Introduction to R for social researchers

Alexandru Cernat

Practical 1

Initial set-up

Before we start let's do the prep work.

1. Open Rstudio
2. Set up a working directory so we have all our files in one place. Go to the menu on the top. Go to "Session" -> "Set Working Directory" -> "Choose Directory". Choose there a folder on your computer to store everything from the session.
3. Open a new script. In the menu go to "File" -> "New File" -> "R Script". **From now on write only in the script**. When you are happy with what you wrote you can select the code you want to run and press the "Run" button. Alternatively you can use the shortcut "ctrl + enter". Remember to save your script regularly. You can use this to reproduce your work and as a back up if something goes wrong. Also, remember to add comments so it's easier to understand what you did in the future.

Bellow you will have some tasks to do. Use the slides provided as inspiration for how to solve them. Sometimes I might show you some code that you should also copy and run on your computer before progressing. **If you get stuck ask for help!**

Creating objects

We will start with some basic tasks of creating objects in `R`.

Create a numeric vector called "num_vct" that includes the numbers: 5, 10, 150, 8. Also, print to see how it looks like. Also, divide the vector by 5 and see what you get.

Create a logical vector called "log_vct" that includes values: TRUE, FALSE, F, F and T. Print it. Calculate the mean of the vector. Why do you think you get that result for the mean?

Create a string vector called "str_vct" that includes: "I am", "loving", "R".

Run the code bellow to apply the function `paste()` to it. What do you think this function does?

```
paste(str_vct, collapse = " ")
```

Use the `typeof()` command on the three objects you created above. Do they give you the output you expect?

Create a list called "list1" with the three objects you created (num_vct, log_vct, str_vct).

Give the names "a", "b", "c" and "d" to the elements of the numeric vector ("num_vct").

Create a factor with the values 1, 0, 0, 0, 1 where 0 represents "Did not vote" and 1 represents "Voted". Call it "fct_var" and do a table of it.

Do a matrix with all the numbers between 1 and 12 that has 4 rows and save it as "matrix1". Print it.

Try to do the matrix again but this time use the option "byrow = T". Save the result as "matrix2" and print it. What do you think is the effect of the option?

Create a dataset called "df" with three variables:

- age: 14, 80, 35, 65
- sex: "m", "f", "f", "m"
- vote: FALSE, TRUE, TRUE, FALSE

Print the result.

Add a new row to the dataset with an individual that is 21, male, and did not vote. Save the new dataset as "df2". Print the result.

Add a column to "df2" called "work" that takes values: FALSE, FALSE, TRUE, TRUE, TRUE. Save the data as "df3" and print it.

Selecting elements

Now that we created different types of objects let's practice selection.

Select the second element of "str_vct".

Select the third element of "log_vct".

Select the first and third element of "num_vct" (in one command).

Select the second element of "num_vct" using its name ("b").

For the "matrix1" object you created before try to (separately and without saving the result):

- select the first row
- select the second and third column
- select the element in the second row and third column
- exclude rows 1 and 3

For the "df3" object you created before try to (separately and without saving the result):

- select the "vote" variable. Try doing it once using the name and once using the column number
- exclude the second and third rows as well as the forth variable (in one command)
- select the rows with age below 65

Practical 2

Let's practice a few more things that you have learned.

Try to order the "df3" data you created in the first practical by age. Try to do it both going from the smallest age to the highest one and the other way around.

Next, try to remove the "sex" variable from the "df2" dataset.

Using "df3" replace the values on the work variable with FALSE if age is larger than 64.

Describe data

You are next going to apply what you have learned to some real data. We will start out by using the "swiss" data that is in the memory of `R` by default. This has some statistics on Swiss regions at about 1888. You can print it like any object. You can also find out more about it using the `?swiss`:

```
# print the data
swiss

# find out more about the data
?swiss
```

Use the commands you have learned before to explore the "swiss" data (dim, head, tail, View, summary). *The data already exists in the memory of `R` so you don't need to import it.*

Calculate the mean and variation of the "Fertility" and "Infant.Mortality" variables.

Modify variables

Next we are going to create some new variables by recording the existing ones. First, let's look at the "Catholic" variable using a histogram:

```
hist(swiss$Catholic)
```

Imagine you wanted to calculate the percentage of non-catholics in each region. Make such a variable and add it to the dataset as "non_catholic". *Tip: you can do 100 - the "Catholic" variable.*

To check the recode we can make a plot of the two variables:

```
plot(swiss$Catholic, swiss$non_catholic)
```

How do you interpret the plot? Do you think the recoding worked as expected?

Imagine that we want to calculate proportions instead of percentages. Make a new variable, "non_catholic_prop", by dividing the previous variable by 100. Then plot "non_catholic" and "non_catholic_prop" (like we did above) to check the result.

Do a histogram of the "Education" variable.

Imagine we want to dichotomize the education variable. We want to make a variable with two categories: "1" if "Education" is bellow 12 and "2" if it is above 12. Use the `ifelse()` command to make such a variable and call it "ed_level".

Do a table of the new variable as well as a cross-table between the original variable and the new one.

To make the new variable easier to understand let's make it a factor. Create a new factor variable "ed_level_fct" which takes "ed_level" and adds the labels "Low" and "High".

Do a cross-table between "ed_level" and "ed_level_fct". Was the coding correct?

Bonus track

If there is time you can play with the European Social Survey. Download the "ess9_raw.csv" in your working directory. Then, use the command you learned in the presentation to import it.

Make a smaller dataset in which you only keep the following variables: "idno", "cntry", "gndr", "eduyrs", "ppltrst", "vote", "happy".

Explore the data using the commands you learned above (head, tail, dim, view and summary).

Recode the sex and vote variables like we did in the slides and try to find out if there are differences between man and women in the voting likelihood.

Do a histogram of the "ppltrst" (How much you trust people, larger value = more trust).

Calculate the average trust in the entire data. Then calculate average trust in Great Britain ("GB" in the "cntry" variable) and in Russia ("RS" in the "cntry" variable). Are there differences between countries in the amount of trust in other people?

Do the same for happiness ("happy" variable).

Save the final dataset you used as a csv file called "ess9_small.csv".