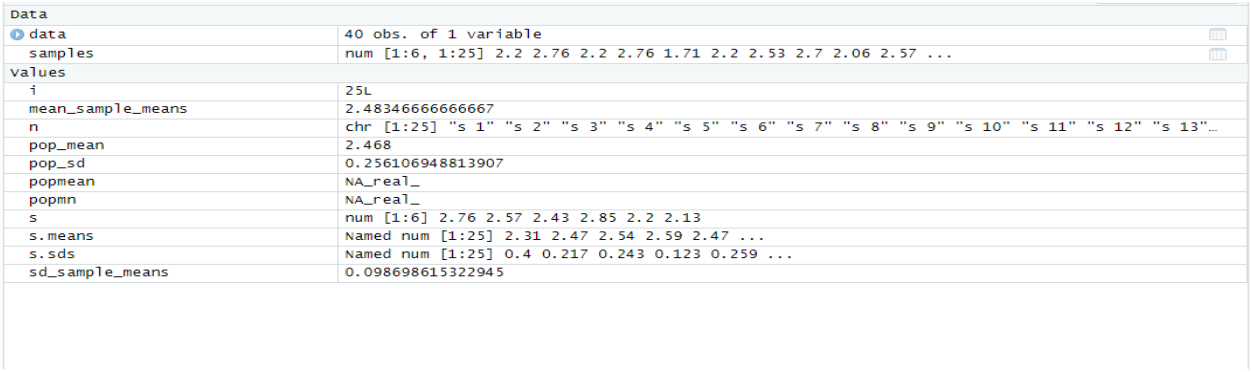
Exercise

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.

Answers



Data	
data	40 obs. of 1 variable
samples	num [1:6, 1:25] 2.2 2.76 2.2 2.76 1.71 2.2 2.53 2.7 2.06 2.57 ...
values	
i	25L
mean_sample_means	2.48346666666667
n	chr [1:25] "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"…
pop_mean	2.468
pop_sd	0.256106948813907
popmean	NA_real_
popmn	NA_real_
s	num [1:6] 2.76 2.57 2.43 2.85 2.2 2.13
s.means	Named num [1:25] 2.31 2.47 2.54 2.59 2.47 ...
s.sds	Named num [1:25] 0.4 0.217 0.243 0.123 0.259 ...
sd_sample_means	0.098698615322945

```
> setwd("C:\\Users\\IT24104216\\Desktop\\IT24104216_Lab_08")
> ## Importing the dataset from folder
> data <- read.table("Laptopsweights.txt", header=TRUE)
> fix(data)
> attach(data)
The following object is masked from data (pos = 3):

    Weight.kg.

The following object is masked from data (pos = 4):

    Weight.kg.

The following object is masked from data (pos = 5):

    Weight.kg.

The following object is masked from data (pos = 6):

    Weight.kg.

> ## Getting population mean and standard deviation
> pop_mean <- mean(Weight.kg.)
> pop_sd <- sd(Weight.kg.)
> ## Create null vectors
> samples <- c()
> n <- c()
> ## Generate 25 random samples of size 6 with replacement
> for(i in 1:25) {
+   s <- sample(Weight.kg., 6, replace=TRUE)
+   samples <- cbind(samples, s)
+   n <- c(n, paste('s', i))
+ }
```

```
> ##Giving column names to samples
> colnames(samples) <- n
> ## Getting sample mean and standard deviation
> s.means <- apply(samples, 2, mean)
> s.sds <- apply(samples, 2, sd)
> ## Getting mean and standard deviation of sample mean
> mean_sample_means <- mean(s.means)
> sd_sample_means <- sd(s.means)
> ## Display comparisons
> pop_mean
[1] 2.468
> mean_sample_means
[1] 2.483467
> pop_sd
[1] 0.2561069
> sd_sample_means
[1] 0.09869862
>
```