Practical 02

FYCS Descriptive Statistics

Aim : To obtain frequency distribution and data presentation using R

a. Frequency Distribution

A frequency distribution is a summary of the number of times different values occur in a dataset. It organizes data into classes or intervals and shows the count (frequency) of observations falling into each class. The purpose of creating a frequency distribution is to provide a compact and organized way to understand the distribution of values within a dataset.

Frequency distributions are commonly used in statistics to provide a concise summary of large datasets, making it easier to identify patterns, trends, or central tendencies. They are often presented in tabular form or visualized using histograms, bar charts, or other graphical representations.

To convert a raw data to discrete grouped data we use the table( ) function:

```
# Given data
x = c(2,3,12,2,9,17,3,15,15,15,3,8,9,2,8)

# Create a frequency table
frequency = table(x)

# display the result
frequency
```

```
## x
##  2  3  8  9 12 15 17
##  3  3  2  2  1  3  1
```

```
# for displaying the frequency table vertically
print(cbind(frequency))
```

```
##    frequency
## 2          3
## 3          3
## 8          2
## 9          2
## 12         1
## 15         3
## 17         1
```

To convert raw data to a continuous grouped data we use the following functions :

c( ), log10( ), length( ), max( ),min( ) , seq( ), cut( ) ,table( ), cbind( )

```
# Given data
x = c(19, 03, 12, 23, 05, 11, 25, 17, 22, 19, 21, 06, 20, 18, 27, 07, 25, 21, 10,
12, 07, 07)

# Calculate k and w from sturge formula
k = 1 + 3.322 * log10(length(x))
w = (max(x) - min(x)) / k

# Define breaks
breaks = seq(3, 28, 5)

# Create intervals
intervals = cut(x, breaks, right = FALSE)
# cut() function assigns each data to respective class.

# Calculate frequency
frequency = table(intervals)

# Display the result
result = cbind(frequency)
print(result)
```

```
##         frequency
## [3,8)           6
## [8,13)          4
## [13,18)         1
## [18,23)         7
## [23,28)         4
```

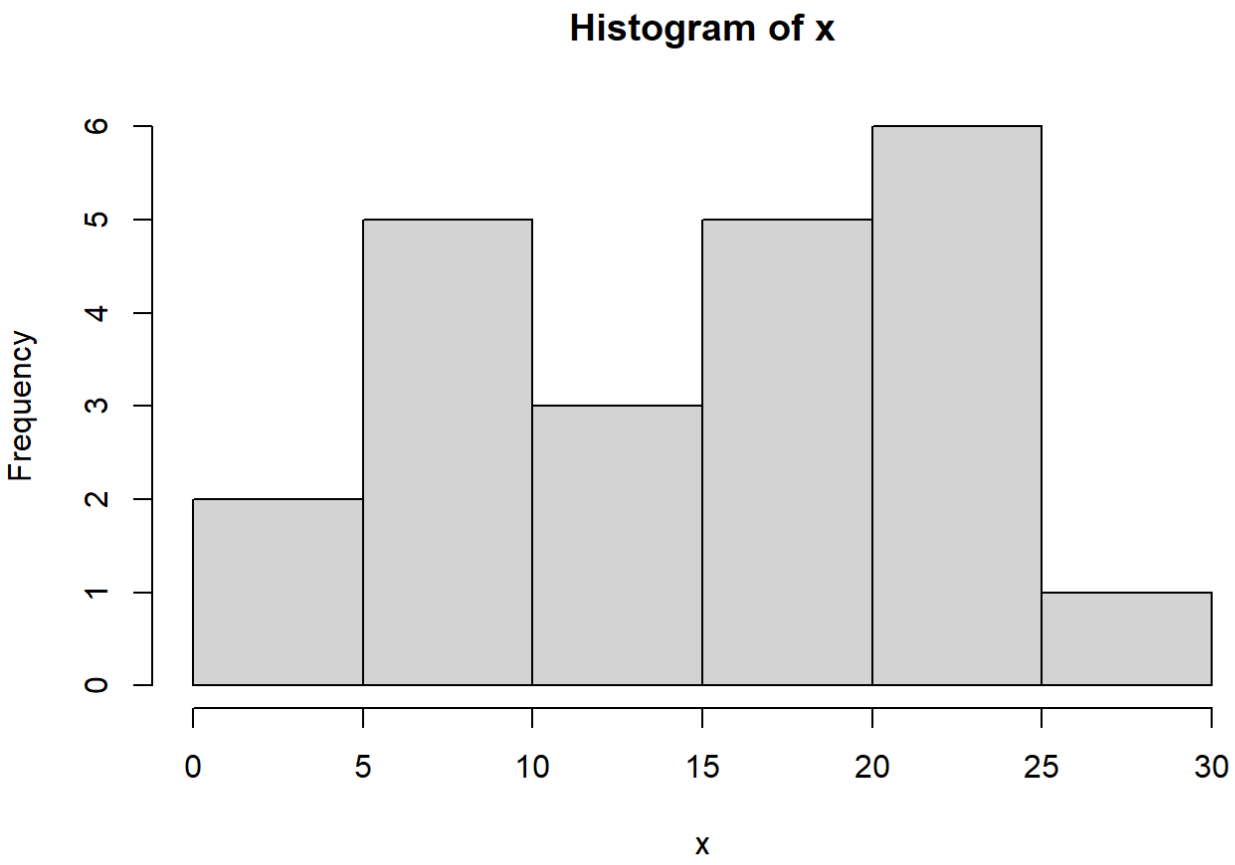
b. Diagrams

Q1) Draw a histogram for the following data.

19, 03, 12, 23, 05, 11, 25, 17, 22, 19, 21, 06, 20, 18, 27, 07, 25, 21, 10, 12, 07, 07

```
# histogram type 1
x = c(19, 03, 12, 23, 05, 11, 25, 17, 22, 19, 21, 06, 20, 18, 27, 07, 25, 21, 10,
12, 07, 07)

hist(x)
```



Q2) For the given distribution below, draw a histogram representing data

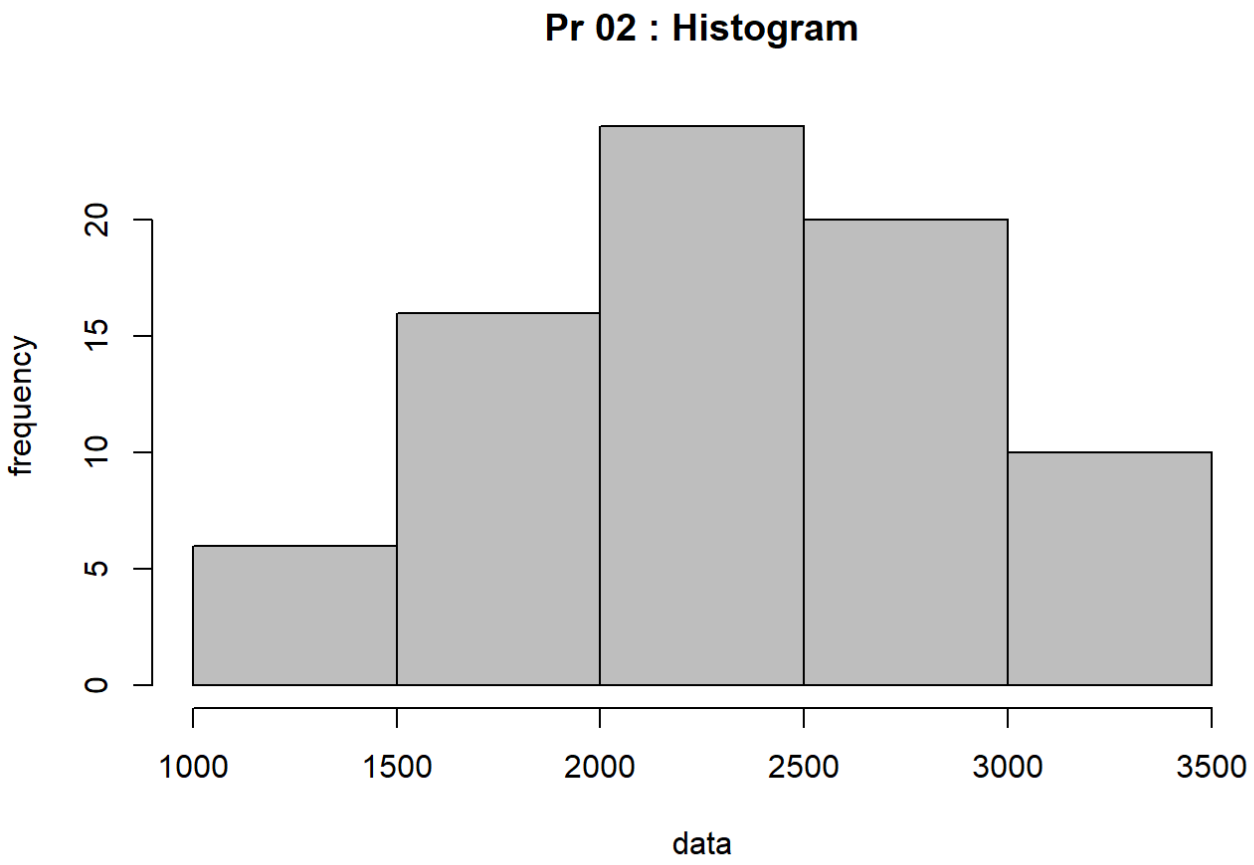
House rent	1000-1500	1500-2000	2000-2500	2500-3000	3000-3500
# Families	06	16	24	20	10

```
# Given data
lb <- c(1000, 1500, 2000, 2500, 3000)
ub <- c(1500, 2000, 2500, 3000, 3500)
cm <- (lb + ub) / 2
f <- c(6, 16, 24, 20, 10)

# Create grouped continuous data
y <- rep(cm, f)

# Define breaks
breaks <- c(lb[1], ub)

# Plot histogram
hist(y,
     breaks = breaks,
     col = 'grey',
     ylab = 'frequency',
     xlab = 'data',
     main = 'Pr 02 : Histogram')
```

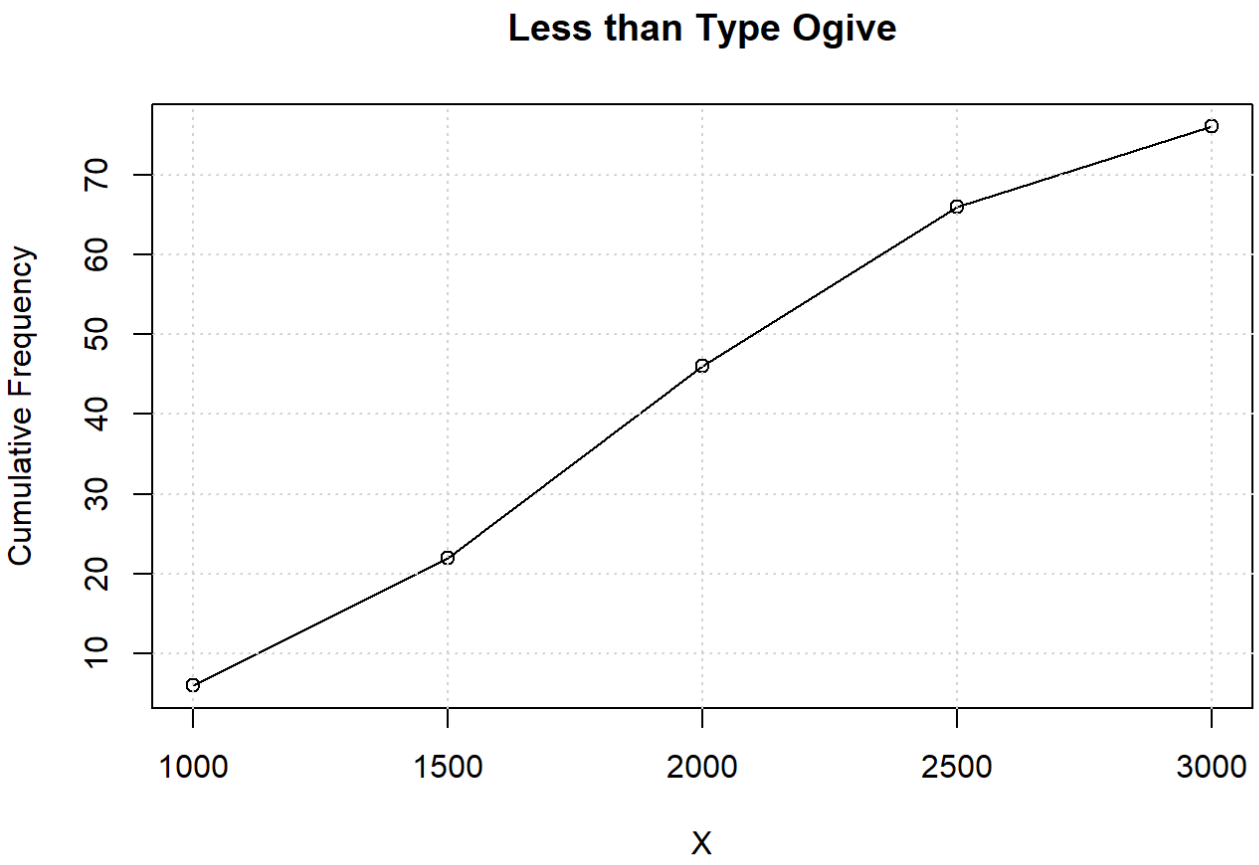


c. Graphs

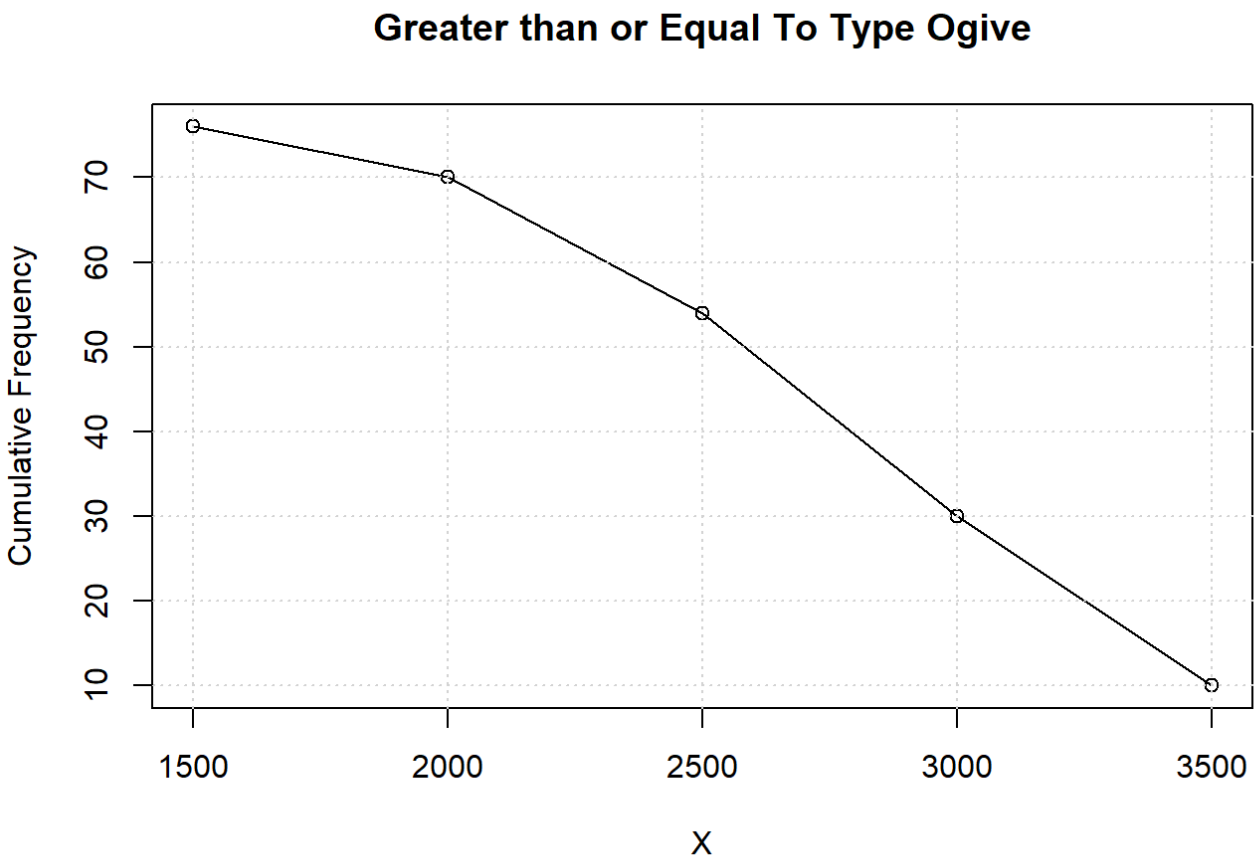
Q1) Draw ogive curves for the following data

```
# Given data
lb <- c(1000, 1500, 2000, 2500, 3000)
ub <- c(1500, 2000, 2500, 3000, 3500)
f <- c(6, 16, 24, 20, 10)

# Less than type cumulative curve
lcf <- cumsum(f)
plot(lb,
     lcf,
     "line",
     main = "Less than Type Ogive",
     xlab = "X",
     ylab = "Cumulative Frequency")
points(lb,lcf)
grid()
```




```
# Greater than or equal to type cumulative curve
mcf <- rev(cumsum(rev(f)))
plot(ub,
     mcf,
     "line",
     main = "Greater than or Equal To Type Ogive",
     xlab = "X",
     ylab = "Cumulative Frequency")
points(ub,mcf)
grid()
```


Greater than or Equal To Type Ogive
Cumulative Frequency
10
20
30
40
50
60
70
1500
2000
2500
3000
3500
X