Sri Lanka Institute of Information Technology

Lab Submission
Worksheet No-08

IT24103538

Hettiarachchi H P P

Probability and Statistics - IT2120

B.Sc. (Hons) in Information Technology

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

```
data <- read.table("Exercise - LaptopsWeights.txt", header = TRUE)
# R converts "Weight(kg)" to "Weight.kg."
laptop_weights <- data$Weight.kg.
n_population <- length(laptop_weights)
population_mean <- mean(laptop_weights)
# Adjust R's sample sd to get the population sd
population_sd <- sd(laptop_weights) * sqrt((n_population - 1) / n_population)

print(paste("Population Mean:", population_mean))
print(paste("Population Standard Deviation:", population_sd))
```

```
> n_population <- length(laptop_weights)
> population_mean <- mean(laptop_weights)
> # Adjust R's sample sd to get the population sd
> population_sd <- sd(laptop_weights) * sqrt((n_population - 1) / n_population)
> print(paste("Population Mean:", population_mean))
[1] "Population Mean: 2.468"
> print(paste("Population Standard Deviation:", population_sd))
[1] "Population Standard Deviation: 0.252885349516337"
```

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

```
samples <- c()
number_of_samples <- 25
sample_size <- 6

for (i in 1:number_of_samples) {
  s <- sample(laptop_weights, sample_size, replace = TRUE)
  samples <- cbind(samples, s)
}
colnames(samples) <- paste("Sample", 1:number_of_samples)

sample_means <- apply(samples, 2, mean)
sample_sds <- apply(samples, 2, sd)

print(sample_means)
print(sample_sds)
```

```
> samples <- c()
> number_of_samples <- 25
> sample_size <- 6
>
> for (i in 1:number_of_samples) {
+   s <- sample(laptop_weights, sample_size, replace = TRUE)
+   samples <- cbind(samples, s)
+ }
> colnames(samples) <- paste("Sample", 1:number_of_samples)
>
> sample_means <- apply(samples, 2, mean)
> sample_sds <- apply(samples, 2, sd)
>
> print(sample_means)
 Sample 1  Sample 2  Sample 3  Sample 4  Sample 5  Sample 6  Sample 7
 2.291667  2.416667  2.538333  2.355000  2.523333  2.446667  2.560000
 Sample 8  Sample 9 Sample 10 Sample 11 Sample 12 Sample 13 Sample 14
 2.260000  2.571667  2.643333  2.463333  2.493333  2.380000  2.503333
Sample 15 Sample 16 Sample 17 Sample 18 Sample 19 Sample 20 Sample 21
 2.456667  2.598333  2.403333  2.451667  2.441667  2.536667  2.435000
Sample 22 Sample 23 Sample 24 Sample 25
 2.268333  2.403333  2.485000  2.378333
> print(sample_sds)
 Sample 1  Sample 2  Sample 3  Sample 4  Sample 5  Sample 6  Sample 7
0.4132029 0.3704952 0.1133872 0.3602638 0.2765261 0.1502886 0.2304778
 Sample 8  Sample 9 Sample 10 Sample 11 Sample 12 Sample 13 Sample 14
0.1751571 0.2091331 0.1447296 0.2082947 0.1562903 0.2110924 0.2563331
Sample 15 Sample 16 Sample 17 Sample 18 Sample 19 Sample 20 Sample 21
0.2471976 0.2595702 0.1681269 0.3055105 0.3360605 0.2578113 0.2066640
Sample 22 Sample 23 Sample 24 Sample 25
0.3336715 0.2301014 0.2609023 0.2278962
```

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)
sd_of_sample_means <- sd(sample_means) # This is the Standard Error

print(paste("Mean of the 25 Sample Means:", mean_of_sample_means))
print(paste("Standard Deviation of the 25 Sample Means (Standard Error):", sd_of_sample_means))

# Compare with population parameters
theoretical_standard_error <- population_sd / sqrt(sample_size)

print(paste("Population Mean:", population_mean))
print(paste("Theoretical Standard Error (Population SD / sqrt(n)):", theoretical_standard_error))
```

```
> print(paste("Mean of the 25 Sample Means:", mean_of_sample_means))
[1] "Mean of the 25 Sample Means: 2.4522"
> print(paste("Standard Deviation of the 25 Sample Means (Standard Error):", sd_of_sample_means))
[1] "Standard Deviation of the 25 Sample Means (Standard Error): 0.0983259413417474"
>
> # Compare with population parameters
> theoretical_standard_error <- population_sd / sqrt(sample_size)
>
> print(paste("Population Mean:", population_mean))
[1] "Population Mean: 2.468"
> print(paste("Theoretical Standard Error (Population SD / sqrt(n)):", theoretical_standard_error))
[1] "Theoretical Standard Error (Population SD / sqrt(n)): 0.103240011623401"
```