

# Sri Lanka Institute of Information Technology



Lab Submission

<Lab sheets No 5>

<IT24104167>

<Jayawardena K.D>

**Probability and Statistics | IT2120**

B.Sc. (Hons) in Information Technology

```

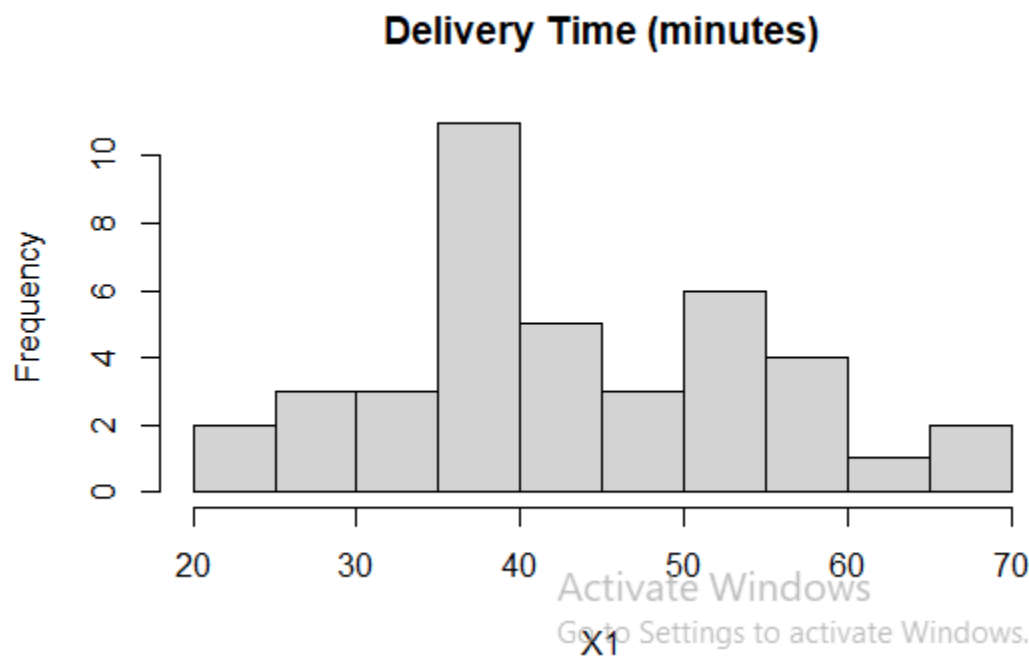
1 getwd()
2 setwd('C:\\Users\\it24104167\\Downloads\\IT24104167')
3 getwd()
4
5 ##import dataset
6 Delivery_Times <- read.table("Exercise - Lab 05.txt", header = TRUE, sep = ",")
7 Delivery_Times
8 ##to get dataset in new window
9 fix(Delivery_Times)
10 ##rename variables in dataset
11 names(Delivery_Times) <-c("X1")
12 fix(Delivery_Times)
13 ##call variable by their name
14 attach(Delivery_Times)
15
> getwd()
[1] "C:/Users/it24104167/Downloads/IT24104167"
> setwd('C:\\Users\\it24104167\\Downloads\\IT24104167')
> getwd()
[1] "C:/Users/it24104167/Downloads/IT24104167"
> ##import dataset
> Delivery_Times <- read.table("Exercise - Lab 05.txt", header = TRUE, sep = ",")
> Delivery_Times
  Delivery_Time_.minutes.
1                      34
2                      54
3                      47
4                      29
5                      39
6                      61
7                      20
8                      40
9                      57
10                     36
11                     38
12                     44
13                     59
14                     38
15                     40
16                     40
17                     67
18                     66
19                     55
20                     48
21                     52
22                     59
23                     35
24                     56
25                     32
26                     38
27                     54

```

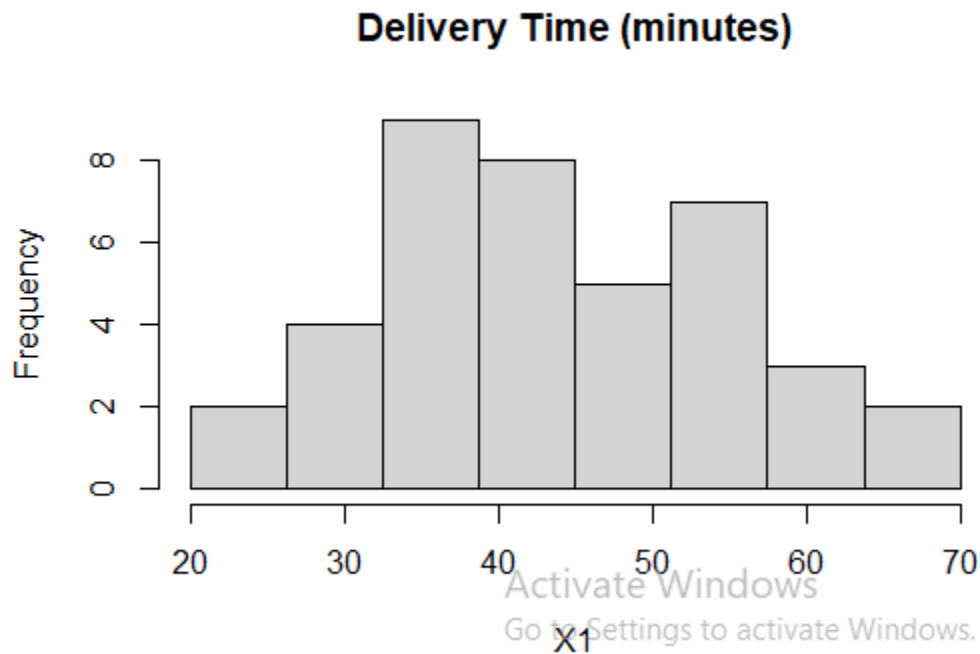
```

27          54
28          30
29          43
30          36
31          42
32          20
33          27
34          38
35          54
36          43
37          45
38          51
39          36
40          47
> ##to get dataset in new window
> fix(Delivery_Times)
> ##rename variables in dataset
> names(Delivery_Times) <-c("X1")
> fix(Delivery_Times)
> ##call variable by their name
> attach(Delivery_Times)
16 ##histogram
17 histogram <-hist(X1, main="Delivery Time (minutes)")
18
19 ##Draw a histogram using 9 classes where the lower limit is 20 and an upper limit of 70.
20 histogram <-hist(X1, main="Delivery Time (minutes)", breaks = seq(20,70, length=9),right = FALSE)
> ##histogram
> histogram <-hist(X1, main="Delivery Time (minutes)")
>

```



```
> ##Draw a histogram using 9 classes where the lower limit is 20 and an upper limit of 70.
> histogram <- hist(x1, main="Delivery Time (minutes)", breaks = seq(20,70, length=9),right = FALSE)
> |
```



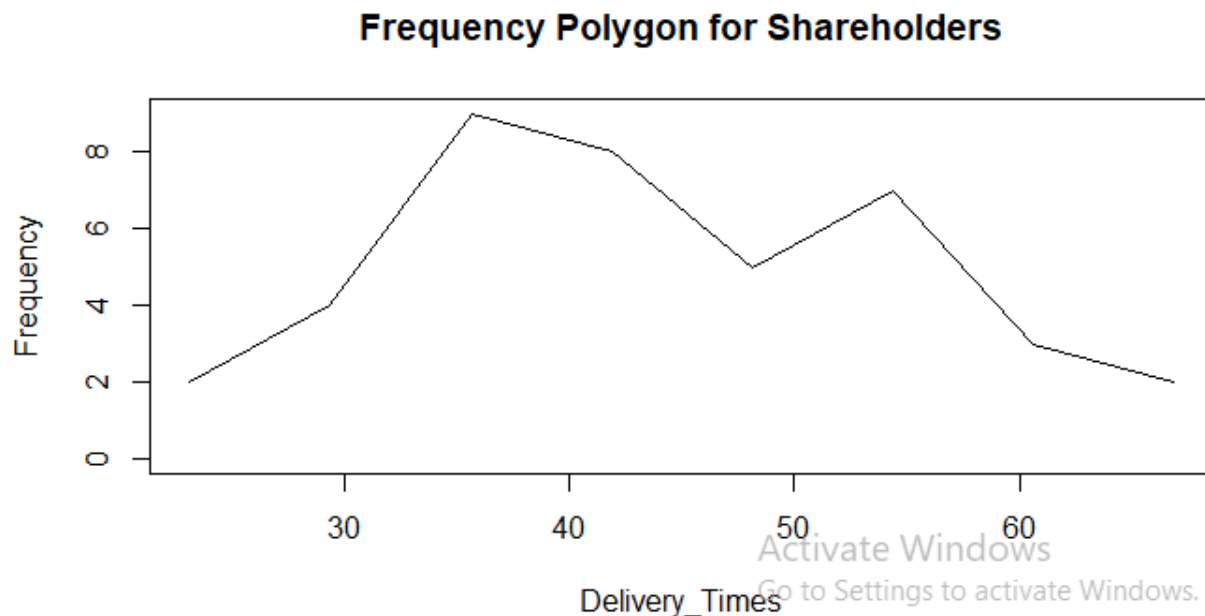
```
22 ##Construct the frequency distribution for the above specification.
23 ##assign class limits of frequency distribution into a variable called "breaks"
24 breaks <- round(histogram$breaks)
25 breaks
26
27 ##assign class limits of frequencies of the histogram into a variable called "freq"
28 freq <- histogram$counts
29 freq
30
31 ##assign mid points of each class into a variable called "mids"
32 mids <- histogram$mids
33 mids
34
35 ##creating the variable called "Classes" for frequency distribution
36 classes <- c()
37
38 ##creating a "for loop" to assign classes of the frequency distribution into 'classes' variable created above
39 for (i in 1:length(breaks)-1) {
40   classes[i] <- paste0("[" ,breaks[i] ,",",breaks[i+1],")")
41 }
42
43 ##Obtaining frequency distribution by combing the values of "classes" & "freq" variables
44 ##cbind command used to merge the columns with same length
45 cbind(Classes = classes, Frequency = freq)
46
```

```

> ##Construct the frequency distribution for the above specification.
> ##assign class limits of frequency distribution into a variable called "breaks"
> breaks <- round(histogram$breaks)
> breaks
[1] 20 26 32 39 45 51 58 64 70
> ##assign class limits of frequencies of the histogram into a variable called "freq"
> freq <- histogram$counts
> freq
[1] 2 4 9 8 5 7 3 2
> ##assign mid points of each class into a variable called "mids"
> mids <- histogram$mids
> mids
[1] 23.125 29.375 35.625 41.875 48.125 54.375 60.625 66.875
> ##creating the variable called "classes" for frequency distribution
> classes <- c()
> ##creating a "for loop" to assign classes of the frequency distribution into 'classes' variable created above
> for (i in 1:length(breaks)-1) {
+   classes[i] <- paste0("[",breaks[i],",",breaks[i+1],")")
+ }
> ##Obtaining frequency distribution by combing the values of "classes" & "freq" variables
> ##cbind command used to merge the columns with same length
> cbind(classes = classes, Frequency = freq)
  classes Frequency
[1,] "[20,26)"    "2"
[2,] "[26,32)"    "4"
[3,] "[32,39)"    "9"
[4,] "[39,45)"    "8"
[5,] "[45,51)"    "5"
[6,] "[51,58)"    "7"
[7,] "[58,64)"    "3"
[8,] "[64,70)"    "2"
> |
47 ##Portray the distribution in the form of a frequency polygon.
48 ##draw frequency polygon to the same plot
49 lines(mids,freq)
50
51 ##draw frequency polygon in a new plot
52 plot(mids,freq, type = 'l', main = "Frequency Polygon for Shareholders", xlab = "Delivery_Times", ylab = "Frequency", ylim = c(0,max(freq)))
53
> lines(mids,freq)
> ##draw frequency polygon in a new plot
> plot(mids,freq, type = 'l', main = "Frequency Polygon for Shareholders", xlab = "Delivery_Times", ylab = "Frequency", ylim = c(0,max(freq)))
> |

```

Activate V



```

47 ##Portray the distribution in the form of a frequency polygon.
48 ##draw frequency polygon to the same plot
49 lines(mids,freq)
50
51 ##draw frequency polygon in a new plot
52 plot(mids,freq, type = 'l', main = "Frequency Polygon for Shareholders", xlab = "Delivery_Times", ylab = "Frequency", ylim = c(0,max(freq)))
53
54 ##Portray the distribution in a cumulative frequency polygon (ogive).
55 ##using "cumsum" command to get cumulative frequencies
56 cum.freq <- cumsum(freq)
57 cum.freq
58
59 ##create a null variable called "new"
60 new <- c()
61
62 ##using "for" loop to store cumulative frequencies in order to get the ogive
63- for(i in 1:length(breaks)){
64-   if(i==1){
65-     new[i]=0
66-   }else{
67-     new[i]=cum.freq[i-1]
68-   }
69- }
70
71 ##draw cumulative frequency polygon in a new plot
72 plot(breaks, new, type = 'l', main = "Cumulative Frequency Polygon for Shareholders", xlab = "Delivery_Times", ylab = "Cumulative Frequency", ylim = c(0,max(cum
73
74 ##obtain upper limit of each class along with its cumulative frequency in a table
75 cbind(upper = breaks, CumFreq = new)
76 plot(breaks, new, type = 'o', main = "Cumulative Frequency Polygon for Shareholders", xlab = "Delivery_Times", ylab = "Cumulative Frequency", ylim = c(0,max(cum
77
78
> ##Portray the distribution in the form of a frequency polygon.
> ##draw frequency polygon to the same plot
> lines(mids,freq)
> ##draw frequency polygon in a new plot
> plot(mids,freq, type = 'l', main = "Frequency Polygon for Shareholders", xlab = "Delivery_Times", ylab = "Frequency", ylim = c(0,max(freq)))
> ##Portray the distribution in a cumulative frequency polygon (ogive).
> ##using "cumsum" command to get cumulative frequencies
> cum.freq <- cumsum(freq)
> cum.freq
[1] 2 6 15 23 28 35 38 40
> ##create a null variable called "new"
> new <- c()
> ##using "for" loop to store cumulative frequencies in order to get the ogive
> for(i in 1:length(breaks)){
+   if(i==1){
+     new[i]=0
+   }else{
+     new[i]=cum.freq[i-1]
+   }
+ }
> ##draw cumulative frequency polygon in a new plot
> plot(breaks, new, type = 'l', main = "Cumulative Frequency Polygon for Shareholders", xlab = "Delivery_Times", ylab = "Cumulative Frequency", ylim = c(0,max(cu
m.freq)))
> ##obtain upper limit of each class along with its cumulative frequency in a table
> cbind(upper = breaks, CumFreq = new)
      upper CumFreq
[1,]    20         0
[2,]    26         2
[3,]    32         6
[4,]    39        15
[5,]    45        23
[6,]    51        28
[7,]    58        35
[8,]    64        38
[9,]    70        40
> plot(breaks, new, type = 'o', main = "Cumulative Frequency Polygon for Shareholders", xlab = "Delivery_Times", ylab = "Cumulative Frequency", ylim = c(0,max(cu
m.freq)))
> |

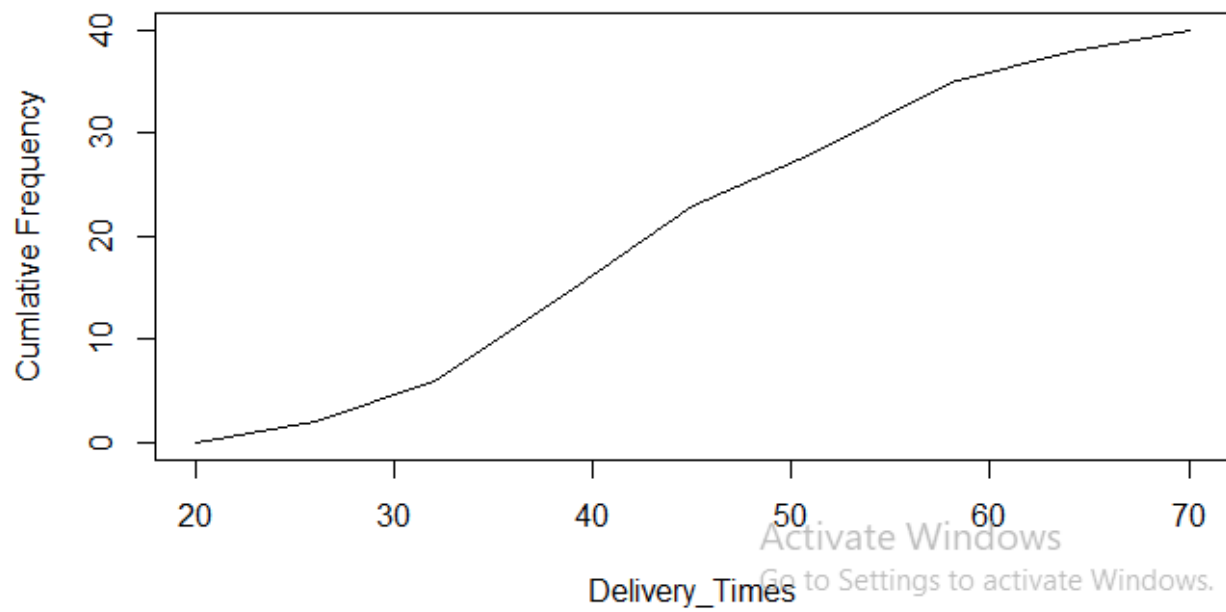
```

Activate Windows

Go to Settings to activate Windows.

Go to Settings to activate Windows.

**Cumulative Frequency Polygon for Shareholders**



**Cumulative Frequency Polygon for Shareholders**

