W02-1: CSV I/O and some simple modifications

This worksheet welcomes you to build your first R script. After completing this worksheet you should have gained some experience in accessing and modifying selected features of data frames and feel confident to read and write standard CSV files.

Things you need for this worksheet

- R — the interpreter can be installed on any operation system. For Linux, you should use the r-cran packages supplied for your Linux distribution. If you use Ubuntu, this is one of many starting points. If you use windows, you could install R from the official CRAN web page.
- R Studio — we recommend to use R Studio for (interactive) programming with R. You can download R Studio from the official web page.
- your data sets from W01-1: Getting and organizing data

Your first R script?

The R script which should be compiled in the content of this worksheet might be your first one. Don't worry. No matter if you use a simple text or R editor or the more advance R Studio, everything is about problem-oriented working, deconstructing the task into simple to handle pieces and keeping the focus on what you actually want. If you get that straight, then it is quite easy to find solutions for your problems. Since this is the first time, we won't leave you in the dark but help you in staying focused.

Learning log assignments

As always, please add these entries to your today's learning log at teachwiki:

- Favorite aspect of the session (if any)
- Superfluous aspect of the session (if any)
- Eureka effect (if any)
- Links to what I've learned so far (if any)
- Questions (if any)

For more information see this short howto.

As today's special, please complete the following assignment:

In W01-1: Getting and organizing data you have downloaded a chronology of natural disasters from the EM-DAT data base. This is your data set for this worksheet.

Please write an R script for the following tasks and upload it to your learning log.

As promised, the problem of today assignment is not one huge task but it comes already deconstructed in something like a sorted list of code blocks which only have to be implemented in the specified order:

1. Read the natural disasters chronology CSV file into a data frame
2. Check if reading has been successful by printing the first two rows to the standard output (standard output means: the R console)
3. To check if the *total_affected* column of the data set is actually the sum of columns *affected*, *injured* and *homeless*, add a new column called *control_affected* and use it to store the results of an arithmetic operation which is appropriate for this check (keep it simple, this is just something out of “+”, “-”, “*”, “/”)
4. rename the column *affected* to *other_affected* (see tip below)
5. write the modified data frame to a CSV file with the values sorted by continent and country.

 Please upload the first 5 lines of your output data set to your learning log.

For this task, it does not matter, if you have implemented your R script in such a way that only the first 5 lines are written to the output file in step 5 above or if you just open it afterwards and only copy the first 5 lines into your log.

Step 4 might be the most difficult sub-task but if you do not want to try a solution with e.g. the `colnames` function you might be happy to hear that the position of the *control_affected* column does not matter as long as the *affected* column does no longer appear after finishing step 4.