Experiment No.5
Data Visualization using Hive/PIG/R/Tableau/.
Date of Performance: 22/08/23
Date of Submission: 05/09/23

Aim: Data Visualization using Hive/PIG/R/Tableau/.

Theory:

Data visualization is the technique used to deliver insights in data using visual cues such as graphs, charts, maps, and many others. This is useful as it helps in intuitive and easy understanding of the large quantities of data and thereby make better decisions regarding it. Data Visualization in R Programming Language

The popular data visualization tools that are available are Tableau, Plotly, R, Google Charts, Infogram, and Kibana. The various data visualization platforms have different capabilities, functionality, and use cases. They also require a different skill set . This article discusses the use of R for data visualization.

R is a language that is designed for statistical computing, graphical data analysis, and scientific research. It is usually preferred for data visualization as it offers flexibility and minimum required coding through its packages.

Consider the following *airquality* data set for visualization in R:

Ozone	Solar R.	Wind	Temp	Month	Day
41	190	7.4	67	5	1
36	118	8.0	72	5	2
12	149	12.6	74	5	3
18	313	11.5	62	5	4

NA	NA	14.3	56	5	5
28	NA	14.9	66	5	6

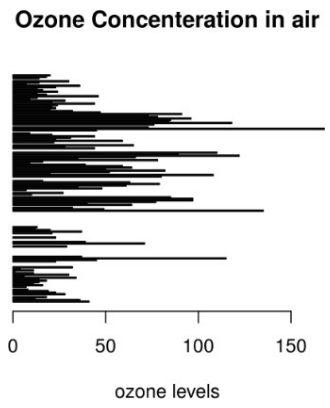
1.Bar Plot

There are two types of bar plots- horizontal and vertical which represent data points as horizontal or vertical bars of certain lengths proportional to the value of the data item. They are generally used for continuous and categorical variable plotting. By setting the horiz parameter to true and false, we can get horizontal and vertical bar plots respectively.

Example 1:

```
# Horizontal Bar Plot for

# Ozone concentration in airbarplot(airquality$Ozone,

    main = 'Ozone Concenteration in air',xlab = 'ozone levels', horiz = TRUE)
```

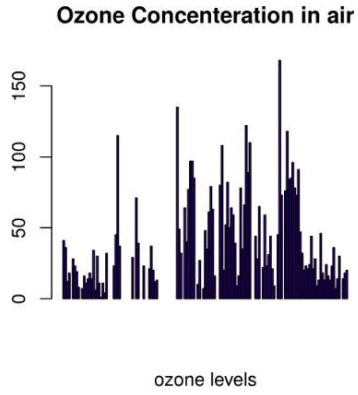
Output:



Example 2:

```
# Vertical Bar Plot for

# Ozone concentration in air
```

```
barplot(airquality$Ozone, main = 'Ozone Concenteration in air',xlab = 'ozone levels', col ='blue',
    horiz = FALSE)
```



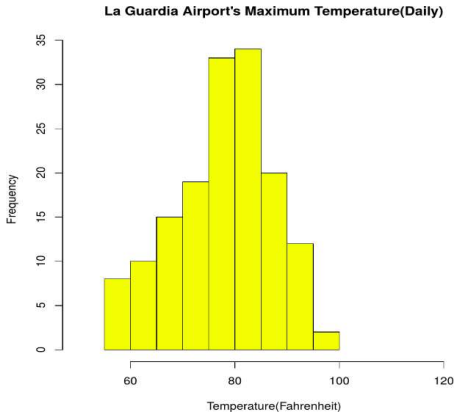
2. Histogram

A histogram is like a bar chart as it uses bars of varying height to represent data distribution. However, in a histogram values are grouped into consecutive intervals called bins. In a Histogram, continuous values are grouped and displayed in these bins whose size can be varied.

Example:

```
# Histogram for Maximum Daily Temperaturedata(airquality)


hist(airquality$Temp, main ="La Guardia Airport's\ Maximum Temperature(Daily)",
   xlab ="Temperature(Fahrenheit)",xlim = c(50, 125), col ="yellow",freq = TRUE)
```

Output:

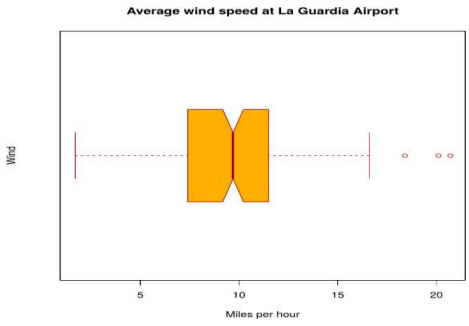


3. Box Plot

The statistical summary of the given data is presented graphically using a boxplot. A boxplot depicts information like the minimum and maximum data point, the median value, first and third quartile, and interquartile range.

Example:

```
# Box plot for average wind speeddata(airquality)

boxplot(airquality$Wind, main = "Average wind speed\at La Guardia Airport",

    xlab = "Miles per hour", ylab = "Wind",col = "orange", border = "brown", horizontal =

    TRUE, notch = TRUE)
```



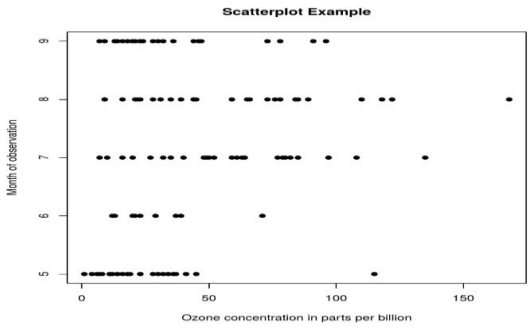
4. Scatter Plot

A scatter plot is composed of many points on a Cartesian plane. Each point denotes the value taken by two parameters and helps us easily identify the relationship between them.

Example:.

```
# Scatter plot for Ozone Concentration per monthdata(airquality)

plot(airquality$Ozone, airquality$Month,main ="Scatterplot Example",

   xlab ="Ozone Concentration in parts per billion",ylab =" Month of observation ", pch = 19)
```

5. Heat Map



Heatmap is defined as a graphical representation of data using colors to visualize the value ofthe matrix. heatmap() function is used to plot heatmap.

Syntax: heatmap(data)

Parameters: data: It represent matrix data, such as values of rows and columnsReturn: This

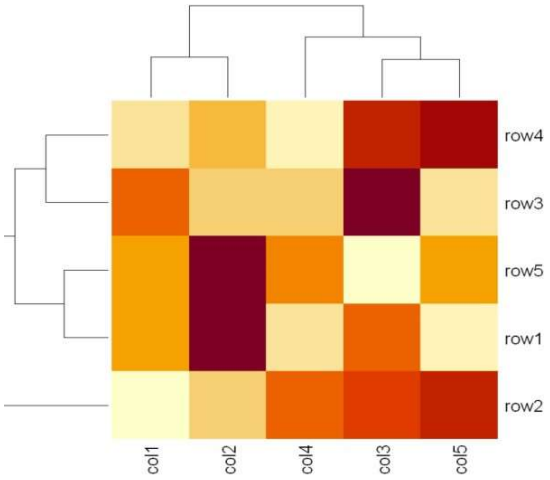
function draws a heatmap.

```
# Set seed for reproducibility# set.seed(110)

# Create example data

data <- matrix(rnorm(50,  0, 5), nrow = 5, ncol = 5)
```

Column names

colnames(data) <- paste0("col", 1:5) rownames(data) <- paste0("row", 1:5)# Draw a heatmap

heatmap(data)



6. Map visualization in R

Here we are using maps package to visualize and display geographical maps using an R programming language.

Read dataset and convert it into# Dataframe

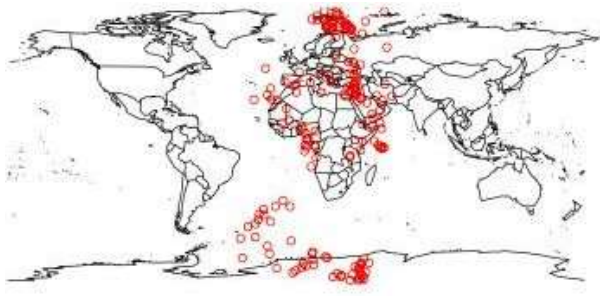
data <- read.csv("worldcities.csv")df <- data.frame(data)

Load the required librarieslibrary(maps)

map(database = "world")

marking points on map

points(x = df$lat[1:500], y = df$lng[1:500], col = "Red")install.packages("maps")



7. 3D Graphs in R

Here we will use preps() function, This function is used to create 3D surfaces in perspectiveview. This function will draw perspective plots of a surface over the x–y plane.

Syntax: persp(x, y, z)

Parameter: This function accepts different parameters i.e. x, y and z where x and y are vectors defining the location along x- and y-axis. z-axis will be the height of the surface in the matrix z.

Return Value: persp() returns the viewing transformation matrix for projecting 3D coordinates (x, y, z) into the 2D plane using homogeneous 4D coordinates (x, y, z, t).

Adding Titles and Labeling Axes to Plotcone <- function(x, y){

sqrt(x ^ 2 + y ^ 2)

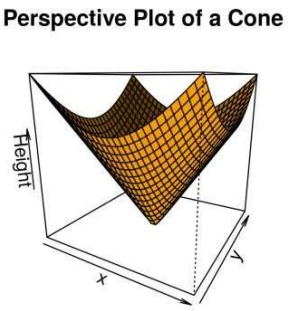
}

prepare variables.

x <- y <- seq(-1, 1, length = 30)z <- outer(x, y, cone)

plot the 3D surface

```
# Adding Titles and Labeling Axes to Plotpersp(x, y, z,
main="Perspective Plot of a Cone",zlab = "Height",
theta = 30, phi = 15,
col = "orange", shade = 0.4)
```



CONCLUSION:

Data visualization is a critical component of modern data analytics, and utilizing tools like Hive, PIG, R, and Tableau can greatly enhance the process of extracting insights from large datasets. Hive and PIG are invaluable for data preparation and transformation in big data environments, allowing users to efficiently manage and structure data before visualization. R, on the other hand, is a powerful statistical programming language, ideal for in-depth analysis and statistical visualization. Tableau excels in creating interactive, user-friendly visualizations that enable users to explore data intuitively. When combined, these tools offer a comprehensive approach to data visualization, from data wrangling to the creation of engaging, actionable visual insights. However, the choice of tool should be guided by specific project requirements, data size, and user preferences, ensuring that the most suitable tool is selected for the task at hand. Ultimately, the effective integration of Hive, PIG, R, and Tableau in data visualization can provide organizations with a robust and versatile framework for uncovering valuable insights from their data.