Data classification and diagrammatic representation

```
#Pie Chart
x <- c(21, 62, 10, 53)
labels <- c("London", "New York", "Singapore", "Mumbai")
pie(x,labels)
#Pie chart with title
x <- c(21, 62, 10, 53)
labels <- c("London", "New York", "Singapore", "Mumbai")
pie(x, labels, main = "City pie chart", col = rainbow(length(x)))

#Bar plot
H <- c(7,12,28,3,41)
barplot(H)

#Bar plot with labels
H <- c(7,12,28,3,41)
M <- c("Mar","Apr","May","Jun","Jul")
barplot(H,names.arg=M,xlab="Month",ylab="Revenue",col="blue",main="R
evenue chart",border="red")

#Box plot
input <- mtcars[,c('mpg','cyl')]
print(head(input))
boxplot(mpg ~ cyl, data = mtcars, xlab = "Number of Cylinders", ylab
= "Miles Per Gallon", main = "Mileage Data")

#Histogram
v <-  c(9,13,21,8,36,22,12,41,31,33,19)
hist(v,xlab = "Weight",col = "yellow",border = "blue")

#Line graph
# plot(v,type,col,xlab,ylab)
v <- c(7,12,28,3,41)
```

```
plot(v,type = "o")
```

```
                    #Line Graph with heading
v <- c(7,12,28,3,41)
plot(v,type = "o", col = "red", xlab = "Month", ylab = "Rain fall",
main = "Rain fall chart"


#Scatter plots
#plot(x, y, main, xlab, ylab, xlim, ylim, axes)
input <- mtcars[,c('wt','mpg')]
print(head(input))
plot(x = input$wt,y = input$mpg,
   xlab = "Weight",
   ylab = "Milage",
   xlim = c(2.5,5),
   ylim = c(15,30),
   main = "Weight vs Milage"
)


                    #ALL scatter plots together
pairs(~wt+mpg+disp+cyl,data = mtcars, main = "Scatterplot Matrix")


#####################################################
        Categorical variables in R are stored into a factor.
factor(x = character(), levels, labels = levels, ordered =
is.ordered(x))
```

R does not use the terms *nominal*, *ordinal*, and *interval/ratio* for types of variables.

In R, nominal variables can be coded as variables with *factor* or *character* classes.

Continuous variable/Interval/ratio data can be coded as variables with *numeric* or *integer* classes. An *L* used with values to tell R to store the data as an integer class.

We can code ordinal data as either numeric or factor variables, depending on how we will be summarizing, plotting, and analyzing it.

```
sex <- factor(c("male", "female", "female", "male"))

levels(sex)

nlevels(sex)
```

Nominal Categorical Variable

```
# Create a color vector

color_vector <- c('blue', 'red', 'green', 'white', 'black',
'yellow')

# Convert the vector to factor

factor_color <- factor(color_vector)

factor_color
```

Ordinal Categorical Variable

```
# Create Ordinal categorical vector

day_vector <- c('evening', 'morning', 'afternoon', 'midday',
'midnight', 'evening')

# Convert `day_vector` to a factor with ordered level

factor_day <- factor(day_vector, order = TRUE, levels =c('morning',
'midday', 'afternoon', 'evening', 'midnight'))

# Print the new variable

factor_day

## Levels: morning < midday < afternoon < evening < midnight

# Append the line to above code

# Count the number of occurence of each level

summary(factor_day)

##Continuous Data

dataset <- mtcars

class(dataset$mpg)
```

```
##Importing data from excel
df <- read.table("<FileName>.txt", header = TRUE)
dat <- read.csv(
file = "data.csv",
header = TRUE,
sep = ",",
dec = "."
)

##Easy enough: at the beginning of your script, simply
add library(readxl) to the list of libraries you are loading. And
tidyverse package by using install.package(tidyverse) command
df <- read_excel("<name and extension of your file>")

install.packages("readxl")
f1 <- read_xlsx(file.choose())
library(readxl)
install.packages("tidyverse")
data("salaries",package= "car")
install.packages("pacman")

# Then load the package by using either of the following:
require(pacman)  # Gives a confirmation message.
library(pacman)  # No message.

# Or, by using "pacman::p_load" you can use the p_load
# function from pacman without actually loading pacman.
# These are packages I load every time.
pacman::p_load(pacman, dplyr, GGally, ggplot2, ggthemes,
        ggvis, httr, lubridate, plotly, rio, rmarkdown, shiny,
        stringr, tidyr)
```

```
library(datasets)  # Load/unload base packages manually

# CLEAN UP ####################################################


# Clear packages

p_unload(dplyr, tidyr, stringr) # Clear specific packages

p_unload(all)  # Easier: clears all add-ons

detach("package:datasets", unload = TRUE)  # For base


# Clear console

cat("\014")  # ctrl+L


# Clear environment

rm(list = ls())


# Clear packages

detach("package:datasets", unload = TRUE)  # For base


# Clear plots

dev.off()  # But only if there IS a plot


# Clear console

cat("\014")  # ctrl+L


# Clear mind :)
```

Exercises

1 Draw a histogram of sepal width from dataset iris. How would you describe this?

2 Create a csv file for the following data

```
##   Source     Cu
## 1  Site1 19.700
## 2  Site2 10.643
## 3  Site1 33.792
## 4  Site2  5.353
## 5  Site2 19.890
## 6  Site2 26.966
```

Draw a boxplot to show atmospheric copper concentration by sites.

3 Create a csv file for the following data

```
##      City ProductA ProductB ProductC
## 1 Seattle       23       11       12
## 2  London       89        6       56
## 3   Tokyo       24        7       13
## 4  Berlin       36       34       44
## 5  Mumbai        3       78       14
```

Draw a bar graph to show total sales across different cities.

4 The dataset swiss contains a standardized fertility measure and various socioeconomic indicators for each of 47 French-speaking provinces of Switzerland in about 1888.

a. Draw a scatterplot of Fertility against %Catholic. Which kind of areas have the lowest fertility rates?
b. Discuss the relationship between the variables Education and Agriculture.

5 Write a R program to change the first level of a factor with another level of a given factor.

6 Write a R program to create an ordered factor from data of minimum 20 elements consisting of the names of months.