lab4

chaithra

23/09/2021

Intro to Data Science - Lab 4

IST687 Section M002

Professor Anderson

Enter your name here: Chaithra Kopparam Cheluvaiah

#Select one of the below and add needed information

1. I did this homework by myself, with help from the book and the professor.

```
set.seed(123)
sampleSize <- 30
studentPop <- rnorm(20000, mean=20, sd=3) # it generates a vector of normally distributed
# random numbers of size 20000 which will have mean of 20 and standard deviation of 3
undergrads <- sample(studentPop, size = sampleSize, replace=TRUE) # creating sample with size 30
# and replace the sample into the population
grads <- rnorm(sampleSize, mean= 25, sd=3) # generating a vector of normally distributed
# random numbers of size 30
if (runif(1)>0.5) { testSample <- grads} else {testSample <- undergrads} # generate one random value
# between 0 and 1 and accordingly decide testSample should be grads or undergrads
mean(testSample) # returns the average of test sample generated by the previous code
```

```
## [1] 19.46393
```

2. Generate 10 samples from the "undergrads" dataset

```
sample(undergrads, size=10, replace=FALSE)
```

```
##  [1] 20.78143 19.02448 14.81608 20.58074 21.53082 18.77646 23.25561 19.88288
##  [9] 18.06872 15.77235
```

3. Generate 10 new samples and take the mean of that sample

```
testMean <- mean(sample(undergrads, size=10, replace=FALSE))
testMean
```

```
## [1] 20.64041
```

4. Repeat this process 3 times (i.e., generate a sample and take the mean 3 times, using the replicate function)

```
replicate(3,mean(sample(undergrads,size=10,replace=FALSE)))
```

```
## [1] 20.06306 19.53600 19.68166
```

5. Generate a list of sample means from the population called "undergrads"

```
replicate(100,mean(sample(undergrads,size=10,replace=FALSE)))
```

```
##   [1] 20.44243 21.57265 18.72135 19.64541 20.70442 19.59023 18.96526 19.49218
##   [9] 18.96285 18.25319 19.60977 19.46285 18.40900 19.33495 20.61943 20.79078
##  [17] 17.99310 19.04966 20.12940 18.90670 19.03761 19.37586 19.24108 20.60066
##  [25] 19.67494 18.72065 19.24757 18.89767 18.98884 19.19827 19.48709 19.50608
##  [33] 20.95845 19.37816 18.30753 19.41552 19.57424 20.46287 19.38802 18.93010
##  [41] 19.56669 19.81587 20.03654 19.72381 18.91821 19.79987 19.19229 19.41679
##  [49] 19.20274 18.89881 18.50964 19.72894 18.56740 18.75030 19.03572 20.13606
##  [57] 20.90022 19.16884 19.90168 18.28545 18.55450 19.21248 19.33452 20.58819
##  [65] 18.98717 18.39516 19.31001 19.60386 20.43733 18.67803 19.98685 19.24865
##  [73] 20.46636 20.53082 19.32387 18.89741 20.21833 19.16308 18.98281 19.84681
##  [81] 19.55590 18.41502 19.19436 19.96380 18.12822 19.09086 19.20134 19.98830
##  [89] 21.13171 19.57857 19.48440 19.75453 19.34813 19.67955 19.02748 19.71790
##  [97] 20.93485 20.64335 19.49284 20.86317
```

6. Once you have your list of sample means generated from undergrads, the trick is to compare mean(testSample) to that list of sample means and see where it falls. Is it in the middle of the pack? Far out toward one end? Here is one hint that will help you: In chapter 7, the quantile() command is used to generate percentiles based on thresholds of 2.5% and 97.5%. Those are the thresholds we want, and the quantile() command will help you create them.

```
means <- replicate(10000,mean(sample(undergrads,size=10,replace=FALSE)))
quant <- quantile(means, probs = c(0.05,0.95))
quant
```

```
##       5%      95%
## 18.35031 20.60197
```

7. Your code should have a print() statement that should say either, "Sample mean is extreme," or, "Sample mean is not extreme."

```r
if ((mean(testSample) < quant[1]) || (mean(testSample) > quant[2])){
  print("Sample Mean is Extreme")
}else{

  print("Sample mean is not extreme")
}
```

```
## [1] "Sample mean is not extreme"
```

8. Add a comment stating if you think the testSample are undergrad students. Explain why or why not.

```r
# Yes, testSample are undergrad students
# from the above calculated quartiles, 95% of sample means are higher than 18.31257
# and lower than 20.59990
# mean of testSample is 19.46393, it makes the testSample closer to the mean age of undergrads
```

9. Repeat the same analysis to see if the testSample are grad students

```r
means <- replicate(10000,mean(sample(grads,size=10,replace=FALSE)))
quant <- quantile(means, probs = c(0.05, 0.95))
quant
```

```
##       5%      95%
## 23.42106 25.96551
```

```r
if ((mean(testSample) < quant[1]) || (mean(testSample) > quant[2])){
  print("Sample Mean is Extreme")
}else{

  print("Sample mean is not extreme")
}
```

```
## [1] "Sample Mean is Extreme"
```

```r
# from the above calculated quartiles of grads means, 95% of sample means are higher than 23.38948
# and lower than 25.94915
# since, the mean(testSample) is 19.46393, testSample is not closer to the mean age of grads
# Therefor, testSample is not a sample from the grads
```