Modern workflows in data science

Assignment 2

Alexandru Cernat

In this assignment we will be playing with Rmarkdown. We will start with some basic stuff and then do a batch report. By the end of the assignment you'll do 35 reports!

We will use real world data again. This time you will use the European Value Study (EVS) from 2017 and look at attitudes towards gender roles and immigration.

Tasks

1. **Get EVS data**. The data you want is ZA7500. You can find it here: https://search.gesis.org/research_data/ZA7500 (**10%**)
2. **Write an overall report as a pdf**.

The two variables of interest are v72 and v80 (I'll let you discover what exactly they measure). Other variables you should keep are age, education, sex and country. For the overall report I would like to have a short intro about the research and data (one paragraph), a section with descriptives that includes max two tables (e.g., one for continuous variables and one for categorical variables), a section that includes two graphs: how the two variables of interest change with age, and a section with two regression models explaining the two dependent variables. The model should include: age, age squared, sex and education. The results from both models should be presented in a single table. Add a sentence or two interpreting each table and graph.

Grading of this section:

- clean data for analysis (**20%**)
- descriptive tables, graphs and regression tables (**20%**)
- two version of the report, one showing no R code (for policy makers) and one that also shows your code (for statisticians) (**10%**)

2. **Country level automated report in html**.

For the final task you will create a report for each country in the EVS (33 reports in total). We want the same tables and graphs as in the previous report but now it should be only based on the data from the country the report is on (i.e., the report on Estonia should only be based on the Estonian data). You can have a general intro that would apply to all the reports (e.g., the data that you use and your research question). You don't have to interpret the rest of the tables and graphs (the descriptives and change with age) but now you have to create dynamic interpretation of the regression model. I want you to focus on the sex variable in the two models and write a paragraph that says what's the effect on the dependent. It should say if it's a positive or negative relationship, what is the coefficient and if the relationship is statistically significant.

Grading for this section

- the 33 reports and the syntax that produces them (**20%**)
- dynamic (i.e., changes depending on the data) interpretation of results (**10%**)

3. **Presentation** of reports (e.g., labels and captions), Github repo structure and commits (**10%**)

As in the previous exercise you will now create a new github repo and work on this assignment. You can save your code and the reports in github and me and the TA will check them out after you invite us as collaborators. Again, don't share the data.

Top tips

- organize the repo in a similar way to what you did before or how I did it in my example solution
- it's up to you how you organize things but I still used a "master" `R` script for data cleaning and for running the batch report. Additionally, I had three R markdown documents: two for the second task and one that is the basis for the dynamic reports (task 3)
- I treated the two dependent variables as continuous (although they have a small number of categories). So for the practical you can use OLS and calculate means and standard deviations
- there are lots of education variables. I used one that had just three categories: low, medium and high education
- for the regression table you can have a look at the `texreg` package (although there are a few different ways to do nice tables as covered in the videos)
- for the dynamic reports you will use html. The nice thing about that is that you can include interactive components. If you feel adventurous transform your two graphs into interactive graphs using the `ggplotly()` command from the `plotly` package. *Being able to extend ggplot in such a way is one of the things that makes it prefferable to other packages.*
- the working directory for R markdown is by default the same folder where the .Rmd file is. If you want to move up a folder (e.g. go to the repo level if the Rmd is in a subfolder) you can use `"../"` in a path. For example, `"../data/data.csv"` goes up a folder and then in the data subfolder.