Exam in PSY-3006 - Indepdendent study: Visualization in R with ggplots2

Candidate number: 22

The repository is available at https://github.com/a-asen/PSY-3006-Visualization (as part of the course requirement). Word count is : `INPUT WORD COUNT HERE, THX` corresponding to approximately `PAGES`

{{ < pagebreak > }}

Visualizing

Visualizing data is a key tool in understanding data and data analysis. It is a simple trick, but highly efficient for both researchers and laymen to understand what the data and data analysis is telling us. It can condense highly complicated data structures that could be impossible for any humans to understand. Modern technological advancement has made data visualizing easier and more capable. Moreover, the development of programs and software using open source (or derivates of it), makes it possible for anyone to learn data visualization. R is a prominent program for data analysis and data visualization. It is an open source program with a vast amount of packages. One of the most important packages for R is the "ggplot2" library. This is a package created to make data visualization follow simple rules of "grammer" for "visualizing".

The Grammer of graphics

"The Grammer of Graphics" by Leland Wilkinson describes a framework for what a "grammer" of graphics should look like. The book gives a basic foundation for how to structure grammer to create graphs. Moreover, the book proposes various systems and how these systems relate to each other. The book do not rely on a specific coding language, since it is meant to give a basic understanding of how to structure the *grammer* for visualizing.

Course goal

In this course, I utilize the ggplot2 package in R to visualize data. The ggplot2 package builds on the foundation that "The Grammer of Graphics" lay. This is done to learn how to visualize data in R using the "grammer" of visualizing. The benefit of this framework is that most things should be able to be visualized with a solid understanding of the package. Moreover, the course requires the use of version control software, like Git, and reporting tools like Quarto. Git makes it possible to store version of the code. This makes it possible to go back in time (so to speak) to fix bugs that might have arisen during coding. Moreover, it is an important tool in collective coding efforts, as it can be incredibly hard to coordinate code from different people. Quarto is a reporting tool built on Rmarkdown, which is again built from Markdown. Markdown is lightweight markup language that is used to structure the format of a text file with simple symbols. For instance, the title "course goal" is started with two hashtags (##) which create a "title heading level 2" (three hashtags would create a level 3 heading - see "Circle visualization"). RMarkdown incorporate R code with the basic markdown structure, such that the text file can be easily structured and include R code and its output. Quarto builds on RMarkdown by incorporating its own syntax and improving certain features. For instance, by making it possible to do inline codes using the "`". Inline code makes it possible to implement code within a line of text. This is particularly handy if you are doing data analysis, but change one aspects which may require the rewriting of numbers.

Circle visualization

A typical plot has an x and y axis, often portrayed in a rectangular way. However, it is also possible to visualize data in a circular way. The circle plot changes the normal portrayal of the rectangular plot by *connecting back to itself*. Plotting in this way is not widespread because of obvious problems by visualizing in this way, as we will see throughout this exploration. However, there are some areas that can benefit by displaying information in a circular way. The best use-cases for circle plots are in data that naturally follow a circular pattern. This can, for instance, data over seasons, days and hours. All these aspects typically follow a recurring aspect. Indeed, Wilkinsons (2006, page 213) note that circle plots can be used for displaying seasonal data. In the first part, consisting of two exercises, of this course, I will visualizing data in circular ways.

Ex1: Seasonal sleep

The first part relates to visualizing fictive seasonal data in a circular way. The data pertains to sleep data of a fictive participant that sleeps longer during the winter period and shorter during the summer period, with a lot of variation throughout the year. The exercise explores how to visualize the sleep data in a simple and effective manner. Through that exploration, I found that visualizing all the data points is a bit difficult throughout a year. It lead to problems

in interpreting what data point was related to which day. However, by summarizing the data over each week and adding an error bar, we are able to visualize sleep in a more intelligible way. Although visualizing in a circular way can yield more pleasing figures, the loss of accurate data representation might not be worth it. Indeed, visualizing in a normal (rectangular way) yielded a greater overview of the distribution throughout the year. In this plot, we could easily see the changes in sleep amount over time. However, the relationship between seasons may be less intuitive. Hence, there are advantages and disadvantages by visualizing data through a season using circle plots. To further explore the utility of circle plots, I turn to visualizing the sleep time of participants in a sleep experiment.

Ex2: Sleep times

For the second exercise, I investigated how sleep times can be visualized in a circular way.

For the two first exercises (ex1, ex2), I focus on circular visualization. In the first exercise I start by visualizing sleep data throughout a year. The data for this exercise has been generate (see "src" scripts). In the second exercise, visualize sleep data from an experiment. There, I try to indicate the difference between two sleep conditions - one sleep deprived and one with normal sleep.

Multivariate visualization

For the two latter exercises (ex3 and ex4), I focus on multivariate visualization. Especially in relation to the joint distribution of two continuous variables of a categorical and a continuous variable.

Ex3: Continuous-continuous

- In this report I investigate distribution plots. More specifically joint distributions called "multivariate visualization". I utilize patchwork and ggside to explore how to plot individual distributions around a joint distribution space.

Ex4: Categorical-continuous

- More meaningful splitting of distributions over certain responses

Exercise reports: