Sardar Patel Institute of Technology, Mumbai

Department of Electronics and Telecommunication Engineering

T.E. Sem-V (2018-2019)

ETL54-Statistical Computational Laboratory

Lab-4:Data Visualization and R Graphics

Name: Yash Hemant Patil UID: 2016120036

Objective: To create a range of graphs to summarize your data and results

Outcomes:

1. To create boxplot, scatter plots, including correlation plots
2. To create line graphs, pie charts and bar charts
3. To save graphs as files on disk (png, jpg etc.)
4. To choose the right type of chart for your specific objectives and how to implement it in R using ggplot2.

System Requirements: Ubuntu OS with R and RStudio installed and ggplot2 Introduction to Visualization: Data visualization is an art of how to turn numbers into useful knowledge. [1] Graphs are a powerful way to present your data and results in a concise manner. Whatever kind of data you have, there is a way to illustrate it graphically. A graph is more readily understandable than words and numbers, and producing good graphs is a vital skill. Some graphs are also useful in examining data so that you can gain some idea of patterns that may exist; this can direct you toward the correct statistical analysis. Selecting the Right Chart Type: There are four basic presentation types:

1. Comparison
2. Composition
3. Distribution
4. Relationship

To determine which amongst these is best suited for your data, we suggest you should answer a few questions like,[1]

How many variables do you want to show in a single chart? How many data points will you display for each variable? Will you display values over a period, or among items or groups?

Do refer the following picture and understand how select a right chart type.

[**Courtesy: Dr. Andrew Abela**]

In your day-to-day activities, you'll come across the below listed 7 charts most of the time.

1. Scatter Plot

2. Histogram

3. Bar & Stack Bar Chart

4. Box Plot

5. Area Chart

6. Heat Map

7. Correlogram

[1] **Pre-reading material and understand (for 60 minutes)**

Refer the following website;

https://www.r-bloggers.com/7-visualizations-you-should-learn-in-r/

[2] **Read and perform laboratory on data visualization with R Graphics [2 hr.]**

Refer the following website:

http://r-statistics.co/Top50-Ggplot2-Visualizations-MasterList-R-Code.html

Laboratory Session:

Procedure:

1. Open RStudio

2. Go to RConsole (>)

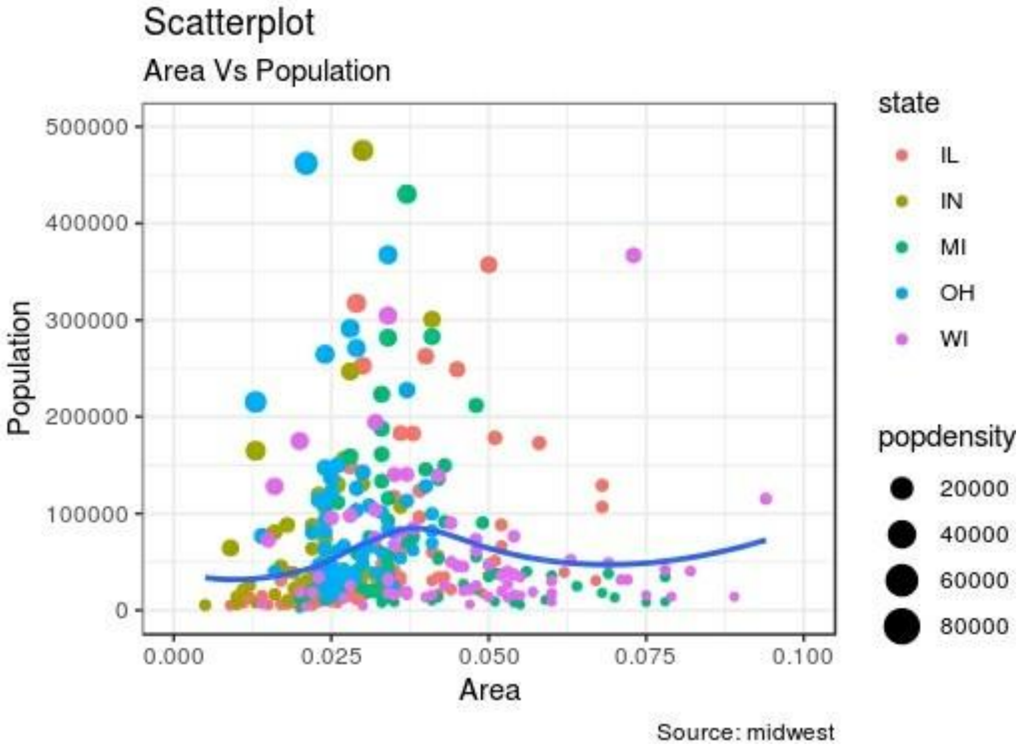
Practice the following 8 categories of plots (graphs):

1. Correlation:

Code:

```
options(scipen=999)

library(ggplot2)
theme_set(theme_bw()) # pre-set the bw theme.
data("midwest", package = "ggplot2")

# Scatterplot
gg <- ggplot(midwest, aes(x=area, y=poptotal)) +
  geom_point(aes(col=state, size=popdensity)) +
  geom_smooth(method="loess", se=F) +
  xlim(c(0, 0.1)) +
  ylim(c(0, 500000)) +
  labs(subtitle="Area Vs Population",
       y="Population",
       x="Area",
       title="Scatterplot",
       caption = "Source: midwest")

plot(gg)
```

Graph:

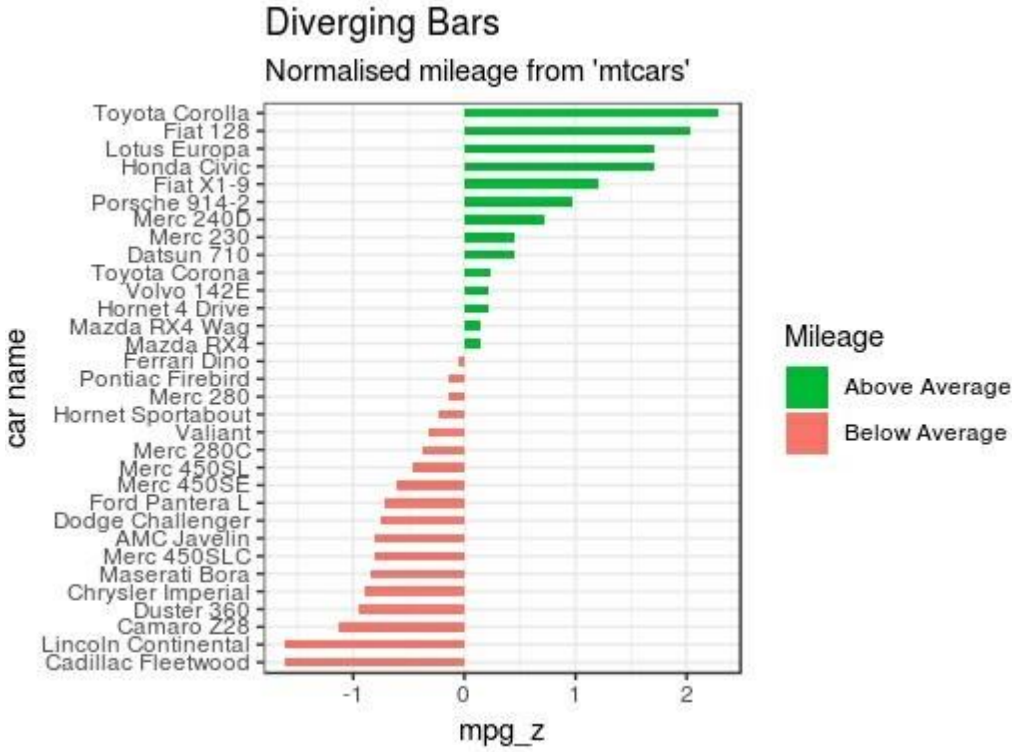


2. Deviation:

Code:

```
library(ggplot2)

theme_set(theme_bw())

# Data Prep
data("mtcars") # load data
mtcars$`car name` <- rownames(mtcars)
mtcars$mpg_z <- round((mtcars$mpg - mean(mtcars$mpg))/sd(mtcars$mpg), 2)
mtcars$mpg_type <- ifelse(mtcars$mpg_z < 0, "below", "above")
mtcars <- mtcars[order(mtcars$mpg_z), ]
mtcars$`car name` <- factor(mtcars$`car name`, levels = mtcars$`car name`)

ggplot(mtcars, aes(x=`car name`, y=mpg_z, label=mpg_z)) +
  geom_bar(stat='identity', aes(fill=mpg_type), width=.5) +
  scale_fill_manual(name="Mileage",
                    labels = c("Above Average", "Below Average"),
                    values = c("above"="#00ba38", "below"="#f8766d")) +
  labs(subtitle="Normalised mileage from 'mtcars'",
       title= "Diverging Bars") +
  coord_flip()
```

Graph:



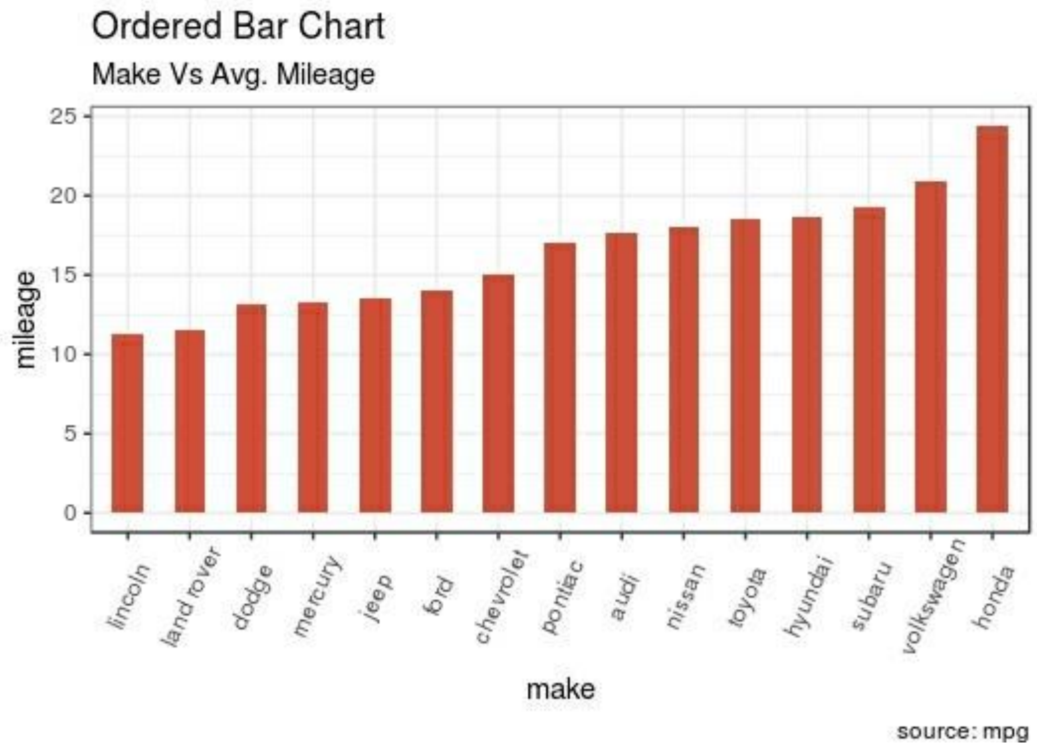
3.

Ranking:

Code:

```
# Prepare data: group mean city mileage by manufacturer.

cty_mpg <- aggregate(mpg$cty, by=list(mpg$manufacturer), FUN=mean)

colnames(cty_mpg) <- c("make", "mileage")
cty_mpg <- cty_mpg[order(cty_mpg$mileage), ]
cty_mpg$make <- factor(cty_mpg$make, levels = cty_mpg$make)
head(cty_mpg, 4)

library(ggplot2)

theme_set(theme_bw())

# Draw plot
ggplot(cty_mpg, aes(x=make, y=mileage)) +
  geom_bar(stat="identity", width=.5, fill="tomato3") +
  labs(title="Ordered Bar Chart",
       subtitle="Make Vs Avg. Mileage",
       caption="source: mpg") +
  theme(axis.text.x = element_text(angle=65, vjust=0.6))
```

Graph:

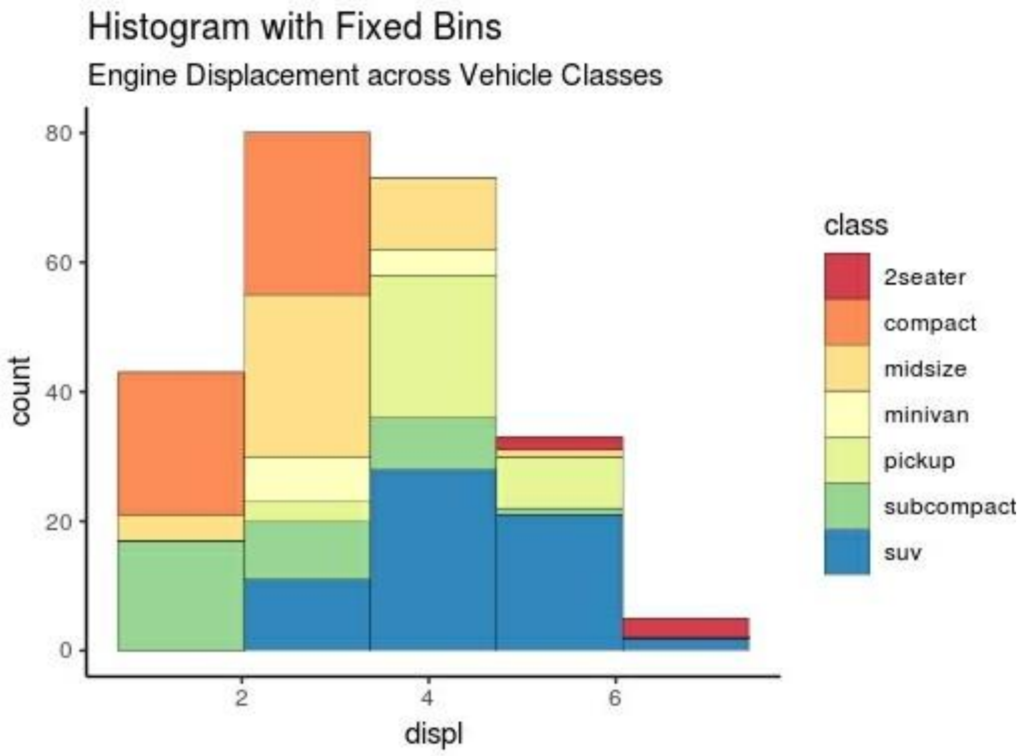


4. Distribution:

Code for Histogram Plot:

```
library(ggplot2)

theme_set(theme_classic())

# Histogram on a Continuous (Numeric) Variable
g <- ggplot(mpg, aes(displ)) + scale_fill_brewer(palette = "Spectral")

g + geom_histogram(aes(fill=class),
                   binwidth = .1,
                   col="black",
                   size=.1) + # change binwidth
  labs(title="Histogram with Auto Binning",
       subtitle="Engine Displacement across Vehicle Classes")

g + geom_histogram(aes(fill=class),
                   bins=5,
                   col="black",
                   size=.1) +  # change number of bins
  labs(title="Histogram with Fixed Bins",
       subtitle="Engine Displacement across Vehicle Classes")
```

Graph for Histogram Plot:

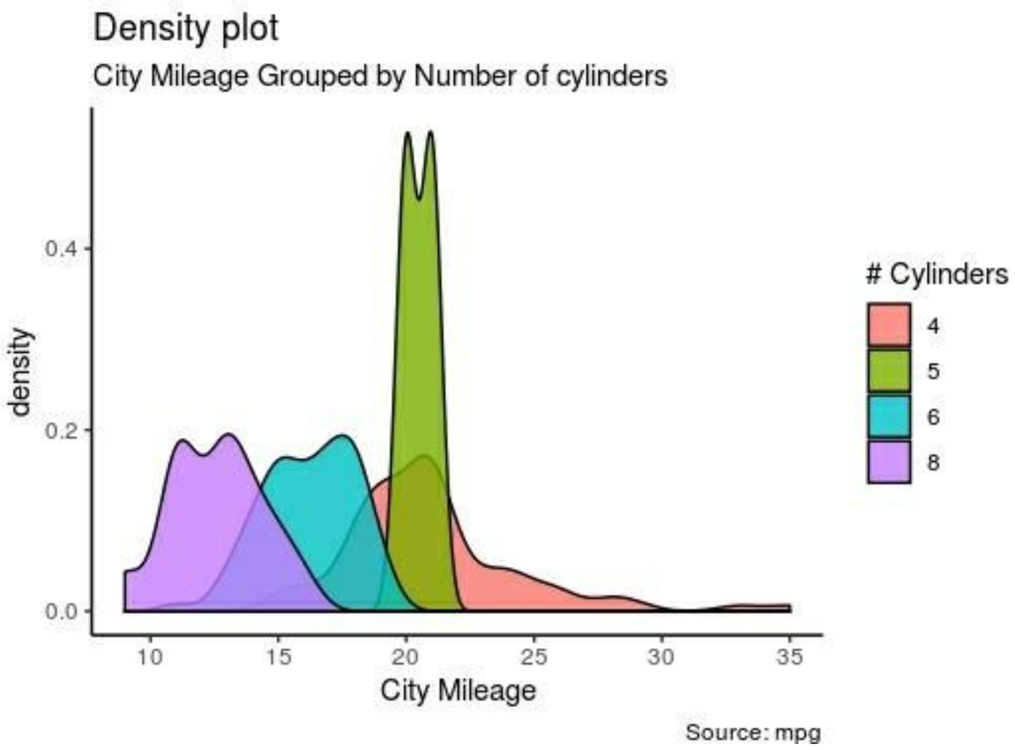


Code

for Density Plot:

```
library(ggplot2)
theme_set(theme_classic())

# Plot
g <- ggplot(mpg, aes(cty))
g + geom_density(aes(fill=factor(cyl)), alpha=0.8) +
   labs(title="Density plot",
       subtitle="City Mileage Grouped by Number of cylinders",
       caption="Source: mpg",
       x="City Mileage",
       fill="# Cylinders")
```

Graph for Density Plot:

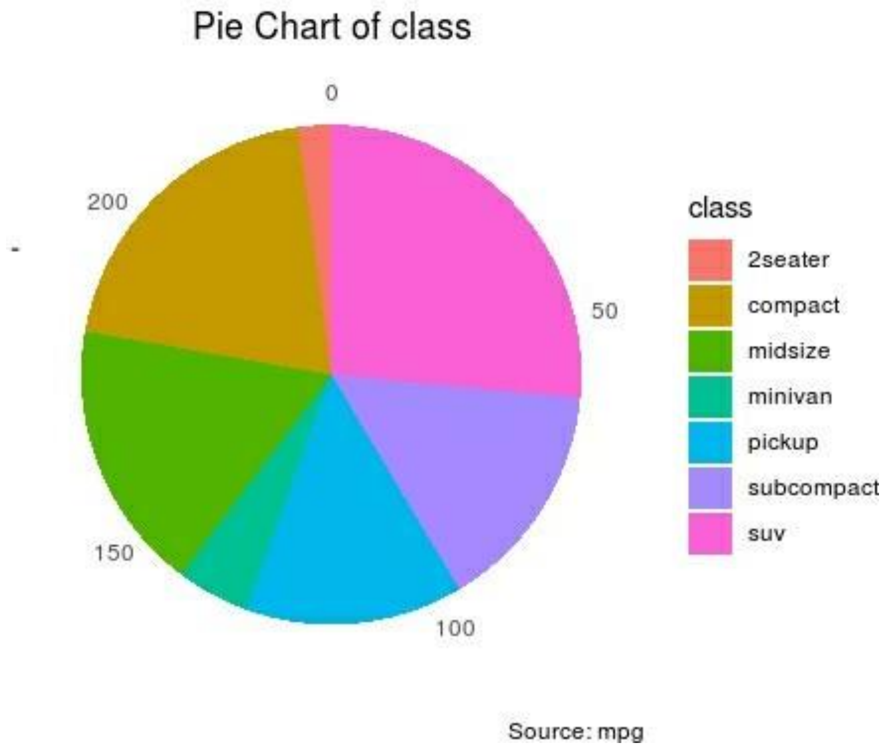


5. Composition:

Code:

```
library(ggplot2)

theme_set(theme_classic())

# Source: Frequency table
df <- as.data.frame(table(mpg$class))
colnames(df) <- c("class", "freq")
pie <- ggplot(df, aes(x = "", y=freq, fill = factor(class))) +
  geom_bar(width = 1, stat = "identity") +
  theme(axis.line = element_blank(),
        plot.title = element_text(hjust=0.5)) +
  labs(fill="class",
       x=NULL,
       y=NULL,
       title="Pie Chart of class",
       caption="Source: mpg")

pie + coord_polar(theta = "y", start=0)

# Source: Categorical variable.
# mpg$class
pie <- ggplot(mpg, aes(x = "", fill = factor(class))) +
  geom_bar(width = 1) +
  theme(axis.line = element_blank(),
        plot.title = element_text(hjust=0.5)) +
  labs(fill="class",
       x=NULL,
       y=NULL,
       title="Pie Chart of class",
       caption="Source: mpg")

pie + coord_polar(theta = "y", start=0)
```

Graph:



Source: mpg

Describe following with respect to data visualization:

1. When to use scatter plot, histogram,bar and stack charts,box plot, Area chart?

Answer: The most frequently used plot for data analysis is undoubtedly the scatterplot. Whenever you want to understand the nature of relationship between two variables, invariably the first choice is the scatterplot.

Histogram is used when you have lots and lots of data points and want to study where and how the data points are distributed.

2. What is Heatmap?

Answer: A heat map (or heatmap) is a graphical representation of data where the individual values contained in a matrix are represented as colors. It is a bit like looking a data table from above. It is really useful to display a general view of numerical data, not to. extract specific data point.

3. What is correlogram?

Answer: In the analysis of data, a correlogram is an image of correlation statistics. For example, in time series analysis, a correlogram, also known as an autocorrelation plot, is a plot of the sample autocorrelations versus (the time lags). ... The correlogram is a commonly used tool for checking randomness in a data set.

4. How to summarize lots of data?

5. Run this script in R script editor and describe:.

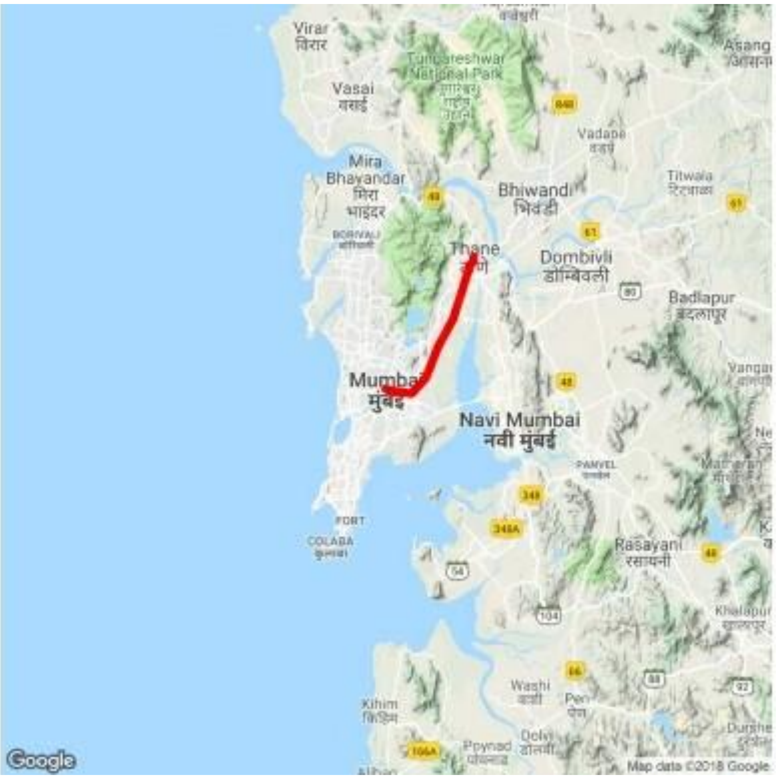
Code:

```
require(ggmap)

require(ggplot2)

from='Mumbai'

to='Thane'

route_df <- route(from, to, structure = 'route', mode = 'driving')

qmap('Mumbai', zoom = 10) +

  geom_path(

    aes(x = lon, y = lat),  colour = 'red', size = 1.5,

    data = route_df, lineend = 'round')

mapdist(from = 'Mumbai', to = 'Thane')
```

Result:

```
 from    to     m     km    miles seconds  minutes
1 Mumbai Thane 22738 22.738 14.12939    1844 30.73333
     hours
```

```
1 0.5122222
```

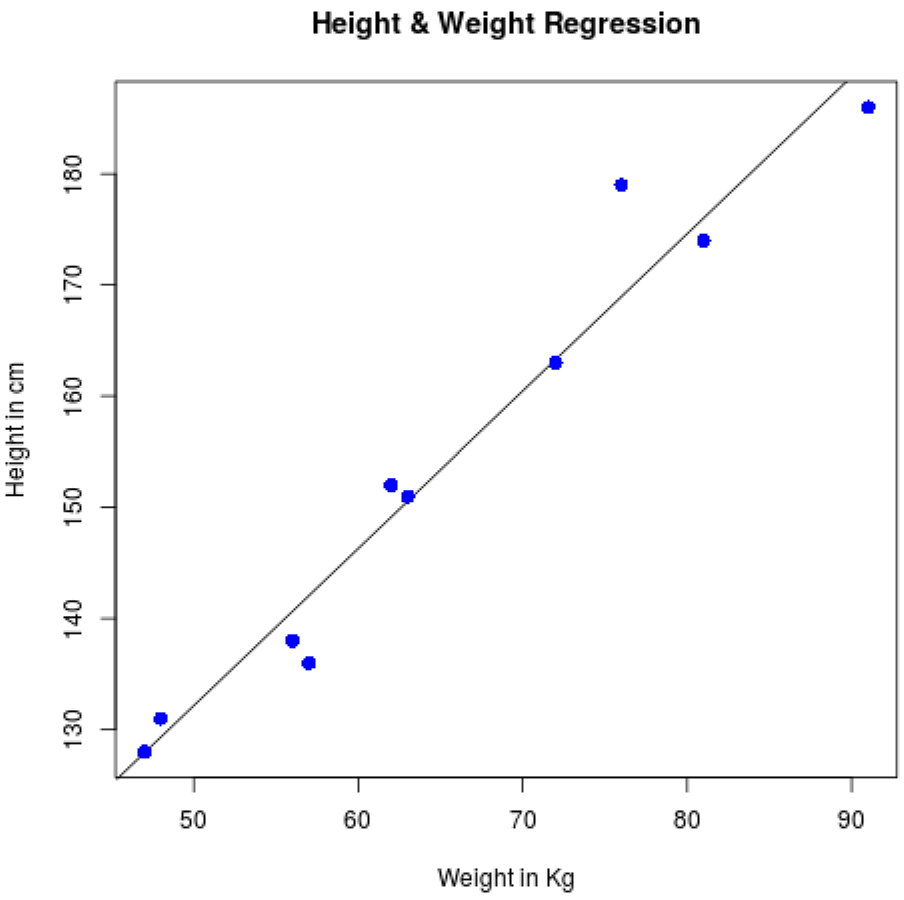


6. Run in Rscript Editor and describe it.

Code:

```
x <- c(151, 174, 138, 186, 128, 136, 179, 163, 152, 131)
y <- c(63, 81, 56, 91, 47, 57, 76, 72, 62, 48)
relation <- lm(y~x)
png(file = "linearregression.png")
plot(y,x,col = "blue",main = "Height & Weight Regression",
abline(lm(x~y)),cex = 1.3,pch = 16,xlab = "Weight in Kg",ylab = "Height in cm")
dev.off()
```

Result:



Conclusion:

ggplot2 is the most elegant and aesthetically pleasing graphics framework available in R.The ggmap package provides facilities to interact with the google maps api and get the coordinates (latitude and longitude) of places you want to plot. We can also zoom into the map by setting the zoom argument. The default is 10.

Thus we learned about Correlation, Deviation, Ranking, Distribution, Composition.

The list of given visualizations sorts the visualizations based on its primary purpose. Primarily, there are 8 types of objectives we can construct plots.

References:

[1] **7 Visualizations You Should Learn in R**

https://www.r-bloggers.com/7-visualizations-you-should-learn-in-r/

[2] **Top 50 ggplot2 Visualizations**

http://r-statistics.co/Top50-Ggplot2-Visualizations-MasterList-R-Code.html

5