

## Day 3 Lab Manual

### UNIVARIATE ANALYSIS IN R - MEASURES OF CENTRAL TENDENCYExercise:

#### I. ARITHMETIC MEAN

a) Write suitable R code to compute the average of the following values.

12,7,3,4.2,18,2,54,-21,8,-5

code:-

```
values <- c(12, 7, 3, 4.2, 18, 2, 54, -21, 8, -5)
```

```
mean(values)
```

```
# Compute the mean after trimming 3 values from each end#
```

```
print("USING TRIM")
```

```
mean(values, trim = 0.3)
```

```
values <- c(12, 7, 3, 4.2, 18, 2, 54, -21, 8, -5, NA)
```

```
mean(values, na.rm = TRUE)
```

b) Compute the mean after applying the trim option and removing 3 values from eachend.

c) Compute the mean of the following vector .

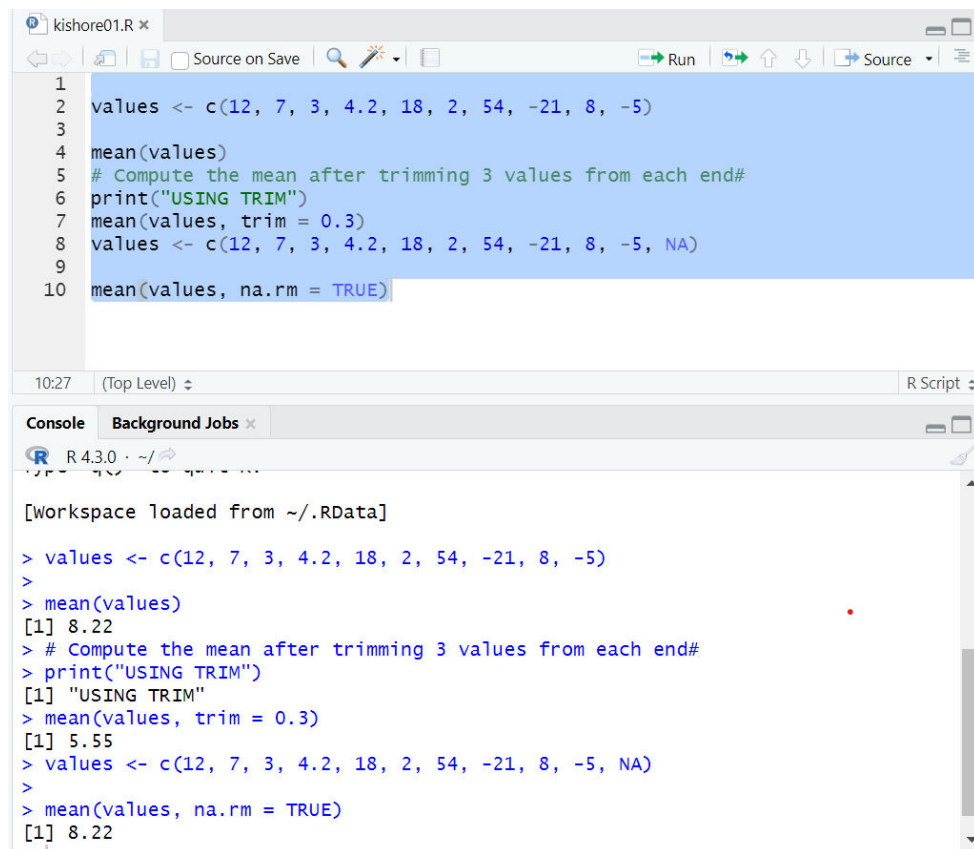
(12,7,3,4.2,18,2,54,-21,8,-5,NA)

```
#If there are missing values, then the mean function returns NA.
```

```
# Find mean dropping NA values.
```

```
#To drop the missing values from the calculation use na.rm = TRUE
```

SAMPLE INPUT/OUTPUT:-



The screenshot shows an RStudio interface. The top pane displays R code for calculating the mean of a vector with and without trimming. The bottom pane shows the console output of these commands.

```
1 values <- c(12, 7, 3, 4.2, 18, 2, 54, -21, 8, -5)
2
3
4 mean(values)
5 # Compute the mean after trimming 3 values from each end#
6 print("USING TRIM")
7 mean(values, trim = 0.3)
8 values <- c(12, 7, 3, 4.2, 18, 2, 54, -21, 8, -5, NA)
9
10 mean(values, na.rm = TRUE)
```

Console Output:

```
[Workspace loaded from ~/.RData]
> values <- c(12, 7, 3, 4.2, 18, 2, 54, -21, 8, -5)
>
> mean(values)
[1] 8.22
> # Compute the mean after trimming 3 values from each end#
> print("USING TRIM")
[1] "USING TRIM"
> mean(values, trim = 0.3)
[1] 5.55
> values <- c(12, 7, 3, 4.2, 18, 2, 54, -21, 8, -5, NA)
>
> mean(values, na.rm = TRUE)
[1] 8.22
```

## II.MEDIAN

Write suitable R code to compute the median of the following values.

12,7,3,4.2,18,2,54,-21,8,-5

CODE:-

```
values <- c(12, 7, 3, 4.2, 18, 2, 54, -21, 8, -5)

median(values)
```

SAMPLE INPUT/OUTPUT:-

```
1 # Define a vector containing the values
2 values <- c(12, 7, 3, 4.2, 18, 2, 54, -21, 8, -5)
3
4 # Compute the median of the values
5 median(values)

> median(values)
[1] 5.6
```

### III. MODE

Calculate the mode for the following numeric as well as character data set in R.

(2,1,2,3,1,2,3,4,1,5,5,3,2,3) , ("o","it","the","it","it")

#### UNIVARIATE ANALYSIS IN R - MEASURES OF DISPERSION

CODE:-

```
numeric_data <- c(2,1,2,3,1,2,3,4,1,5,5,3,2,3)

get_mode <- function(x) {

  tab <- table(x)

  as.numeric(names(tab)[tab == max(tab)])

}

get_mode(numeric_data)

char_data <- c("o","it","the","it","it")

get_mode <- function(x) {

  tab <- table(x)

  as.character(names(tab)[tab == max(tab)])

}

get_mode(char_data)
```

SAMPLE INPUT/OUTPUT:-

```
kishore01.R*
Source on Save Run Source
1 numeric_data <- c(2,1,2,3,1,2,3,4,1,5,5,3,2,3)
2 get_mode <- function(x) {
3   tab <- table(x)
4   as.numeric(names(tab)[tab == max(tab)])
5 }
6 get_mode(numeric_data)
7 char_data <- c("o","it","the","it","it")
8 get_mode <- function(x) {
9   tab <- table(x)
10  as.character(names(tab)[tab == max(tab)])
11 }
12 get_mode(char_data)
13

13:1 (Top Level) R Script

Console Background Jobs
R 4.3.0 · ~/
> numeric_data <- c(2,1,2,3,1,2,3,4,1,5,5,3,2,3)
> get_mode <- function(x) {
+   tab <- table(x)
+   as.numeric(names(tab)[tab == max(tab)])
+ }
> get_mode(numeric_data)
[1] 2 3
> char_data <- c("o","it","the","it","it")
> get_mode <- function(x) {
+   tab <- table(x)
+   as.character(names(tab)[tab == max(tab)])
+ }
> get_mode(char_data)
[1] "it"
>
```

#### Exercise: 4

Download mpg dataset which contains Fuel economy data from 1999 and 2008 for 38

popular models of car from the URL given below.

<https://vincentarelbundock.github.io/Rdatasets/datasets.html>

Answer the following queries

- i) Find the car which gives maximum city miles per gallon
- ii) Find the cars which gives minimum disp in compact and subcompact class

CODE:-

```
library(utils)

download.file("https://github.com/tidyverse/ggplot2/blob/main/data-raw/mpg.csv", "mpg.csv")

mpg <- read.csv("mpg.csv")
```

```
max_city_mpg <- max(mpg$cty)
```

```
max_city_mpg_car <- mpg[mpg$cty == max_city_mpg, "model"]
```

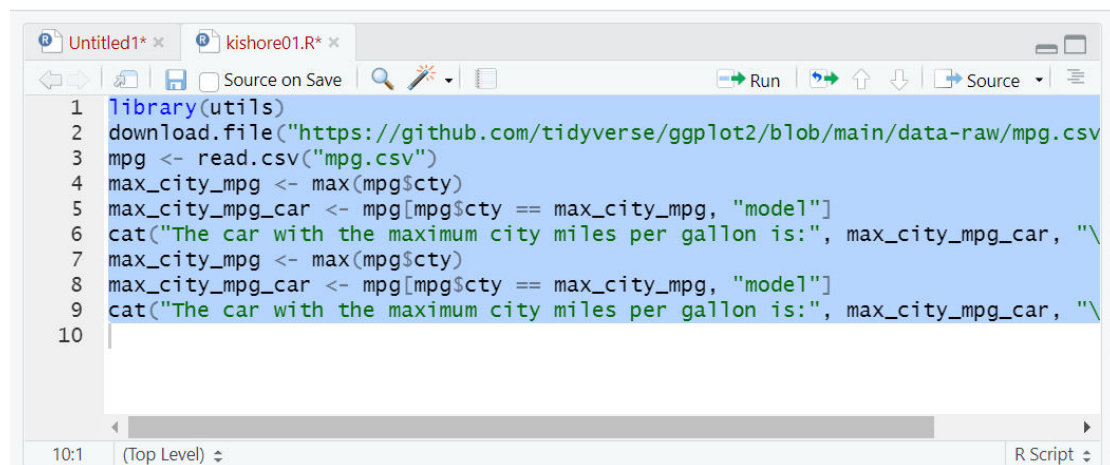
```
cat("The car with the maximum city miles per gallon is:", max_city_mpg_car, "\n")
```

```
max_city_mpg <- max(mpg$cty)
```

```
max_city_mpg_car <- mpg[mpg$cty == max_city_mpg, "model"]
```

```
cat("The car with the maximum city miles per gallon is:", max_city_mpg_car, "\n")
```

SAMPLE INPUT/OUTPUT:-



```
1 library(utils)
2 download.file("https://github.com/tidyverse/ggplot2/blob/main/data-raw/mpg.csv")
3 mpg <- read.csv("mpg.csv")
4 max_city_mpg <- max(mpg$cty)
5 max_city_mpg_car <- mpg[mpg$cty == max_city_mpg, "model"]
6 cat("The car with the maximum city miles per gallon is:", max_city_mpg_car, "\n")
7 max_city_mpg <- max(mpg$cty)
8 max_city_mpg_car <- mpg[mpg$cty == max_city_mpg, "model"]
9 cat("The car with the maximum city miles per gallon is:", max_city_mpg_car, "\n")
10
```

#### Exercise: 5

Use the same dataset as used in Exercise 4 and perform the following queries

- i) Find the standard deviation of city miles per gallon
- ii) Find the variance of highway miles per gallon

#### Exercise 6

Use the same dataset and perform the following queries

- i) Find the range of the disp in the data set mpg
- ii) Find the Quartile of the disp in the data set mpg
- iii) Find the IQR of the disp column in the data set mpg

#### Exercise 7

```
#Install Library
```

```
library(e1071)
```

a. Find the skewness of city miles per mileage in the data set mpg ?

Use qplot function and display the graph for the city miles per mileage columnb. Find the kurtosis of city miles per mileage in the data set mpg