

PS Lab 05 - IT24102768

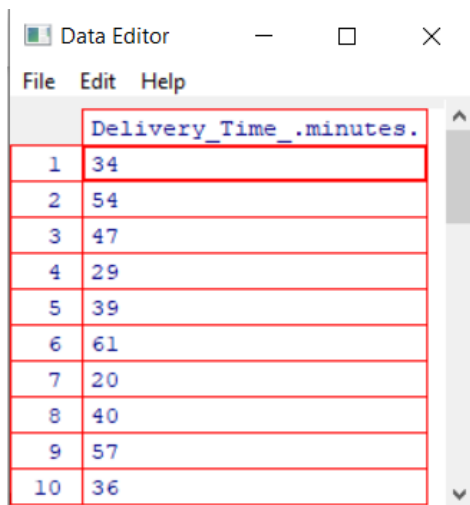
Senadeera H.P.R.L.C.

Exercise

Part (1)

```
1 setwd("C:\\Users\\DELL\\Desktop\\IT24102768-PS")
2
3 Delivery_Times <- read.table("Exercise - Lab 05.txt", header = TRUE)
4 fix(Delivery_Times)
5 attach(Delivery_Times)
```

```
> setwd("C:\\Users\\DELL\\Desktop\\IT24102768-PS")
> Delivery_Times <- read.table("Exercise - Lab 05.txt", header = TRUE)
> fix(Delivery_Times)
```

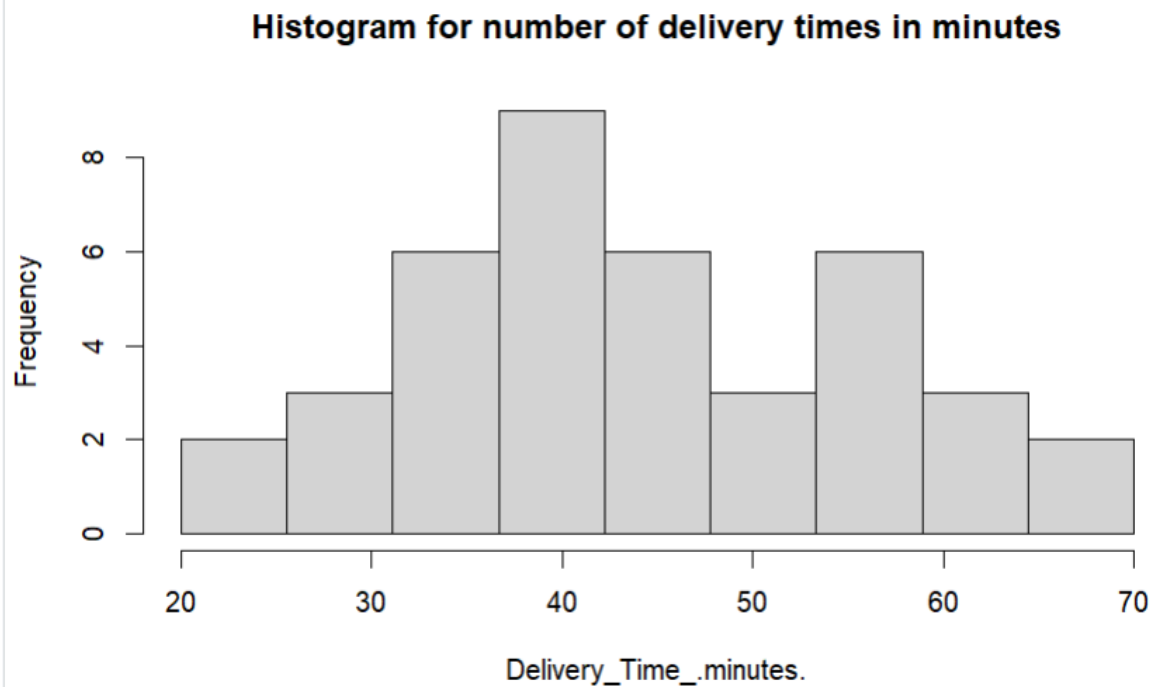


	Delivery_Time_.minutes.
1	34
2	54
3	47
4	29
5	39
6	61
7	20
8	40
9	57
10	36

Part (2)

```
7 histogram <- hist(Delivery_Time_.minutes.,
8                   main="Histogram for number of delivery times in minutes",
9                   breaks = seq(20,70, length = 10), right = FALSE)
10
```

```
> View(Delivery_Times)
> histogram <- hist(Delivery_Time_.minutes.,
+                   main="Histogram for number of delivery times in minutes",
+                   breaks = seq(20,70, length = 10), right = FALSE)
> |
```



Part (3)

```
11 # Histogram has a bimodal shape with peaks around 35-40 minutes and 55-60 minutes.  
12 #Distribution is roughly symmetric and slight right-skew.  
13
```

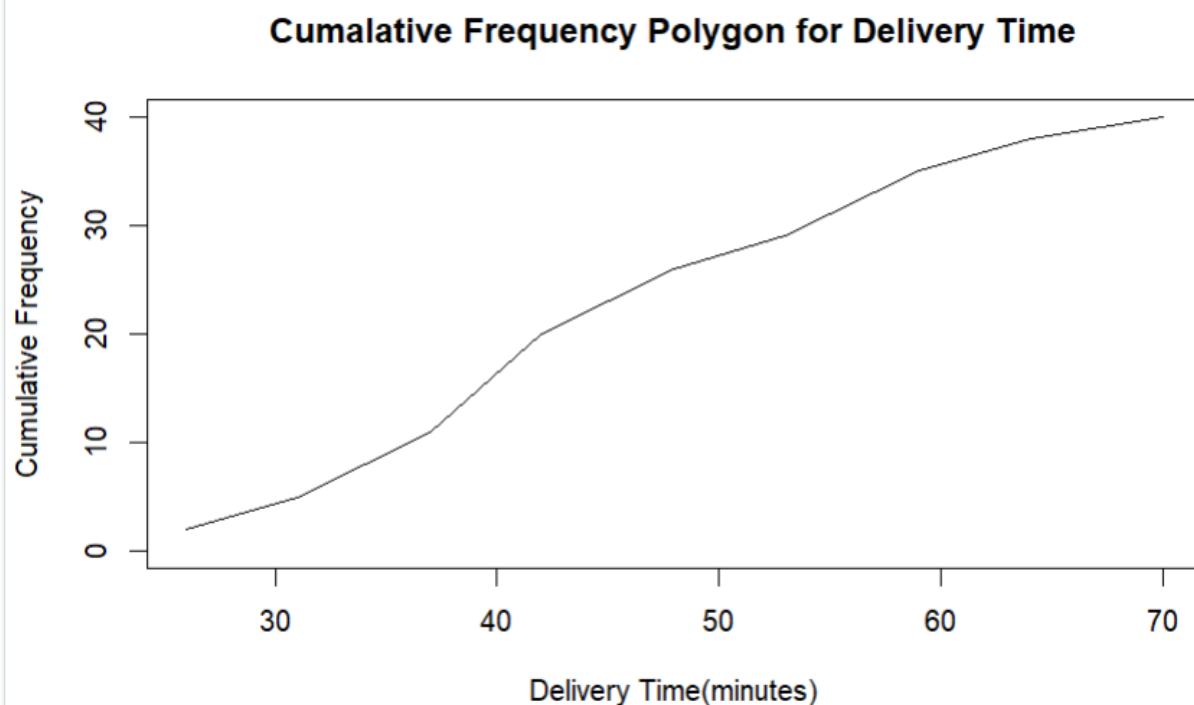
Part (4)



```
14 breaks <- round(histogram$breaks)  
15 frequency <- histogram$counts  
16 cum.frequency <- cumsum(frequency)  
17  
18 plot (breaks[-1], cum.frequency, type = 'l',  
19       main = "Cumalative Frequency Polygon for Delivery Time",  
20       xlab="Delivery Time(minutes)",  
21       ylab = "Cumulative Frequency",  
22       ylim = c(0, max (cum.frequency)))  
23  
24 detach(Delivery_Times)
```

```

> breaks <- round(histogram$breaks)
> frequency <- histogram$counts
> cum.frequency <- cumsum(frequency)
>
> plot (breaks[-1], cum.frequency, type = 'l',
+       main = "Cumalative Frequency Polygon for Delivery Time",
+       xlab="Delivery Time(minutes)",
+       ylab = "Cumulative Frequency",
+       ylim = c(0, max (cum.frequency)))
>
> detach(Delivery_Times)

```



Data		
Delivery_Times	40 obs. of 1 variable	
histogram	List of 6	
Values		
breaks	num [1:10]	20 26 31 37 42 48 53 59 64 70
cum.frequency	int [1:9]	2 5 11 20 26 29 35 38 40
frequency	int [1:9]	2 3 6 9 6 3 6 3 2