

## ASSIGNMENT-8

AIM: Visualise the data using R/Python by plotting graphs for assignments 2 & 3.

## OBJECTIVES:

- i. To understand & apply analytics concepts of big data using R/Python.
- ii. To study detailed data visualisation in R.

## THEORY:

## 1. R-PIE-CHARTS:

In R, pie chart is created using `pie()` function which takes 2 positive nos. as a vector i/p.

The additional parameters are used to control labels, color, title, etc.

## Syntax:

`pie(x, label, radius, main, col, clockwise)`

`x` → vector containing values

`label` → used to give description.

`radius` → radius of pie

`main` → title of chart

`col` → color palette

`clockwise` → logical value indicated if slices are drawn clockwise or anticlockwise.

Ex:

`x ← c(21, 62, 10, 58)`

`labels ← c("London", "New York", "Singapore", "Mumbai")`



# Plot the chart  
`pie(x, label)`

# Give chart file a name  
`png(file = "city.png")`

# Save file  
`dev.off()`

## 2. R-BAR-CHARTS:

A bar chart represents data in rectangular bars with length of bars proportional to the value of variable. R uses `barplot()` for this.

Syntax:

`barplot(H, xlab, ylab, main, names, arg, col)`

`H` → vector

`xlab` → x label

`ylab` → y label

`main` → title

`names, arg` → vector of names

`col` → colour of bars.

ex:

```
H <- c(7, 12, 28, 3, 41)
barplot(H)
```

## 3. R-BOX PLOTS :

Box plots are used to measure how well distributed data is in a dataset. It divides dataset into 5 quartiles. This graph represents min, max, median, 1<sup>st</sup> quartile, 3<sup>rd</sup> quartile.



Syntax:

`boxplot(x, data, notch, varwidth, names, main)`

`x` → vector

`data` → dataframe

`notch` → logical value

`varwidth` → logical value

`names` → group labels

`main` → title.

ex:

`input ← mtcars[, c('mpg', 'cyl')]`

#### 4. R- HISTOGRAMS:

A histogram represents the frequencies of values of a variable bunched ranges. R uses `hist()` function.

Syntax:

`hist(v, main, xlab, xlim, ylim, col, border)`

`v` → vector

`main` → title

`col` → colour

`border` → border colour

`xlab` → x label

`xlim` → range of values on x-axis

`ylim` → range of values on y-axis

ex:

`v ← c(9, 13, 1, 18, 2, 5, )`

`hist(v, xlab = "weight", col = "yellow",`

`border = "blue")`



**CONCLUSION :**

In this assignment, we visualised data using  
by plotting graphs.