Exercise sheet: Day 1

Vangelis Theodorakis, Fatemeh Behjati, Julien Gagneur, Marcel Schulz

09 April, 2021

Contents

1 Vectors

First, create three named numeric vectors of size 10, 11 and 12 respectively in the following manner:

1) One vector with the "colon" approach: *from:to*
2) One vector with the `seq()` function: *seq(from, to)*
3) And one vector with the `seq()` function and the `by` argument: *seq(from, to, by)*

For easier naming you can use the vector `letters` or `LETTERS` which contain the latin alphabet in small and capital, respectively. In order to select specific letters just use e.g. `letters[1:4]` to get the first four letters. Check their types. What is the outcome? Where do you think the difference comes from?

2 Factors

1) Create a character vector consisting of three annotations *Mutant-1, Mutant-2, Control*.
2) Using this annotation vector, create a factor where each annotation is repeated 4 times in a sequential manner (*Mutant-1, Mutant-2, Control, Mutant-1, Mutant-2, Control, ...*). In addition, the levels are the sorted annotation values.
3) Print the results.

3 Data tables

The purpose of this exercise is to get familiarized with data.table and try out some of its useful features.

3.1 Basic operations

Please follow the steps listed below:

1) load the library called *dslabs*
2) Access the database called *brexit_polls*. You can take a look at the the *help* documentation of this database (*?brexit_polls*) to learn about its content.

For example:

column name	Description
pollster	Pollster conducting the poll.
poll_type	Online or telephone poll.
samplesize	Sample size of poll.
remain	Proportion voting Remain.
leave	Proportion voting Leave.

3) Inspect this data by checking properties such as the class type, the number of rows and columns, its column names, the unique values in the *poll_type* column.
4) Create a new variable called *brexit_DT* and assign the data.table converted version of *brexit_polls*.

3.2 More exciting operations

Continue from the previous part and perform the following actions:

5) From *brexit_DT* get the counts of Online and Telephone polls
6) What are the mean and median values of the *samplesize*
7) Add a new column *remain_polls* to *brexit_DT* that holds the multiplication of *samplesize* to *remain*
8) What is the range of values in this newly created column?
9) How do the mean values of *undecided* look like when grouped by *pollster*? How do they look like when grouped by *poll_type*?
10) Remove the column *remain_polls* created in step 7.

4 Looping

1) Initialize a variable called *counter* by 0.
2) Using a for loop that iterates 10 times,
   - create a random number drawn from a uniform distribution with *min=0* and *max=5*.
   - whenever this random number is bigger than or equal to 1, increment *counter* by 1.
3) Print the final value in *counter*.

5 Functions

1) Write a function named *get_counts* that takes a GTEx data table as input and outputs the total counts of rows that the sample tissue type (*SMTS*) is *Heart* and the sample analysis freeze (*SMAFRZE*) is *RNASEQ*.
2) How about if you try the same but for *Blood*.
   - If this task was too easy, can you modify your function such that instead of taking only one argument, it takes two additional ones, one for the *SMTS* and another for *SMAFRZE*. Iterate over all possible values of *SMTS* (**Hint:** *unique(data$SMTS)*) and call your function by providing the sample tissue type.

6 R Markdown

Downloaded and stored the *sample_annotation.tsv* file from Google drive. Then, create an Rmarkdown file and perform the following tasks: 1) Read the *sample_annotation.tsv* file. 2) Create a new variable containing the counts of each *tissue* existing in the data. 3) Use the *barplot* function to plot the number of tissue types in the GTEx data. 4) Try to sort the bars according to the tissue counts (optional).