Introducing R/Tidyverse to Clinical Statistical Programming

*MBSW 2018*

*Freeman Wang, @freestatman*

*2018-05-15   
  
Slides available at*[*https://bit.ly/2KNKALU*](https://bit.ly/2KNKALU)

layout: true .footer[Slides available at <https://bit.ly/2KNKALU> ]

layout: true

Disclaimer

1. All the data and info in this talk are public (Twitter, GitHub).
   * CDISC example data were downloaded from: [GitHub](https://github.com/phuse-org/phuse-scripts/tree/master/data)
2. This talk represents my own views, not those of BSSI.
   * BSSI does not have an opinion of which tool you should use: e.g. SAS vs R, or R/base vs R/Tidyverse.

Why? Why so popular (1/2)

* **Not** about the good-looking plots, or the fancy manipulation functions
* Content-driven and communication-focused workflow (logic-flow)
* Concisely expresses human logic as R code
  + Fast human logic I/O
  + Yourself ↔️ team / customer
  + Past you ➡️ present you
* Seamlessly align multiple layers of logic, across analysis objective, programming, and output

Why? Why so popular (2/2)

* structured domains of workflow, and well-defined verb/vocabulary within each domain
* grammer of data manipulation (dplyr)
* grammer of data visualizaiton (ggplot2)
* grammer of statistics (not mature yet… SAS is the standard.)
* consistent design:
* **learn it once, use it everywhere**

How? Tidy principles

1. Tidy data (Shared data structures)
2. Tidy programming API (Compose simple pieces)
3. The pipe! (functional programming for Human logic)
4. Tidy statistics

Tidyverse: more packages

Clinical programming is one of Data Science

.footer[<http://r4ds.had.co.nz/>]

A “Real” Tidyverse [Workflow](http://r4ds.had.co.nz/introduction-1.html)

What? Tidy data

* Each row is an observation
* Each column is a variable
* Clnical data
* Long-format is commonly seen in data storage, e.g. SDTM/ADaM
* Wide-format is commonly seen for DEA, modeling, and visualization
* Align manipulaiton, statistical and visualization logic with tidy data

What? Grammer of data manipulation

* dplyr, key verbs
* select (common verb in SQL)
* mutate (e.g. case\_when)
* filter
* group\_by
* summarize
* arrange
* Translatable to SQL
* [Cheatsheet](https://github.com/rstudio/cheatsheets/raw/master/data-transformation.pdf)

Example of Why, How&What

* Align your manipulation visually and logically
* Convert your raw coding to more consumable coding (shorten the logic gap)

What? Tidyverse extended families

From the community \* [ggplot2 extention packages](http://www.ggplot2-exts.org/gallery/) - survminer, cowplot, etc \* [plotly](https://plotly-book.cpsievert.me/) \* [summarytools](https://github.com/dcomtois/summarytools) \* [janitor](https://github.com/sfirke/janitor) \* [tidyversity](https://github.com/mkearney/tidyversity) \* [jsmisc](https://github.com/strengejacke/sjmisc) \* More bioconductor packages buys in!

Example

**library**(haven)

**library**(tidyverse)

iris <- haven::read\_sas('data/iris.sas7bdat')

adsl <- Hmisc::sasxport.get("data/adam/cdisc/adsl.xpt")

## Processing SAS dataset ADSL ..

adsl %>%

select(usubjid, contains('trt')) %>%

DT::datatable(options = list(pageLength = 3))

Show  entries

Search:

|  | **usubjid** | **trt01p** | **trt01pn** | **trt01a** | **trt01an** | **trtsdt** | **trtedt** | **trtdur** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 01-701-1015 | Placebo | 0 | Placebo | 0 | 2014-01-02 | 2014-07-02 | 182 |
| 2 | 01-701-1023 | Placebo | 0 | Placebo | 0 | 2012-08-05 | 2012-09-01 | 28 |
| 3 | 01-701-1028 | Xanomeline High Dose | 81 | Xanomeline High Dose | 81 | 2013-07-19 | 2014-01-14 | 180 |

Showing 1 to 3 of 254 entries

Previous12345…85Next

The pipe! %>%

* Conceptually the same with Unix pipe snytax
* Push the LHS output into the 1st argument of the RHS function
* Natural representation of human logic
* Each layer of process is a function
* Enbrace Functional programming
* Similar philosophy to ggplot2
* Grammar of Graphics

#TidyTuesday

#TidyTuesday

Discussion

* R/Tidyverse is fast growing
  + Adpoting new idea
  + Some rare API change caused some pain for R package developers (OK for general users)
* Enviroment/Namespace control is a common R problem
  + Loaded functions may be over-writen by loading other packages
  + More robust usage is to add package namespace: dplyr::select()