SEC 1-FA1 GROUP 1-SIGUE, JP

2025-01-30

Write the skewness program, and use it to calculate the skewness coefficient of the four examination subjects in results.txt (results.csv). 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
results <- read.table("results.txt", header = TRUE)

skew <- function(x) {
  xbar <- mean(x, na.rm = TRUE)
  sum2 <- sum((x-xbar) ** 2, na.rm = TRUE)
  sum3 <- sum((x-xbar) ** 3, na.rm = TRUE)
  skew <- (sqrt(length(x))* sum3)/(sum2 ** (1.5))
  skew
}

pearson_skewness <- function(x){
  mean_x <- mean(x, na.rm = TRUE)
  median_x <- median(x, na.rm = TRUE)
  sd_x <- sd(x, na.rm = TRUE)

  skew <- (3 * (mean_x - median_x)) / sd_x
  return(skew)
}

skewness_results <- data.frame(
    arch1 = c(skew(results$Arch1) ,pearson_skewness(results$Arch1)),
    prog1 = c(skew(results$Prog1) ,pearson_skewness(results$Prog1)),
    arch2 = c(skew(results$Arch2) ,pearson_skewness(results$Arch2)),
    prog2 = c(skew(results$Prog2) ,pearson_skewness(results$Prog2))
  )

rownames(skewness_results) <- c("Normal Skewness", "Pearson Skewness")

print(skewness_results)
```

```
##                        arch1      prog1     arch2      prog2
## Normal Skewness  -0.5195368 -0.3362643 0.4558875 -0.3125144
## Pearson Skewness -0.6069042 -0.6432290 0.5421286 -0.3562908
```

What can you say about these data?

Using Normal Skewness, Prog1, Prog2, and Arch2 are approximately symmetric, while Arch1 is moderately skewed to the left, meaning more students scored higher than lower.

Is it a reasonable approximation?

Pearson's skewness values are somewhat close to the normal skewness values but generally more extreme. This suggests that Pearson's formula is a good approximation but slightly overestimates skewness magnitude.

For the class of 50 students of computing detailed in Exercise 1.1, use R to

form the stem-and-leaf display for each gender, and discuss the advantages of this representation compared to the traditional histogram

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

print("First-Year Java Programming Exam Scores of Females")
```

```
## [1] "First-Year Java Programming Exam Scores of Females"
```

```
print(stem(Females))
```

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

```
print("First-Year Java Programming Exam Scores of Males")
```

```
## [1] "First-Year Java Programming Exam Scores of Males"
```

```
print(stem(Males))
```

```
##
##   The decimal point is 1 digit(s) to the right of the |
##
##   3 | 001257
##   4 | 1224899
```

```
##    5 | 01113668
##    6 | 4457
##    7 | 5
##    8 | 6
##
## NULL
```

A stem-and-leaf display organizes data by splitting numbers into stems and leaves, keeping the original values visible for easy comparison. It's simple to create, highlights individual data points, and helps identify patterns, but histograms might be more useful for large datasets.

construct a box-plot for each gender and discuss the findings

```r
boxplot(Females, Males, names = c("Females", "Males"),
        main = "First-Year Java Programming Exam Scores",
        ylab = "Scores",
        ylim = range(0,100),
        col = c("pink", "lightblue"))
```

First−Year Java Programming Exam Scores
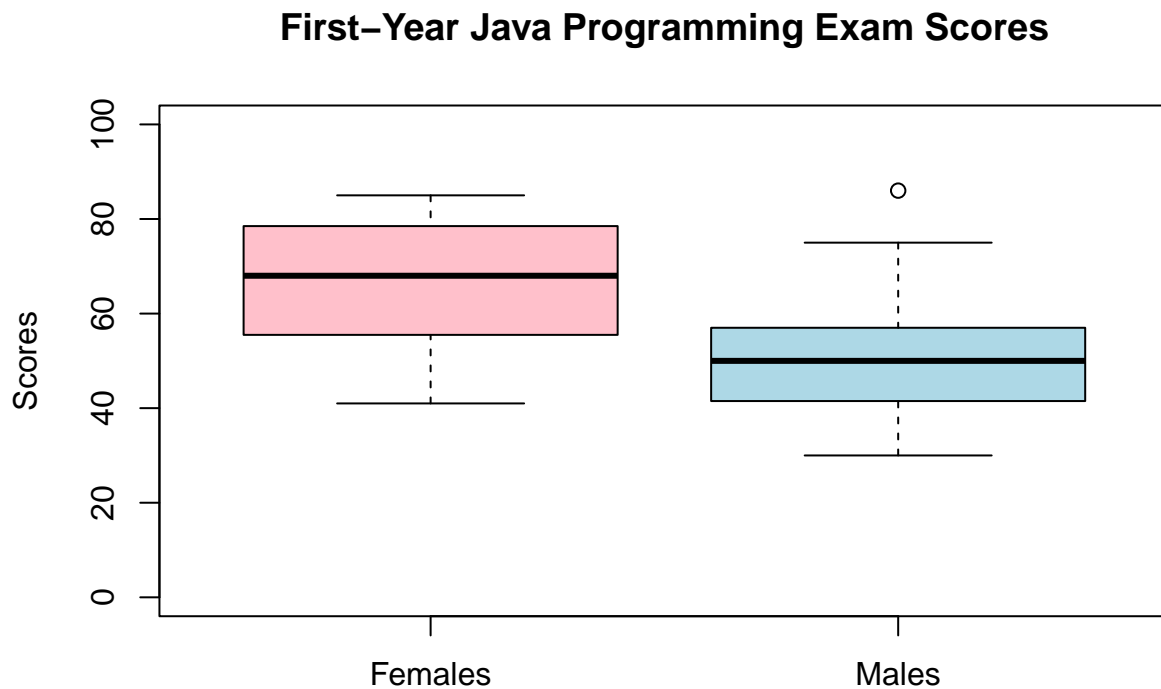
The graph shows that the females, as a group, performed better than the males. However, the female score distribution exhibits greater variability than the males while the male scores are more around their median.