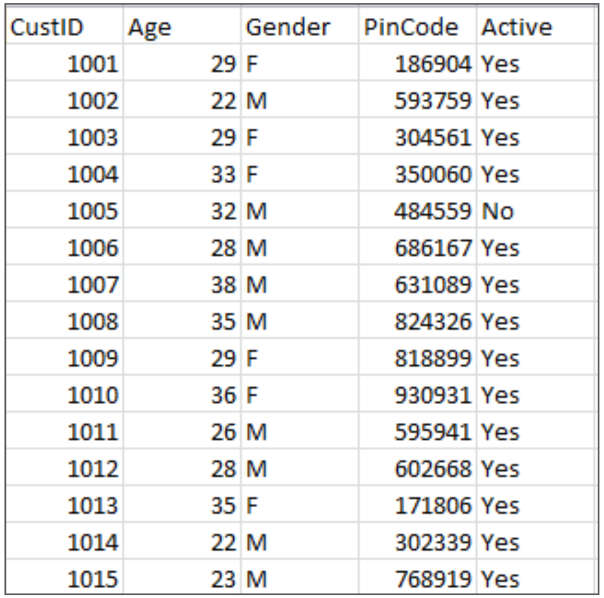
- Grammar of Graphics represents an abstraction of graphics ideas/objects
- Think “verb”, “noun”, “adjective” for graphics
- Allows for a “theory” of graphics on which to build new graphics and graphics objects
- Shorten the distance from mind to page
- Plots are made up of aesthetics (size, shape, color) and geoms (points, lines)

Import Telecom Data Sets

```
    #Import two data Sets

➢ demographic<-read.csv(file.choose(), header=TRUE)
  head(demographic)


➢ transaction<-read.csv(file.choose(), header=TRUE)

     head(transaction)
```

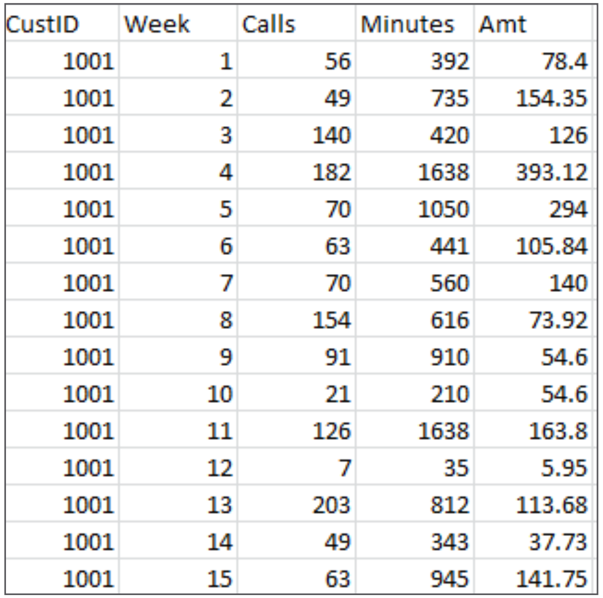
Data Snapshots

Demographic



CustID	Age	Gender	PinCode	Active
1001	29	F	186904	Yes
1002	22	M	593759	Yes
1003	29	F	304561	Yes
1004	33	F	350060	Yes
1005	32	M	484559	No
1006	28	M	686167	Yes
1007	38	M	631089	Yes
1008	35	M	824326	Yes
1009	29	F	818899	Yes
1010	36	F	930931	Yes
1011	26	M	595941	Yes
1012	28	M	602668	Yes
1013	35	F	171806	Yes
1014	22	M	302339	Yes
1015	23	M	768919	Yes

Transactions



CustID	Week	Calls	Minutes	Amt
1001	1	56	392	78.4
1001	2	49	735	154.35
1001	3	140	420	126
1001	4	182	1638	393.12
1001	5	70	1050	294
1001	6	63	441	105.84
1001	7	70	560	140
1001	8	154	616	73.92
1001	9	91	910	54.6
1001	10	21	210	54.6
1001	11	126	1638	163.8
1001	12	7	35	5.95
1001	13	203	812	113.68
1001	14	49	343	37.73
1001	15	63	945	141.75

Aggregate and Merge

#Aggregating and Merging

```
tcalls<-aggregate(Calls~CustID, data=transaction, FUN=sum)
   head(tcalls)
```

```
working<-merge(demographic, tcalls, by=("CustID"), all=TRUE)
   head(working)
```

```
working$age_group<-cut(working$Age, breaks=c(0,30,45,Inf), labels=c("18-30","30-45",">45"))
  head(working)
```

Simple Bar Chart

A **Bar Chart** is the simplest and basic form of graph.

In this graph, for each data item, we simply draw a ‘bar’ showing its value

Simple Bar Chart: It is a type of chart which shows the values of different categories of data as rectangular bars with different lengths.
The values are generally :

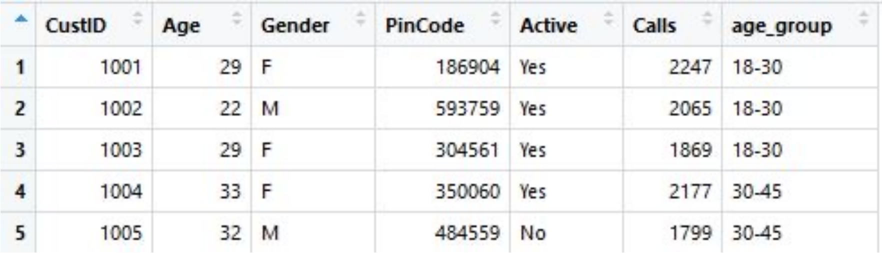
- Frequency
- Mean
- Totals
- Percentages

Simple Bar Chart

# Simple bar chart of count of customers by age group

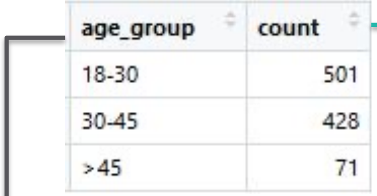
- **ggplot(working , aes (x = age_group)) + geom_bar()**

`ggplot()`	*is a function in ggplot2 which yields different types of plots*
`working`	*is the data to be used*
`aes()`	*specifies the variables to be used on each axis*
`geom_bar()`	*is used to call shapes and colors*

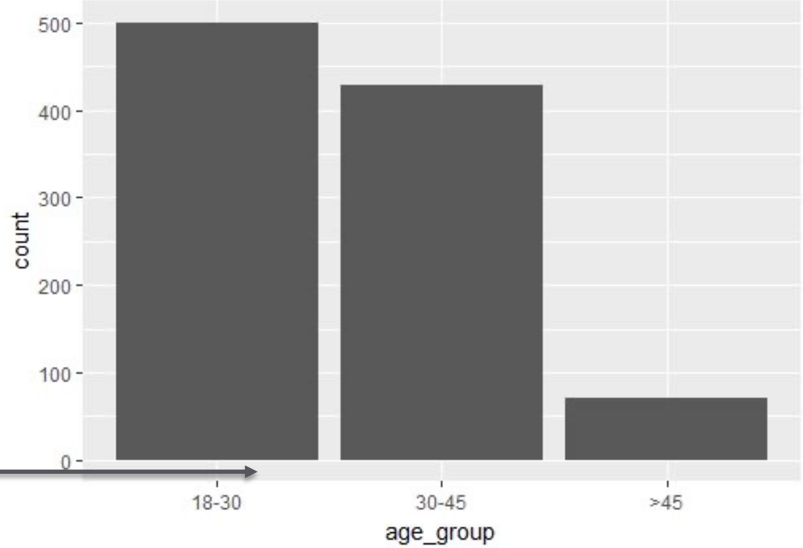


	CustID	Age	Gender	PinCode	Active	Calls	age_group
1	1001	29	F	186904	Yes	2247	18-30
2	1002	22	M	593759	Yes	2065	18-30
3	1003	29	F	304561	Yes	1869	18-30
4	1004	33	F	350060	Yes	2177	30-45
5	1005	32	M	484559	No	1799	30-45

geom_bar() transforms the data with count stat which returns a data set of age_group values and count

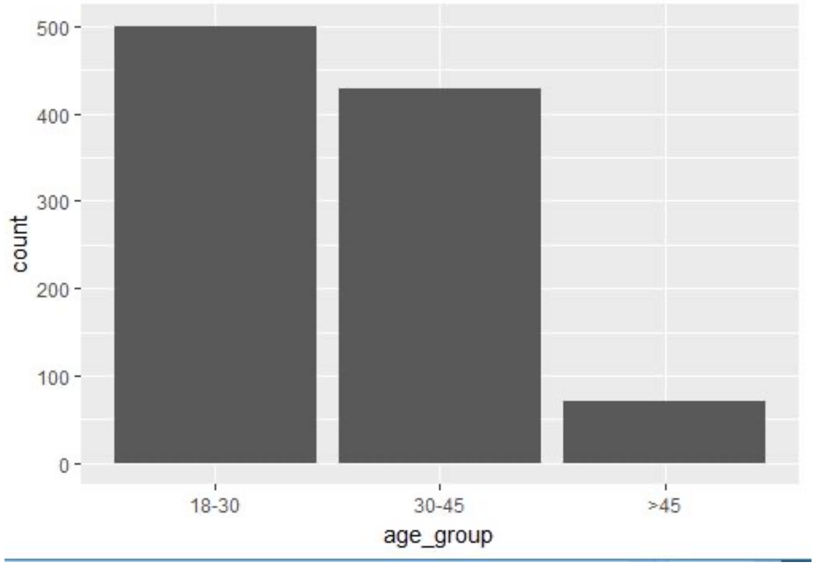


age_group	count
18-30	501
30-45	428
>45	71



Simple Bar Chart

- ggplot(working , aes (x = age_group)) + geom_bar()



Simple Bar Chart..

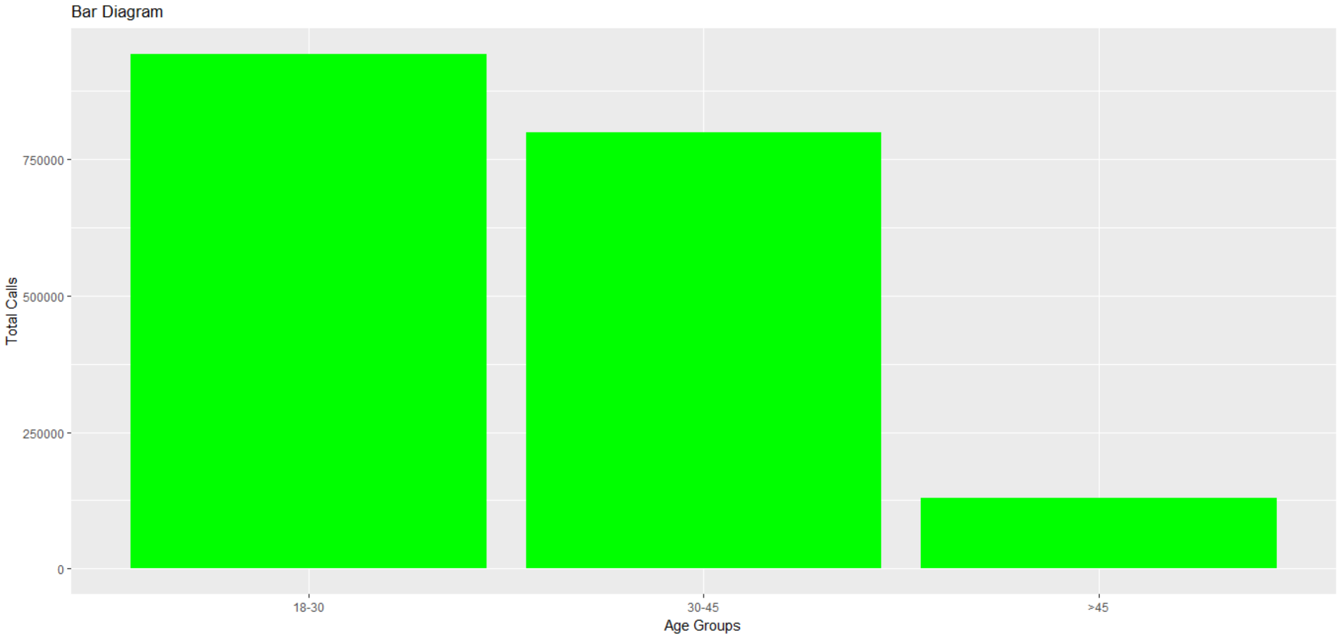
#Bar chart showing age groups on x axis and total calls on y axis

```
ggplot(working,aes(x=age_group,y=Calls))+
          geom_bar(stat="identity",fill="green")+
          labs(x="Age Groups",y="Total Calls",title="Bar Diagram")
```

`ggplot()`	*is a function in ggplot2 which yields different types of plots*
`working`	*is the data to be used*
`aes()`	*specifies the variables to be used on each axis*
`geom_bar()`	*is used to call shapes and colors*
`stat="identity"`	*using the height of the bar will represent the values in a column of the data frame*
`labs()`	*Used to give labels/titles*

Simple Bar Chart
R Output

#Bar chart showing total calls for customers in each age group



Simple Bar Chart
Change Order of the Bars

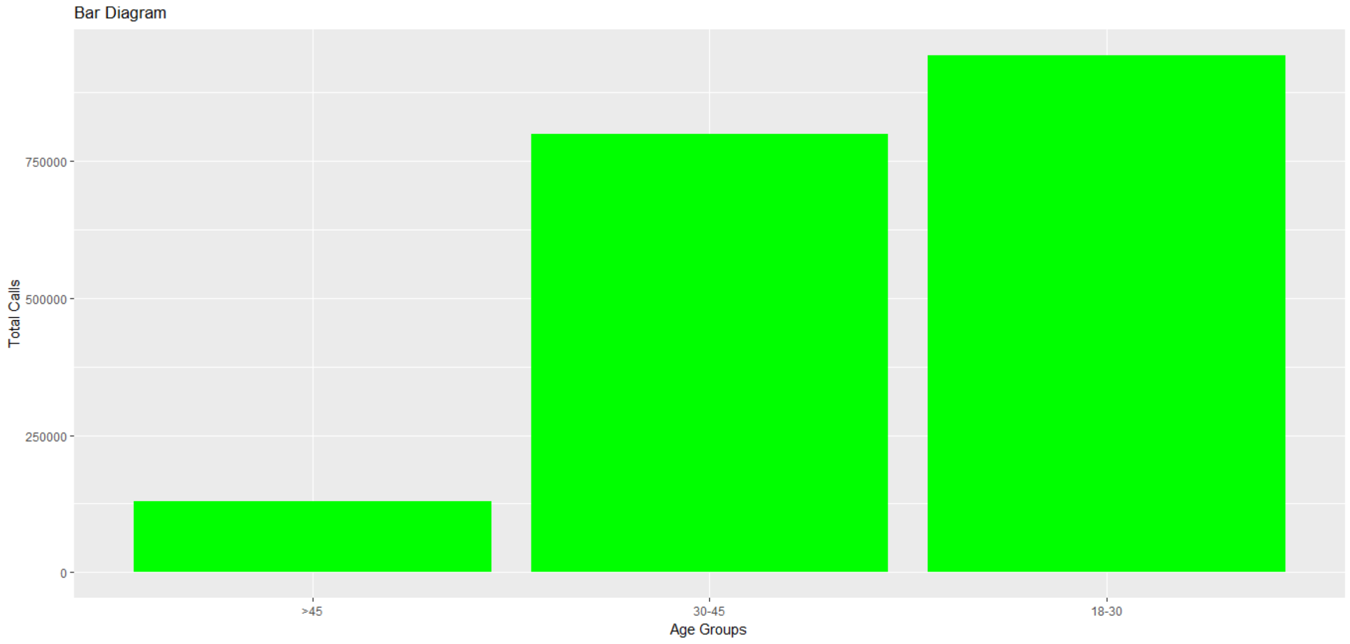
# Order bars as per value

```
ggplot(working,aes(reorder(age_group,Calls),Calls))+
          geom_bar(stat="identity",fill="green")+
          labs(x="Age Groups",y="Total Calls", title="Bar Diagram")
```

reorder() orders levels of a factor variable (First argument) by values of a second variable, usually numeric (Second argument)

Simple Bar Chart
R Output

#Bar chart (ordered by value)



Simple Bar Chart
Horizontal View

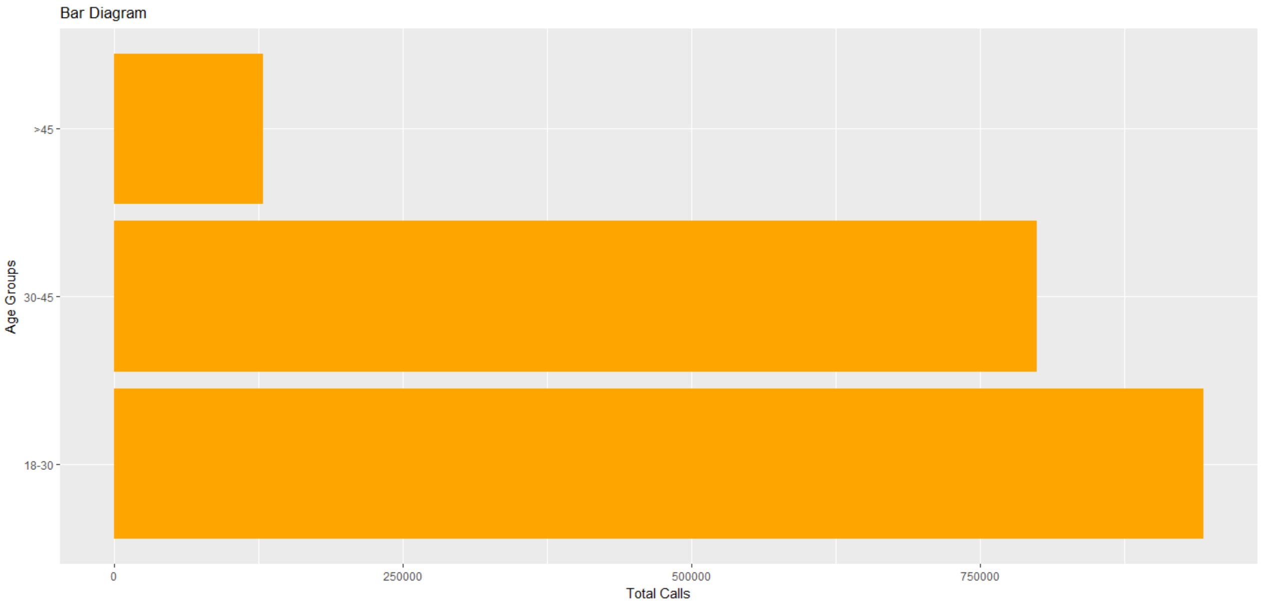
#Make bar graph horizontally oriented

```
ggplot(working,aes(x=age_group,y=Calls))+
        geom_bar(stat="identity",fill="orange")+
        labs(x="Age Groups",y="Total Calls",title="Bar Diagram")+
        coord_flip()
```

# coord_flip makes bar graph horizontal

Simple Bar Chart
R Output

#Horizontal Bar Plot by Age Group



Stacked Bar Chart

#Stack the plot with Gender

- **ggplot(working, aes(x=age_group))+**

  geom_bar(aes(fill=Gender))+

  labs(x = "Age Group", y="No. of customers", title="Stacked bar chart")

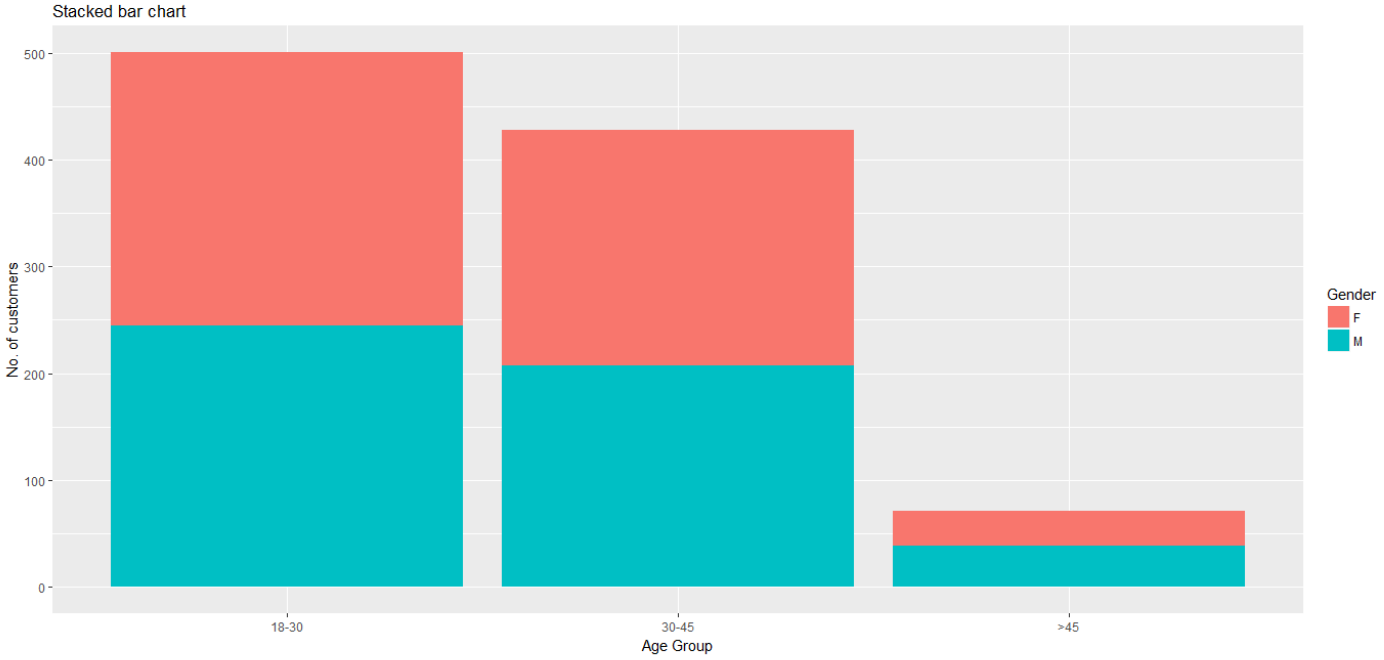
#Normalized Bars

Add position="fill" to geom_bar() to produce stacked bar with normalized height

geom_bar(aes(fill=Gender),position="fill")

Stacked Bar Chart
R Output

#Stack the plot with Gender



DATA SCIENCE INSTITUTE

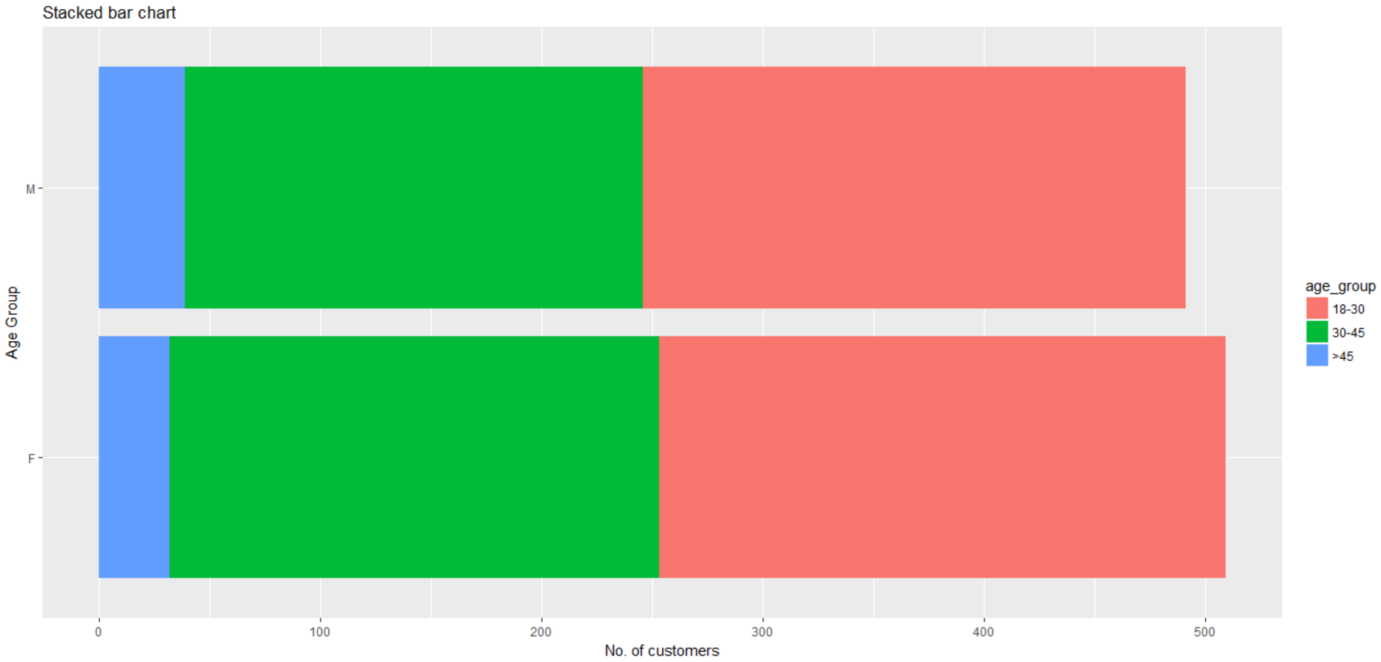
Stacked Bar Chart
Horizontal View

#Try with flipping the coordinate and stacking with age group on Gender

```
➢ ggplot(working, aes(x=Gender))+
           geom_bar(aes(fill=age_group))+
           labs(x="Age Group", y="No. of customers", title="Stacked bar chart")+
           coord_flip()
```

Stacked Bar Chart
R Output

#Try with flipping the coordinate and stacking with age group on Gender

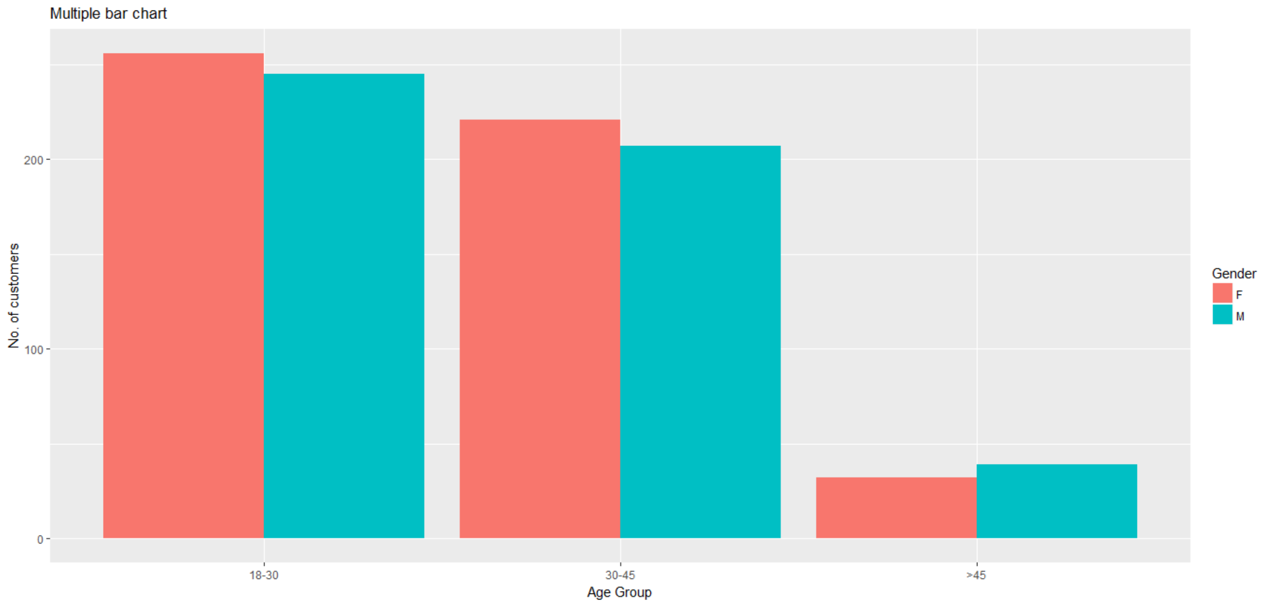


Multiple Bar Chart

```
#Multiple bars, side by side

ggplot(working, aes(x=age_group))+
        geom_bar(aes(fill=Gender), position="dodge")+
        labs(x="Age Group", y="No. of customers", title="Multiple bar chart")
```

Multiple Bar Chart
R Output

#Multiple bars, side by side



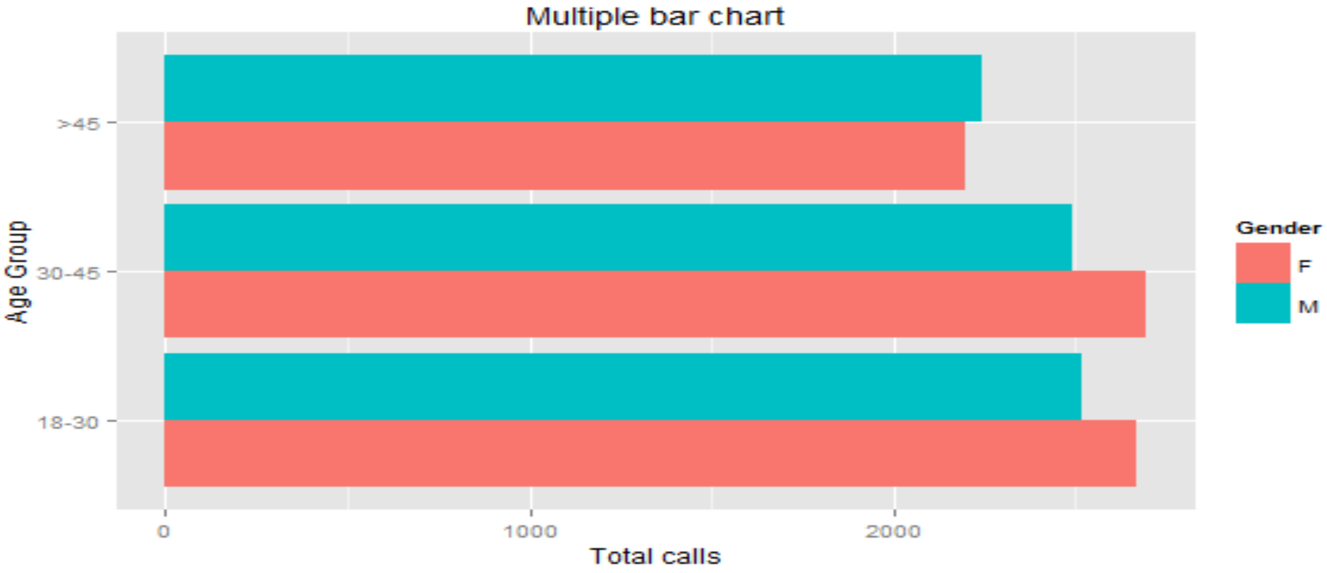
Multiple Bar Chart
Horizontal View

#Try with total calls on y-axis and flipped coordinate

```
ggplot(working, aes(x=age_group,y=Calls))+
        geom_bar(stat="identity",aes(fill=Gender), position="dodge")+
        labs(x="Age Group", y="Total calls", title="Multiple bar          chart")+
        coord_flip()
```

Multiple Bar Chart
R Output

#Try with total calls on y-axis and flipped coordinate



Box Plot

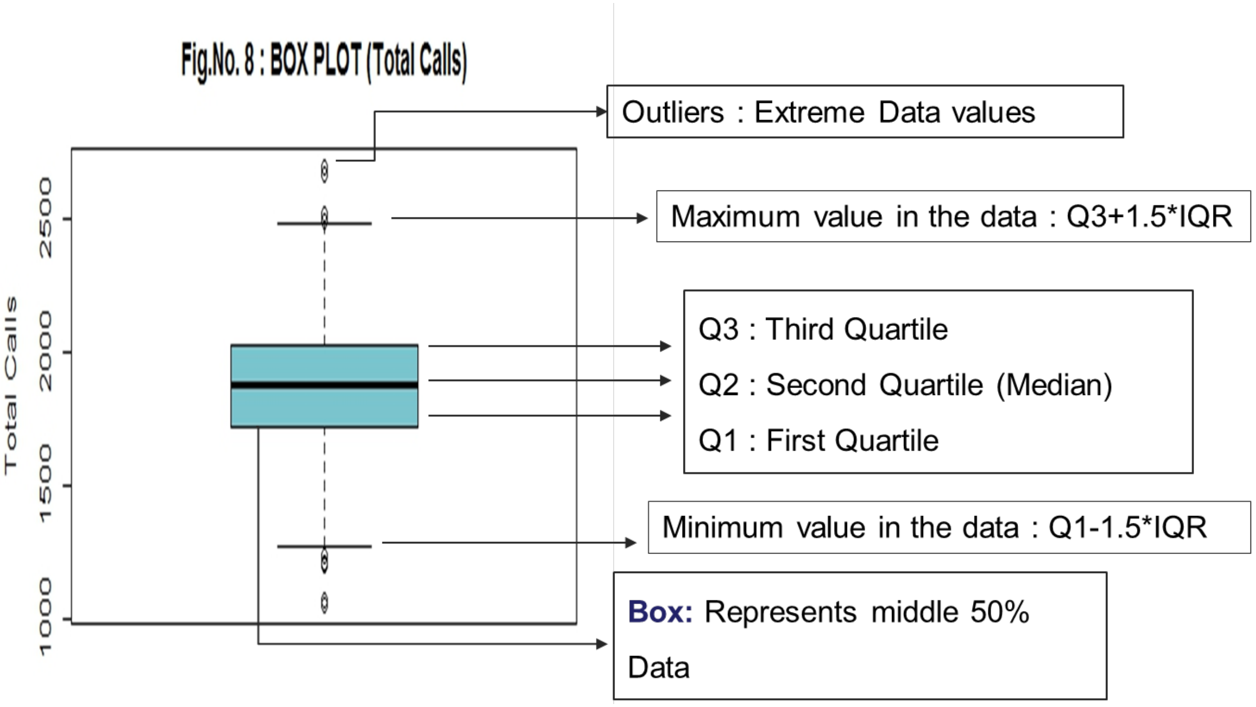
Box and Whisker plot summarizes data graphically using 5 measures:

- Minimum
- The Three Quartiles : Q1, Q2 (i.e. Median) and Q3
- Maximum.

Advantages of a Box Plot :

- A boxplot is particularly effective when comparing two sets of data
- It shows us the shape of the data

Box Plot



Outliers : Extreme Data values

Maximum value in the data : Q3+1.5*IQR

Q3 : Third Quartile

Q2 : Second Quartile (Median)

Q1 : First Quartile

Minimum value in the data : Q1-1.5*IQR

Box: Represents middle 50% Data

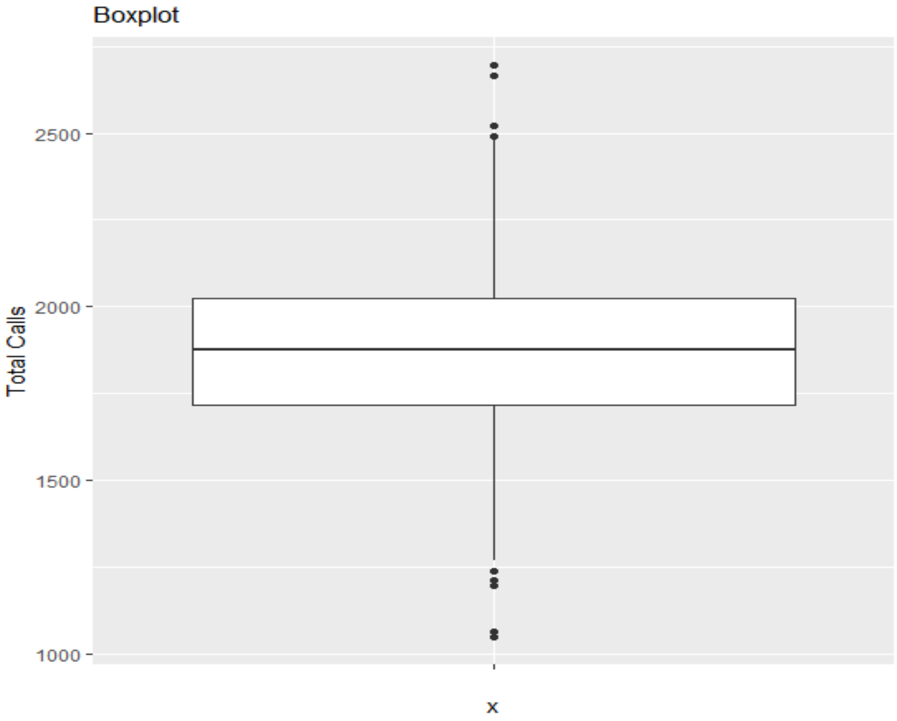
This plot shows that the distribution of total call is very much symmetric

Box Plot

```
# Box plot for variable ‘Calls’

ggplot(working, aes(x="", y=Calls))+
            geom_boxplot()+
            labs( y="Total Calls", title="Boxplot")
```

Box Plot
R Output

# Box plot for variable ‘Calls’

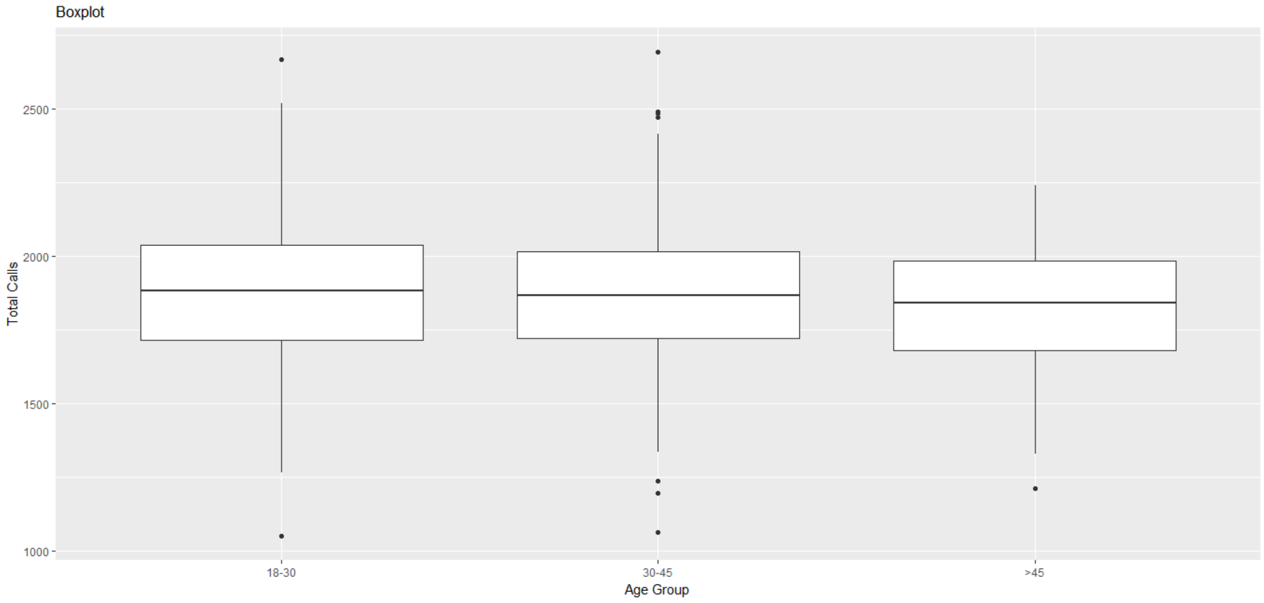


Box Plot
By Age Group

```
# Box plot by Age group

➢ ggplot(working, aes(x=age_group, y=Calls))+
          geom_boxplot()+
          labs(x="Age Group", y="Total Calls", title="Boxplot")
```

Box Plot
By Age Group R Output

- # Box plot by Age group



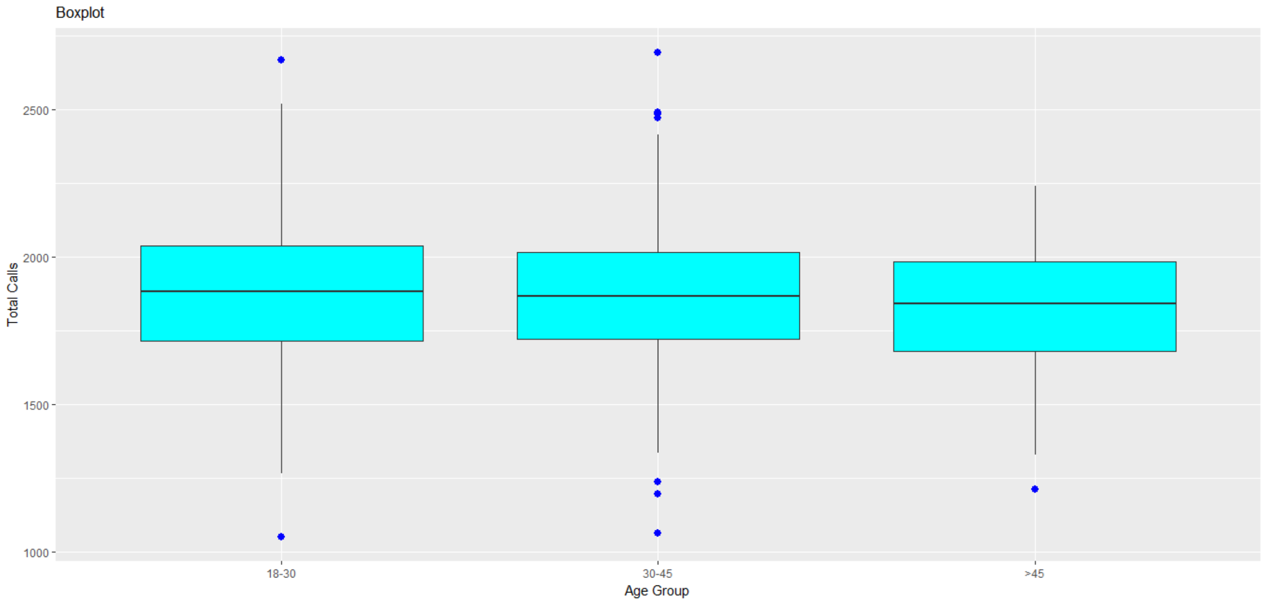
Box Plot By Age Group
Enhance the plot

#Box plot by Age group; colour the boxes & outlier

```
ggplot(working, aes(x=age_group, y=Calls))+
        geom_boxplot(fill=5, outlier.colour="blue", outlier.size=2.5)+
        labs(x="Age Group", y="Total Calls", title="Boxplot")
```

Box Plot By Age Group
Enhance the plot R Output

#By Age group; colour the boxes & outlier



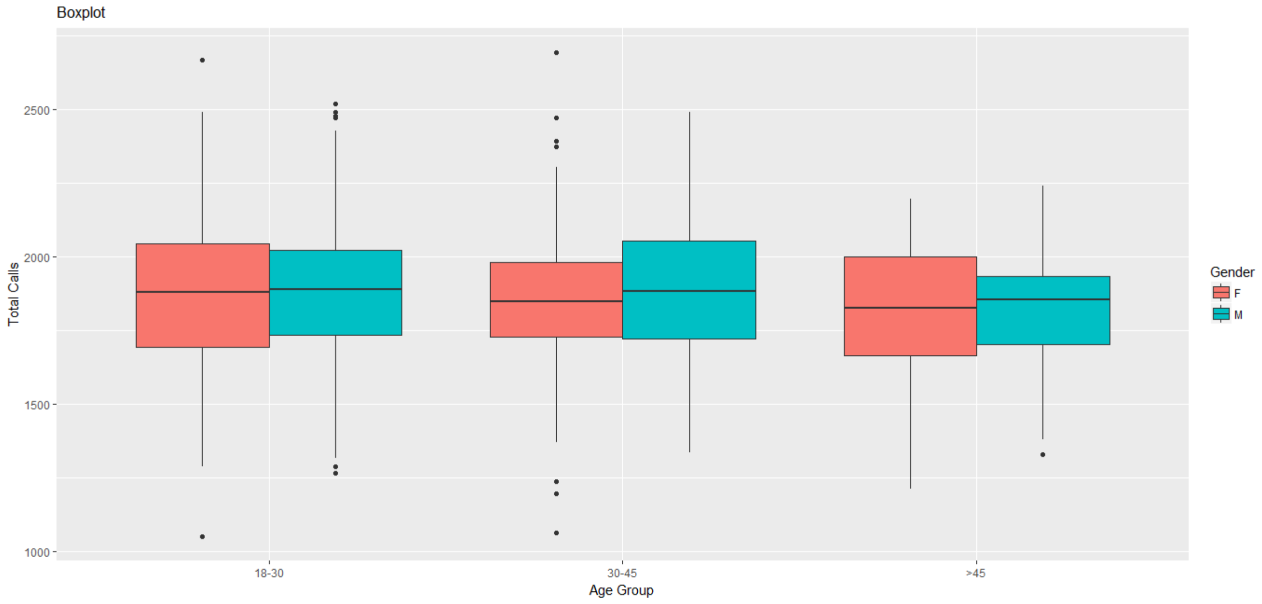
Box Plot
# Adding Gender Facet

#Adding Gender Facet

```
➢ ggplot(working, aes(x=age_group, y=Calls))+
        geom_boxplot(aes(fill=Gender))+
        labs(x="Age Group", y="Total Calls", title="Boxplot")
```

Box Plot
# Adding Gender Facet R Output

#Adding Gender Facet



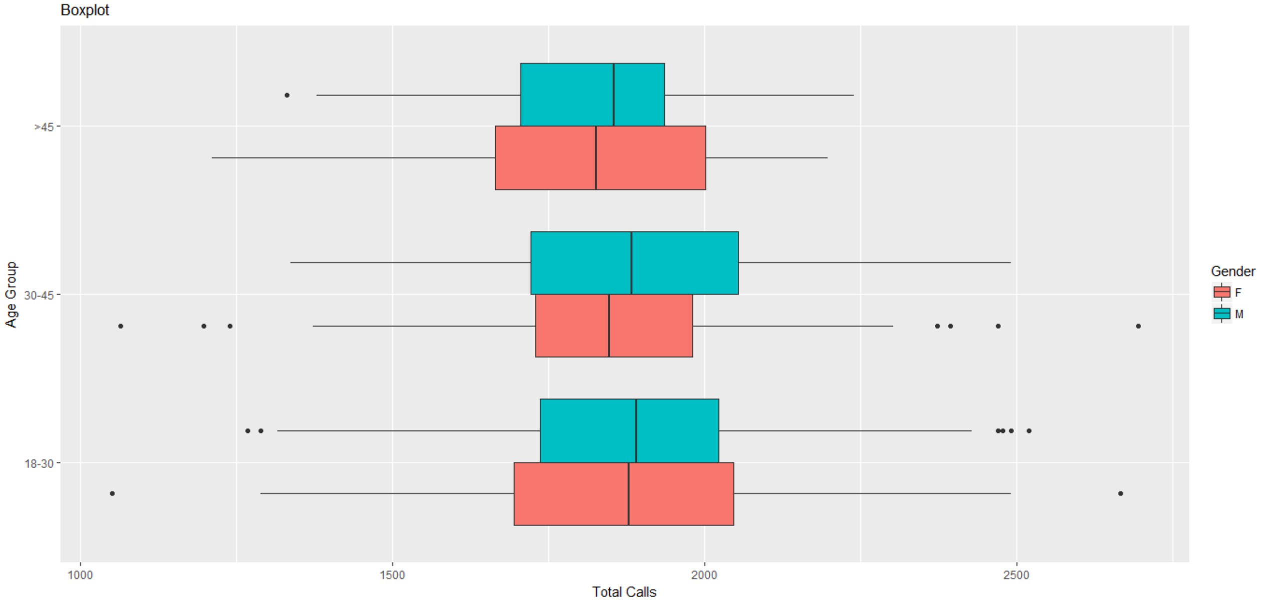
DATA SCIENCE INSTITUTE

Box Plot
Horizontal View

```
# Box plot:Horizontal View

ggplot(working, aes(x=age_group, y=Calls))+
        geom_boxplot(aes(fill=Gender))+
        coord_flip()+
        labs( y= “Total Calls", x="Age Group", title="Boxplot")
```

Box Plot
Horizontal View: R Output

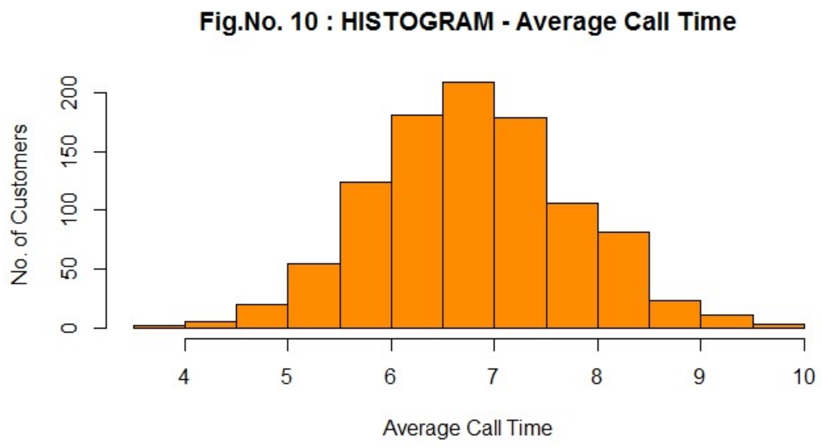
# Box plot: Horizontal View



Histogram

A **Histogram** is similar to a bar chart but is used to display continuous data. Therefore we will use a continuous scale with no 'gaps' between the bars.

It is generally used to check the Normality of the data.



- ***This plot shows that the distribution of Average Call Time is very much symmetric***

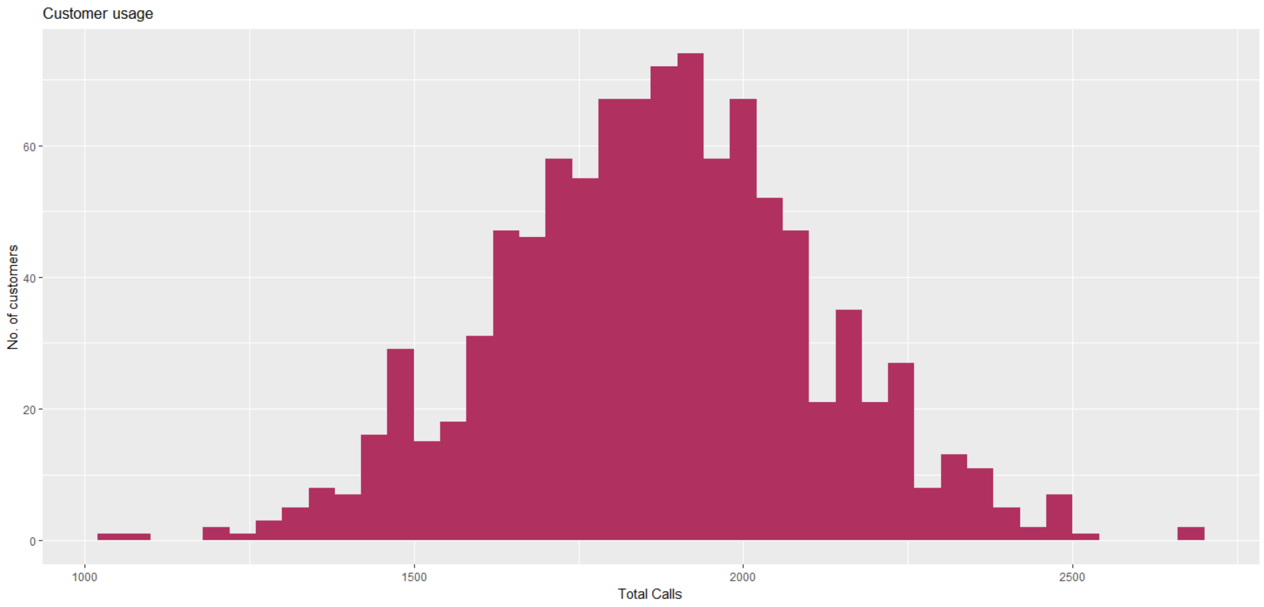
Histogram

#Histogram for variable ‘Calls’

```
ggplot(working, aes(x=Calls))+
          geom_histogram(binwidth=40, fill="maroon")+
          labs(x=“Total Calls", y="No. of customers", title="Customer usage")
```

# Histogram
R Output

#Histogram for variable 'Calls'



Data Visualisation 2

DATA SCIENCE
INSTITUTE

What will we learn

- Scatterplot with regression line
- Bubble chart
- Scatterplot matrix (GGally)

Data Import

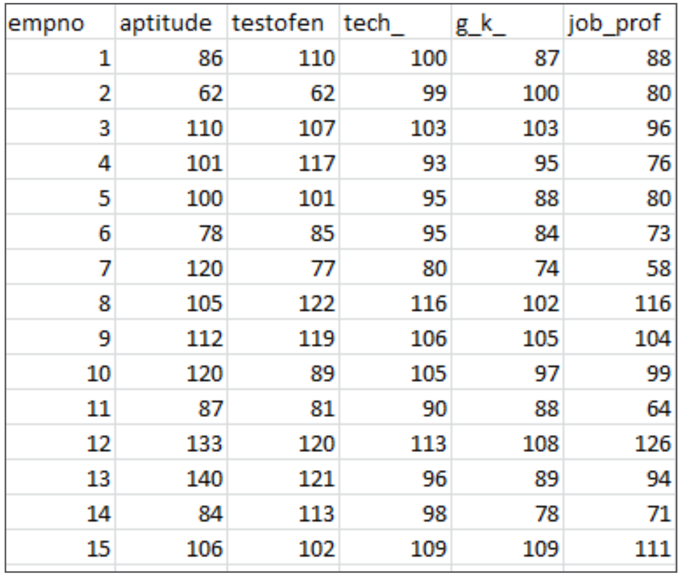
- Import

  ➢job<-read.csv(file.choose(), header=TRUE)

  ```
  head(job)
  ```

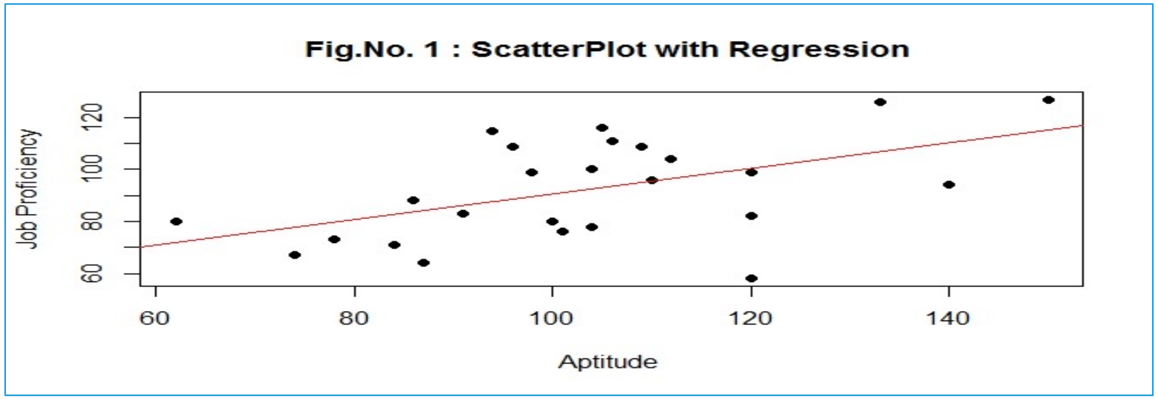
Data Snapshot

Job Proficiency Data



empno	aptitude	testofen	tech_	g_k_	job_prof
1	86	110	100	87	88
2	62	62	99	100	80
3	110	107	103	103	96
4	101	117	93	95	76
5	100	101	95	88	80
6	78	85	95	84	73
7	120	77	80	74	58
8	105	122	116	102	116
9	112	119	106	105	104
10	120	89	105	97	99
11	87	81	90	88	64
12	133	120	113	108	126
13	140	121	96	89	94
14	84	113	98	78	71
15	106	102	109	109	111

Scatterplot with regression line



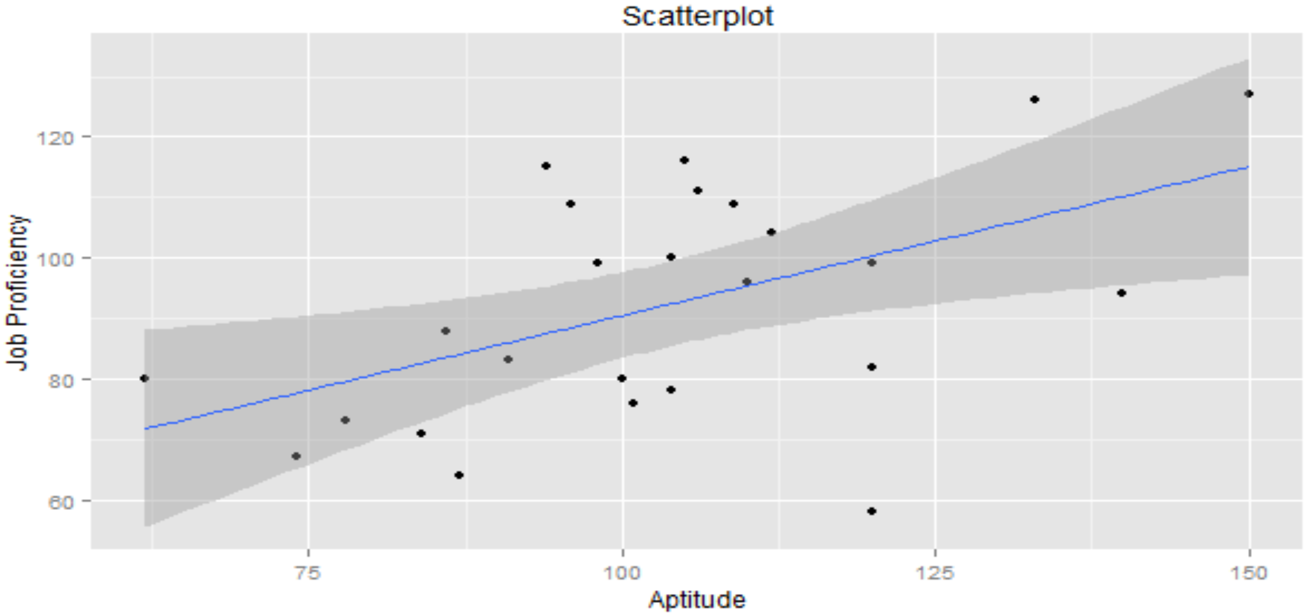
We can observe here that as the aptitude score increases job proficiency also increases.

Also for a given aptitude score, the job proficiency can be estimated and vice-a-versa using the regression line

Scatterplot with regression line

```
➢ggplot(job, aes(x=aptitude, y=job_prof))+
        geom_point()+geom_smooth(method="lm")+
        labs(x="Aptitude", y="Job Proficiency", title="Scatterplot")
```

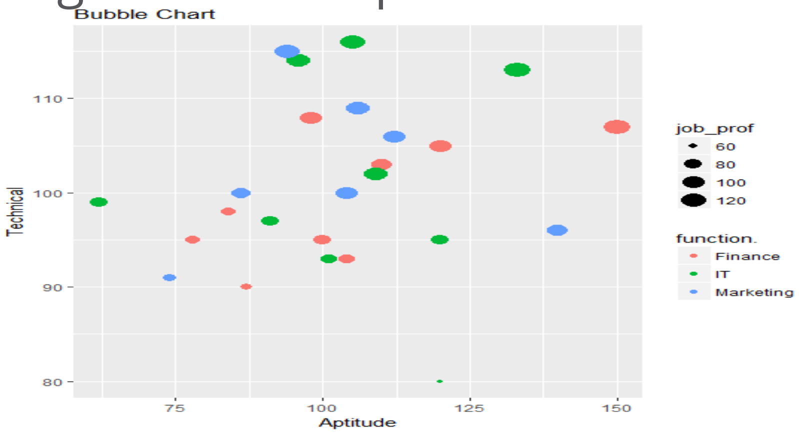
Scatterplot with regression line
R Output



DATA SCIENCE INSTITUTE

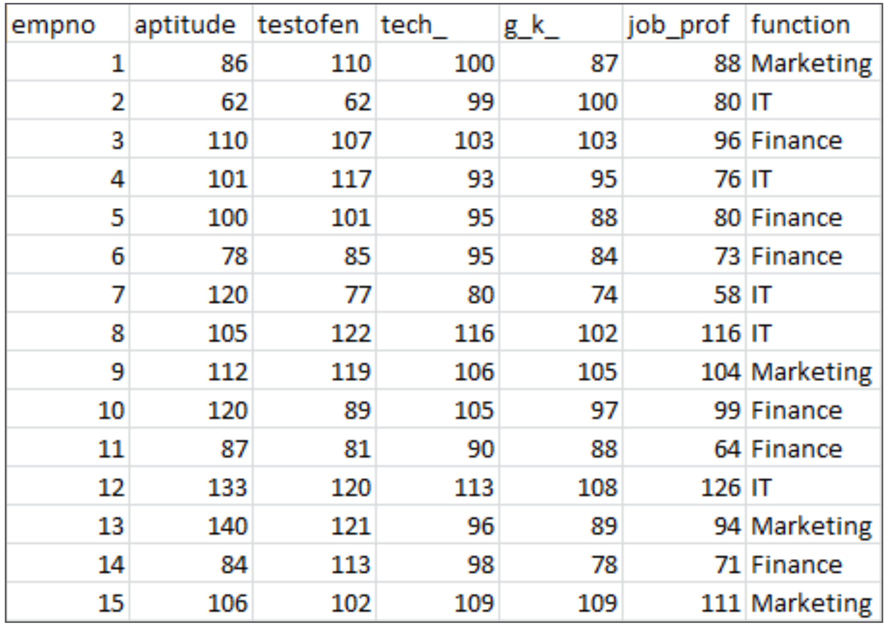
Bubble Chart

- **Bubble chart** is generally used instead of a scatter plot if your data object has three dimensions.
- The sizes of the bubbles are determined by the values in the third variable of the data series.
- Additional information can be provided by incorporating the color aspect.



Data Snapshot

JOB PROFICIENCY DATA for Bubble Chart)



empno	aptitude	testofen	tech_	g_k_	job_prof	function
1	86	110	100	87	88	Marketing
2	62	62	99	100	80	IT
3	110	107	103	103	96	Finance
4	101	117	93	95	76	IT
5	100	101	95	88	80	Finance
6	78	85	95	84	73	Finance
7	120	77	80	74	58	IT
8	105	122	116	102	116	IT
9	112	119	106	105	104	Marketing
10	120	89	105	97	99	Finance
11	87	81	90	88	64	Finance
12	133	120	113	108	126	IT
13	140	121	96	89	94	Marketing
14	84	113	98	78	71	Finance
15	106	102	109	109	111	Marketing

Bubble Chart

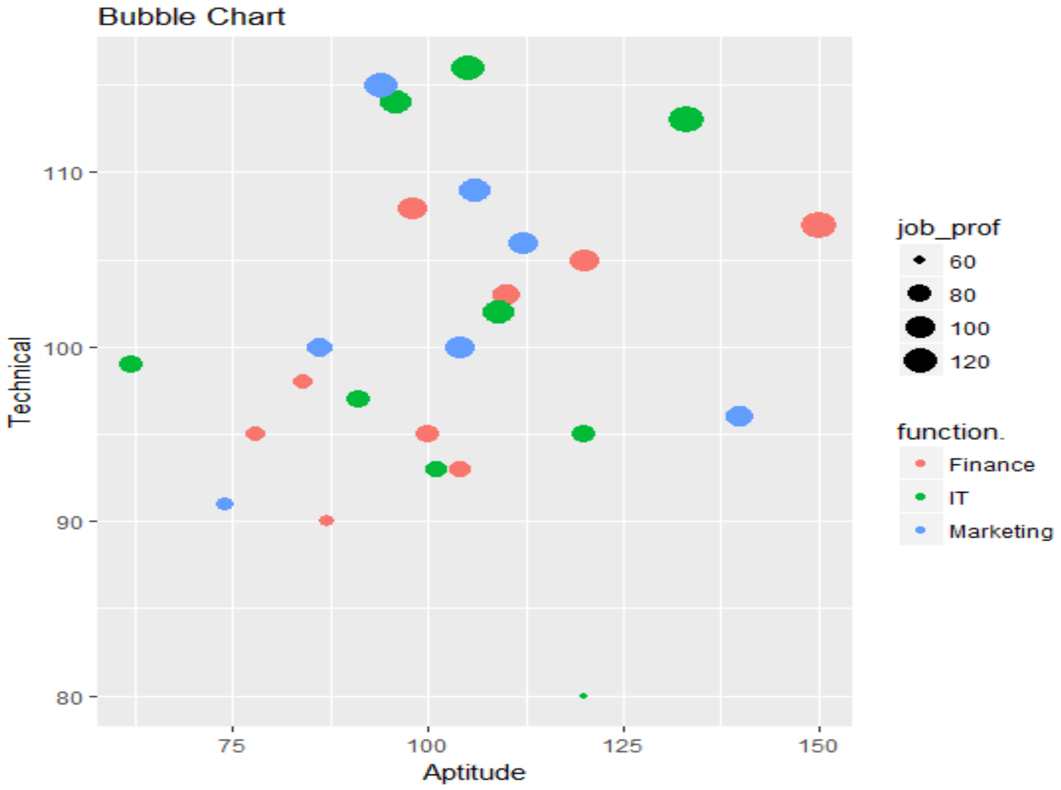
#Import data

- job2<-read.csv(file.choose(),header=T)

#Bubble Chart

- qplot(aptitude, tech_, data=job2, size=job_prof, color=function., xlab="Aptitude", ylab="Technical", main="Bubble Chart")

Bubble Chart
R Output



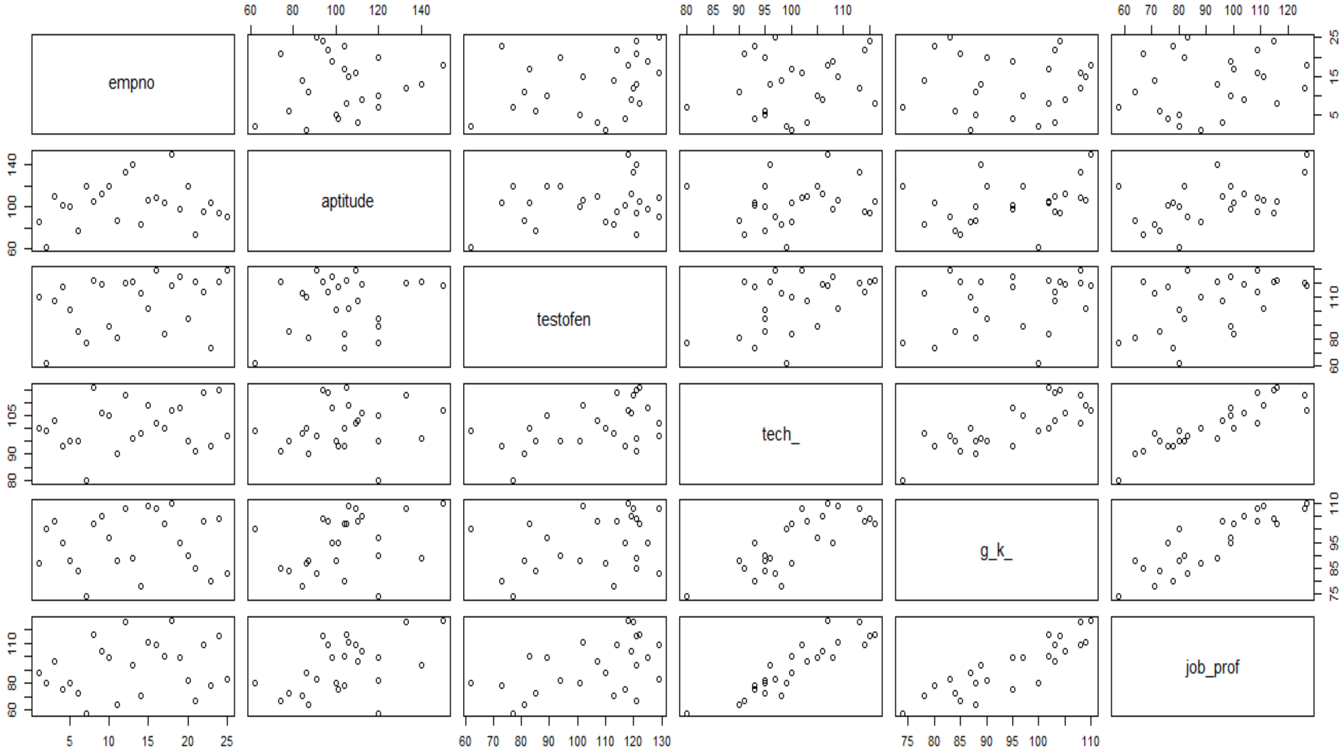
DATA SCIENCE INSTITUTE

Scatter Plot Matrix

- pairs(job)

Yields a scatter plot of pairwise correlations of all variables from the data.

Scatter Plot Matrix
R Output



Scatter Plot Matrix

#Installing GGally package

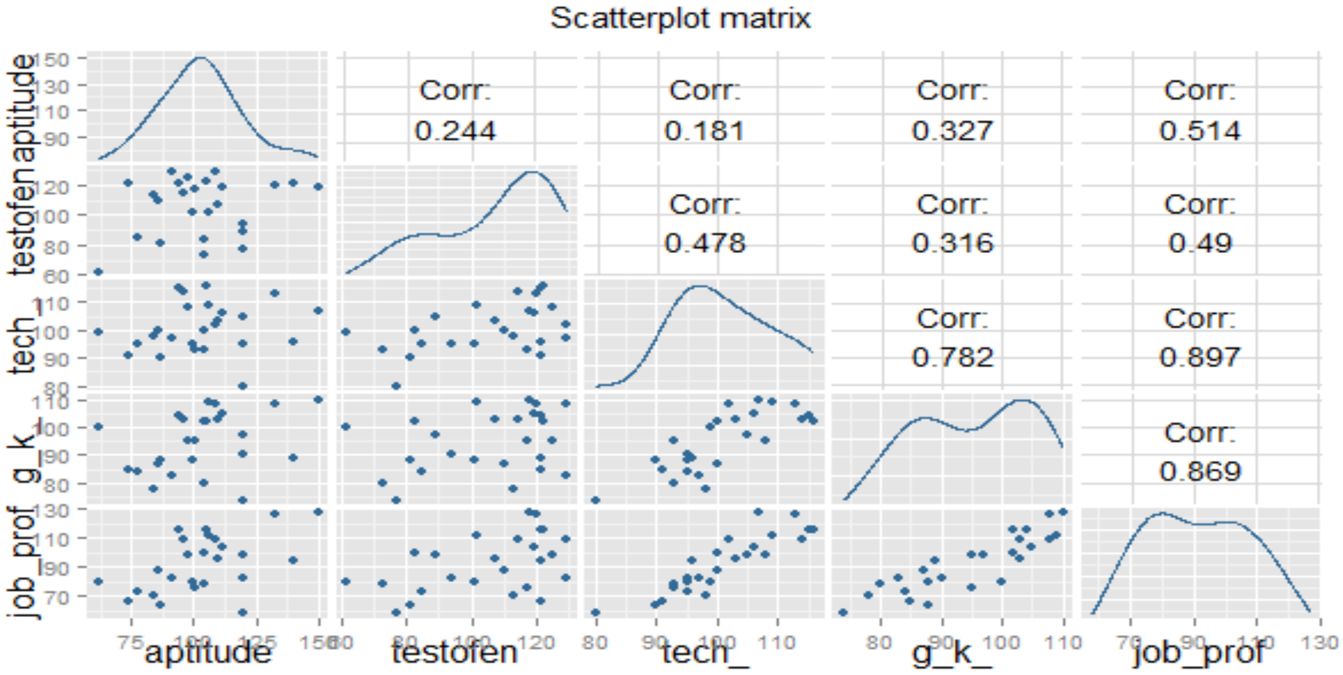
- install.packages(“GGally”)

  library(GGally)

- ggpairs(job[,c("aptitude", "testofen", "tech_", "g_k_", "job_prof")], title="Scatterplot matrix”)

DATA SCIENCE
INSTITUTE

Scatter Plot Matrix
R Output


Scatterplot matrix
aptitude
testofen
tech_
g_k_
job_prof
Corr: 0.244
Corr: 0.181
Corr: 0.327
Corr: 0.514
Corr: 0.478
Corr: 0.316
Corr: 0.49
Corr: 0.782
Corr: 0.897
Corr: 0.869