R exercices
Basic object manipulation and inspection

Gilles Tredan

Abstract

1 Vectors

1.1 Let's start simple

- Create the following vectors $e1 = (2, 5, 0, 8)$, $e2 = (1, 2, 3, ..., 200)$, $e3 = (-200, -202, -204, ... - 210)$, $e4 = (2, 4, 8, 16, 32, 64, 128)$.

- Create the vector v of 50 elements such that $v[2i] = -1$, $v[2i + 1] = 1$,

- Create the vector $e5 = (1, 2, ...210)$ by concatenating $e2$ and $e3$.

- Read seq's help vignette: `?seq`

- Create the vector $e7$ containing 70 equally spaced values between 0 and 1.

- Create the vector $e7$ containing 10 times the sequence $e1$. Tip: use `rep`.

- What is the result of the operation $e2 - e3$? This is called *recycling*, and this is dangerous.

1.2 Character vectors

- Create a vector `vowels` containing all vowels

- `letters` is a character vector containing all the letters in alphabetical order.

- What does `letters %in% vowels` do ?

- Extract the number of each vowel. Tip: use `which`

- Extract the number of each non-vowel

- What are the letters right after a vowel ?

- Create the string `myname` containing your name (in lower case)

- Use `strsplit` to extract individual letters of your name. Observe that `strsplit` returns a list. Access its first element.

- Give for each character of your name its number in the alphabet.

- Do the same with your right neighbor's name.

- Who is on average lower in the alphabet ?

2 DataFrames

2.1 Cute animals.

- Create a dataframe that contains 3 columns: the alphabet letters, the number of each letter, and a binary variable voyel.

- Extract the lines of the dataframe corresponding to your name.

- Let's work on the msleep dataset. It contains information about the sleeping patterns of many animals:

- Examine the dataset: `head, str, names, summary`

- Sanity check: make sure that animals are either awake either asleep over the course of $24hrs$

- What is the animal that sleeps the most ? Tip: `which.max`

- How many animals of less than $100g$ and sleep more than half a day ?

- What is the average brain weight/body weigth ratio ? Tip: use the `na.omit` *function* or the `na.rm` *option*

- Who has the highest such ratio ?

2.2 Endangered animals: let us consider the case of Threatened or extinct animals

We'll assume that the ranking is the following: lc < domesticated < cd < nt < vu < en

- create a copy of `msleep` and reorder its factors according to the presented order. To create an ordered factor, use the `ordered=TRUE` option of the factor function

- use the ordered factors to compare the average weight to isolate threatened animals and compute their average weight

- compare this weight to the weight of the remaining animals.

- add a boolean column to the dataframe, `threatened`, that is true if the animal is threatened.

2.3 Functions

- Create a function returning taking a name as input, and returning the corresponding letter numbers.

- What happends if you give an empty "" name to this function ? Fix accordingly.

- What happends if you give to this function non-standard characters ? Fix the problem. Tip: choose one strategy: `regexp`, `NA`, `substr` and `tolower`

- Create a function that take an animal name as input, and returns its genus under the form "The Goat is a Capri" if it is in the msleep dataset, and "I don't know" otherwise.

- The `browser()` command allows debugging. When R encounters this call, it stops the execution of a function and provides the user with a different `Browse` prompt. Place a `browser` somewhere in your function, and run this function. Print some local variables. You can continue the execution with `n`, and quit the browser mode with `Q`.