DATA 505: Homework 2

Weeks 2 and 3 (classes and types; data frames; descriptive statistics and summarizing data)

2025-09-16

Context

In this homework, you will reinforce concepts from weeks 2 and 3. Using what you learned in week 2, you will practice inspecting, subsetting, and cleaning data, as well as understanding how R handles different types of data. Then, using what you learned in week 3, you will practice producing descriptive summaries of the variables in a dataset.

References

- Lecture 2
- Lecture 3
- Chapter 5 of: Grolemund, G. (2014). Hands-On Programming with R. O'Reilly Media. https://rstudio-education.github.io/hopr/
- Chapter 5 of: Navarro, D. J. (2019). Learning statistics with R: A tutorial for psychology students and other beginners (Version 0.6.1). University of Adelaide. Retrieved from https://learningstatisticswithr.com. Link to pdf

Coding Tasks

Create an R script and use it to answer the following questions.

Part 1: Data frames, types, classes, coercion

1. Reading and Inspecting Data

In this lab we will explore a dataset containing size measurements for 3 species of penguins from the Palmer Archipelago in Antarctica.

a. Use the command like below to read the penguins data into R as a data frame called `penguins` (see the link above).

```
penguins <- read.csv(
  "https://raw.githubusercontent.com/allisonhorst/palmerpenguins/refs/heads/main/inst/extdata
)
```

b. Use `class()`, `typeof()`, and `str()` to inspect the `penguins` object. What do you notice about the types and classes of the columns?

c. Use `names()` to list all column names in the data frame.

2. Subsetting and Accessing Data

a. Use the `$` operator to extract the `body_mass_g` column and store it as a new object called `body_mass`. What is the type of this object?

b. Use the `[["body_mass_g"]]` syntax to extract the same column. Is the result the same as above?

c. Use `length()` to determine how many body mass values are in the data set.

d. Extract the body mass of the first penguin using vector subsetting.

e. Use logical subsetting to extract the body mass values for all penguins on the `"Dream"` island.

3. Working with Types and Coercion

a. Use `typeof()` to check the type of the `body_mass` vector. If it is not numeric, use `as.numeric()` to coerce it and assign the result back to `body_mass`.

b. Update the `penguins` data frame so that the `body_mass_g` column is numeric (if it is not already).

c. Use `typeof()` and `class()` to check the type and class of the `species` column. If it is not a factor, convert it using `as.factor()`.

d. Use `levels()` to list the possible values of `species`.

4. Data Frame Operations

a. Create a new data frame containing only penguins with a body mass of at least 4000 grams.

b. Create a new data frame containing only the `species` and `island` columns for penguins with a non-missing body mass.

c. How many penguins have missing (`NA`) values for body mass? Use `is.na()` and `sum()`.

5. Challenge: Creating a Categorical Variable

a. Create a new factor column in the `penguins` data frame called `mass_group` with three levels:

- "light" for body mass less than 3500 grams
- "medium" for body mass between 3500 and 4500 grams (inclusive)
- "heavy" for body mass greater than 4500 grams

b. Display the `species`, `body_mass_g`, and `mass_group` columns for all penguins.

Part 2: Descriptive statistics and summarizing data

1. Exploring distributions

a. For the `body_mass_g` variable:

- Calculate the mean, median, and standard deviation.
- What are the minimum and maximum values?
- What are the 25th and 75th percentiles (first and third quartiles)? What is the interquartile range (IQR)?
- Create a histogram of `body_mass_g`.
- Write a brief description of the distribution: Is it symmetric, skewed, or does it have outliers?

2. Summarizing categorical variables

a. Use `table()` to create a frequency table for the `species` variable.

b. Use `prop.table()` to calculate the proportion of each species.

c. Describe the distribution of species in the dataset.

3. Challenge: Comparing groups

a. Calculate the mean and median `body_mass_g` for each species (hint: use `tapply()` or `aggregate()`).

b. Create side-by-side boxplots of `body_mass_g` by species.

c. Write a short interpretation: Which species tends to be heavier? Are there differences in spread or outliers?