A Shiny app for bivariate Choropleths

1 Introduction

This app was created using shiny (Chang, Cheng, Allaire, Xie, & McPherson, 2018) with R version 3.5.2 (2018-12-20). Its purpose is to create bivariate choropleths. Creating bivariate choreopleths can be a bit tricky and somewhat time-consuming, especially for newbies in R and similar programs. Therefore, an app might be a convenient way to simplify the creation of bivariate choreopleths.

The app was not specifically created for psychological purposes. However, data visualization and exploration are gaining importance for psychological research. The current app can be a helpful tool for exploring potential relationships between two variables across different geographical areas or may be used to visualize the findings of such. The app can be applied to any topic, including psychological questions, depending on the data. For example, a health psychologist may be interested in one of the many eurostat datasets concerned with health (e.g. self-perceived health).

At this point, it has to be noted, that data exploration tools should never be used for fishing, which can severely bias results. Furthermore, data visualization can not in any way replace statistical testing of previously defined hypotheses.

2 Bivariate Choropleths

The bivariate choropleths created in the app are being created following the instructions by Art Steinmetz (2017). The variables compared in the plot are being normalized on the population sizes. This is important because otherwise, population size differences of the geographic areas would be reflected in the choropleth (Joshua Stevens, 2015). Both variables are categorized into three groups. Subsequently, a single variable combining the classified variables is created. This variable contains nine (3*3) groups, each corresponding to a single color. The geographical shapes, corresponding to the geographic areas in the dataset, are then colored based on this variable. All bivariate choropleths are produced using the package tmap (Tennekes, 2018) and plotted on a world map using leaflet (Cheng, Karambelkar, & Xie, 2018).

3 Own dataset

The user can choose to upload a dataset. However, so far only excel datasets (.xls or .xlsx) can be uploaded. Moreover, the data need to be cleaned beforehand. The dataset should be given in wide format.

After uploading a dataset, a table is rendered displaying the data. This may help the user to select different variables in the sidepanel. The user needs to select two variables that are of interest and a variable with the

geographic areas (either German federal states or European countries). Optionally, a variable containing some kind of time information (e.g. years) can be specified (optional), in which case any level of the variable can be selected. Furthermore, the user needs to select the map that will be used to plot the data. So far, only maps of European countries and German federal states are available. Maps can not be uploaded by the user. The levels of the dataset variable containing the geographic area needs to correspond to the map.

If a time variable and any level of the variable are selected, the dataset will be filtered based on the level of this variable (e.g. only data from the year 2015 will be used going forward). The variables that are compared in the bivariate choropleth will be normalized on the variable containing the population sizes for each area.

After the "Create plot" button has been clicked, the bivariate choreopleth and a legend will be rendered. A mapshot of the bivariate choreopleth can be downloaded.

4 Eurostat data

Instead of uploading an own dataset, the user can choose to use any eurostat dataset. The app allows to search for eurostat datasets by entering a search term into a search bar. Once the user has found a dataset of interest, he or she can enter the code of the dataset into a second text box. The dataset will then be automatically downloaded. Access to eurostat data is obtained via the eurostat package (Lahti, Huovari, Kainu, & Biecek, 2017).

After a dataset has been downloaded, a table displaying the data will be rendered. First of all, the user has to select the variables which are of interest (either two different variables or the same variable twice).

In the next step, a table with column filters is rendered. The user needs to filter all variables which are not blocked. The user should select only one level per column and a year between 1990 and 2016. If these conditions are not met, the filters can not be applied. The columns containing the previously selected variables as well as the variables with the geograpic areas and values are blocked and can not be filtered.

Based on the applied filters, the eurostat data are then subsetted and reshaped from long to wide format. Subsequently, the levels of the previously selected variables of interest are shown and need to be selected.

After the "Create plot" button has been clicked, a variable with the population density will be added to the data on which the selected levels will be normalized. Currently, the population density is retrieved from the eurostat dataset `"demo_r_d3dens"`. However, population densities of different European countries are only available for years between 1990 and 2016, which is a limitation. Also, it has to be noted, that all rows from the dataset for which the geographical area does correspond to any area given in the `"demo_r_d3dens"` dataset will be removed.

Finally, the bivariate choropleth and a legend will be rendered. A mapshot of the bivariate choropleth may be downloaded. Furthermore, the final dataset can also be downloaded and may be uploaded to the app later on to reproduce the same plot.

References

Art Steinmetz. (2017). *Exploring the distribution of NYC Pre-K seats vs. neighborhood income.* Retrieved 19.02.2019, from `https://rpubs.com/apsteinmetz/prek`

Chang, W., Cheng, J., Allaire, J. J., Xie, Y., & McPherson, J. (2018). *shiny: Web Application Framework for R.* Retrieved from `https://CRAN.R-project.org/package=shiny`

Cheng, J., Karambelkar, B., & Xie, Y. (2018). *leaflet: Create Interactive Web Maps with the JavaScript 'Leaflet' Library.* Retrieved from `https://CRAN.R-project.org/package=leaflet`

Joshua Stevens. (2015). *Bivariate Choropleth Maps: A How-to Guide.* Retrieved 19.02.2019, from `http://www.joshuastevens.net/cartography/make-a-bivariate-choropleth-map/`

Lahti, L., Huovari, J., Kainu, M., & Biecek, P. (2017). eurostat R package. , *9*(1), 385–392. Retrieved from `https://journal.r-project.org/archive/2017/RJ-2017-019/index.html`

Tennekes, M. (2018). tmap: Thematic Maps in R. *Journal of Statistical Software*, *84*(6), 1–39.