Lab2

Exercises

Irimie Fabio

Contents

Exercise 1

- A:

  Create the Lab2 project. Use the same structure used for Lab1:

  - scripts,
  - plots,
  - data

- B:

  Install the palmerpenguins package, load the penguins dataset or, alternatively, download the .RData object from moodle and import it after placing it inside the data directory of the project (hint: use the load() function).

- C:

  Compute the mean, the standard deviation, and the median for the numeric variables of the dataset.

- D:

  Create a function called stat_auto that simultaneously returns both the mean and the standard deviation of a given vector (hint: return an object of type list or simply a vector). Then try it on the same numeric variables in C. to check the results (hint: if you obtain NA maybe you forgot to remove NA terms in the vector).

- E:

  Create a function called stat_manual that simultaneously returns both the mean and the standard deviation of a given vector without using the mean() and the sd() functions (hint: you can use length(), sum(), and na.omit() functions). Then try it on the same numeric variables in C. to check the results.

B

```
library(palmerpenguins)
data(penguins)
```

C

```
# Means
cat("Means: \n")
## Means:
colMeans(penguins[, c(3:6, 8)], na.rm = TRUE)
##    bill_length_mm     bill_depth_mm flipper_length_mm       body_mass_g
##          43.92193          17.15117         200.91520        4201.75439
##              year
##        2008.02907
cat("\n")

# Medians
cat("Medians: \n")
## Medians:
sapply(penguins[, c(3:6, 8)], median, na.rm = TRUE)
##    bill_length_mm     bill_depth_mm flipper_length_mm       body_mass_g
##             44.45             17.30            197.00           4050.00
##              year
##           2008.00
cat("\n")

# Standard deviations
cat("Standard deviations: \n")
## Standard deviations:
sapply(penguins[, c(3:6, 8)], sd, na.rm = TRUE)
##    bill_length_mm     bill_depth_mm flipper_length_mm       body_mass_g
##         5.4595837         1.9747932        14.0617137       801.9545357
##              year
##         0.8183559
cat("\n")
```

D

```
stat_auto <- function(vec, na.rm = FALSE) {
  if (na.rm) {
    mean <- mean(vec, na.rm = TRUE)
    sd <- sd(vec, na.rm = TRUE)

    return(list("mean" = mean, "sd" = sd))
  }

  mean <- mean(vec)
  sd <- sd(vec)

  return(list("mean" = mean, "sd" = sd))
}

sapply(penguins[, c(3:6, 8)], stat_auto, na.rm = TRUE)
```

```
##      bill_length_mm bill_depth_mm flipper_length_mm body_mass_g year
## mean 43.92193       17.15117      200.9152          4201.754    2008.029
## sd   5.459584       1.974793      14.06171          801.9545    0.8183559
```

E

```
stat_manual <- function(vec, na.rm = FALSE) {
  if (na.rm) {
    sum <- sum(vec, na.rm = TRUE)
    mean <- sum / na.omit(length(vec))

    sum <- sum((vec - mean)^2, na.rm = TRUE)
    denom <- na.omit(length(vec)) - 1
    varianza <- sum / denom

    sd <- sqrt(varianza)

    return(list("mean" = mean, "sd" = sd))
  }

  sum <- sum(vec)
  mean <- sum / length(vec)

  sum <- sum((vec - mean)^2)
  denom <- length(vec) - 1
  varianza <- sum / denom

  sd <- sqrt(varianza)

  return(list("mean" = mean, "sd" = sd))
}

sapply(penguins[, c(3:6, 8)], stat_manual, na.rm = TRUE)
```

```
##      bill_length_mm bill_depth_mm flipper_length_mm body_mass_g year
## mean 43.66657       17.05145      199.7471          4177.326    2008.029
## sd   5.449612       1.971543      14.06909          799.985     0.8183559
```