QF301. Homework #1.

2021-09-08

I pledge on my honor that I have not given or received any unauthorized assistance on this assignment/examination. I further pledge that I have not copied any material from a book, article, the Internet or any other source except where I have expressly cited the source.

By filling out the following fields, you are signing this pledge. No assignment will get credit without being pledged.

Name: Siddharth Iyer

CWID: 10447455

Date: 9/3/2021

Question 1 (6pt)

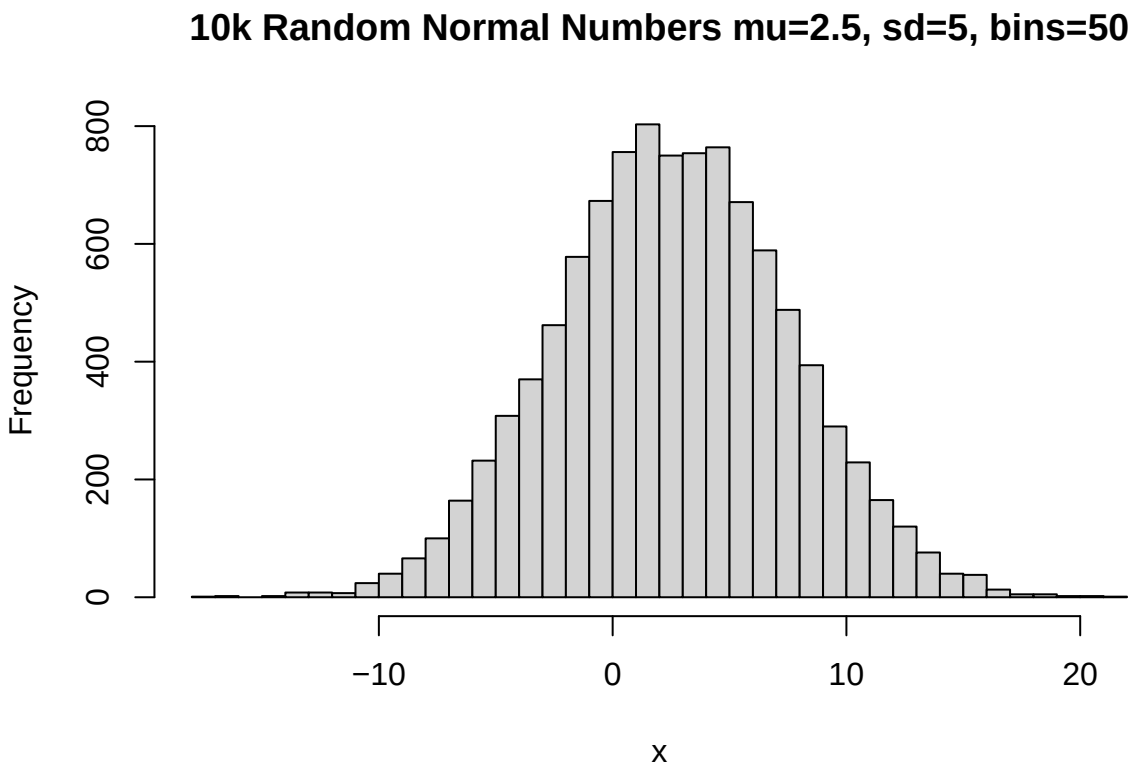
Question 1.1

```
CWID = 10447455
personal = CWID %% 10000
set.seed(personal)
```

Generate a vector `x` containing 10,000 realizations of a random normal variable with mean 2.5 and standard deviation 5.0, and plot a histogram of `x` using 50 bins.

Solution:

```
x = rnorm(10000, mean = 2.5, sd = 5)
hist(x, breaks = 49, main="10k Random Normal Numbers mu=2.5, sd=5, bins=50")
```



Question 1.2

Calculate the mean and standard deviation of these 10,000 values. Do your answers make sense?

Solution:

```
mean(x)
```

```
## [1] 2.56105
```

```
sd(x)
```

```
## [1] 4.984974
```

With a mean of 2.561 and sd of 4.98, this result makes sense because it approximately matches our initial conditions.

Question 1.3

Using the `sample` function, take out 10 random samples of 500 observations each (with replacement). Create a vector of the means of each sample. Calculate the mean of the sample means and the standard deviation of the sample means. What do you observe about these results?

Solution:

```
rep_samples = replicate(10, {
  s = sample(x, 500, replace = T)
  mean(s)
})

rep_samples
```

```
##  [1] 2.757211 2.399806 2.438735 2.373273 2.384784 2.350967 2.968136 2.433883
##  [9] 2.217665 2.458273
```

```
mean(rep_samples)
```

```
## [1] 2.478273
```

```
sd(rep_samples)
```

```
## [1] 0.2190261
```

I notice that the mean (2.478) is approximately the same as the original sample, but the standard deviation is roughly 5/sqrt(500) = 0.2236067. This makes sense, because as the sample size gets smaller, the absolute size of variation will be smaller.

Question 2 (6pt)

Question 2.1

Create a script that creates a vector of the values from 1 to 50 using a for loop.

Solution:

```
vec = c()
for(i in 1:50){
  vec = c(vec, i)
}
vec
```

```
##  [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
## [26] 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50
```

Question 2.2

Create a script that creates a vector of the values from 1 to 50 however you like (just make it different from the above approach)

Solution:

```
# numbers 1-50 in order
new_vec = 1:50
new_vec
```

```
##  [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
## [26] 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50
```

```
# numbers 1-50 out of order but includes all of them n stuff idk
new_vec1 = sample(1:50, size=50, replace=FALSE)
new_vec1
```

```
##  [1]  4 21 49 22 19 15 38  5 36 23 18 35 47 28 25 37 14 39 27 26 31  6  1 24  7
## [26] 42 43 12 20 40 30  3 50  9 17 10 45 16  8 33 29 46 11 48 13 44 34 41  2 32
```

Question 3 (6pt)

Question 3.1

Download and use the "fakedataeasy.csv" file found under Modules section of Canvas. Using this data set, find the mean and standard deviation of each of the variables in this data set.

Solution:

```
df = read.csv(file = "fakedataeasy.csv")
head(df)
```

```
##          y        x5       x6        x7       x8
## 1 359685.4 10.198039 78366.33 260.14274 2852.492
## 2 300760.6 10.392305 65462.75 231.00657 2706.307
## 3 283085.4 10.099505 61782.08 469.51484 2293.194
## 4 232826.3 10.295630 50743.57 481.80357 2396.048
## 5 312396.8  9.949874 67930.96  10.46035 2647.687
## 6 283502.8  9.949874 61812.65 300.33971 2234.187
```

```
for(i in 1:ncol(df)){
  dat = df[,i]
  cat("Variable ", i, "\n")
  cat(mean(dat), "\n")
  cat(sd(dat), "\n")
  cat("\n")
}
```

```
## Variable  1
## 287330
## 36017.43
```

```
##
## Variable  2
## 9.988701
## 0.4944624
##
## Variable  3
## 62557.63
## 7857.351
##
## Variable  4
## 221.878
## 219.6931
##
## Variable  5
## 2507.562
## 315.4524
```