Lab 1: Data handling, visualization and simple linear regression
FMSF90F: Statistical learning and visualization, spring 2023

Global temperatures and green-house gases

Introduction

To get started with data handling and visualisation of data in R you will in this project illustrate the change in both greenhouse gas emissions and global average temperatures during the last century.

Preparations

Before the lab starts prepare by doing the following Sections in advance:

- 1 Organize your files
- 2 Load the data
- 3 Transform the data (1)

Data

Provided on the course web page:

Filename	Description
`Temp_BE.csv`	Global average temperature from Berkeley Earth[1]
`GHG_1880_2014.csv`	Greenhouse gas emissions from PRIMAP [2]
`GHGunits.csv`	Units for the GHG_1880_2014.csv data set

1 Organize your files

For good reproducibility of your analysis you must organize your files in a good manner. For this course you should set up a course folder with subfolders for data, projects etc.

(a). Create a course folder at a good position in your file system, eg. (userXX/courses/StatLearning). If you haven't done so already.)

(b). Create a subfolder for Data (userXX/courses/StatLearning/Data).

(c). Open Rstudio and create a R-project (Button to the right) "in an existing directory" (select the course folder eg userXX/courses/StatLearning). (Those of you that are familiar with Git can of course use Git for version control, but that will not be part of the course.)

[1] https://berkeleyearth.org/data/ Subsection "Land + Ocean (1850 – Recent): annual summary"

[2] https://dataservices.gfz-potsdam.de/pik/showshort.php?id=escidoc:4736895
> Download data > PRIMAP-hist_v2.1.zip

(d). Use command `getwd()` to see what is your working directory. Make sure it is the course folder.)

(e). Create a new Rmd (or Quarto if you prefer that) file in File > New file > R Markdown (Or "Quarto document"). Select a good Title and save the file as "Lab1".

(f). R studio has now created an autogenerated template file called "Lab1.Rmd" that you can test by pressing the Knitr-button (yarn + sticks).

(g). Make sure the output is what you expected.

(h). Add code in the ”setup” chunk to load the libraries you need, eg. `tidyverse` that is required for read_csv, ggplot and data transformation with dplyr, see Figure 1. You typically add more libraries here as you work along and feel the need.

You should in the following remove all pre-made parts of the template that are not interesting to you, and add the text and "code-chunks" you need for your analysis/report. To create more "chunks" use CTRL+ALT+I or use the ”+C” icon above the script. Remember to save and run the file often.

(You might want to change some settings for output etc., but you can wait until you feel the demand.)

2 Load the data

Next step is to load the data:

(a). First save the data from Canvas to the "Data"-folder you created in your course folder.

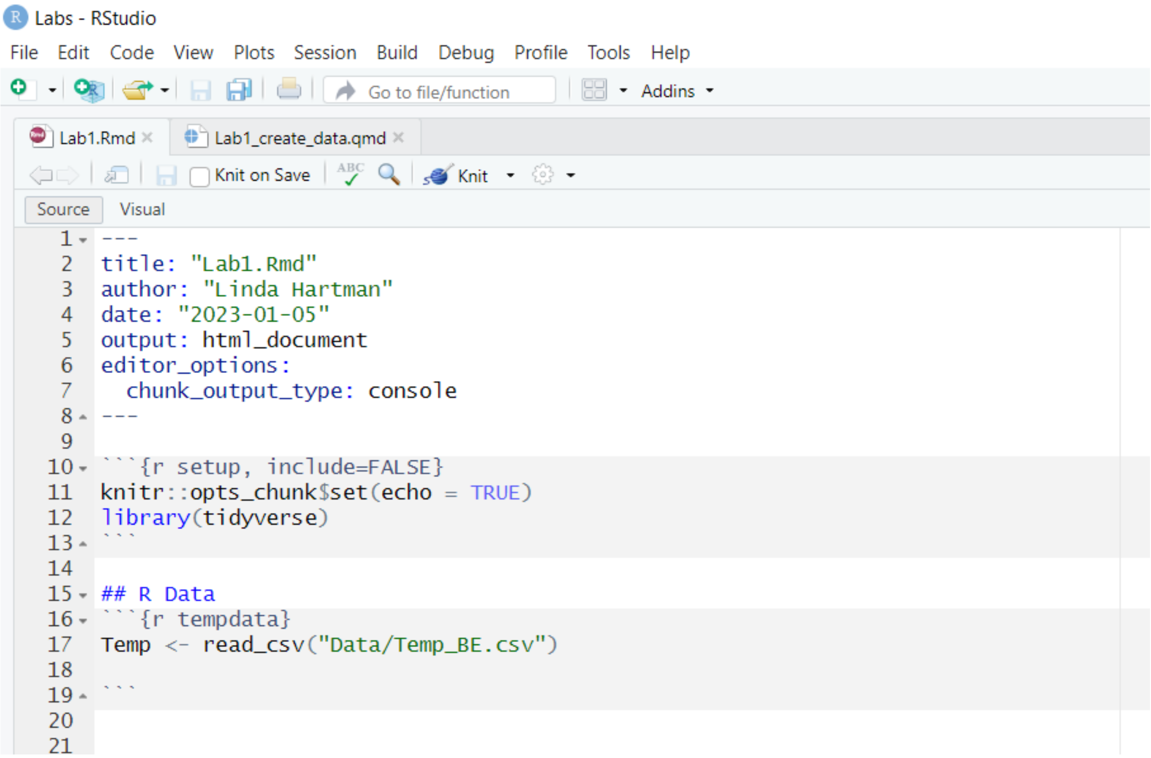
(b). Now import the data files "Temp_BE.csv" & "GHG_1880_2014.csv" in R (as in the Lecture demos). One easy way to import new data is to use File > Import dataset > From text (readr). Use the menues, and when you are satisfied with the result copy the code in the "Code Preview" and paste that to your Rmd-file in a new chunk, see Figure 1, chunk ”tempdata”.

Voluntary challenge:
In the course folder the data was already transformed to data frames. If you have time (and are experienced in R), you can instead download the data directly from the sources (see first page) and work a bit more to import and manipulate it to R.

3 Transform the data (1)

Next step is to prepare the data for the analysis.

(a). The Berkeley Earth data (Temp_BE.csv) gives the annual anomalies from global mean temperature for the period Jan 1951-Dec 1980. For that refernce period the same researchers have estimated the global mean temperature to be 14.105 $^\circ C$ with 95% CI $(14.080^\circ C,\ 14.130^\circ C)$. Make a new column in Temp_BE with absolute estimates of global temperatures, calculated from the 1 year anomalies .



~~~
---
title: "Lab1.Rmd"
author: "Linda Hartman"
date: "2023-01-05"
output: html_document
editor_options:
  chunk_output_type: console
---

```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
library(tidyverse)
```

## R Data
```{r tempdata}
Temp <- read_csv("Data/Temp_BE.csv")

```


~~~

Figure 1: Add ”library” loadings in a setup chunk, and data loading and preparation on the top of the Rmd-report (before the code for any summaries, plotting or analysis). Remember to save the ”import” code to your Rmd-document if you used the Import wizard of Rstudio. (Here done in chunk `tempdata`).

4 Plot your data (1)

(a). Use (your modified) Temp_BE and make a scatterplot of global mean temperature vs year. (Hint: use `geom_point` from ggplot2.). Try different sizes for the markers and select the one you prefer.

(b). try different ”themes” by adding eg. `+ theme_bw()` or `+ theme_void()` to the code where you create the ggplot figure.

(c). Add a smoother to visualize the general trend (Hint: `geom_smooth`). Read the documentation and make sure you understand what type of smoother was used.

(d). *Extra: change the visibility of the smoother. (Hint: set `alpha=0.2` in the call to `geom_smooth`.)*

Make sure you have title + labels and titles (incl units when relevant) on all axes + legend to colors, and adequate sizes of markers, lines and text.

5 Transform the data (2)

Next step in preparing the data for the analysis is to select the relevant data for Greenhouse gases (GHG) and to merge the GHG data to the temperature data (a). In the GHG-dataset

- Select only variables `year` and `KYOTOGHG`.
- Change the variable name from `KYOTOGHG` to `GHG`.
- Merge the Global temperature data with the modified GHG-data using `year` as the identifying variable. (Hint: `left_join(...,by="year")`)
- Calculate accumulated GHG-emissions per year (after 1880 as that is the first year with data for GHG emissions).

  - `mutate(GHG_cum=cumsum(GHG))` is a natural first guess, but does not fulfil the needs in this case. Why? What happens?
  - `mutate(GHG_cum=cumsum(replace_na(GHG, 0))` will do the trick.
    Make sure you understand the mechanism and why it is relevant here. Is the result correct for all years in the data?

Note: It is regarded good practice to separate the data preparation from the analysis. For large projects it is adviced to do data preparation in a separate script/report, and to save the prepared analysis dataset on disk, ready to be imported at the beginning of all analysis scripts/reports. For small projects as in this course the data preparation can be made within the same R-script, but it is adviced to do all data preparation in the beginning of the script = before any analysis starts, to have full control over what data is analysed. Thus, be sure to put the code you just created in a chunk above the plot you made of temp vs year.

6 Plot your data (2)

(a). Make a scatter plot of GHG-emission vs year

(b). Make scatter plots of temperature vs yearly GHG-emissions and temperature vs accumulated GHG emissions for years 1880 - 2014. Do any of these indicate that a linear relation is a reasonable model?

(c). *Voluntary challenge: let the color of the marker indicate what year the measurements were taken*

Again make sure you have titles (incl units when relevant) and labels on all axes + legend to colors, and adequate sizes of markers, lines and text.

7 Simple linear regression

(a). Use your conclusion from the previous section to fit a linear regression model to `GHGcum` after 1930.

(b). Check the model fit by plotting and interpreting

  (a) residuals vs predicted values from the model (see lecture notes)

  (b) *If you have time:* investigate leverage and studentized residuals

(c). Interpret the model. Look at the parameter estimates with confidence intervals and judge if you can verify that temperature increases with increasing GHG-emissions?

(d). For each data-point save the upper and lower limit of the 95%-prediction interval (Hint: use `predict` with a wise setting for `interval`.) Interpret these limits in words, and be clear to your self about the difference between prediction intervals and confidence intervals.

(e). Plot the data again, this time with prediction bands added to the plot (see lecture demo for code).