v2 Data Visualization

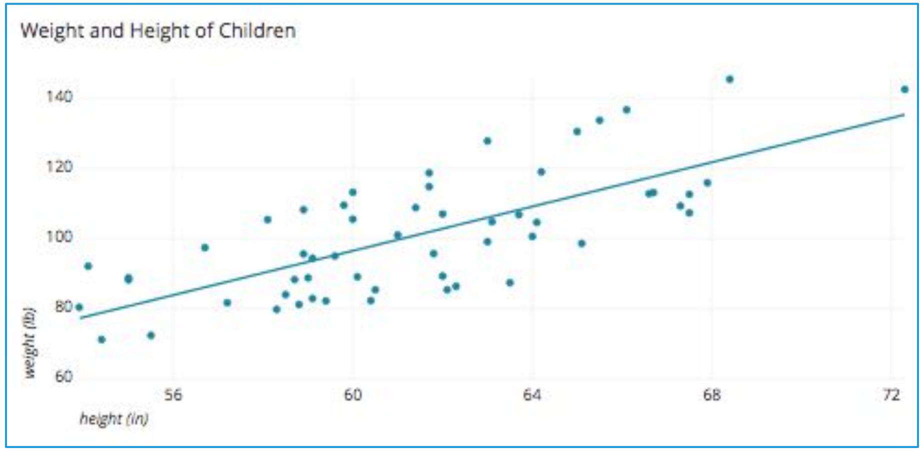
Visualizing Relationships in R

Contents

1. Summarizing Data in Diagrams
    - i. Scatterplot with Regression Line
    - ii. Scatterplot Matrix
    - iii. Bubble Chart
    - iv. Heat Map
    - v. Trend Line
    - vi. Motion Chart
2. Application Areas

Scatter Plot

A **scatter plot** is a two-dimensional data visualization that uses dots to represent the values obtained for two different variables - one plotted along the x-axis and the other plotted along the y-axis. For example this scatter plot shows the height and weight for a set of children.



Each dot represents one child with his or her height measured along the x-axis and weight measured along the y-axis

Scatter plots are used when you want to see how two variables are correlated. In the height and weight e.g., the chart wasn't just a simple log of the height and weight of a set of children, but it also visualized the relationship between height and weight - namely that weight increases as height increases. Notice that the relationship isn't perfect, some taller children weight less than some shorter children, but the general trend is pretty strong and we can see that weight is correlated with height.

Case Study

Let us try and see the correlation between Aptitude score of an employee and how is job performance/Proficiency

Background

A company has the scores of various attribute tests of their employees

Objective

To understand the factors contributing to the Job Proficiency of an employee.

To see the relationship between these various factors

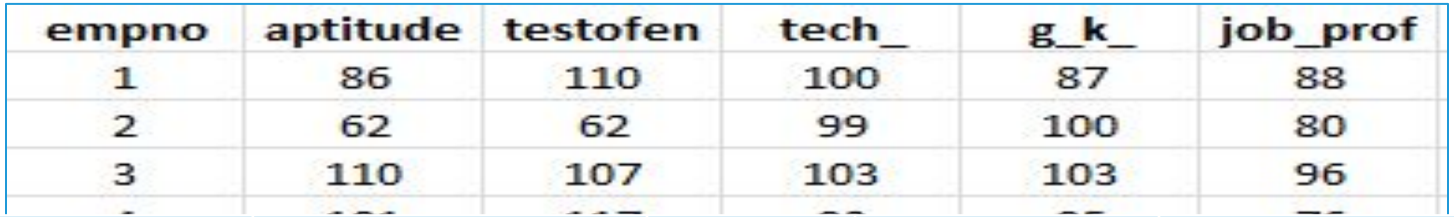
Sample Size

25

Data Snapshot

JOB PROFICIENCY DATA

Variables



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

Observations

Columns	Description	Type	Measurement	Possible values
empno	Employee No	Numeric	-	-
aptitude	Aptitude	Numeric	-	positive values
testofen	Test of English	Numeric	-	positive values
tech_	Technical Score	Numeric	-	positive values
g_k_	General Knowledge	Numeric	-	positive values
job_prof	Job Proficiency	Numeric	-	positive values

ScatterPlot with Regression Line in R

```
# Importing Data
job<-read.csv("JOB PROFICIENCY DATA.csv", header=TRUE)
attach(job)
```

attach() is used to call the data in R with help of which in further codes specifying the data repetitively can be avoided .

```
#Scatterplot with Regression Line
plot(aptitude,job_prof, main="Fig.No. 1 : ScatterPlot with Regression
     Line", xlab="Aptitude ", ylab="Job Proficiency", pch=19)
abline(lm(job_prof~aptitude), col="darkorange")
```

- ❑ **plot()** in base R yields different types of plots
- ❑ **aptitude** is one of the variable for plot
- ❑ **job_prof** is another variable to be plotted
- ❑ **main=** provides the user defined name of the chart. It has to be put in double quotes
- ❑ **xlab=** provides a user defined label for the variable on X axis
- ❑ **ylab=** provides a user defined label for the variable on Y axis
- ❑ **pch=** gives various shapes for the data points on the plot

- ❑ **abline()** in base R yields different types of lines on plot
- ❑ **lm()** provides the liner regression line of the first variable mentioned on the second
- ❑ **col=** provides the color of the line plotted

ScatterPlot with Regression Line in R

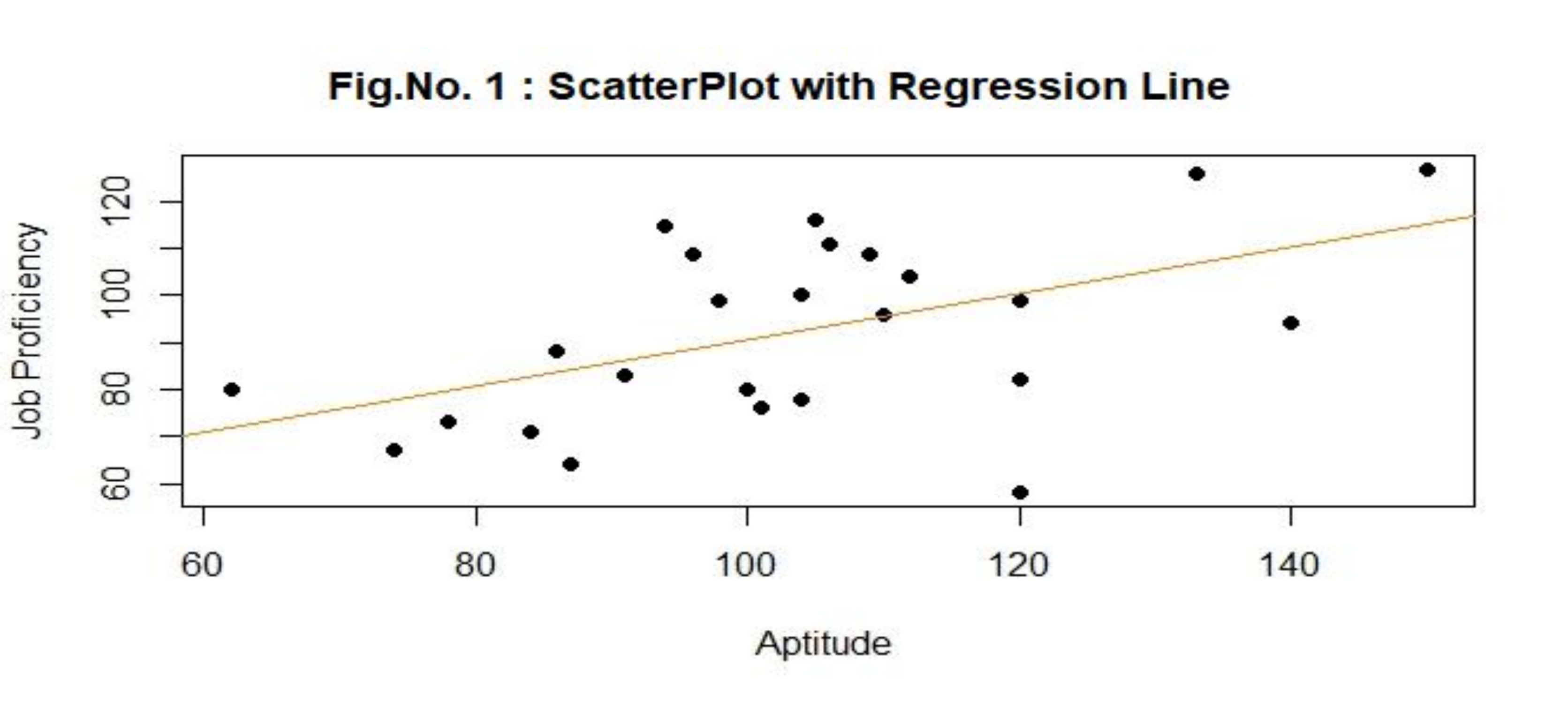


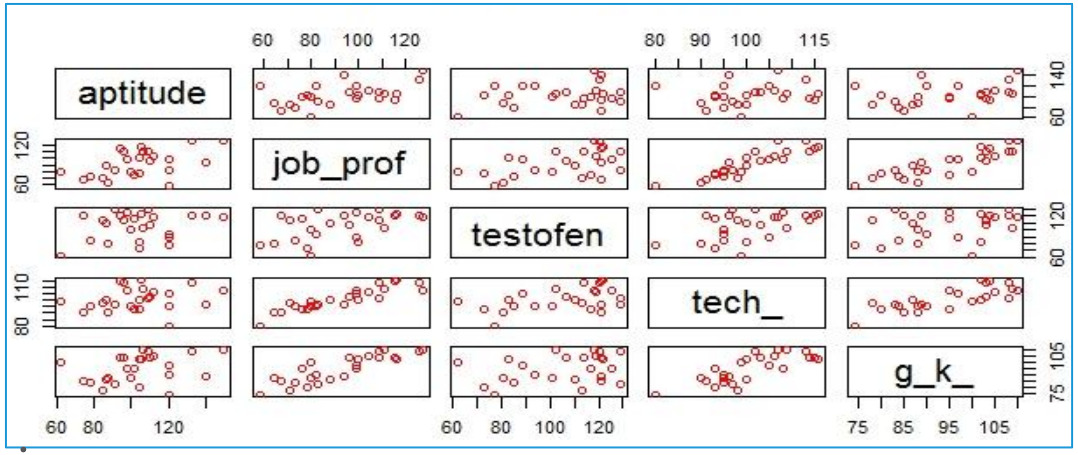
Fig.No. 1 : ScatterPlot with Regression Line

Interpretation :

- Scatter plot above shows that, as the aptitude score increases job proficiency also increases.
- For a given aptitude score, the job proficiency can be estimated and vice-a-versa using the regression line.

Scatter Plot Matrix

Scatter Plot Matrix gives the Scatterplot diagram of multiple variables with each other, all in one chart. It is used to determine if you have a linear correlation between multiple variables.



```
# ScatterPlot Matrix
```

```
pairs(~aptitude+job_prof+testofen+tech_+g_k_,data=job, main="Fig.No. 2 :
ScatterPlot Matrix",col="darkorange")
```

- **pairs()** in base R are used to plot pairwise comparison
- ~ each variable name to be plotted followed by a "+" sign needs to mentioned
- **main=** provides the user defined name of the chart.

Scatter Plot Matrix in R

```
# Output
```

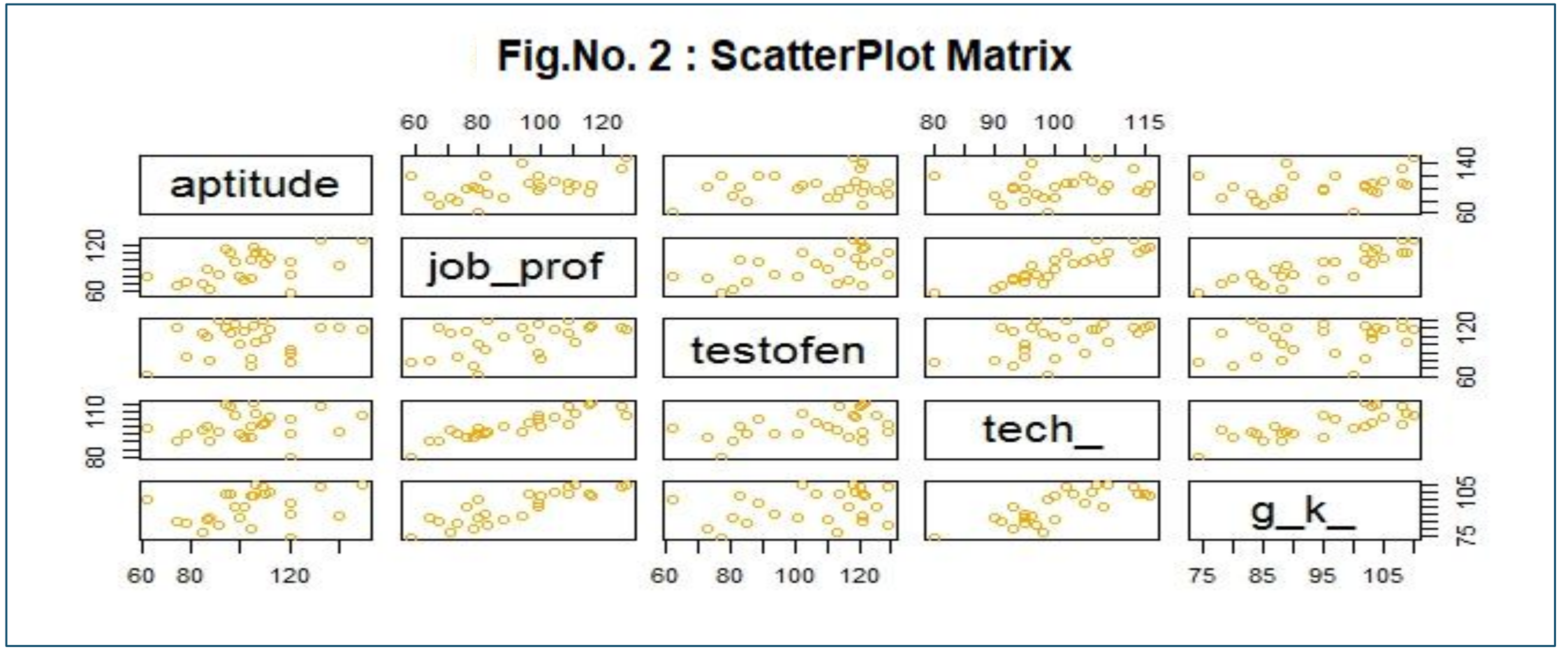


Fig.No. 2 : ScatterPlot Matrix

Interpretation :

- Scatter plot matrix above shows that, as the aptitude score, English language score, technical score and general knowledge score increases job proficiency also increases.
- Technical score and GK score has slight positive relation but other variables are not related to each other.

Scatter Plot Matrix in R using package “GGally”

```
#Installing and calling the package “Ggally” :
install.packages("GGally")
library(GGally)
```

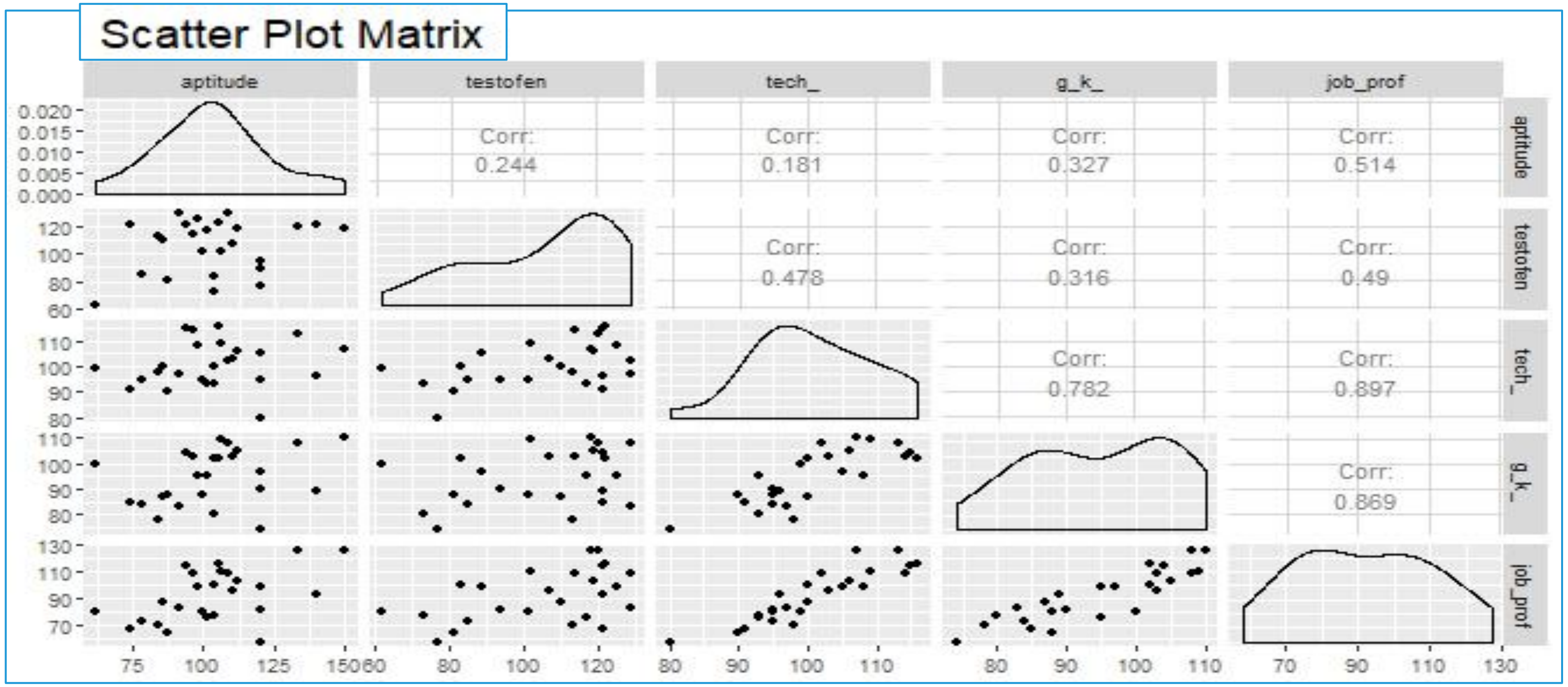
GGally is the best package we can use to plot an effective Scatter Plot in R

```
#ScatterPlot Matrix
ggpairs(job[,c("aptitude","testofen","tech_","g_k_","job_prof")],
title = "Scatter Plot Matrix")
```

- ❑ **ggpairs()** is the function used to call the variables for which the pairwise comparison chart needs to be plotted.
- ❑ **job[]** is the name of the data of which the variables need to be plotted
- ❑ **title =** provides the user defined name of the chart

Scatter Plot Matrix in R using package “GGally”

This plot shows the strength of relation through correlation coefficient and also the distribution of each variable.



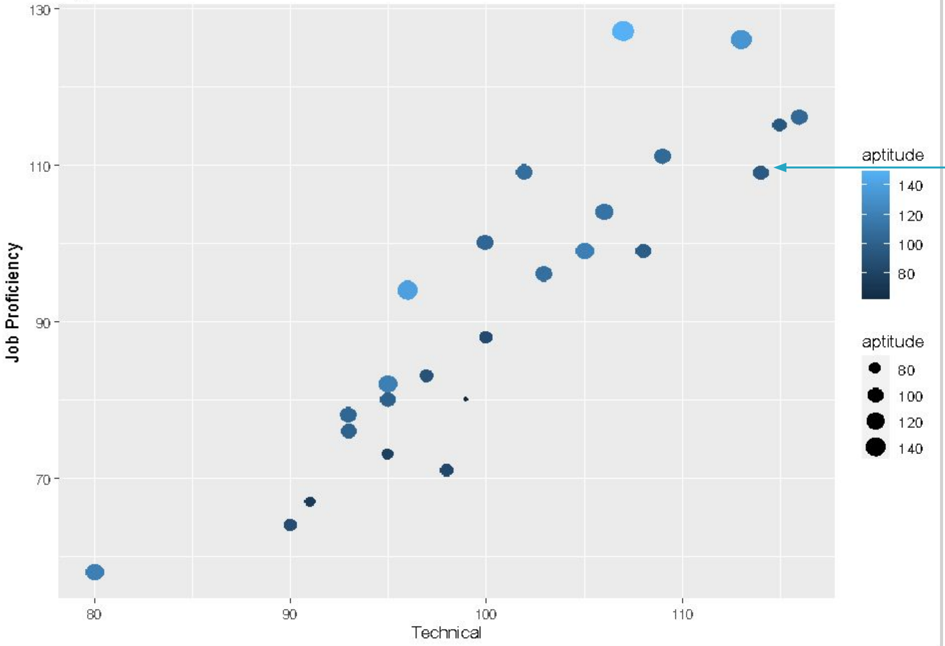
Interpretation :

- Technical and GK score have high correlation with job proficiency as compared to other variables.
- Technical and GK score also share high positive relation with each other.
- Aptitude score graph is symmetric.

Bubble Chart

Bubble chart is generally used instead of a scatter plot if your data has three data series that each contain a set of values.
The sizes of the bubbles are determined by the values in the third data series.

Fig. No. 3 : BUBBLE CHART



Interpretation :

- Here we observe that as Technical score increases Job Proficiency also increases , however, Aptitude score does not show any such consistent direction.

Bubble Chart in R

Package **ggplot2** will be used to create a bubble chart.

```
# Bubble Chart
```

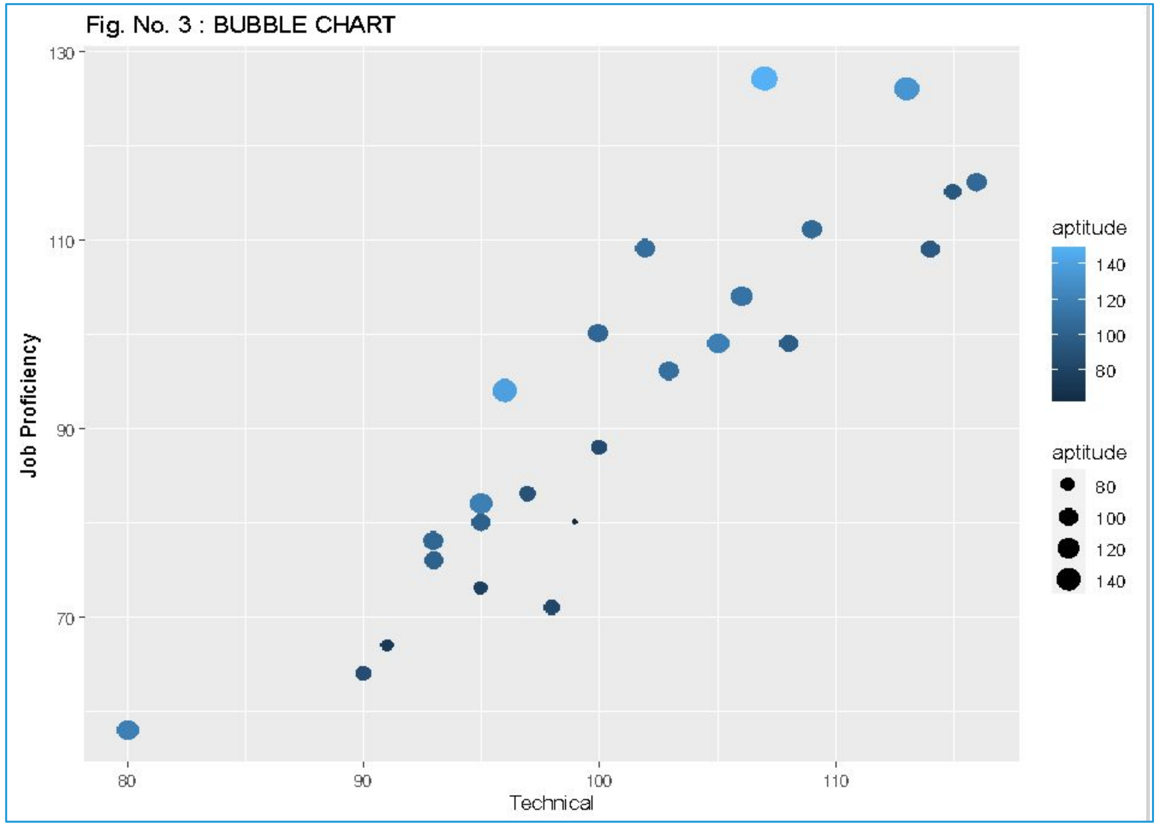
```
install.packages("ggplot2")
library(ggplot2)

qplot(x=tech_, y=job_prof, data=job, color=aptitude, size=aptitude,
xlab="Technical",  ylab="Job Proficiency", main="Fig. No. 3 :
BUBBLE CHART")
```

- **qplot()** in package ggplot2 is used to plot any ‘quick plot’
- **x,y** variables to be plotted on x and y axis
- **data=** data to be used for plotting
- **color=** variable to be considered for the colour of the bubble
- **size=** variable to be considered for the size of the bubble
- **xlab,ylab** labels for x and y axes
- **main=** provides the user defined name of the chart

Bubble Chart in R

```
# Output
```



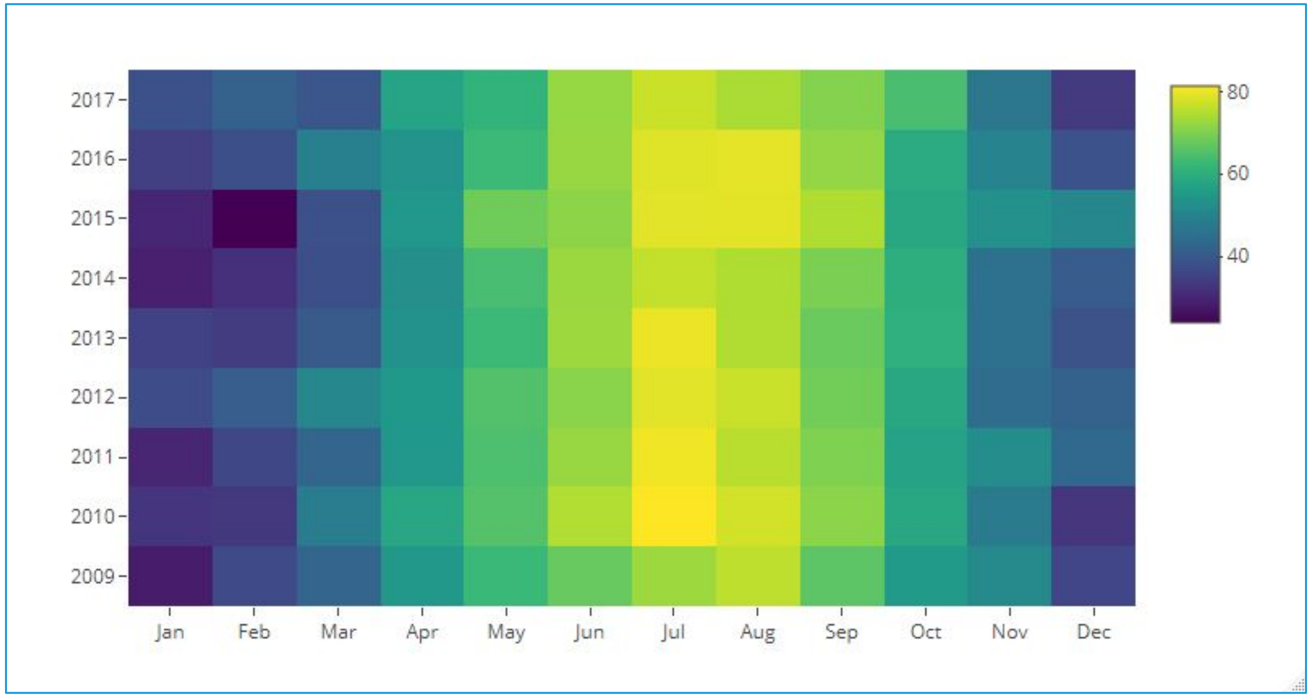
Interpretation :

- Here we observe that as Technical score increases Job Proficiency also increases, however, Aptitude score does not show any such consistent direction.

Heat Map

A Heat Map is a graphical representation of data where the individual values contained in a matrix are represented as colors.

It gives us quick information through color patterns.



In the example given above , we can see the temperature fluctuation in NY across months over the years

Case Study

To get a better understanding of the subject, we shall consider the below case as an example.

Background

NY Temperature varies across months over the years

Objective

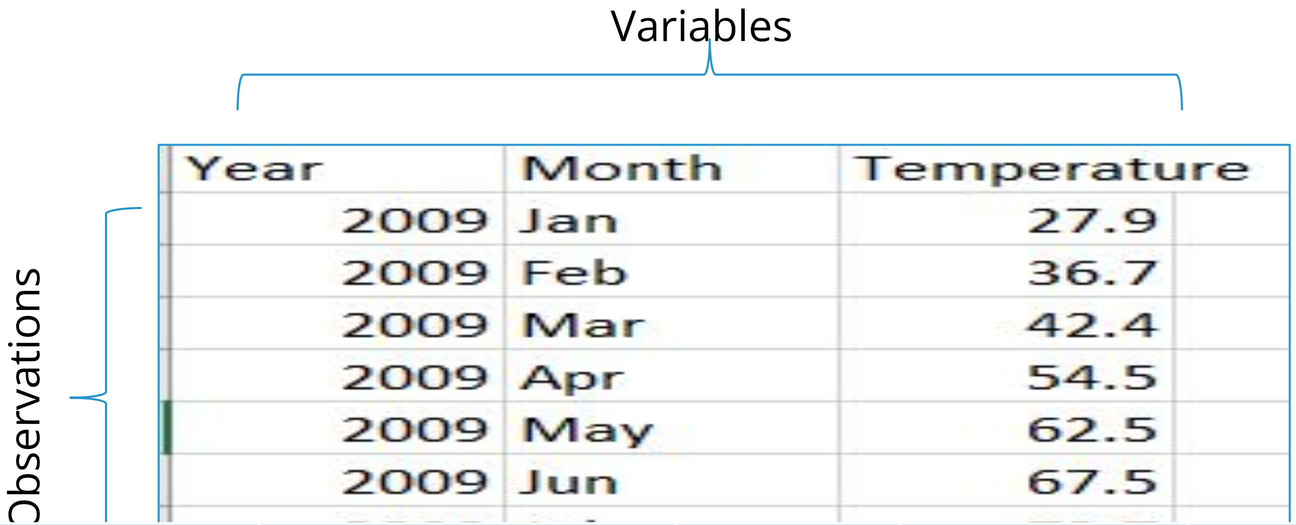
To visually see the hottest months in the years
To see how temperature has fluctuated over the years

Sample Size

108

Data Snapshot

Average Temperatures in NY



Variables

Observations

Year	Month	Temperature
2009	Jan	27.9
2009	Feb	36.7
2009	Mar	42.4
2009	Apr	54.5
2009	May	62.5
2009	Jun	67.5

Columns	Description	Type	Measurement	Possible values
Year	Years listed from 2009-2017	Categorical	2009 – 2017	9
Month	Months of the year	Categorical	Jan - Dec	12
Temperature	Average Temperature in degree Fahrenheit	Numeric	-	-

Heat Map in R

```
# Installing and calling the package
```

```
install.packages("plotly")
library(plotly)
```

```
# Importing Data and Arranging the Months in the right order :
```

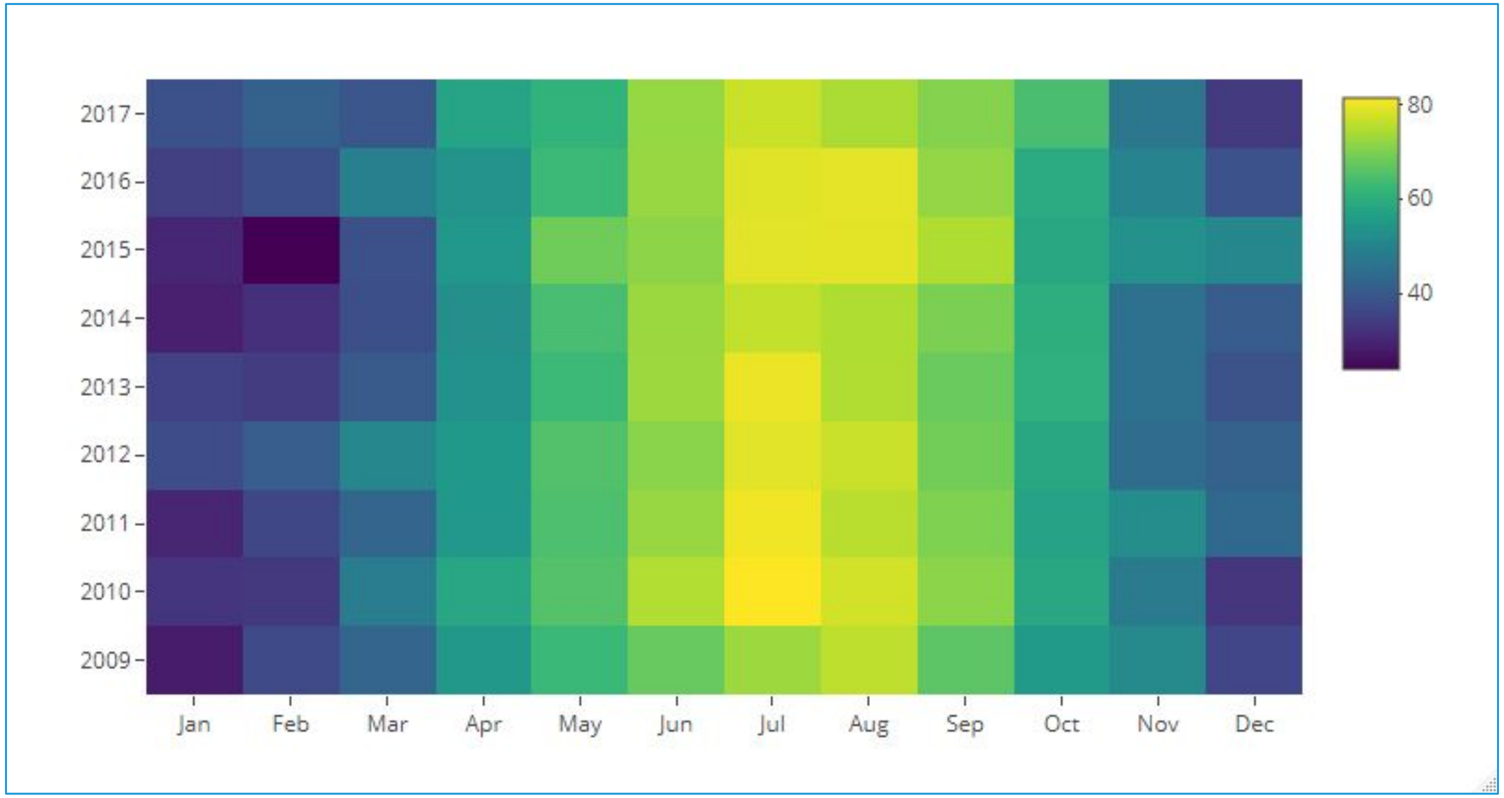
```
heatmapdata<-read.csv("Average Temperatures in NY.csv", header=TRUE)
heatmapdata$Month<-factor(heatmapdata$Month,level=unique(heatmapdata$M
onth))
```

```
# Heat Map
```

```
plot_ly(heatmapdata, x=heatmapdata$Month, y=heatmapdata$Year,
z=heatmapdata$Temperature,
type="heatmap",connectgaps=FALSE,showscale=T)
```

Heat Map in R

```
# Output for Heat Map :
```

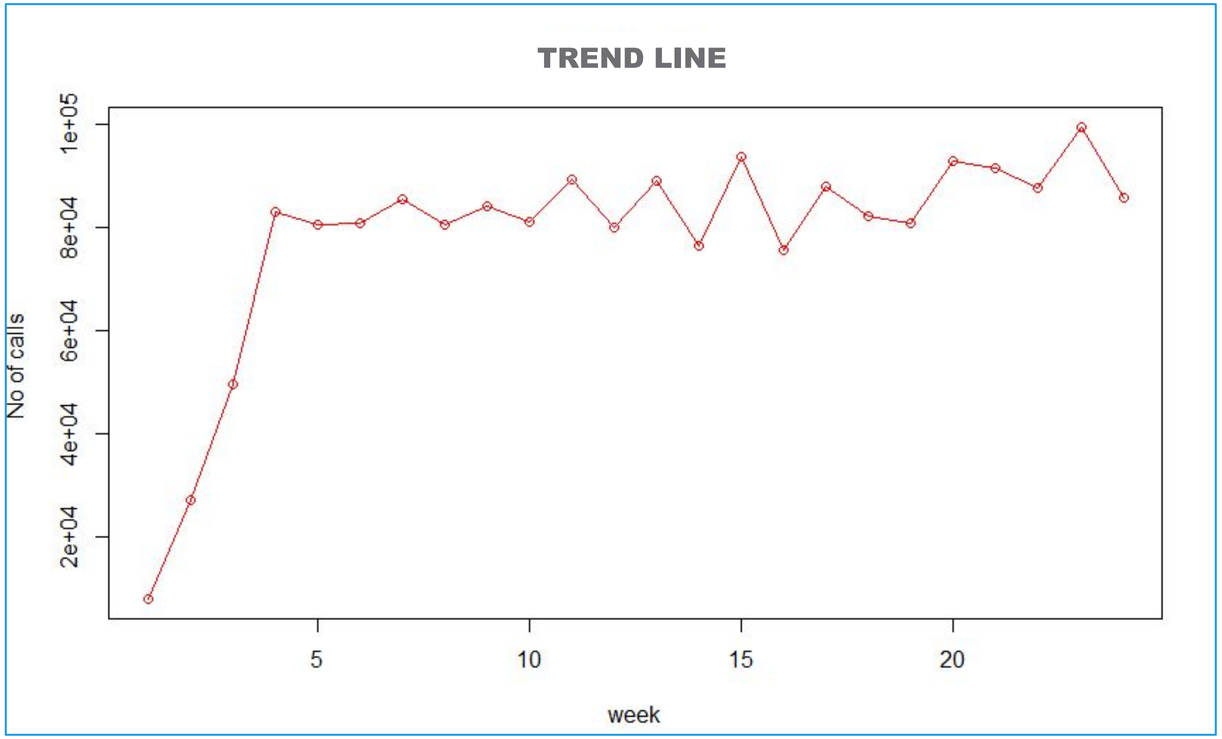


Interpretation :

- Heat map above shows that July is the hottest season across the year .
- 2015 showed a longer hot period as compared to other years extending from may to September

Trend Line

- A Trend Line is a straight line that connects two or more data points and then extends into the future to act as a line of support or resistance.
- It is usually used to plot something over time . It can be used to estimate the future values too



We can observe the increase and decrease in the total number of calls over a period of 24 weeks

Case Study

To get a better understanding of the subject, we shall consider the below case as an example.

Background

Telecom Weekly Data for 24 weeks

Objective

To visually observe the trend of total calls over 24 weeks

Sample Size

21902

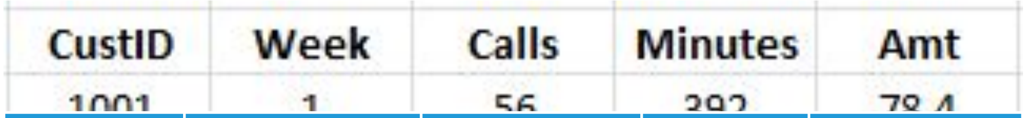
Data Snapshot

Plotting a trendline requires time-element. Consider the following datasets. Week can be taken as the time element.

TelecomData_WeeklyData

Variables

Observations



CustID	Week	Calls	Minutes	Amt
1001	1	56	292	78.4

Columns	Description	Type	Measurement	Possible values
CustID	Customer ID	Numeric	-	-
Week	Week no.	Numeric	1-24	24
Calls	No. of Calls	Numeric	-	positive values
Minutes	Total Minutes	Numeric	Minutes	positive values
Amt	Amount Charged	Numeric	Rs.	positive values

Trend Line in R

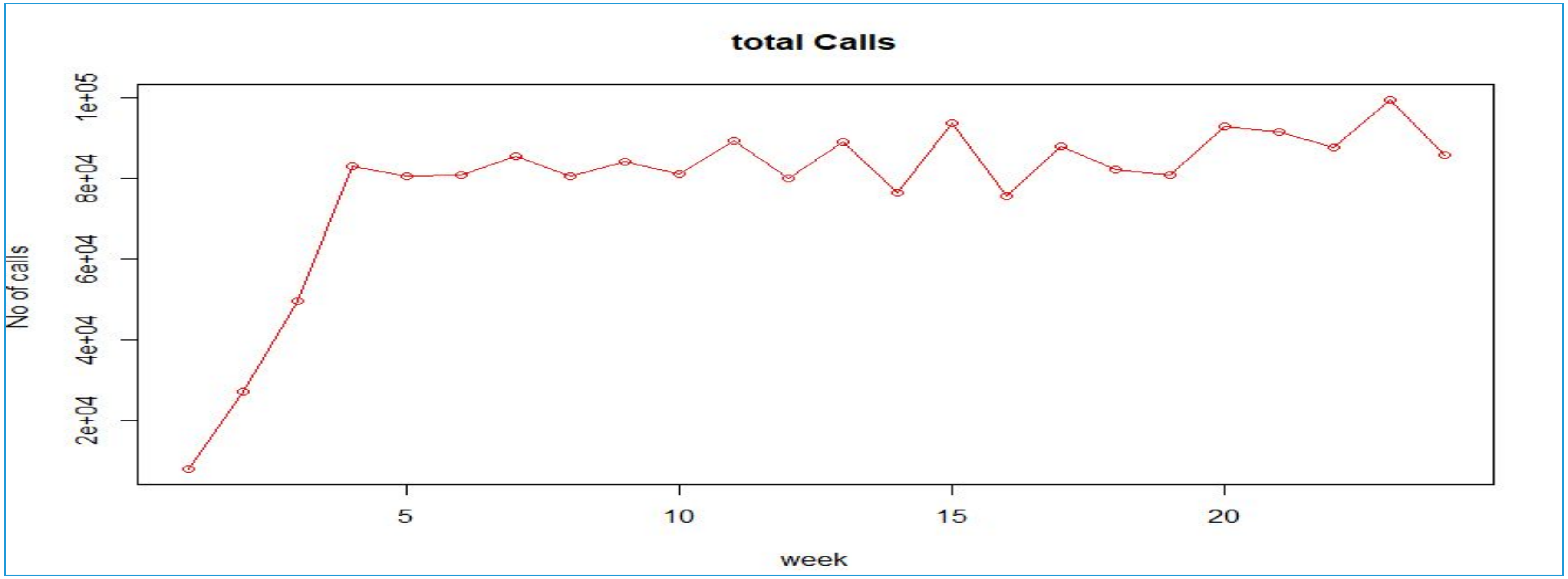
```r
# Importing Data
transaction<-read.csv("TelecomData_WeeklyData.csv", header=TRUE)
# Merging and Formatting Data
trend<-aggregate(Calls~Week, data=transaction, FUN=sum)

# Trend Line
plot(trend, type = "o", col = "red", xlab = "week", ylab = "No of calls",
main = "total Calls")
```

- The basic function is **plot(v,type,col,xlab,ylab)**
- **v** is a vector containing the numeric values.
- **type** takes the value **"p"** to draw only the points, **"l"** to draw only the lines and **"o"** to draw both points and lines.
- **col** is used to give colors to both the points and lines.
- **xlab()** is the label for x axis.
- **ylab()** is the label for y axis.
- **main** is the Title of the chart.

Trend Line in R

```
# Output
```



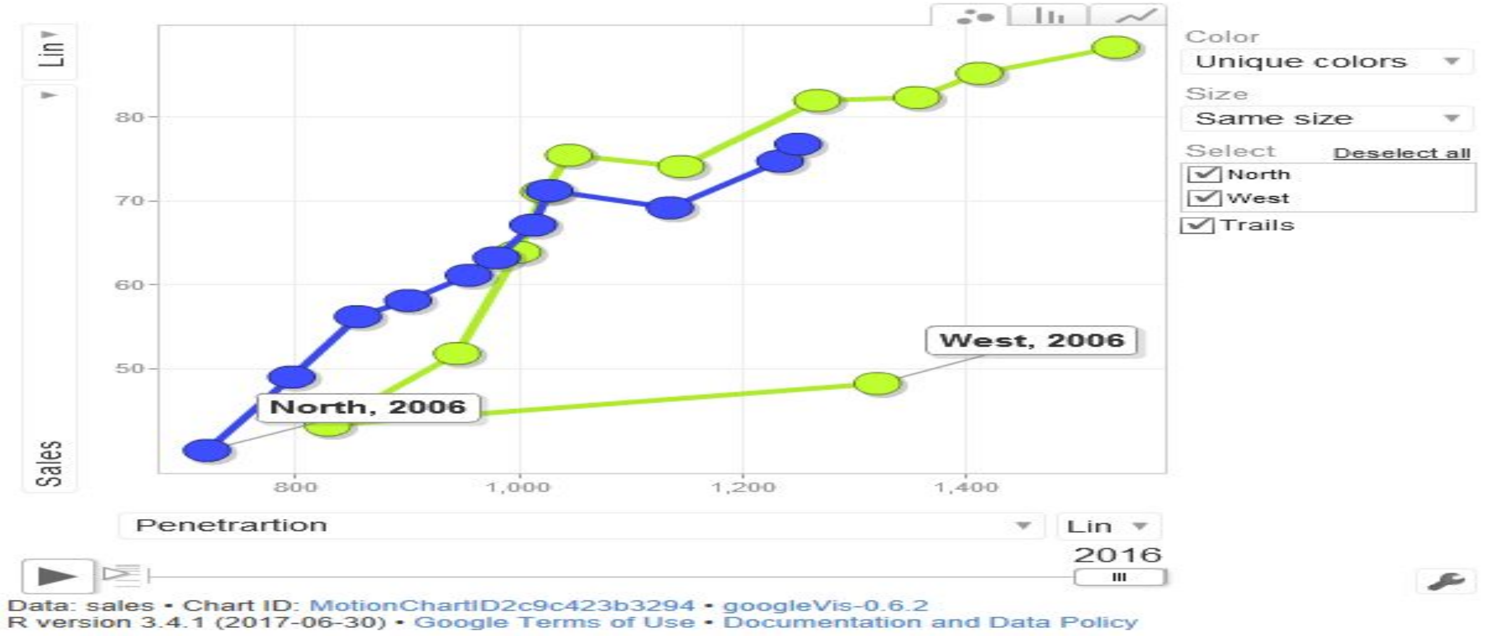
Interpretation :

- Upto first 4 weeks, number of calls increases continuously. After 5th week there are more ups and down in number of calls among customers.

Motion Chart

- A Motion Chart is a dynamic bubble chart which allows efficient and interactive exploration and visualization of longitudinal multivariate Data.
- It allows you to plot the dimension values in your report against up to four metrics across time.

Fig. No. 6 : MOTION CHART



Color: Unique colors

Size: Same size

Select: ☑ North ☑ West ☑ Trails

Penetrartion Lin

2016

Data: sales • Chart ID: MotionChartID2c9c423b3294 • googleVis-0.6.2
R version 3.4.1 (2017-06-30) • Google Terms of Use • Documentation and Data Policy

Case Study

To get a better understanding of the subject, we shall consider the below case as an example.

Background

Sales Data & it’s penetration in each Region over the years

Objective

To visually observe the sales & penetration in motion over the years

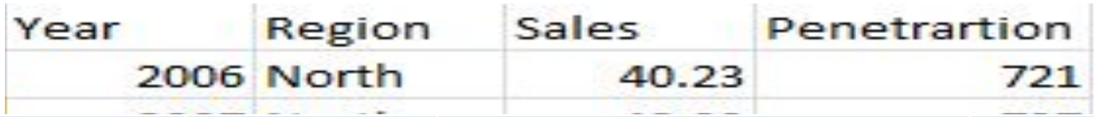
Sample Size

22

Data Snapshot

Sales Data (Motion Chart)

Variables



Year	Region	Sales	Penetrartion
2006	North	40.23	721

Observations

Columns	Description	Type	Measurement	Possible values
Year	Year	Numeric	2006-2016	11
Region	Region	Categorical	North,West	2
Sales	Sales in a particular Year	Numeric	Rs.	Positive values
Penetration	Penetration in a particular Year	Numeric	-	Positive values

Motion Chart in R

```
#Importing Data
```

```
sales<-read.csv("Sales Data (Motion Chart).csv", header=TRUE)
```

```
#Installing and calling the package
```

```
install.packages("googleVis")
library(googleVis)
```

googleVis is the best package we can use to plot an effective Motion Chart in R

```
# Motion Chart
```

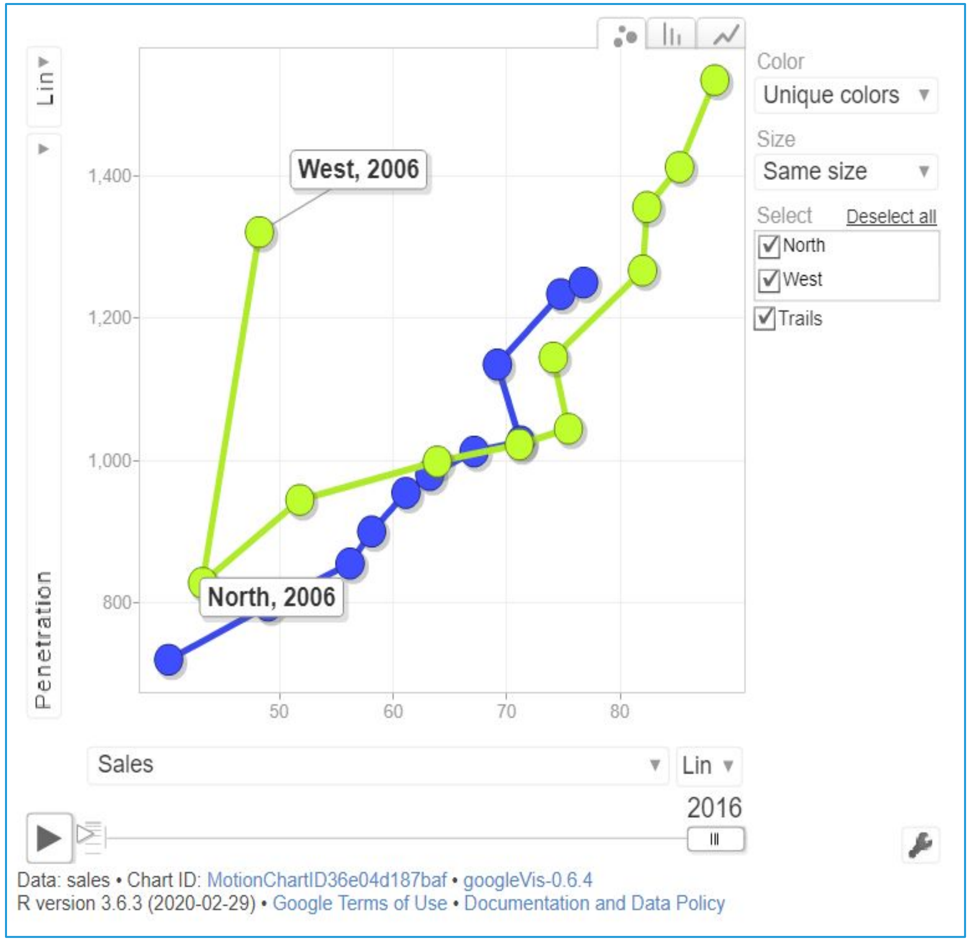
```
mchart<-gvisMotionChart(sales, idvar="Region", timevar="Year")
plot(mchart)
```

- ❑ **gvisMotionChart()** is the function used to create a motion chart
- ❑ **sales** is the data that is used
- ❑ **idvar=** inputs the id variable
- ❑ **timevar=** inputs the time variable

plot() is used to plot the final Motion Chart

Motion Chart in R

```
# Output
```



Color: Unique colors

Size: Same size

Select Deselect all

☑ North

☑ West

☑ Trails

Data: sales • Chart ID: MotionChartID36e04d187baf • googleVis-0.6.4
R version 3.6.3 (2020-02-29) • Google Terms of Use • Documentation and Data Policy

Interpretation :

- Over time, as sales increases penetration has also increased parallelly for both the Regions.

Browser Settings for Motion Chart

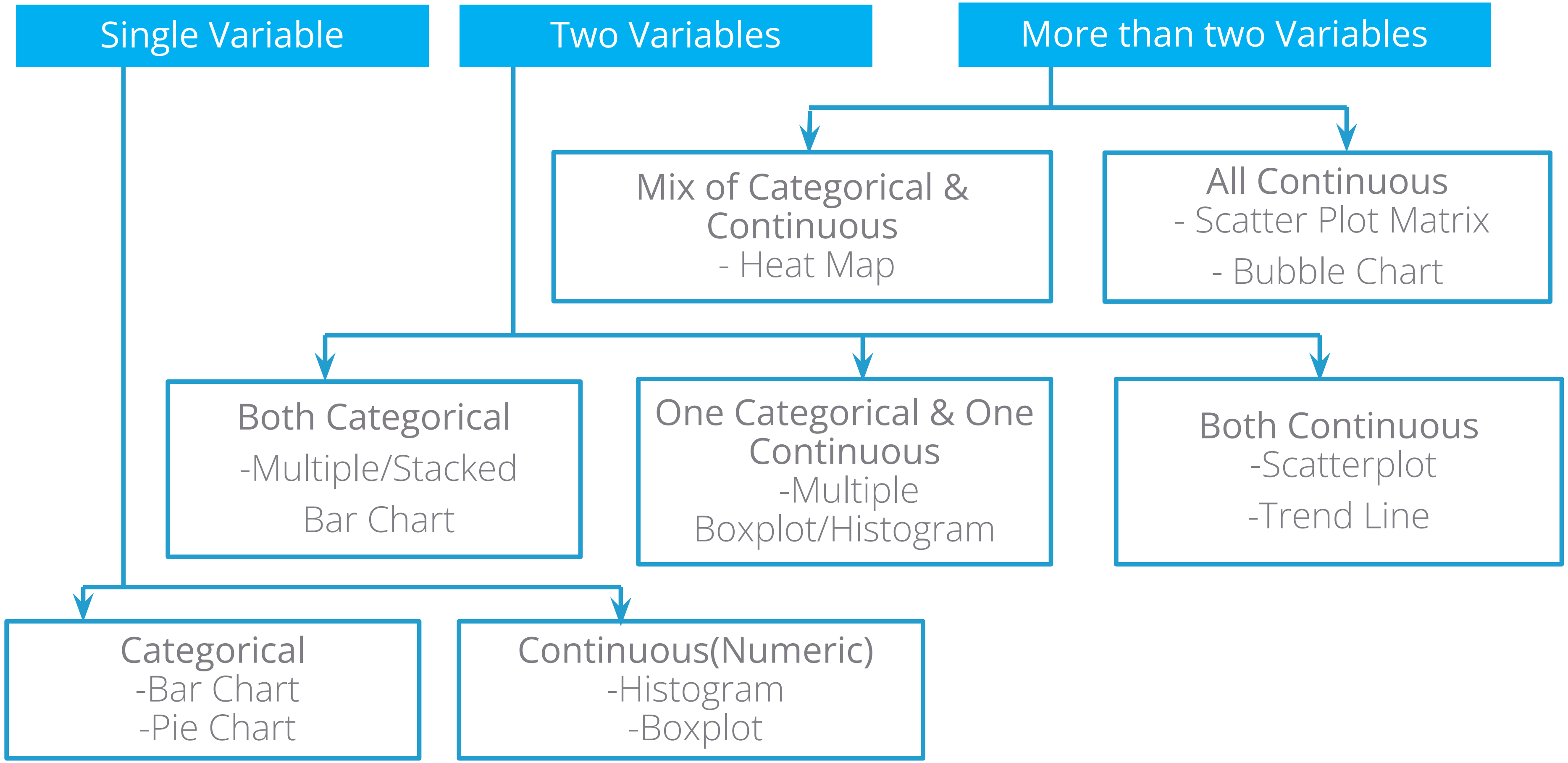
Incase you do not get the output for Motion Chart you will have to do changes in your Chrome Browser settings as follows :

1. Go to the website which opened when you executed the Motion Chart code.
2. To the left of the web address, click the icon that you see: Lock🔒 , Infoⓘ or Dangerous⚠ .
3. Click Site settings.
4. In permission setting change Flash to “allow”. Your changes will save automatically.
5. Then go back to Motion Chart web page & Reload it, you will be able to see the Motion Chart.

If you are using any other browser then make sure that flash player is enabled and updated.

Get an Edge!

Choosing the right graph



Single Variable

Two Variables

More than two Variables

Mix of Categorical & Continuous
- Heat Map

All Continuous
- Scatter Plot Matrix
- Bubble Chart

Both Categorical
-Multiple/Stacked Bar Chart

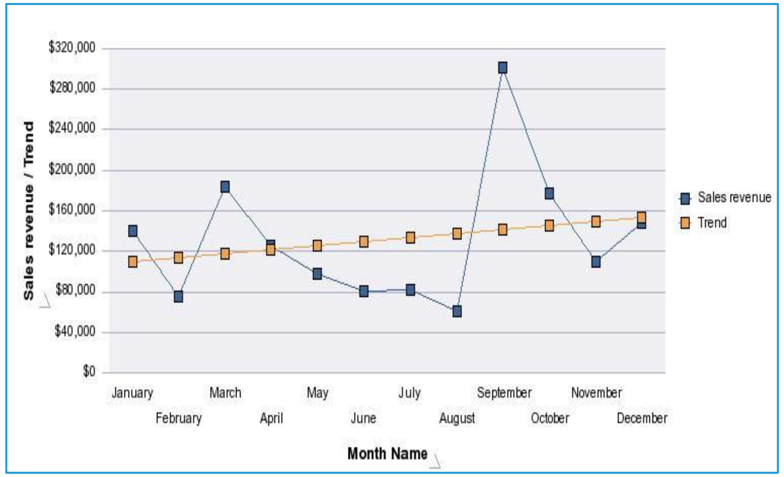
One Categorical & One Continuous
-Multiple Boxplot/Histogram

Both Continuous
-Scatterplot
-Trend Line

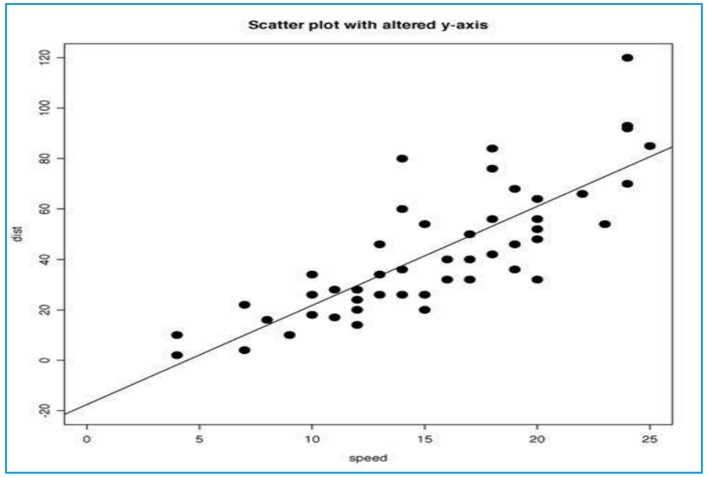
Categorical
-Bar Chart
-Pie Chart

Continuous(Numeric)
-Histogram
-Boxplot

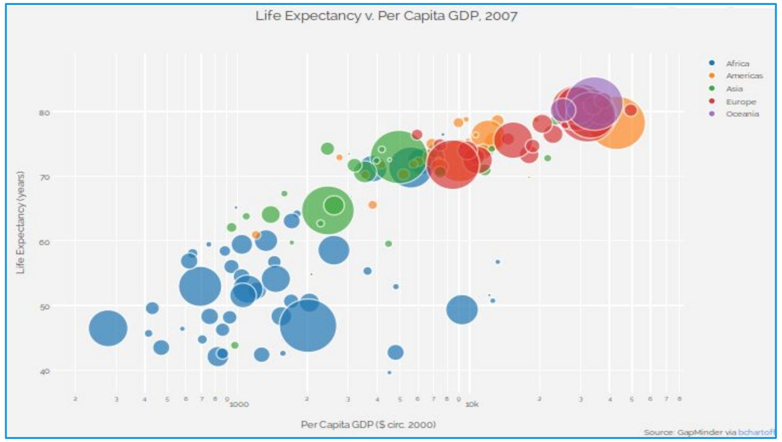
Application Areas



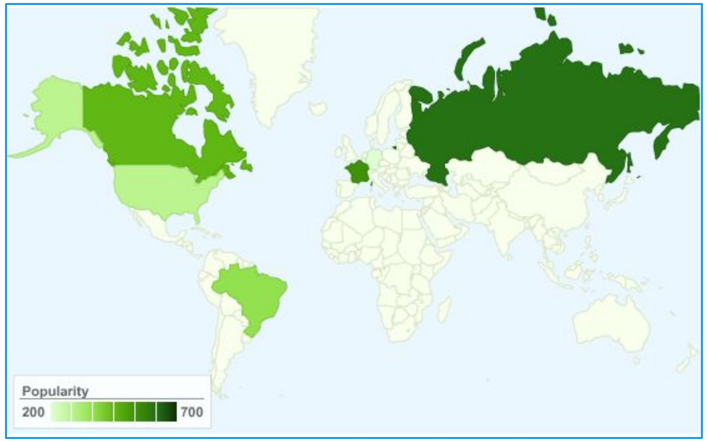
Sales Trends



Distance v/s Speed



Economics



Geographical Studies

Quick Recap

Chart Types and Functions in R

- Scatterplot with Regression Line – **`plot() + abline()`**
- Scatterplot Matrix – **`pairs() or ggpairs()`** from package "GGally"
- Bubble Chart – **`qplot()`** from package "ggplot2"
- Heat Map – **`plot_ly()`** from package "plot.ly“
- Trend Line – **`plot()`**
- Motion Chart - **`gvisMotionChart()`** from package "googleVis"