Lab 3 – Numerical Descriptive Statistics

This lab contains some numbered questions. You are required to submit:

1. One pdf document (Lab 3.pdf) that contains:
   - your written answers to the question
   - the commands you used to answer the question, when asked

2. One R script (Lab 3.R) that contains all the code used to produce your answers. The script must run without errors.

Submit your files to the Lab 3 folder in Learning Hub by 11:59pm next school day.

We are going to analyze data about student absences from school in New South Wales, Australia. This data can be found in the built-in dataset **quine**, which is part of the **MASS** library. Load and view this dataset and get familiar with its help file.

Packages required: MASS, mosaic, dplyr

Functions in MOSAIC Package

The mosaic package was developed by college and university professors to make the syntax of R more approachable. Since R is open-source software, many built-in commands have inconsistent syntax. The mosaic commands are more consistent.

Typical mosaic function syntax:

```
goal(y ~ x , data = mydata, … )
```

- The **goal** is what you want R to do.
- *y* and *x* are the **variables** that R will need to achieve the goal that you want to achieve.
- **mydata** is the dataframe in which these variables are stored.
- The **…** refers to additional options that you can include in the command. We will discuss these later (often includes formatting options, axis titles, etc.)

When you are only working with one variable, the *y* is omitted:

```
goal ( ~x , data = mydata, …)
```

One Quantitative Variable Commands

We can find the mean number of days of absence among the students:

```
> mean(~Days, data=quine)
[1] 16.4589
```

That is, the mean number of days absent was 16.4589.
The following functions follow the same syntax.

Function	What it does
`mean`	calculates the mean
`median`	calculate the median
`sd`	calculate the sample standard deviation
`var`	calculate the sample variance
`min`	determines the minimum
`max`	determines the maximum
`sum`	computes the sum of all the values for that variable
`IQR`	calculate the inter-quartile range
`quantile`	calculate percentiles or quartiles (specify using `probs=k/100`)
`favstats`	Computes our "favourite" statistics: minimum, first quartile, median, third quartile, maximum, mean, standard deviation, sample size, and how many values were missing.
`histogram`	Creates a histogram giving the distribution of the variable of interest.
`bwplot`	Creates a box-and-whisker plot (or boxplot) of the variable of interest.

1. Calculate the mean, median, and standard deviation of **Days** using the function `favstats()`. Record the command and the three numerical statistics asked for.

2. Using functions listed in the above table (not `favstats`), find Pearson's coefficient of skewness (Sk) for the number of days of absence. Is the variable **Days** data symmetric, skewed left, or skewed right?

3. Use `histogram()` to create a histogram for the variable **Days**. Use classes of width 5 starting from zero; add a suitable title that includes the value of n (without hard-coding). How does the shape of the histogram reflect your calculated value of Sk?

4. Use `bwplot()` to create a boxplot for the variable **Days**. Label the X-axis appropriately and provide a title including n (without hard-coding).

5. Find the 25th and 75th percentiles of **Days** and state your result as an interval that contains the *middle 50%* of students: "The middle 50% of students were absent between … and … days."

6. How many outliers are there for variable **Days**? Use an appropriate formula to determine "lower fence" and "upper fence" values to check.

One Quantitative Variable and One Categorical Variable Commands

We may be interested in comparing absences of students in different categories, for instance boys and girls. We can do this by computing statistics for the **Days** field, grouped by **Sex**.

```
> favstats(Days~Sex, data=quine)
  Sex min    Q1 median    Q3 max     mean       sd   n missing
1   F   0  5.00     10 20.25  81  5.22500 15.93100  80       0
2   M   0  5.25     14 27.00  69 17.95455 16.63401  66       0
```

Here we notice a few things:
- There were both boys and girls who had perfect attendance (zero days of absence).
- At each quartile (Q_1, Q_2, Q_3) boys had more absences than girls.
- However, the student with the most absences was female.

All the mosaic functions can be used in this way (grouping by a categorical variable).

7. Create side-by-side boxplots for the number of **Days** of absence, grouped by **Age**. Also calculate `favstats` for **Days** grouped by Age. Make three observations about how days of absence differ between the four groups.

We can also group data by multiple factors; for instance, we can find the mean number of absences, grouped by ethnicity and sex.

```
> mean(Days~Eth+Sex, data=quine)
     A.F      N.F      A.M      N.M
20.92105 10.07143 21.61290 14.71429
```

8. Use appropriate commands to determine which group of students, grouped by age and sex, is most consistent with regards to absences, and which is least consistent.

Measures of Position

The `favstats` function returned the values corresponding to the quartiles: 25%, 50% (median), and 75%. The `quantile` function allows us to generalize to other percentages.

If we are interested in the quartiles for days of absence, use:

```
> quantile(~Days, data=quine)
   0%   25%   50%   75%  100%
 0.00  5.00 11.00 22.75 81.00
```

(Note: you may get an error after this output, because R is missing a package. Install the **Rcpp** package and run the command again and the error should disappear.)

We can obtain different percentile values with the `probs=` argument. For instance, suppose we are interested in the 90th percentiles of absences:

```
> quantile(~Days, probs=0.9, data=quine)
90%
 40
```

We can find multiple percentiles by listing them as sequences:

```
> quantile(~Days, probs=c(0.1, 0.9), data=quine)
10% 90%
  2  40

> quantile(~Days, probs=seq(0.1,0.9, 0.1), data=quine)
10% 20% 30% 40% 50% 60% 70% 80% 90%
  2   5   5   7  11  14  20  27  40
```

9. Calculate the 20th, 40th, 60th, and 80th percentiles for Days of absence, grouped by each of the following categorical variables (This requires three separate commands.) Record each output.
   - learner status
   - sex
   - ethnicity

   Which categorical variable (e.g., Lrn, Sex, Eth) seems to make the greatest difference on the Days of absence at these percentiles?

We can also go in the other direction: that is, we can find the percentile rankings for each data value. We use the `percent_rank` function to do this. (This function is part of the **dplyr** package.) It is not a **mosaic** function, so its syntax is different.

```
> percent_rank(quine$Days)
  [1] 0.08965517 0.47586207 0.55862069 0.17931034 0.17931034
0.53793103
```

This tells us that the first value in the **Days** column represents the 8.965517th percentile of the Days variable (i.e., it is larger than 8.965517% of the variable values).

Percentile rankings are usually given as whole numbers, so format the output accordingly:

```
> round(100*percent_rank(quine$Days))
  [1]   9  48  56  18  18  54  70  72  31  31  61  37  56  31  83  95
 [17]  97  56  63  63  66  90  92  93  41  75  75  81  86  86  89  13
```

This tells us that the first entry in the table is from a student who was absent more often than 9% of students.

10. Give a list of percentile rankings (two decimal places) for the number of days absent for the male F0 students only. Give all the commands you used, as well as your output **as a column**. (Refer to previous labs for relevant functions.)

Z-scores, Chebyshev's Theorem, and the Empirical Rule

It is often useful to know a data value's Z-score (i.e., how many standard deviations that value is away from the mean). Like percentile rankings, Z-scores give a measure of how large or small a data value is relative to the other data values.

In R, the `scale` function (non-mosaic syntax) provides Z-scores of all data in a list. Applying this function to the **Days** column of our table shows, for instance, that the first student value 0.890 standard deviations below the mean:

```
> head(scale(quine$Days))
       [,1]
[1,] -0.89
[2,] -0.34
[3,] -0.15
[4,] -0.71
[5,] -0.71
[6,] -0.21
```

11. Find the percentages of students who were absent for a number of days that were
    - at least 1 standard deviation from the mean
    - at least 2 standard deviations from the mean
    - at least 3 standard deviations from the mean

12. Do your results from Question 9 satisfy Chebyshev's Theorem? Do they satisfy the Empirical Rule? If not, explain why this is with reference to answers to earlier questions.