Exercise

Lab 08

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

```r
setwd("C:\\Users\\IT24102160\\Desktop\\IT24102160")
getwd()

data <- read.table("Exercise - LaptopsWeights.txt", header = TRUE)
laptop_bag_weights <- data[[1]] # Extract the first column (adjust if necessary)

population_mean <- mean(laptop_bag_weights)
population_sd <- sd(laptop_bag_weights)

cat("Population Mean:", population_mean, "\n")
cat("Population Standard Deviation:", population_sd, "\n")

num_samples <- 25
sample_size <- 6

sample_means <- numeric(num_samples)
sample_sds <- numeric(num_samples)

set.seed(42)
for (i in 1:num_samples) {
  sample_data <- sample(laptop_bag_weights, size = sample_size, replace = TRUE)
  sample_means[i] <- mean(sample_data)
  sample_sds[i] <- sd(sample_data)
}

cat("Sample Means:\n", sample_means, "\n")
cat("Sample Standard Deviations:\n", sample_sds, "\n")

mean_sample_means <- mean(sample_means)
sd_sample_means <- sd(sample_means)

cat("Mean of Sample Means:", mean_sample_means, "\n")
cat("Standard Deviation of Sample Means:", sd_sample_means, "\n")
cat("True Population Mean:", population_mean, "\n")
```

36:1 (Top Level) R Script

```
> setwd("C:\\Users\\IT24102160\\Desktop\\IT24102160")
> getwd()
[1] "C:/Users/IT24102160/Desktop/IT24102160"
> data <- read.table("Exercise - LaptopsWeights.txt", header = TRUE)
> laptop_bag_weights <- data[[1]] # Extract the first column (adjust if necessary)
> population_mean <- mean(laptop_bag_weights)
> population_sd <- sd(laptop_bag_weights)
> cat("Population Mean:", population_mean, "\n")
Population Mean: 2.468
> cat("Population Standard Deviation:", population_sd, "\n")
Population Standard Deviation: 0.2561069
> num_samples <- 25
> sample_size <- 6
> sample_means <- numeric(num_samples)
> sample_sds <- numeric(num_samples)
> set.seed(42)
> for (i in 1:num_samples) {
+   sample_data <- sample(laptop_bag_weights, size = sample_size, replace = TRUE)
+   sample_means[i] <- mean(sample_data)
+   sample_sds[i] <- sd(sample_data)
+ }
> cat("Sample Means:\n", sample_means, "\n")
Sample Means:
 2.683333 2.656667 2.621667 2.448333 2.223333 2.568333 2.463333 2.351667 2.246667 2.665
2.476667 2.651667 2.506667 2.585 2.501667 2.501667 2.376667 2.35 2.22 2.32 2.541667 2.49
1667 2.521667 2.475 2.298333
> cat("Sample Standard Deviations:\n", sample_sds, "\n")
Sample Standard Deviations:
 0.1600833 0.110755 0.1444184 0.168928 0.3283697 0.2968782 0.1862973 0.2477431 0.3068985
0.1720174 0.229056 0.1988383 0.2615849 0.2918733 0.1921891 0.2162791 0.3881065 0.2848859
0.2442949 0.2260973 0.1741742 0.1675012 0.2393672 0.1251799 0.3819119
```

```
> mean_sample_means <- mean(sample_means)
> sd_sample_means <- sd(sample_means)
> cat("Mean of Sample Means:", mean_sample_means, "\n")
Mean of Sample Means: 2.469867
> cat("Standard Deviation of Sample Means:", sd_sample_means, "\n")
Standard Deviation of Sample Means: 0.1402073
> cat("True Population Mean:", population_mean, "\n")
True Population Mean: 2.468
> cat("True Population Standard Deviation:", population_sd, "\n")
True Population Standard Deviation: 0.2561069
>
```

Data	
data	40 obs. of 1 variable
Values	
i	25L
laptop_bag_weights	num [1:40] 2.46 2.45 2.47 2.71 2.46 2.05 2.6 2.42…
mean_sample_means	2.46986666666667
num_samples	25
population_mean	2.468
population_sd	0.256106948813907
sample_data	num [1:6] 1.71 2.7 2.23 2.43 2.67 2.05
sample_means	num [1:25] 2.68 2.66 2.62 2.45 2.22 ...
sample_sds	num [1:25] 0.16 0.111 0.144 0.169 0.328 ...
sample_size	6
sd_sample_means	0.140207287019375