

Day 3 Lab Manual

UNIVARIATE ANALYSIS IN R - MEASURES OF CENTRAL TENDENCY

Exercise:

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

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

SOURCE CODE

#a)

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

```
mean(values)
```

#b)

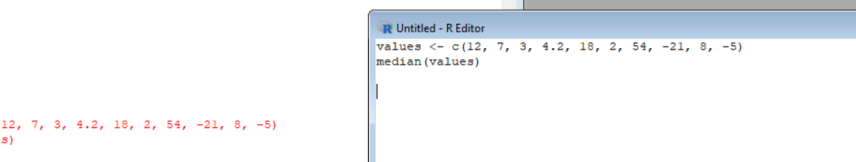
```
values <- c(12, 7, 3, 4.2, 18, 2, 54, -21, 8, -5)
```

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

#c

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

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



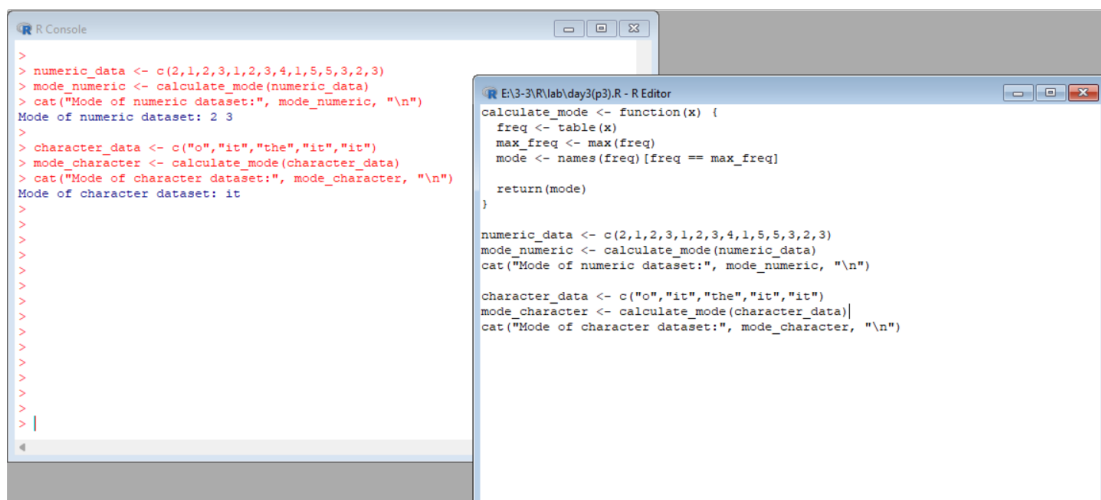
The screenshot displays two windows from the RStudio interface. The 'R Console' window on the left shows a series of prompt characters '>' followed by the execution of the command `median(values)`, which returns the output `[1] 5.6`. The 'Untitled - R Editor' window on the right shows the R script code: `values <- c(12, 7, 3, 4.2, 18, 2, 54, -21, 8, -5)` followed by `median(values)` on a new line.

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")

SOURCE CODE:

```
calculate_mode <- function(x) {  
  
  freq <- table(x)  
  
  max_freq <- max(freq)  
  
  mode <- names(freq)[freq == max_freq]  
  
  return(mode)  
  
}  
  
numeric_data <- c(2,1,2,3,1,2,3,4,1,5,5,3,2,3)  
  
mode_numeric <- calculate_mode(numeric_data)  
  
cat("Mode of numeric dataset:", mode_numeric, "\n")  
  
character_data <- c("o","it","the","it","it")  
  
mode_character <- calculate_mode(character_data)  
  
cat("Mode of character dataset:", mode_character, "\n")
```



The screenshot displays two windows from an R environment. The 'R Console' window on the left shows the execution of the provided R code. It confirms the mode for the numeric dataset as '2 3' and for the character dataset as 'it'. The 'R Editor' window on the right shows the source code being executed, which defines the 'calculate_mode' function and applies it to the specified data sets.

UNIVARIATE ANALYSIS IN R - MEASURES OF DISPERSION

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

SOURCE CODE:

```
mpg <-  
read.csv("https://vincentarelbundock.github.io/Rdatasets/csv/ggplot2/mpg.csv")  
  
head(mpg)  
  
max_city_mpg_row <- which.max(mpg$cty)  
  
max_city_mpg_car <- paste(mpg$manufacturer[max_city_mpg_row],  
mpg$model[max_city_mpg_row])  
  
cat("Car which gives maximum city miles per gallon:", max_city_mpg_car, "\n")  
  
compact_cars <- subset(mpg, class %in% c("compact", "subcompact"))  
  
min_disp_compact <- compact_cars[which.min(compact_cars$displ), "model"]  
  
cat("Car which gives minimum displacement in compact class:", min_disp_compact,  
"\n")  
  
subcompact_cars <- subset(mpg, class == "subcompact")  
  
min_disp_subcompact <- subcompact_cars[which.min(subcompact_cars$displ),  
"model"]  
  
cat("Car which gives minimum displacement in subcompact class:",  
min_disp_subcompact, "\n")
```

The screenshot shows two windows from an R environment. The top window is the R console, and the bottom window is the R script editor.

R Console Output:

```
> head(mpg)
  X manufacturer model displ year cyl   trans drv cty hwy fl  class
1 1      audi     a4    1.8 1999   4   auto(l5)  f  18  29 p compact
2 2      audi     a4    1.8 1999   4 manual(m5)  f  21  29 p compact
3 3      audi     a4    2.0 2008   4 manual(m6)  f  20  31 p compact
4 4      audi     a4    2.0 2008   4 auto(av)   f  21  30 p compact
5 5      audi     a4    2.8 1999   6 auto(l5)   f  16  26 p compact
6 6      audi     a4    2.8 1999   6 manual(m6)  f  18  26 p compact

> max_city_mpg_row <- which.max(mpg$cty)
> max_city_mpg_car <- paste(mpg$manufacturer[max_city_mpg_row], mpg$model[max_c
> cat("Car which gives maximum city miles per gallon:", max_city_mpg_car, "\n")
Car which gives maximum city miles per gallon: volkswagen new beetle

>
> compact_cars <- subset(mpg, class %in% c("compact", "subcompact"))
> min_disp_compact <- compact_cars[which.min(compact_cars$displ), "model"]
> cat("Car which gives minimum displacement in compact class:", min_disp_compact
Car which gives minimum displacement in compact class: civic

>
> subcompact_cars <- subset(mpg, class == "subcompact")
> min_disp_subcompact <- subcompact_cars[which.min(subcompact_cars$displ), "mod
> cat("Car which gives minimum displacement in subcompact class:", min_disp_sub
Car which gives minimum displacement in subcompact class: civic

> |
```

R Script Editor Content:

```
mpg <- read.csv("https://vincentarelbundock.github.io/Rdatasets/csv/ggplot2/mpg.
head(mpg)

max_city_mpg_row <- which.max(mpg$cty)
max_city_mpg_car <- paste(mpg$manufacturer[max_city_mpg_row], mpg$model[max_c
cat("Car which gives maximum city miles per gallon:", max_city_mpg_car, "\n")

compact_cars <- subset(mpg, class %in% c("compact", "subcompact"))
min_disp_compact <- compact_cars[which.min(compact_cars$displ), "model"]
cat("Car which gives minimum displacement in compact class:", min_disp_compact,

subcompact_cars <- subset(mpg, class == "subcompact")
min_disp_subcompact <- subcompact_cars[which.min(subcompact_cars$displ), "model"]
cat("Car which gives minimum displacement in subcompact class:", min_disp_subcomp
```

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

SOURCE CODE:

```
range_disp <- range(mpg$disp)
```

```
cat("Range of disp in the data set mpg:", range_disp, "\n")
```

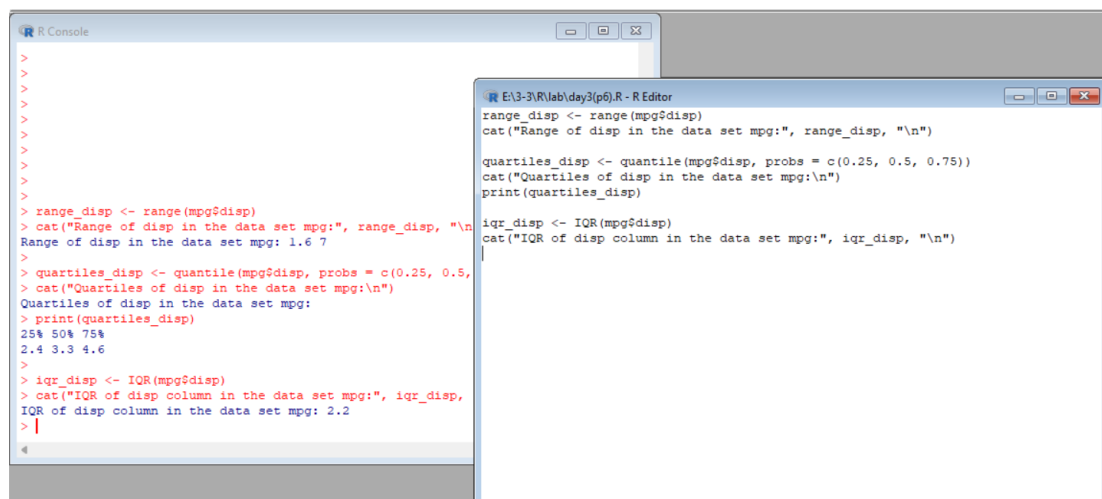
```
quartiles_disp <- quantile(mpg$disp, probs = c(0.25, 0.5, 0.75))
```

```
cat("Quartiles of disp in the data set mpg:\n")
```

```
print(quartiles_disp)
```

```
iqr_disp <- IQR(mpg$disp)
```

```
cat("IQR of disp column in the data set mpg:", iqr_disp, "\n")
```



The screenshot shows two windows from an R environment. The 'R Console' window on the left displays the execution of the code, with output for the range, quartiles, and IQR of the 'disp' variable. The 'R Editor' window on the right shows the source code being executed.

```
>
>
>
>
>
>
>
>
> range_disp <- range(mpg$disp)
> cat("Range of disp in the data set mpg:", range_disp, "\n")
Range of disp in the data set mpg: 1.6 7
>
> quartiles_disp <- quantile(mpg$disp, probs = c(0.25, 0.5, 0.75))
> cat("Quartiles of disp in the data set mpg:\n")
Quartiles of disp in the data set mpg:
> print(quartiles_disp)
25% 50% 75%
2.4 3.3 4.6
>
> iqr_disp <- IQR(mpg$disp)
> cat("IQR of disp column in the data set mpg:", iqr_disp, "\n")
IQR of disp column in the data set mpg: 2.2
> |
```

```
E:\3-3\R\lab\day3(p6).R - R Editor
range_disp <- range(mpg$disp)
cat("Range of disp in the data set mpg:", range_disp, "\n")

quartiles_disp <- quantile(mpg$disp, probs = c(0.25, 0.5, 0.75))
cat("Quartiles of disp in the data set mpg:\n")
print(quartiles_disp)

iqr_disp <- IQR(mpg$disp)
cat("IQR of disp column in the data set mpg:", iqr_disp, "\n")
```

Exercise 7

#Install Library

```
library(e1071)
```

- 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 column

- Find the kurtosis of city miles per mileage in the data set mpg

SOURCE CODE:

```
library(e1071)
```

```
library(ggplot2)
```

```
skew_cty <- skewness(mpg$cty)
```

```
cat("Skewness of city miles per gallon in the data set mpg:", skew_cty, "\n")
```

```
qplot(mpg$cty, geom="histogram", binwidth=2, main="City Miles Per Gallon",  
xlab="Miles Per Gallon")
```

```
kurt_cty <- kurtosis(mpg$cty)
```

```
cat("Kurtosis of city miles per gallon in the data set mpg:", kurt_cty, "\n")
```

