

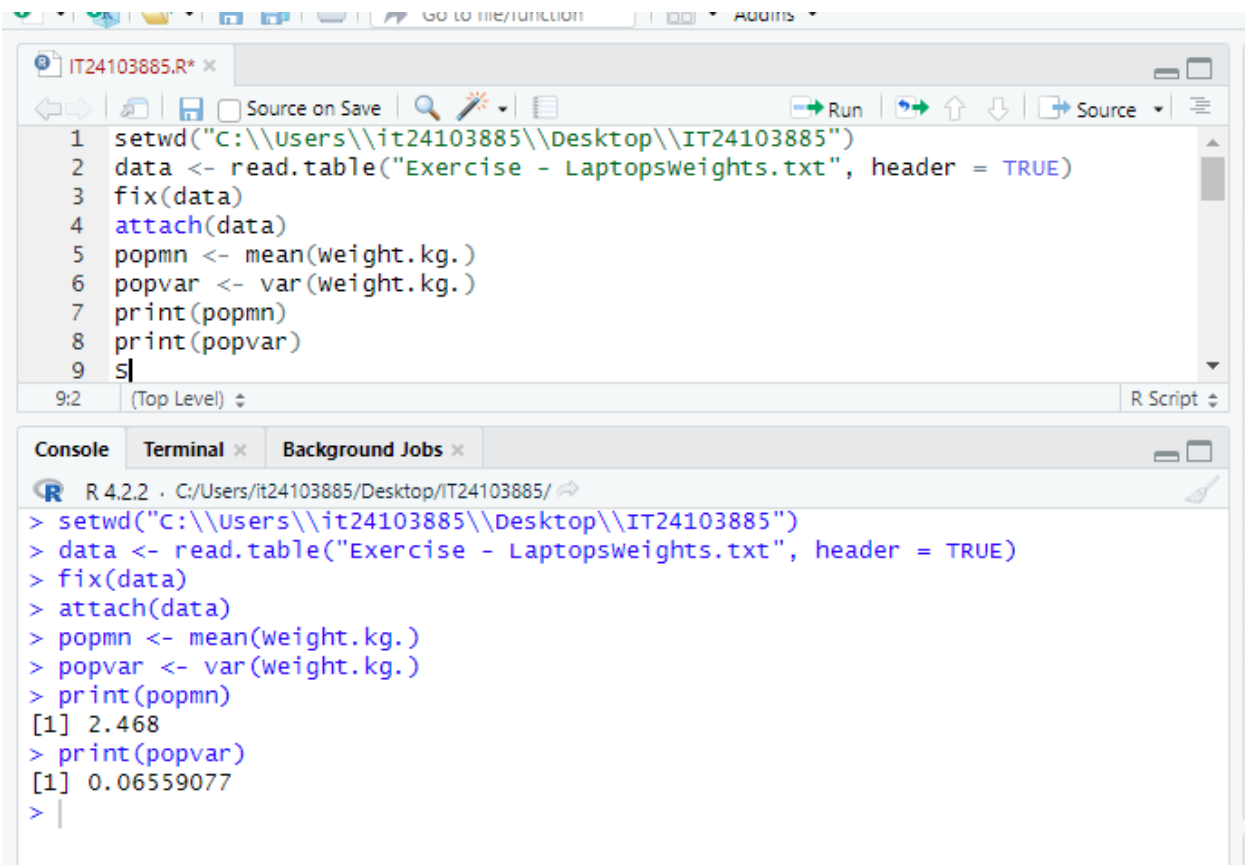
# IT2120 - Probability and Statistics

## Lab Sheet 08

### IT24103885 - Senarathna Y.M.C.S

**Instructions:** Create a folder in your desktop with your registration number (Eg: "IT....."). You need to save the R script file and take screenshots of the command prompt with answers and save it in a word document inside the folder. Save both R script file and word document with your registration number (Eg: "IT....."). After you finish the exercise, zip the folder and upload the zip file to the submission link.

1. Calculate the population mean and population standard deviation of the laptop bag weights.



The screenshot shows an R Studio interface. The top pane displays an R script file named 'IT24103885.R' with the following code:

```
1 setwd("C:\\Users\\it24103885\\Desktop\\IT24103885")
2 data <- read.table("Exercise - Laptopsweights.txt", header = TRUE)
3 fix(data)
4 attach(data)
5 popmn <- mean(weight.kg.)
6 popvar <- var(weight.kg.)
7 print(popmn)
8 print(popvar)
9 s|
```

The bottom pane shows the console output for the same script:

```
R 4.2.2 · C:/Users/it24103885/Desktop/IT24103885/
> setwd("C:\\Users\\it24103885\\Desktop\\IT24103885")
> data <- read.table("Exercise - Laptopsweights.txt", header = TRUE)
> fix(data)
> attach(data)
> popmn <- mean(weight.kg.)
> popvar <- var(weight.kg.)
> print(popmn)
[1] 2.468
> print(popvar)
[1] 0.06559077
> |
```

2. Draw 25 random samples of size 6 (with replacement) and calculate the sample mean and sample standard deviation for each sample.

```
9
10 samples <- c()
11 n <- c()
12 for(i in 1:25) {
13   s <- sample(weight.kg., 5, replace = TRUE)
14   samples <- cbind(samples, s)
15   n <- c(n, paste('s', i))
16 }
17
18 colnames(samples) = n
19 s.means <- apply(samples, 2, mean)
20 s.sd <- apply(samples, 2, sd)
21 print(s.means)
22 print(s.sd)
23
```

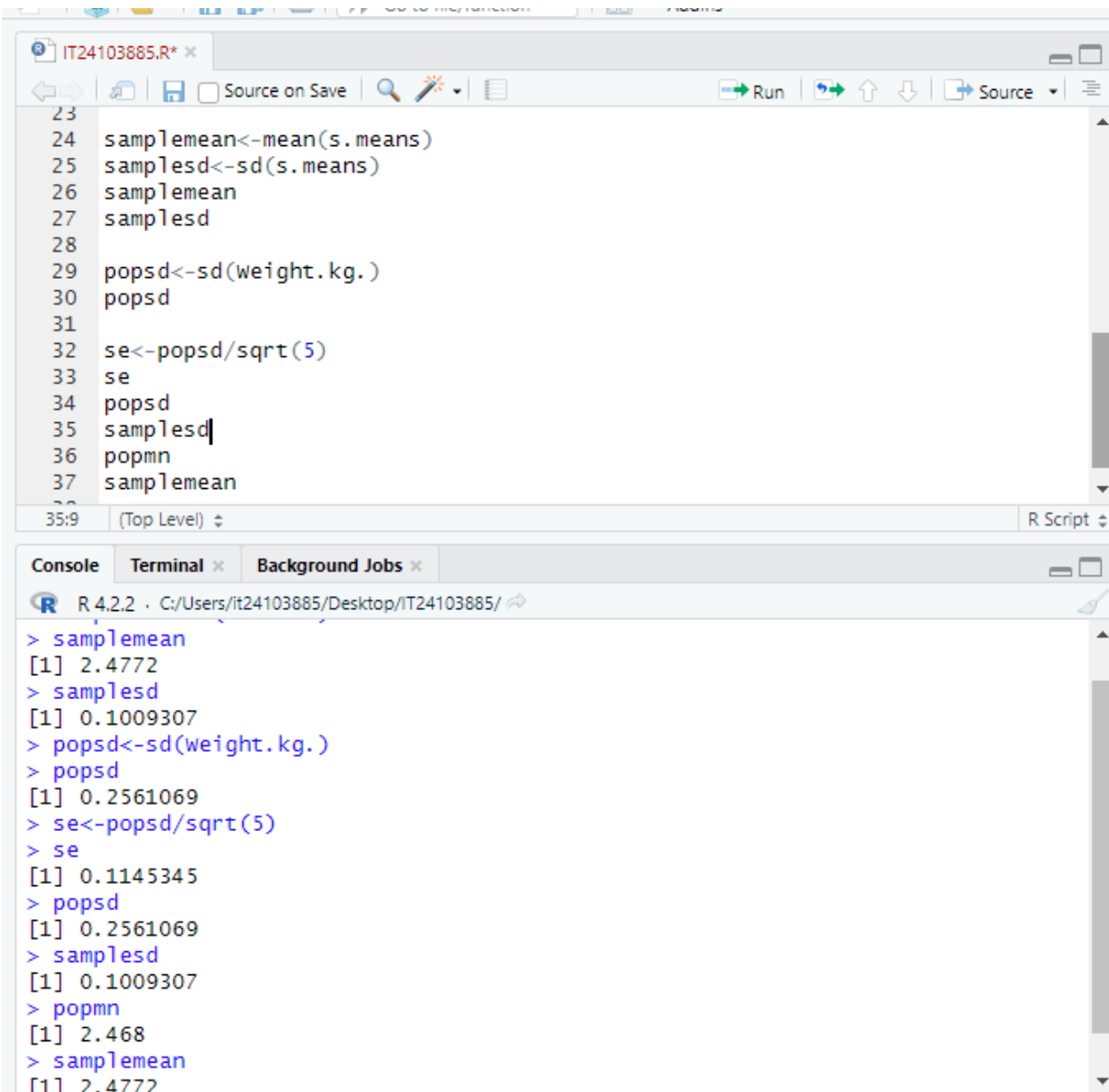
22:12 (Top Level) R Script

Console Terminal Background Jobs

R 4.2.2 · C:/Users/it24103885/Desktop/IT24103885/

```
> s.sd <- apply(samples, 2, sd)
> print(s.means)
  s 1   s 2   s 3   s 4   s 5   s 6   s 7   s 8   s 9   s 10  s 11  s 12
2.284 2.424 2.502 2.378 2.574 2.490 2.516 2.530 2.612 2.316 2.354 2.462
  s 13  s 14  s 15  s 16  s 17  s 18  s 19  s 20  s 21  s 22  s 23  s 24
2.518 2.566 2.274 2.544 2.648 2.502 2.498 2.536 2.410 2.484 2.378 2.594
  s 25
2.536
> print(s.sd)
      s 1      s 2      s 3      s 4      s 5      s 6      s 7
0.38894730 0.23319520 0.44415088 0.26789923 0.23586013 0.21760055 0.37507333
      s 8      s 9      s 10      s 11      s 12      s 13      s 14
0.22056745 0.13809417 0.42045214 0.41295278 0.24478562 0.25083859 0.13939153
      s 15      s 16      s 17      s 18      s 19      s 20      s 21
0.38141841 0.16056151 0.19421637 0.07694154 0.28734996 0.20292856 0.20112185
      s 22      s 23      s 24      s 25
0.18716303 0.46911619 0.20574256 0.17386777
> |
```

3. Calculate the mean and standard deviation of the 25 sample means and state the relationship of them with true mean and true standard deviation.



The screenshot displays the R Studio interface. The top pane shows the source editor with R code for calculating sample statistics. The bottom pane shows the console with the output of the executed code.

```
IT24103885.R* x
Source on Save
Run
Source

23
24 samplemean<-mean(s.means)
25 samplesd<-sd(s.means)
26 samplemean
27 samplesd
28
29 popsd<-sd(weight.kg.)
30 popsd
31
32 se<-popsd/sqrt(5)
33 se
34 popsd
35 samplesd
36 popmn
37 samplemean
38

35:9 (Top Level) R Script
```

Console

```
R 4.2.2 C:/Users/it24103885/Desktop/IT24103885/
> samplemean
[1] 2.4772
> samplesd
[1] 0.1009307
> popsd<-sd(weight.kg.)
> popsd
[1] 0.2561069
> se<-popsd/sqrt(5)
> se
[1] 0.1145345
> popsd
[1] 0.2561069
> samplesd
[1] 0.1009307
> popmn
[1] 2.468
> samplemean
[1] 2.4772
```