Lab 2: Advanced R

Question Sheet

Dr Mercedes Torres Torres

Introduction

This question sheet has a series of exercises designed to help you deepen your knowledge of R. We will focus on creating functions, and reading and writing from different types of files. You will also practice how to implement conditional and iterative structures, like if and for loops.

In this lab session, you will learn to:

- Use control structures
- Write functions in R
- Read from/ Write to file
- Create and use *lists* and *data frames*, including accesing its data, manipulating it and storing it
- Use the *apply()* functions

1 Creating Functions

1. Create a function that will solve quadratic equations in the form $ax^2 + bx + c = 0$. The function will take three parameters (a, b and c) and return a vector x with both solutions to the equation. What is the solution of the following equations?

   a. x2 - 5x - 3 = 0

   b. x2 - 4 = 0

   c. 2x2 + 4x - 4 = 0

   d. 5x2 + 2x - 1 = 0

   e. 6x2 + 11x - 35 = 0

2. Write a function that given a matrix A of logical values, returns the number of TRUE values and the number of FALSE values on a vector v = (True_values False_values).

3. Create a function that given two vectors v and u, returns two vectors: one with the values present in both vectors and one with the values present in one of the vectors and not the other.

4. Create a function that given two lists L and P, returns a list C with the concatenation of both lists, and prints the number of elements of C.

5. Write a function that given a matrix A, returns the maximum value in A. You cannot use the function max().

6. Write a function that given an N-dimensional point x and an MxN-dimensional matrix A, returns the closest point to x in A.

7. Write a function that, given a numerical one-dimentional list L, returns which values inside that list are prime numbers. (Hint: Remember *schoolmath*)

2 Lists and Data Frames

1. Given four lists *Food*, *Nutrition*, *Calories*, and *Prices*:

   a. Create a function called *generateMenu(food, nutrition, cals, prices)* that returns a data frame called *menu* with that information. Call the function with the following lists:

   - *Food*: Toast, Pancakes, Eggs Benedict, Chocolate Muffin, Bacon, Coffee, Tea, Croissant, Oatmeal, Banana, Apple, Sausage, Orange Juice
   - *Nutrition*: Vegan, Vegetarian, Vegetarian, Vegetarian, Meat, Vegan, Vegan, Vegetarian, Vegetarian, Vegan, Vegan, Meat, Vegan
   - *Calories*: 150, 500, 250, 300, 100, 1,1, 260, 2290, 50, 10, 125, 70
   - *Price*: 2.5, 5.75, 8, 4.5, 5, 2, 3.25, 3.75, 3, 1.5, 1.5, 2.75, 2.5

   b. Which food items have lower than 250 calories?

   c. Which of the food items that are Vegetarian are cheaper than £6?

   d. What is the most expensive Vegan item on the menu?

   e. What are the average calories of each of the Nutrition groups?

   f. What is the Nutrition group with less food items?

3 Reading and Writing

1. Modify your *generateMenu(. . .)* function so it saves your data frame menu into a CSV file.

   a. Write a function that, given a *Nutrition* preference, prints in a CSV file which items in the menu comply with it and how much they cost.

   b. Write a function that, given a data frame called *order* which has a variable number of food items and the quantity of these, prints the price of ordering all of the items on screen.

   c. Write a function that given a maximum number of calories, prints a list of the food items that have those calories or less and their nutrition information on a CSV file.

   d. Write a function that given a variable called budget which is the maximum amount of money you can spend, prints on screen the three food items closest in price to budget.

2. We have collected the marks obtained in five different modules (*Biology, Maths, Chemistry, Physics and Programming*) from six different students:

   a. Create a matrix with the marks. What is the average mark for each person? And the standard deviation in their grades? What is the minimum mark for each subject?

   - John: 32, 52, 50, 44, 50
   - Mary: 88, 67, 59, 70, 70
   - Mark: 78, 77, 68, 67, 80
   - June: 89, 90, 81, 89, 87
   - Claire: 61, 65, 50, 78, 50
   - Anthony: 67, 68, 65, 40, 66

   b. Write a function that creates a single data frame compiling all of this information, including module name, student name and gender. Make sure that all rows and columns are named correctly. This function should also save the data frame in a file called “class_marks.csv”.

- John: M, 18
- Mary: F, 19
- Mark: M, 19
- June: F, 19
- Claire: F, 18
- Anthony: M, 18

c. Read this files into a variable called *class2020* and edit it to add a new module: Statistics. Use *class2020* in the remainder of the exercises.

- John: 89
- Mary: 99
- Mark: 76
- June: 85
- Claire: 96
- Anthony: 65

d. Write a function that, given a name, returns the highest score that person has obtained and the name of the module.

e. Write a function that, given a name

   a. If the name is present in the data frame, it returns the average score of that person and the average score of the people with the same age.

   b. If the name is not present, it prints an error message.

f. Write a function that, given a module and a threshold mark, it saves on a TXT the name of people who have obtained less than that mark in the module.

g. Write a function that, given a module, prints the name of the students in descending order according to their marks on the screen.

h. Write a function that, given a module, returns the average score participants according to age.