iNZightTools: Tools for **iNZight**

Tom Elliott

Te Rourou Tātaritanga, Victoria University of Wellington
Department of Statistics, University of Auckland

Abstract

A package to do things.

1 Introduction

iNZight (Elliott et al., inproc) is a graphical user interface (GUI) for easy data exploration and visualisation. As part of its development, we needed wrapper packages to translate between the user interface and R functions.

- GUI has input fields user can population, equivalent to function arguments
- generally, one window talks to one function (but not always)
- often some complicated manipulation to go from user input values to valid function arguments
- doing this in a GUI is bad practice
- instead developed intermediary package that takes simple inputs and produces a `data.frame` output
- overtime this developed to also attach the **tidyverse** (Wickham et al., 2019) code used to perform the action
- simple function/argument interface makes it easier for beginners

There are a range of different classes of methods in **iNZightTools**. Many are data manipulation functions linking to **dplyr** (Wickham et al., 2020) or similar. However there are others that are 'clever' helper functions for choosing the correct function for a given situation.

- import data from a range of file formats with `smart_read()`
- import a survey design from a specification file format with `import_survey()`

2 Designing code-writing functions

- a core part of **iNZightTools** is the implementation of 'code-writing' in almost all of the functions
- there might now be better ways of doing this
- consists of putting together expressions (using)
- arguments are manipulated into the correct form and pasted into the function expression
- finally, 'iNZightTools:::interpolate()' is used to parse the expression and attach the code as an attribute (`code`)
- almost all functions return a `data.frame`
- the **iNZight** GUI can pull the resulting `data.frame` and strip off the code, adding it to the script
- R users could also run the functions and then look at the underlying code used in the background—a great way of getting familiar with the **tidyverse**

3 Importing Data

- import data with the `smart_read()` function
- uses file extension to guess best package and function to use, e.g., `.xlsx` uses `haven::read_excel()` (Wickham and Miller, 2020)
- also handles metadata parsing for comma separated values (CSV)

3.1 Parsing Metadata for Easier Data Distribution

- often data coded (factors as numbers instead of labels)
- users need to refer to information (often external) to first set-up the variables correctly before they can get started with visualisation

- this is hard/not feasible for novice users
- metadata can be included in/distributed with the raw data
- `smart_read()` will parse the metadata and apply transformations
- here are some examples

4 Data wrangling with iNZightTools

- a bunch of methods: filter, sort, aggregate, join
- here are some examples
- and accessing the code

5 Working with Complex Survey Designs

- a hugely important data type
- can be problematic if misspecified
- two helpers: using **srvyr** (Freedman Ellis and Schneider, 2020) to handle survey objects the same as `data.frame`s
- second, can parse survey specification file format distributed with the raw survey data and automatically build a survey object
- here's some examples ...

Acknowledgements

iNZight is free software developed by students of the Department of Statistics, University of Auckland. Recent funding has been received by Te Rourou Tātaritanga (`https://terourou.org`), a research group at Victoria University of Wellington funded by an MBIE Endeavour Grant, ref 62506 ENDRP, with additional funding from iNZight Analytics.

References

T. Elliott, C. Wild, D. Barnett, and A. Sporle. **iNZight**: A graphical user inferface for data visualisation and analysis through R. inproc.

G. Freedman Ellis and B. Schneider. ***srvyr**: **dplyr**-Like Syntax for Summary Statistics of Survey Data*, 2020. URL `https://CRAN.R-project.org/package=srvyr`. R package version 1.0.0.

H. Wickham and E. Miller. ***haven**: Import and Export SPSS, Stata and SAS Files*, 2020. URL `https://CRAN.R-project.org/package=haven`. R package version 2.3.1.

H. Wickham, M. Averick, J. Bryan, W. Chang, L. D. McGowan, R. François, G. Grolemund, A. Hayes, L. Henry, J. Hester, M. Kuhn, T. L. Pedersen, E. Miller, S. M. Bache, K. Müller, J. Ooms, D. Robinson, D. P. Seidel, V. Spinu, K. Takahashi, D. Vaughan, C. Wilke, K. Woo, and H. Yutani. Welcome to the **tidyverse**. *Journal of Open Source Software*, 4(43):1686, 2019. doi: 10.21105/joss.01686.

H. Wickham, R. François, L. Henry, and K. Müller. ***dplyr**: A Grammar of Data Manipulation*, 2020. URL `https://CRAN.R-project.org/package=dplyr`. R package version 1.0.2.