Formative Assessment 1

Elisha Sophia Borromeo and Zyann Lynn Mayo

2025-01-31

GitHub Link: https://github.com/mayonnayz/FA1_Probability.git

Skewness Program

Write the skewness program, and use it to calculate the skewness coefficient of the four examination subjects in results.txt (results.csv).

```r
library(e1071)
```

After installing in R, I loaded the e1071 package as it contains the skewness function:

```r
exam_results <- read.table("results.txt", header = TRUE, na.strings = "NA")
```

This ensures that my text file is read

```r
head(exam_results, 10)
```

As per instruction, I will only be printing the first 10 for readability

```
##    Gender Arch1 Prog1 Arch2 Prog2
## 1       M    99    98    83    94
## 2       M    NA    NA    86    77
## 3       M    97    97    92    93
## 4       M    99    97    95    96
## 5       M    89    92    86    94
## 6       M    91    97    91    97
## 7       M   100    88    96    85
## 8       F    86    82    89    87
## 9       M    89    88    65    84
## 10      M    85    90    83    85
```

```r
exam_results$arch1 <- as.numeric(exam_results$Arch1)
skewness_arch1 <- skewness(exam_results$arch1, na.rm = TRUE)
cat("Skewness for Arch1:", skewness_arch1, "\n")
```

Take note that we must ensure that the arguments are numeric before proceeding

```
## Skewness for Arch1: -0.5063276
```

```r
exam_results$prog1 <- as.numeric(exam_results$Prog1)
skewness_prog1 <- skewness(exam_results$prog1, na.rm = TRUE)
cat("Skewness for Prog1:", skewness_prog1, "\n")
```

```
## Skewness for Prog1: -0.329161
```

```r
exam_results$arch2 <- as.numeric(exam_results$Arch2)
skewness_arch2 <- skewness(exam_results$arch2, na.rm = TRUE)
cat("Skewness for Arch2:", skewness_arch2, "\n")
```

```
## Skewness for Arch2: 0.4423272
```

```r
exam_results$prog2 <- as.numeric(exam_results$Prog2)
skewness_prog2 <- skewness(exam_results$prog2, na.rm = TRUE)
cat("Skewness for Prog2:", skewness_prog2, "\n")
```

```
## Skewness for Prog2: -0.2977574
```

```r
exam_results$Arch1 <- NULL
exam_results$Prog1 <- NULL
exam_results$Arch2 <- NULL
exam_results$Prog2 <- NULL
```

To avoid redundancy, remove repeating columns

```r
summary(exam_results)
```

We can finally view the reults and compare it with Pearson's

```
##     Gender              arch1           prog1           arch2      
##  Length:119         Min.   :  3.00   Min.   :12.00   Min.   : 6.00  
##  Class :character   1st Qu.: 46.75   1st Qu.:40.00   1st Qu.:40.00  
##  Mode  :character   Median : 68.50   Median :64.00   Median :48.00  
##                     Mean   : 63.57   Mean   :59.02   Mean   :51.97  
##                     3rd Qu.: 83.25   3rd Qu.:78.00   3rd Qu.:61.00  
##                      Max.   :100.00   Max.   :98.00   Max.   :98.00  
##                      NA's   :3        NA's   :2       NA's   :4      
##      prog2      
##  Min.   : 5.00  
##  1st Qu.:30.00  
##  Median :57.00  
##  Mean   :53.78  
##  3rd Qu.:76.50  
##  Max.   :97.00  
##  NA's   :8
```

What can you say about these data?

Upon calculating the skewness for each subject and obtaining their coefficients using the skewness function, it has given me information regarding the distribution of the data. It is said that when skewness is close to 0, then the data is close to symmetrical. If it's positive, then it is right-skewed or the distribution usually has a higher mean than its median while negative skewness is the opposite where the median is usually higher than the mean. This can be observed from the data above. We can see that arch2 has a positive coefficient and its mean is higher than its median while the other three subjects are negative, proposing that their median is higher than their mean. Nonetheless, they are all close to zero, they range from -0.5063276 to 0.4423272, which signifies that the distribution is somewhat symmetrical.

Pearson's Skewness Program

Pearson has given an approximate formula for the skewness that is easier to calculate than the exact formula given in Equation 2.1.Write a program to calculate this and apply it to the data in results.txt (results.csv).

```r
library(knitr)
```

This will print the results of my R code chunks

```r
exam_results <- read.table("results.txt", header = TRUE, na.strings = "NA")
head(exam_results, 10)
```

Let me repeat this action so that R can read my file

```
##    Gender Arch1 Prog1 Arch2 Prog2
## 1       M    99    98    83    94
## 2       M    NA    NA    86    77
## 3       M    97    97    92    93
## 4       M    99    97    95    96
## 5       M    89    92    86    94
## 6       M    91    97    91    97
## 7       M   100    88    96    85
## 8       F    86    82    89    87
## 9       M    89    88    65    84
## 10      M    85    90    83    85
```

Arch1 Values

```r
exam_results$arch1 <- as.numeric(exam_results$Arch1)
mean_arch1 <- mean(exam_results$arch1, na.rm = TRUE)
median_arch1 <- median(exam_results$arch1, na.rm = TRUE)
sd_arch1 <- sd(exam_results$arch1, na.rm = TRUE)
```

```r
skewness_arch1 <- 3 * (mean_arch1 - median_arch1) / sd_arch1
skewness_arch1
```

Calculating skewness of arch1 based on Pearson's

```
## [1] -0.6069042
```

Prog1 Values

```r
exam_results$prog1 <- as.numeric(exam_results$Prog1)
mean_prog1 <- mean(exam_results$prog1, na.rm = TRUE)
median_prog1 <- median(exam_results$prog1, na.rm = TRUE)
sd_prog1 <- sd(exam_results$prog1, na.rm = TRUE)
```

```r
skewness_prog1 <- 3 * (mean_prog1 - median_prog1) / sd_prog1
skewness_prog1
```

Calculating skewness of prog1 based on Pearson's

```
## [1] -0.643229
```

Arch2 Values

```r
exam_results$arch2 <- as.numeric(exam_results$Arch2)
mean_arch2 <- mean(exam_results$arch2, na.rm = TRUE)
median_arch2 <- median(exam_results$arch2, na.rm = TRUE)
sd_arch2 <- sd(exam_results$arch2, na.rm = TRUE)
```

```r
skewness_arch2 <- 3 * (mean_arch2 - median_arch2) / sd_arch2
skewness_arch2
```

Calculating skewness of arch2 based on Pearson's

```
## [1] 0.5421286
```

Prog2 Values

```
exam_results$prog2 <- as.numeric(exam_results$Prog2)
mean_prog2 <- mean(exam_results$prog2, na.rm = TRUE)
median_prog2 <- median(exam_results$prog2, na.rm = TRUE)
sd_prog2 <- sd(exam_results$prog2, na.rm = TRUE)
```

```
skewness_prog2 <- 3 * (mean_prog2 - median_prog2) / sd_prog2
skewness_prog2
```

Calculating skewness of prog2 based on Pearson’s

```
## [1] -0.3562908
```

```
exam_results$Arch1 <- NULL
exam_results$Prog1 <- NULL
exam_results$Arch2 <- NULL
exam_results$Prog2 <- NULL
```

Again, to avoid redundancy, I will remove repeating columns

```
summary(exam_results)
```

We may now compare the previous results with this

```
##     Gender              arch1            prog1           arch2      
##  Length:119         Min.   :  3.00   Min.   :12.00   Min.   : 6.00  
##  Class :character   1st Qu.: 46.75   1st Qu.:40.00   1st Qu.:40.00  
##  Mode  :character   Median : 68.50   Median :64.00   Median :48.00  
##                     Mean   : 63.57   Mean   :59.02   Mean   :51.97  
##                     3rd Qu.: 83.25   3rd Qu.:78.00   3rd Qu.:61.00  
##                     Max.   :100.00   Max.   :98.00   Max.   :98.00  
##                     NA's   :3        NA's   :2       NA's   :4      
##      prog2      
##  Min.   : 5.00  
##  1st Qu.:30.00  
##  Median :57.00  
##  Mean   :53.78  
##  3rd Qu.:76.50  
##  Max.   :97.00  
##  NA's   :8      
```

Is it a reasonable approximation?

Comparing the results from the built-in skewness function and the results that were derived from Pearson's skewness approximation, With Pearson's ranging from -0.643229 to 0.5421286 and the skewness function ranging from -0.5063276 to 0.4423272, we can say that it is a reasonable approximation. Note that Arch2 is still the only subject that has a positive skewness and the other three have negatives just like in the skewness function. Still, they are varying values, which reminds us that Pearson's formula is merely an approximation—it is not accurate. Regardless, it gives us an insight regarding the distribution's skewness as it provides close values.

STEM-AND-LEAF AND BOXPLOT GRAPH

```
Female <- c(57, 59, 78, 79, 60, 65, 68, 71, 75,
            48, 51, 55, 56, 41, 43, 44, 75, 78,
            80, 81, 83, 83, 85)

Male <- c(48, 49, 49, 30, 30, 31, 32, 35, 37,
          41, 86, 42, 51, 53, 56, 42, 44, 50,
          51, 65, 67, 51, 56, 58, 64, 64, 75)
```

Assign the score values of both genders into variables "Female" and "Male"

Part A: Form the stem-and-leaf display for each gender, and discuss the advantages of this representation compared to the traditional histogram.

```
stem(Female)
```

Female Stem-and-Leaf

```
##
##   The decimal point is 1 digit(s) to the right of the |
##
##   4 | 1348
##   5 | 15679
##   6 | 058
##   7 | 155889
##   8 | 01335
```

```
stem(Male)
```

Male Stem-and-Leaf

```
##
##   The decimal point is 1 digit(s) to the right of the |
##
```

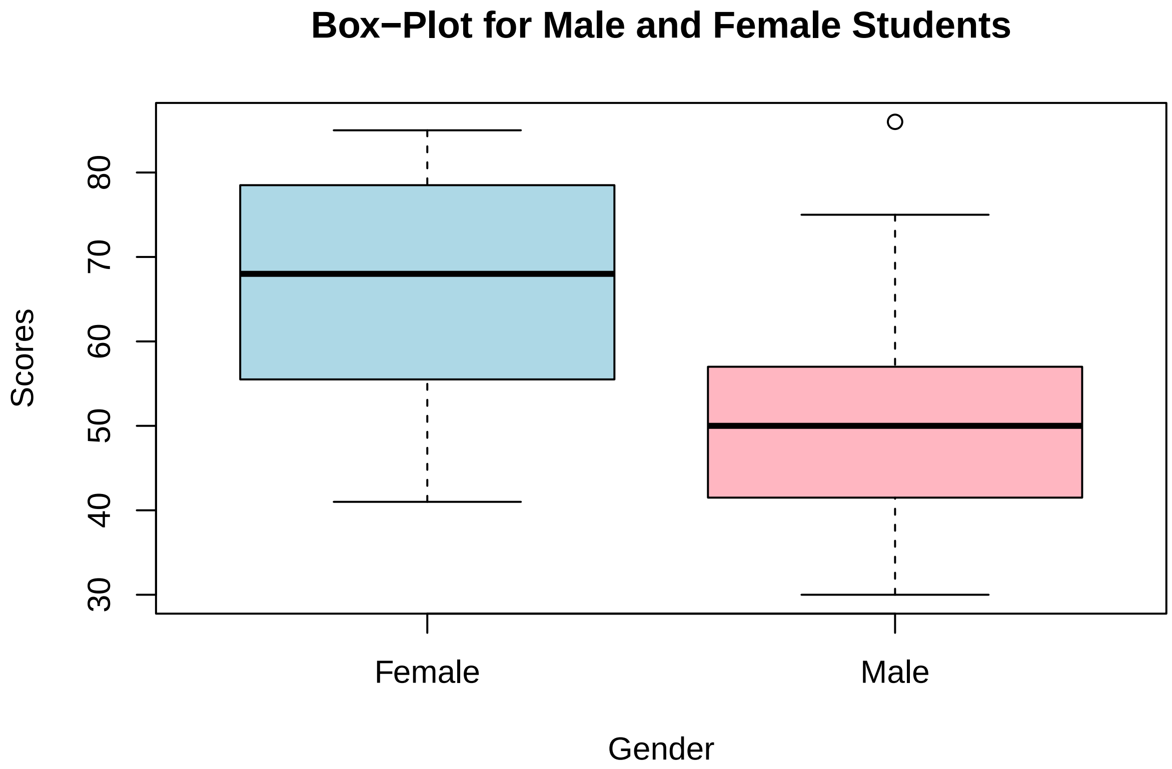
```
##   3 | 001257
##   4 | 1224899
##   5 | 01113668
##   6 | 4457
##   7 | 5
##   8 | 6
```

Both the stem-and-leaf and histogram visualize the distribution of a set of data within a range. Despite this, the advantages of the representation in the stem-and-leaf is that you can see the exact data values and not just grouped within a range of numbers. With this, the repetition or frequency of specific numbers in the data can also be seen. For example, in the traditional histogram, the bar could show a range between numbers 40 and 49, which means that it could contain all numbers from 40 to 49, but with the stem-and-leaf, the actual values are revealed.

Part B: Construct a box-plot for each gender and discuss the findings.

```r
scores <- data.frame(
  score = c(Male, Female),
  gender = factor(c(rep("Male", length(Male)), rep("Female", length(Female))))
)

boxplot(score ~ gender, data = scores,
        main = "Box-Plot for Male and Female Students",
        xlab = "Gender",
        ylab = "Scores",
        col = c("lightblue", "lightpink"))
```



As this reveals the box plot for both female and male scores, let us interpret and compare them. Firstly, it shows that the median of the female scores is slightly below 70, while the males' are close to 50, indicating that females, on average, scored higher than males. As for the spread of the scores determined by the interquartile range (IQR), which is indicated by the areas with color, it shows how there is more variation among the female scores compared to the male score. Additionally, an outlier in the male box plot is evident with a value of 86, the highest score among the 50 students. This points out an unusually extreme score in comparison to the rest of the male scores. To summarize the findings, the box plot highlights that females have more variability in their scores and tend to score higher on average than males, whose scores are more clustered at the lower end.