Sri Lanka Institute of Information Technology

Lab Submission
<Worksheet 8>

<IT24102257>

<Senadheera P.V.S.T.>

Probability and Statictics | IT2120

1. Calculate population mean and variance of the dataset.

```
> setwd("C:\\Users\\Sahan Senadheera\\Desktop\\IT24102257 Lab8 ps")
> data <- read.table("Data - Lab 8.txt", header = TRUE)
> fix(data)
> attach(data)
```

```
> # Question 01
> # Commands "mean" & "var" will compute the mean and the variance for data.
> popmn <-mean(Nicotine)
> popmn
[1] 1.77425
> popvar <- var(Nicotine)
> popvar
[1] 0.1524558
```

2. Get 30 random samples of size 5, with replacement and calculate sample mean and sample variance for each sample.

```
> # Question 02
> # First create null vectors to store sample data sets.
> samples <- c()
> n <- c()
> # The "for" loop will be used to create and assign samples of size 5 for "samples" variable created above.
> # Using "sample" command we can draw a random sample either with replacement or without replacement.
> # By making "replace" argument as TRUE we can create samples with replacement.
> for(i in 1:30){
+   s <- sample(Nicotine,5,replace = TRUE)
+   samples <- cbind(samples,s)
+   n <- c(n,paste("S",i))
+ }
> # Assign column names for each sample created. Names have stored earlier under "n" variable.
> colnames(samples) = n
> # Using "apply" commands we can ask to calculate any function such as mean, variance, etc. row wise or
> # column wise in a matrix.
> #Here, considering the second argument as "2" we can calculate either mean/variance column wise
> #which stored earlier in "samples" variable which is a matrix
> s.means <- apply(samples,2,mean)
> 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  S 13  S 14  S 15  S 16  S 17
1.830 1.324 1.858 1.872 1.776 2.018 1.828 2.114 1.732 1.734 1.532 1.704 2.054 1.538 1.748 1.942 1.938
 S 18  S 19  S 20  S 21  S 22  S 23  S 24  S 25  S 26  S 27  S 28  S 29  S 30
2.082 1.770 1.662 1.636 1.852 1.572 1.924 2.014 1.884 1.798 1.724 1.872 1.824
> s.vars <- apply(samples,2, var)
> s.vars
    S 1     S 2     S 3     S 4     S 5     S 6     S 7     S 8     S 9    S 10    S 11    S 12    S 13
0.11760 0.26128 0.06392 0.12087 0.11023 0.08752 0.07177 0.02743 0.18127 0.14008 0.23142 0.32338 0.04228
   S 14    S 15    S 16    S 17    S 18    S 19    S 20    S 21    S 22    S 23    S 24    S 25    S 26
0.21317 0.05322 0.16112 0.17877 0.16997 0.05240 0.30027 0.04388 0.17992 0.17927 0.12433 0.05233 0.05158
   S 27    S 28    S 29    S 30
0.37757 0.07928 0.15622 0.03883
```

3. Calculate mean and variance of the Sample Means.

```
> ##Question 03
> #Following commands will calculate mean and variance of samples means stored in "s.means" variable.
> samplemean <- mean(s.means)
> samplemean
[1] 1.8052
> samplevars <- var(s.means)
> samplevars
[1] 0.0313312
```

4. Compare and state relationship (if any) Population Mean and the Mean of Sample Means.

```
> #Question 04
> #Compare the population mean and mean of sample means.
> popmn
[1] 1.77425
> samplemean
[1] 1.8052
```

5. Compare and state relationship (if any) Population Variance and the Variance of Sample Means.

```
> # Question 05
> # Compare the population variance and variance of sample means.
> truevar = popvar/5
> truevar
[1] 0.03049117
> samplevars
[1] 0.0313312
```

Exercise

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.

```
> #EXCERSICE
> data <- read.table("Exercise - LaptopsWeights.txt", header = TRUE)
> fix(data)
> attach(data)
```

```
> pop_mean <- mean(Weight.kg.)
> pop_mean
[1] 2.468
> pop_sd <- sd(Weight.kg.)
> pop_sd
[1] 0.2561069
```

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

```
> samples <- c()
> sample_names <- c()
> for (i in 1:25) {
+   s <- sample(Weight.kg., 6, replace = TRUE)
+   samples <- cbind(samples, s)
+   sample_names <- c(sample_names, paste('Sample', i))
+ }
> colnames(samples) <- sample_names
> sample_means <- apply(samples, 2, mean)
> sample_means
 Sample 1  Sample 2  Sample 3  Sample 4  Sample 5  Sample 6  Sample 7  Sample 8  Sample 9 Sample 10
 2.626667  2.475000  2.411667  2.435000  2.651667  2.430000  2.448333  2.465000  2.550000  2.435000
Sample 11 Sample 12 Sample 13 Sample 14 Sample 15 Sample 16 Sample 17 Sample 18 Sample 19 Sample 20
 2.446667  2.423333  2.571667  2.348333  2.515000  2.356667  2.560000  2.428333  2.456667  2.490000
Sample 21 Sample 22 Sample 23 Sample 24 Sample 25
 2.556667  2.310000  2.488333  2.266667  2.416667
> sample_sds <- apply(samples, 2, sd)
> sample_sds
  Sample 1   Sample 2   Sample 3   Sample 4   Sample 5   Sample 6   Sample 7   Sample 8   Sample 9
0.23703727 0.31040296 0.44790252 0.37367098 0.28554626 0.32174524 0.19630758 0.23270153 0.12181954
 Sample 10  Sample 11  Sample 12  Sample 13  Sample 14  Sample 15  Sample 16  Sample 17  Sample 18
0.25398819 0.24196418 0.36059211 0.22560290 0.38809363 0.21667949 0.31935351 0.09919677 0.26783702
 Sample 19  Sample 20  Sample 21  Sample 22  Sample 23  Sample 24  Sample 25
0.11093542 0.21605555 0.19966639 0.53081070 0.21802905 0.39185031 0.37585458
```

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.

```
> mean_of_sample_means <- mean(sample_means)
> mean_of_sample_means
[1] 2.462533
> sd_of_sample_means <- sd(sample_means)
> sd_of_sample_means
[1] 0.09158471
```