Assignment 1

Jaimie Chin

2023-02-01

Question 1

A researcher collects the ages of students in a graduate program to get a sense if how old people are in the program.

a. Create a data set in R using the above data (using data.frame() would help)

```r
# Create a dataframe of participants and their age

ages = data.frame(Participants = c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15),
                  Participant_Age = c(28, 23, 28, 30, 24, 30, 20, 25, 29, 24, 24, 24, 20, 28, 29))

ages
```

```
##    Participants Participant_Age
## 1             1              28
## 2             2              23
## 3             3              28
## 4             4              30
## 5             5              24
## 6             6              30
## 7             7              20
## 8             8              25
## 9             9              29
## 10           10              24
## 11           11              24
## 12           12              24
## 13           13              20
## 14           14              28
## 15           15              29
```

b. What is the mean? What is the median? What is the mode?

```r
# Create a function to find the mode of a set of data
getMode = function(v) {
  uniq = unique(v)
  uniq[which.max(tabulate(match(v, uniq)))]
}
```

```r
# Find the mean of ages in the sample
mean = mean(ages$Participant_Age)
sprintf("The mean age in this sample is: %s", mean)
```

```
## [1] "The mean age in this sample is: 25.7333333333333"
```

```r
# Find the median of ages in the sample
median = median(ages$Participant_Age)
sprintf("The median age in this sample is: %s", median)
```

```
## [1] "The median age in this sample is: 25"
```

```r
# Find the mode of ages in the sample
mode = getMode(ages$Participant_Age)
sprintf("The mode age in this sample is: %s", mode)
```

```
## [1] "The mode age in this sample is: 24"
```

c. What can you say about the three measures of central tendency?

The three measures of central tendency are relatively similar (~25). This tells us that the data set of Participant Ages do not have extraneous outliers that may have affected the mean of ages. Therefore, because all three measures of central tendency are ~25 years of age, it lets us know that the normal or 'average' age for the participants in this sample is 25.

Question 2

Create a random selection of 10 numbers ranging from 1 to 100 (using the sample() function with replace set to TRUE). You'll need to store this sample into an object (i.e., create a variable) so that you can do calculations with the sample.

```r
# Create a sample following the parameters
sample = sample(1:100, 10, replace=TRUE)
sample
```

```
##  [1]  26  35 100  48  11  26   6  94  29  40
```

a. What is the median? What is the mean? Is there a difference between the two? Why?

```r
# Find the mean of the sample
mean = mean(sample)
sprintf("The mean in this sample is: %s", mean)
```

```
## [1] "The mean in this sample is: 41.5"
```

```r
# Find the median of the sample
median = median(sample)
sprintf("The median in this sample is: %s", median)
```

```
## [1] "The median in this sample is: 32"
```

Yes, there is a minor difference between the mean and median. This is most likely the case because the sample of 10 values, while randomly samples, is skewed slightly towards the left – or the sample contains more values closer to 60. This happens because the mean outlines the center of gravity of data set whereas median highlights the middle-most value of the data set.
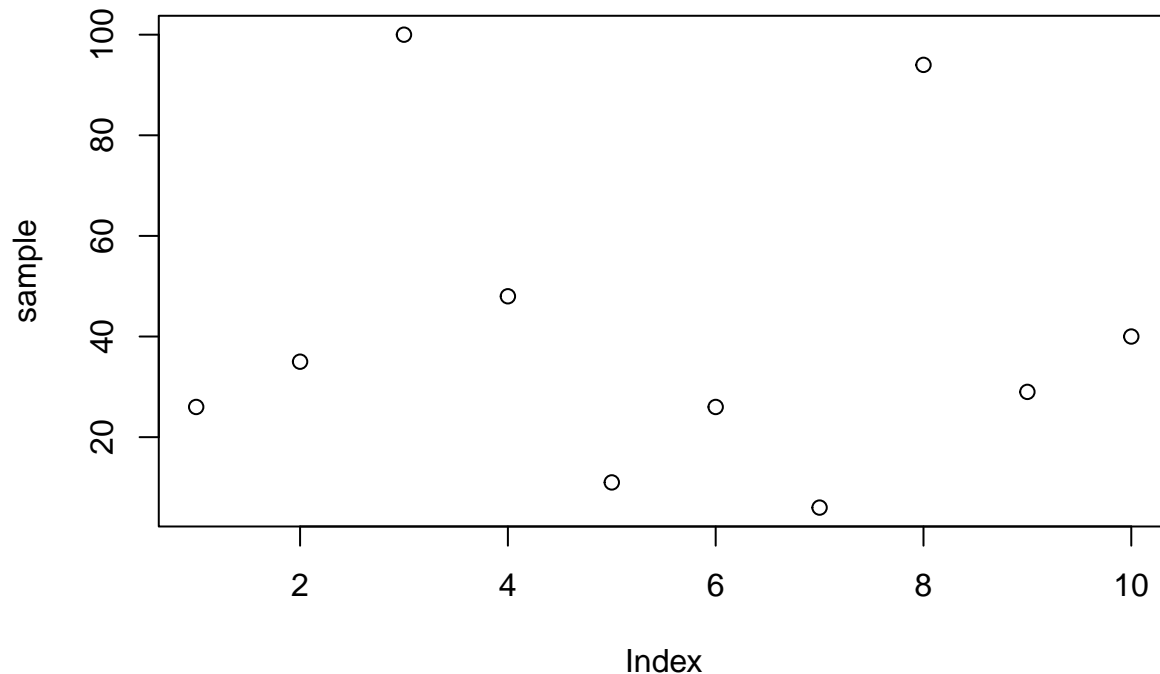
b. Compute the standard deviation

```
# Compute the standard deviation of the sample
sd = sd(sample)
sprintf("The standard deviation of this sample is: %s", sd)
```

```
## [1] "The standard deviation of this sample is: 31.7848671190832"
```

c. Create a plot to visualize these data

```
# Create a scatter plot of the sample values
plot(sample)
```


Now create a random sample of 100 numbers ranging from 1 to 100 (again, using the sample() function with replace set to TRUE).

```
# Create a sample following the parameters
sample = sample(1:100, 100, replace=TRUE)
sample
```

```
##   [1]  57  32  91  70  70  89  96  94  47  22  34  40  64  78  50  36  10  83
##  [19]  73 100  37  14  78  33  10  49  69  30  95  33  53  24  21  81   4  60
##  [37]  15  21  95  50  29  37  44  44  15  41   7  31  68  37  90  98  54  51
##  [55]  72  22  79   1  29  39  68  33  32   6  87   8  68  82  69   6  91  70
##  [73]  41  44  95  47  81  22  61  34  96  78  25  17  65  30  67  73  96  64
##  [91]   6  73  74  78  86  64  44  20   4  37
```

a. What is the median, mean, and standard deviation of this sample?

```
# Find the mean of the sample
mean = mean(sample)
sprintf("The mean in this sample is: %s", mean)
```

```
## [1] "The mean in this sample is: 51.38"
```

```
# Find the median of the sample
median = median(sample)
sprintf("The median in this sample is: %s", median)
```
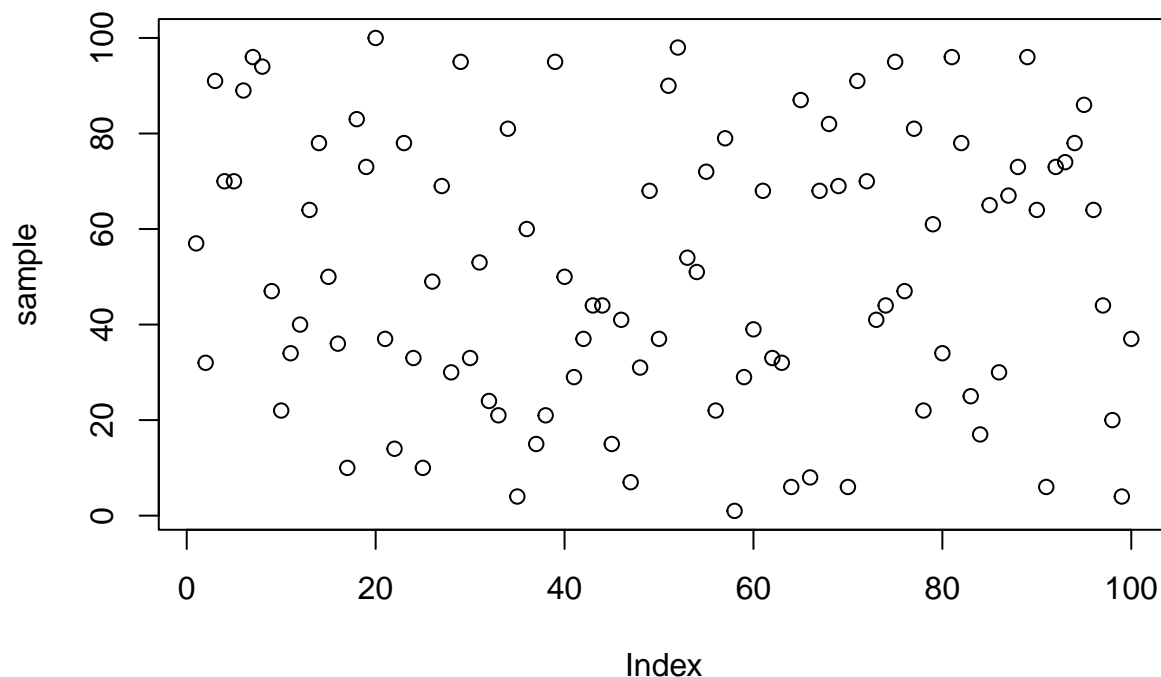
[1] "The median in this sample is: 49.5"

```
# Compute the standard deviation of the sample
sd = sd(sample)
sprintf("The standard deviation of this sample is: %s", sd)
```

[1] "The standard deviation of this sample is: 28.2277104343168"

b. Create a plot to visualize these data

```
# Create a scatter plot of the sample values
plot(sample)
```


c. How do these values from your sample of 100 compare to your first sample of 10? Are they similar/different? Why? [If you expected them to be similar and they are not (or vice versa), why aren't they similar (or different)?]

I expected both the sample of 100 values and 10 values to have no distinct correlation since they are both random samples. Observing both of the plots, the plot with 100 values is more clearly random, but the plot with 10 values also shows a non-correlational relationship between the points. Furthermore, the high standard deviation for the sample with 100 values tells us that the data is greatly dispersed in relation to the mean, which deters a correlation between the values.

What was different was that the sample of 100 values has a greater difference between the mean and median, which tells us that it is even more skewed towards the left compared to the sample of 10 values. This is most likely due to the fact that in a sample with more data values, the mean underestimates the most common values in a negatively skewed distribution. Also, the higher

Question 3

A researcher wants to study the impact of the presence (in the room) of a sweet snack on task completion. 30 participants are given 5 logic problems to solve. Half of the participants are randomly assigned to desks that have only a pencil and the word problems. The remaining participants are assigned to desks with a pencil, the same word problems, and a candy dispenser. Participants are timed and the completion times are recorded. These are the times (in seconds)

```
# Create a dataframe of Candy_Absent group

candy_absent = data.frame(Participant = c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15),
                Reaction_Time = c(501, 536, 659, 317, 530, 523, 381, 573, 535, 509, 604, 704, 370, 44

candy_absent
```

```
##    Participant Reaction_Time
## 1            1           501
## 2            2           536
## 3            3           659
## 4            4           317
## 5            5           530
## 6            6           523
## 7            7           381
## 8            8           573
## 9            9           535
## 10          10           509
## 11          11           604
## 12          12           704
## 13          13           370
## 14          14           440
## 15          15           404
```

```
# Create a dataframe of Candy_Present group

candy_present = data.frame(Participant = c(16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30),
                Reaction_Time = c(690, 691, 510, 586, 675, 470,  533, 693, 440, 614, 475, 374, 500, 47

candy_present
```

```
##    Participant Reaction_Time
## 1           16           690
## 2           17           691
## 3           18           510
## 4           19           586
## 5           20           675
## 6           21           470
## 7           22           533
## 8           23           693
## 9           24           440
## 10          25           614
## 11          26           475
## 12          27           374
## 13          28           500
## 14          29           478
## 15          30           664
```

a. Compute the mean and median for both groups. What do you think about the results you've computed?

```
# Find the mean of reaction times in the Candy Absent group
mean = mean(candy_absent$Reaction_Time)
sprintf("The mean reaction time in the Candy-Absent group is: %s", mean)
```

```
## [1] "The mean reaction time in the Candy-Absent group is: 505.733333333333"
```

```
# Find the median of reaction time in the Candy Absent group
median = median(candy_absent$Reaction_Time)
sprintf("The median reaction time in the Candy-Absent group: %s", median)
```

```
## [1] "The median reaction time in the Candy-Absent group: 523"
```

```
# Find the mean of reaction times in the Candy Present group
mean = mean(candy_present$Reaction_Time)
sprintf("The mean reaction time in the Candy-Present group is: %s", mean)
```

```
## [1] "The mean reaction time in the Candy-Present group is: 559.533333333333"
```

```
# Find the median of reaction time in the Candy Present group
median = median(candy_present$Reaction_Time)
sprintf("The median reaction time in the Candy-Present group: %s", median)
```

```
## [1] "The median reaction time in the Candy-Present group: 533"
```

Observing the mean and median for both groups, it is clear that the group with Candy-Present has generally higher values of reaction time. Meaning, the group with candy on average took more time to solve the word-problems.

b. If you changed the highest score in the Candy-Absent group to be 10 times the original value, what would happen to the mean? What about median?

I would expect the mean to be largely affected by this change as the mean is more sensitive to outliers. I would expect nothing to change with the median as the order and number of values will remain the same, which would not affect the middle value of the group sample.

```
# Find the index of the max reaction time in the Candy Absent group
max_time = which.max(candy_absent$Reaction_Time)

# Create a copy of the candy_absent data frame and change the value of the highest score
candy_absent_copy = cbind(candy_absent)
candy_absent_copy = replace(candy_absent_copy$Reaction_Time, candy_absent_copy$Participant == max_time,

# Find the mean of reaction times in the altered Candy Absent group
mean = mean(candy_absent_copy)
sprintf("The mean reaction time in the altered Candy-Absent group is: %s", mean)
```

```
## [1] "The mean reaction time in the altered Candy-Absent group is: 928.133333333333"
```

```
# Find the median of reaction time in the altered Candy Absent group
median = median(candy_absent_copy)
sprintf("The median reaction time in the altered Candy-Absent group: %s", median)
```

```
## [1] "The median reaction time in the altered Candy-Absent group: 523"
```

c. If you changed the highest score in the Candy-Present group to be one tenth is original value, what would happen to the mean? What about the median?

I would expect the mean to be still affected by this change as the mean is more sensitive to outliers and scaling a value to be smaller (one tenth of its original value), would be an outlier. I would expect the median to change as well, because now the highest score will be the smallest score, which affects the order of the values, which would affect the middle value of the group sample.

```r
# Find the index of the max reaction time in the Candy Present group
max_time = which.max(candy_present$Reaction_Time)
max_time = candy_present$Participant[[max_time]]

# Create a copy of the candy_present data frame and change the value of the highest score
candy_present_copy = cbind(candy_present)
candy_present_copy = replace(candy_present_copy$Reaction_Time, candy_present_copy$Participant == max_ti

# Find the mean of reaction times in the altered Candy Present group
mean = mean(candy_present_copy)
sprintf("The mean reaction time in the altered Candy-Present group is: %s", mean)
```

```
## [1] "The mean reaction time in the altered Candy-Present group is: 517.953333333333"
```

```r
# Find the median of reaction time in the altered Candy Present group
median = median(candy_present_copy)
sprintf("The median reaction time in the altered Candy-Present group: %s", median)
```

```
## [1] "The median reaction time in the altered Candy-Present group: 510"
```

Question 4

A teacher developed a new studying technique and was interested in seeing if it worked. To test this, they had students take a test before using the studying technique (to get a baseline score) and then had students take a similar test one week later (after using the studying technique).

```r
# Create a dataframe of scores

scores = data.frame(Participant = c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13),
                    Pre_Test = c(4, 5, 6, 7, 4, 7, 5, 9, 6, 8, 2, 8, 3),
                    Post_Test = c(6, 6, 5, 8, 8, 5, 6, 9, 4, 8, 5, 7, 2))

scores
```

```
##    Participant Pre_Test Post_Test
## 1            1        4         6
## 2            2        5         6
## 3            3        6         5
## 4            4        7         8
## 5            5        4         8
## 6            6        7         5
## 7            7        5         6
## 8            8        9         9
## 9            9        6         4
## 10          10        8         8
## 11          11        2         5
## 12          12        8         7
## 13          13        3         2
```

a. Plot participants' pre- and post-test scores in a single plot. What do you see? Does there appear to be a relationship between their scores before and after using the studying technique?

```
# Plot Pre-Test & Post-Test
plot(scores$Participant, scores$Pre_Test, pch=19, col="red", xlab = "Participant", ylab = "Score")
points(scores$Participant, scores$Post_Test, pch=19, col = "blue")
legend("topleft", legend = c("Pre-Test", "Post-Test"),
       pch = 19, col = c("red", "blue"))
```
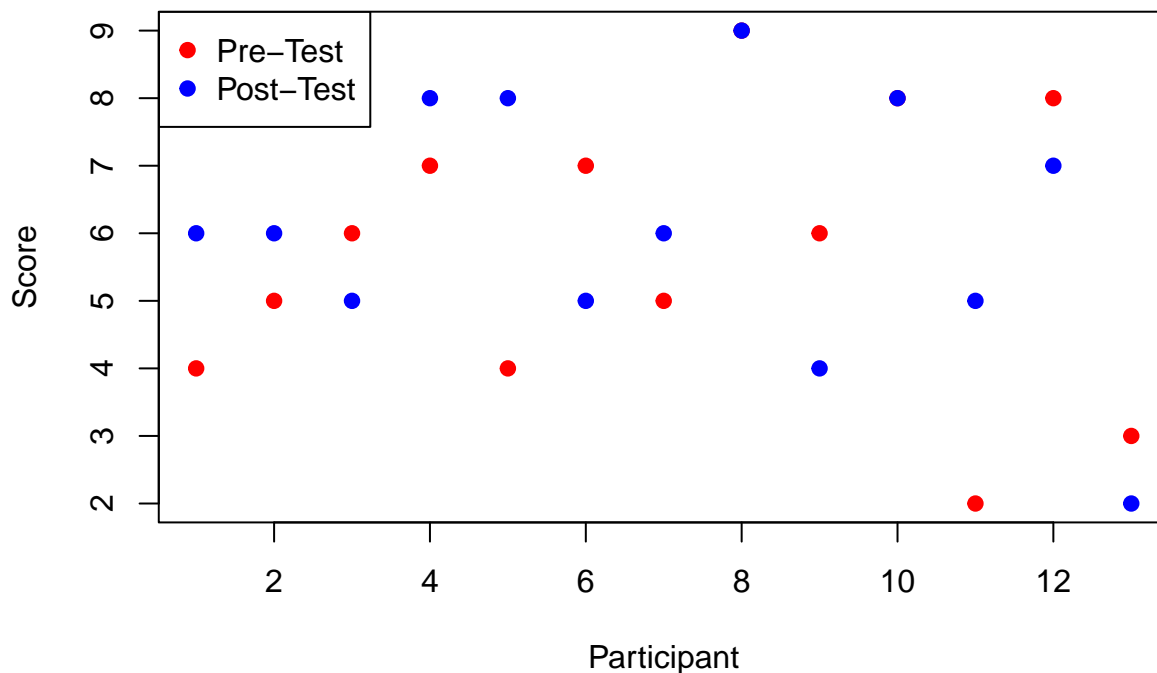

It is difficult to discern from the plot if there is a relationship between scores before and after using the studying technique. Vaguely, it appears that there is somewhat of a positive correlation between pre- and post-test scores – meaning students who performed higher on the pre-test are likely to also perform higher on the post-test.

b. Now check to see if there is a correlation between participants' pre- and post-test scores. What is the correlation? You can use the cor() function in R.

```
# Find the correlation between participants' pre- and post-test scores
corr = cor(scores$Pre_Test, scores$Post_Test)
sprintf("The correlation between participants' pre- and post-test scores is: %s", corr)
```

```
## [1] "The correlation between participants' pre- and post-test scores is: 0.581405769696434"
```

c. What do you think of this teacher's experiment to test their studying technique? Do you think they did a good experiment to yield appropriate data to test the effectiveness of their studying technique?

No, I do not think the teacher's experiment to test the *effectiveness* of their studying technique was a good experiment. Using a correlation to find the relationship between the pre- and post-test tells us nothing about how the *studying technique* positively effected the scores of students – it only tells us how the pre- and post-test scores are related (may be dependent on student's innate ability).

If yes, why? Be concrete about what you think the teacher did right.

If no, give one suggestion (besides running with a larger sample size) the teacher could do to improve their experiment?

I would suggest for the teacher to implement a control group that does not utilize the studying technique and test group that does utilize the studying technique. By doing this, the teacher can clearly use a comparison of mean scores (t-test) to identify how effective using the studying method is compared to not using the studying method.