Description, implementation and validation of a user interface for complex datasets in the social sciences

Ludwig-Maximilians-Universität München

Fakultät für Mathematik, Informatik und Statistik

Bachelor of Statistics

Bachelor's Thesis

30. Juni 2018

Author
Samuel Lippl

Supervisor
Dr. Fabian Scheipl

Selbständigkeitserklärung

Ich erkläre hiermit, dass ich die vorliegende Arbeit selbständig angefertigt, alle Zitate als solche kenntlich gemacht sowie alle benutzten Quellen und Hilfsmittel angegeben habe.

Samuel Lippl, 15.09.2018

Abstract

This report will provide an overview over outlier detection in R. It starts by discussing some general principles of outlier detection. Linear methods and their nonlinear extensions are presented next. Along the general methods, examples of application and an appropriate methodology in R are introduced. The report will be concluded by a discussion of method evaluation as well as a comparison of the different introduced algorithms.

Contents

1 | Introduction

One of the main advantages of the statistical programming language R (**?**) lies in its conciseness. With just one line of code, it is possible to build a linear model, visualize a variable's distribution or conduct complex modifications on several datasets. This is possible because R is a domain-specific language and is therefore able to make strong assumptions – many people will need to build a linear model or read in a csv file and it is therefore sensible to create custom functions for these purpose.

An important property of this conciseness is that the code is still easy to read. Two features that are especially important for this both rely on the specific domain of statistical analysis for which R was created:

- Specialized functions: `read.csv` essentially calls `read.table` with a few modified parameters. Nonetheless, it is immediately clear what this line of code is supposed to achieve.
- Default values: The user does not need to specify every single parameter of a function. For instance, it is helpful that `read.table` contains the parameter `na.strings` that allows the user to specify values that encode `NA`s. However, in most cases, `NA`s are encoded by the string `NA` or a missing value ^[1]. By setting default values, the user only needs to think about this parameter when the file structure is out of the ordinary.

^[1:] The latter is only implemented in `read_delim` from the package `readr` but the advantage of default values remains valid nevertheless.

These advantages are certainly not unique to R. They are designed to minimize the expected time a user needs to spend with coding his decisions while maintaining easy reproducibility of his work. On the other hand, if there are more complicated tasks to undertake, a consistent interface allows the user to do that, as well.

A good example for this concept, in my mind, is the package `stringr` (**?**). Functions like `str_trim` (trim whitespace) or `str_to_title` (capitalize) make special use cases easily accessible. On the other hand, `str_replace` allows more complicated operations with regular expressions using the same consistent interface.

Things, however, start to fall apart when one attempts to modify default values. This is possible by setting the global options in R; however, relying on these makes reproducibility harder. On the other hand, one could write new functions to solve this problem. This is, however, more laborious than such an endeavour needs to be.

A good example of this are datasets with many variables as they occur in the social sciences. As an example, I will consider the Varieties of Democracy v-dem.net dataset which produces indicators of democracy (**?**, **?**). It contains many variables on different aspects of democracy with values per country and year. If one wishes to visualize the development of this variable over time, a simple line plot often makes sense. Consider, for instance, the variable which characterizes the freedom of religion on a scale between 0 and 4 for Germany:

```
df_vdem %>%
  filter(country_name == "Germany") %>%
  ggplot(aes(year, v2clacfree_osp)) +
  geom_line()
```

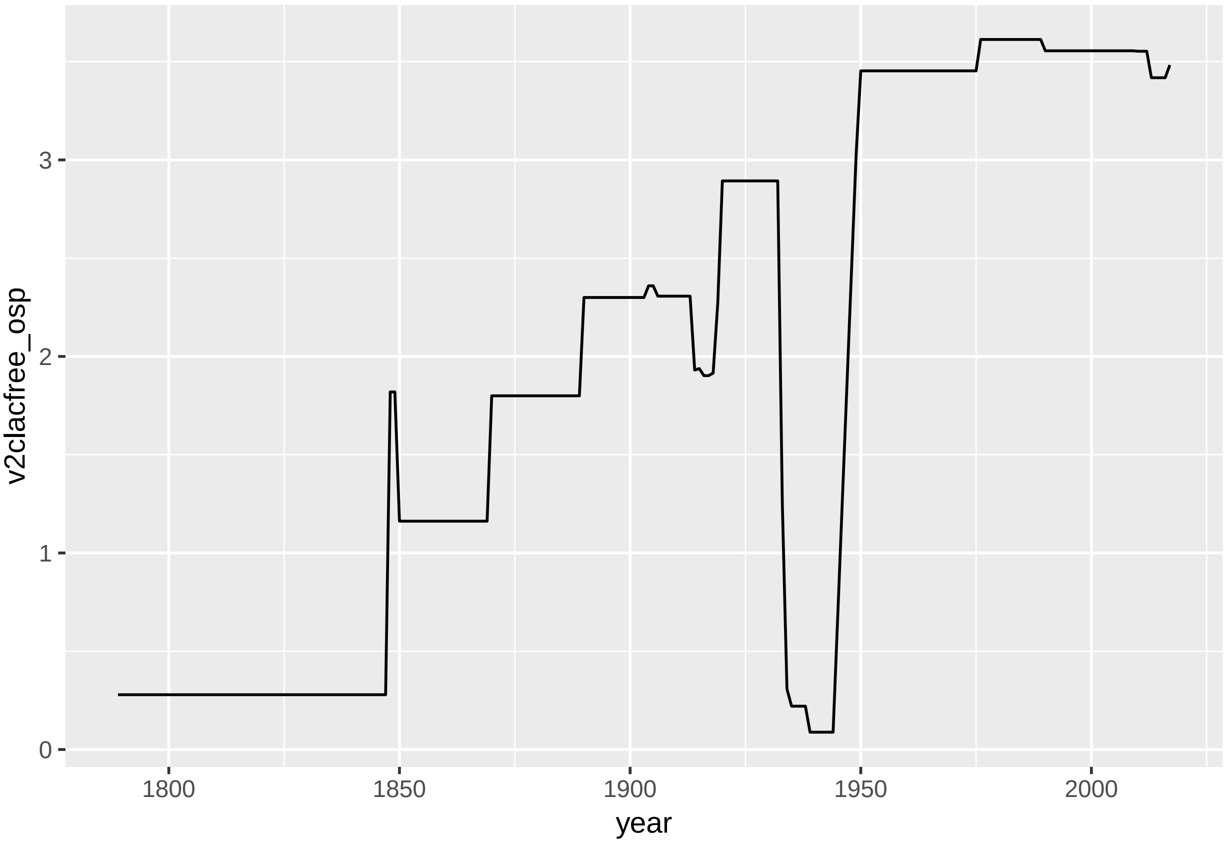

Figure 1.1: Example: Freedom of religion in Germany over time

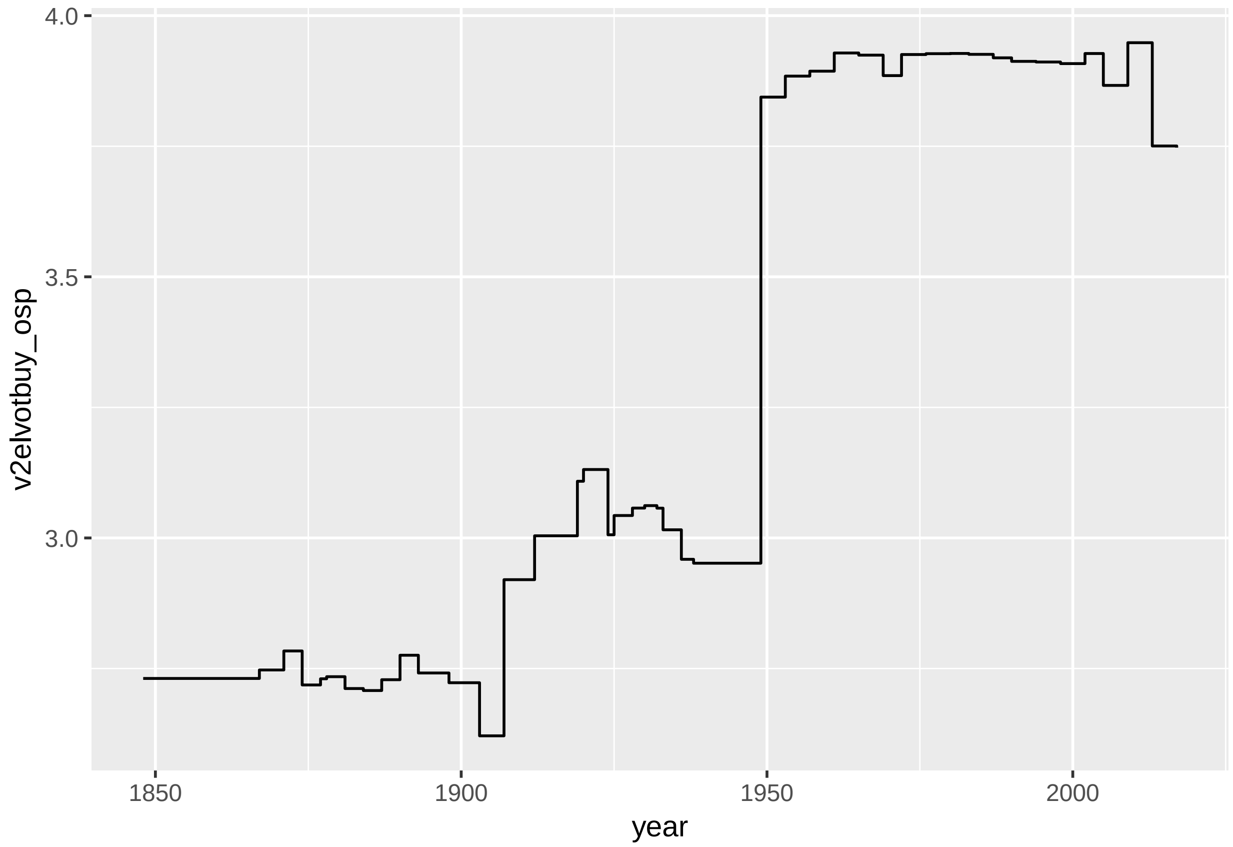

Figure 1.2: Example: Election vote buying in Germany over time

Although there are considerable changes within a single year, freedom of religion is a continuous value and linear interpolation of this development within a year makes sense.

Considering the variable `v2elvotbuy_osp`, however, a line plot makes less sense. This variable captures whether there was evidence of vote buying during a national election and is therefore only present in years where there has been a national election. A step plot seems more sensible in this case as the represented value would always refer to the last election.

```
df_vdem %>%
  filter(country_name == "Germany") %>%
  select(year, v2elvotbuy_osp) %>%
  na.omit() %>%
  ggplot(aes(year, v2elvotbuy_osp)) + geom_step()
```

Furthermore, the scale titles should be modified to show an interpretable variable name, the scale should in many be standardized to depict the entire range between 0 and 4.

In summary, there are many considerations one needs to implement in such a visualization. Therefore, every time the statistician needs to implement such a visualization, she needs to think about these questions again, which is time expensive and makes interactive user interfaces impossible. This problem is not limited to visualization; another example would be descriptive tables of a linear model or a report summarizing all covariates that have been used.

In summary, R provides amazing opportunities to to outsource everyday thought processes in data analysis. However, adapting these mechanisms for application-specific thought processes is expensive and difficult. A broad framework for such an adaptation would enable researchers to think about certain decisions (like the visualization of a specific variable) once and then be done with it. Both the researcher himself and his colleagues who might not need to think about this at all would benefit from this.

In this Bachelor's Thesis, I describe such a framework, implement it as the package `tectr` in R and apply it to the V-DEM dataset. The #methods discusses some details regarding the package construction and the dataset before we get a #example in the third chapter. The #concept will present the framework and the implementation in a more specific way. #application presents the application of `tectr` to the V-DEM dataset and the #summary summarizes the thesis and discusses the next steps regarding `tectr`.

2 Methodology

This chapter introduces the V-DEM dataset and discuss the methodological background of `tectr`'s construction.

2.1 V-DEM

2.1.1 Introduction to the database

The Varieties of Democracy Institute is concerned with measuring different aspects of democracy. It distinguishes between seven high-level principles: electoral, liberal, participatory, deliberative, egalitarian, majoritarian and consensual. These are measured by a variable in the interval $[0, 1]$ and consist of several mid- and low-level indices. The low-level indices are coded with the help of several country experts. These receive a questionnaire. Most questions can be answered by an ordinal scale of five alternatives. Consider, as an example, the variable "Disclosure of campaign donations":

> Question: Are there disclosure requirements for donations to national election campaigns? 0: No. There are no disclosure requirements. 1: Not really. There are some, possibly partial, disclosure requirements in place but they are not observed or enforced most of the time. 2: Ambiguous. There are disclosure requirements in place, but it is unclear to what extent they are observed or enforced. 3: Mostly. The disclosure requirements may not be fully comprehensive (some donations not covered), but most existing arrangements are observed and enforced. 4: Yes. There are comprehensive requirements and they are observed and enforced almost all the time.

The answers are then analyzed for inter-coder reliability and a standardized average of the responses together with a confidence interval which contains 68 % of the probability mass is created. Lower-level indices are created from these answers which are summarized in mid-level and then high-level indices. An overview over the structure can be found in appendix D of the codebook (**?**). The database contains data on 201 countries between 1789 and 2017. (**?**, **?**)

2.1.2 `vdem.tectr`

I have created the package `vdem.tectr` which contains the country-year dataset. It can be downloaded via github.com/sflippl/vdem.tectr:

```r
# install.packages("devtools")
devtools::install_github("sflippl/vdem.tectr")
```

The package contains three datasets:

- `df_vdem`: This dataset contains all variables from the varieties of democracy dataset where interval variables are numeric and categorical variables are saved as factors or ordered factors where appropriate.

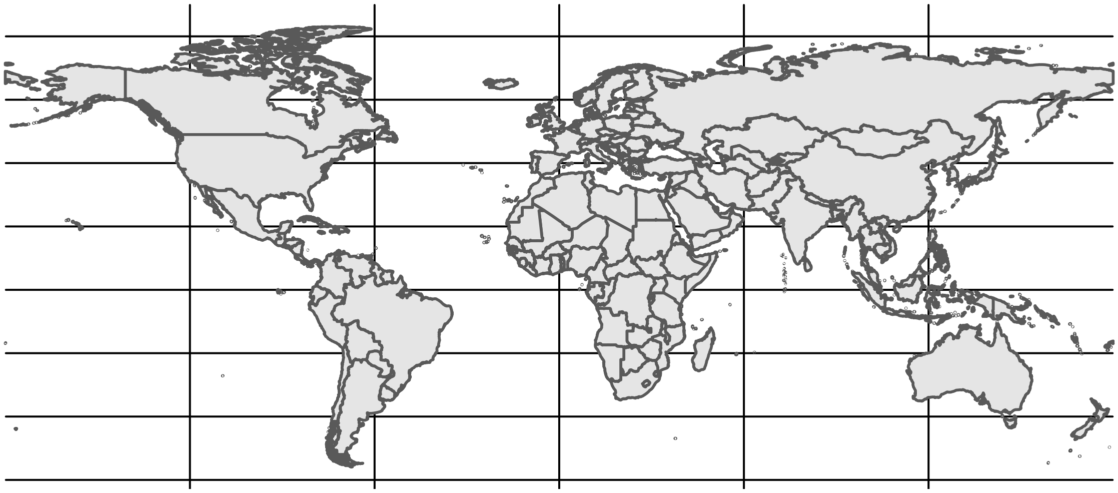

Figure 2.1: (#fig:vdem_spatial)Country borders in 2017 in the V-Dem database

- `vdem_spatial`: This simple features object (**?**) contains the polygon shapes of the different countries for every year between 1945 and 2017. I have used the CShapes dataset (**?**, **?**), sovereignty- and state-level maps data from www.naturalearthdata.com and the details from the document on country coding units from V-Dem (**?**). Note that the coded country borders by V-Dem do not constitute any endorsement of controversial entities such as Zanzibar.

```
vdem_spatial %>%
  filter(end_year == 2017) %>%
  ggplot() +
  geom_sf() +
  theme_map()
```

- `vdem` which contains the variables from `df_vdem`, the country shapes from `vdem_spatial` and further metainformation (see below)

Details on these datasets and the reproducible code can be found in the folder “data-raw” in the package.

2.2 Package construction

The package has been constructed with the packages `devtools` (**?**), `roxygen2` (**?**) and `testthat` (**?**).

3 | Effective explicitness

We describe our methods in this chapter.

4 | Applications

Some significant applications are demonstrated in this chapter.

4.1 Example one

4.2 Example two

5 | Final Words

We have finished a nice book.

Bibliography

Coppedge, M., Gerring, J., Knutsen, C. H., Lindberg, S. I., Skaaning, S.-E., Teorell, J., Ciobanu, V., and Olin, M. (2018a). V-Dem Country Coding Units v8. Technical report, Variaties of Democracy (V-Dem) Project.

Coppedge, M., Gerring, J., Knutsen, Carl Henrik Lindberg, S. I., Skaaning, S.-E., Teorell, J., Altman, D., Bernhard, M., Fish, S. M., Cornell, A., Dahlum, S., Gjerløw, H., Glynn, A., Hicken, A., Krusell, J., Lührmann, A., Marquardt, K. L., McMann, K., Mechkova, V., Medzihorsky, J., Olin, M., Paxton, P., Pemstein, D., Pernes, J., von Römer, J., Seim, B., Sigman, R., Staton, J., Stepanova, N., Sundström, A., Tzelgov, E., Wang, Y.-t., Wig, T., Wilson, S., and Ziblatt, D. (2018b). V-Dem Codebook v8.

Oppedge, M., Gerring, J., Knutsen, Carl Henrik Lindberg, S. I., Skaaning, S.-E., Teorell, J., Altman, D., Bernhard, M., Fish, S. M., Cornell, A., Dahlum, S., Gjerløw, H., Glynn, A., Hicken, A., Krusell, J., Lührmann, A., Marquardt, K. L., McMann, K., Mechkova, V., Medzihorsky, J., Olin, M., Paxton, P., Pemstein, D., Pernes, J., von Römer, J., Seim, B., Sigman, R., Staton, J., Stepanova, N., Sundström, A., Tzelgov, E., Wang, Y.-t., Wig, T., Wilson, S., and Ziblatt, D. (2018). V-Dem Country-Year Dataset v8.

Pebesma, E. (2018). sf: Simple Features for R.

Pemstein, D., Marquardt, K. L., Tzelgov, E., Wang, Y.-t., Krusell, J., and Miri, F. (2018). The V-Dem Measurement Model: Latent Variable Analysis for Cross-National and Cross-Temporal Expert-Coded Data.

R Core Team. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria.

Weidmann, N. B. and Gleditsch, K. S. (2010). Mapping and Measuring Country Shapes. The R Journal, 2(1):18–24.

Weidmann, N. B., Kuse, D., and Gleditsch, K. S. (2010). The geography of the international system: The CShapes dataset. International Interactions, 36(1):86–106.

Wickham, H. (2011). testthat: Get Started with Testing. The R Journal, 3:5–10.

Wickham, H. (2018). stringr: Simple, Consistent Wrappers for Common String Operations.

Wickham, H., Danenberg, P., and Eugster, M. (2018a). roxygen2: In-Line Documentation for R.

Wickham, H., Hester, J., and Chang, W. (2018b). devtools: Tools to Make Developing R Packages Easier.