Simmons-Week10-Lab.R

jesse

2024-03-26

#q1: this is to set seed. By doing so, the initiation point is always the same, not random.  
set.seed(2)   
# assign the value 30 to the variable "sampleSize"  
sampleSize <- 30  
  
#q2: randomly generate 20000 numbers that obeyed normal distribution with mean 20 and standard deviation 3.   
# Then assign those numbers to the vector "studentPop".  
studentPop <- rnorm(20000,mean=20,sd=3)  
summary(studentPop)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 7.774 17.982 20.023 20.016 22.041 31.089

#investigate studentPop now. How many rows? What are the values look like? Are they close to the mean value 20?  
#q3: draw sampleSize (30 sample defined above) from studentPop and assign the 30 numbers to "undergrads".  
undergrads <- sample(studentPop,size=sampleSize,replace=TRUE)  
mean(undergrads)

## [1] 19.52719

#q4: create a sample of graduate students--randomly generate 30 numbers (use sampleSize) that obeyed normal distribution with mean 25 and sd 3. See the mean is 5 years older than the undergraduate sample apparently.   
# Assign the numbers to the vector "grads".  
# Sample size is sampleSize (==30), mean is 25, standard deviation is 3.   
grads <- rnorm(sampleSize,mean=25,sd=3)  
mean(grads)

## [1] 24.89729

#q5: Randomly assign either the grads sample or the undergrads sample to testSample, depending on the value generated by runif(1).  
# "runif(1)" would generate a random number between 0 and 1.   
# If the number is greater than 0.5, assign grads sample to testSample. Otherwise, assign undergrads sample to testSample.  
  
  
if (runif(1)>0.5) { testSample <- grads } else { testSample <- undergrads }  
  
#q6: calculate the mean of "testSample" What is the mean of testSample?  
mean(testSample)

## [1] 24.89729

#After you run this code, the variable “testSample” will contain either a sample of undergrads or a sample of grads. The line before last “flips a coin” by generating one value from a uniform distribution (by default the distribution covers 0 to 1) and comparing it to 0.5. The question you must answer with additional code is: Which is it, grad or undergrad?  
quantile(testSample, probs = c(.025, .975))

## 2.5% 97.5%   
## 19.50119 29.68256

#q7: Generate 100 sample means from studentPop and assign it as mySample  
 mySample <- replicate(100, mean(studentPop) )  
  
  
#q8: Compare mean(testSample) to that list of sample means (mySample) and see where it falls. quantile() function helps display data distribution. Produce quantiles on thresholds 2.5% and 97.5%.  
quantile(mySample, probs = c(.025, .975))

## 2.5% 97.5%   
## 20.01569 20.01569

#q9: if the sample mean is less than quantiles on thresholds 2.5% or greater than quantiles on thresholds 97.5%, then it can be definded as extreme. Otherwise it is not extreme.  
 if (mean(mySample) < quantile(testSample, probs=0.025) | mean(mySample) > quantile(testSample, probs=0.975))  
 { print("Sample mean is extreme") } else { print("Sample mean is not extreme") }

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