

Using R for QC of Tables

Marcus Trincia, PHASTAR, Cambridge MA, USA

ABSTRACT

The R® programming language is no longer a tool for the future generation of clinical programmers, rather it has become a staple for this one. Due to the magnitude of open-source libraries available for R, it is difficult to find an area it cannot enrich. In this paper, we combine the familiarity of SAS® within the pharmaceutical industry with the versatility of R to provide a method for performing quality control under a double programming model. We use a combination of easily accessible SAS macros, R custom functions and libraries, and simple command-line interface to implement standard processes including dataset comparison between R and SAS files, log checks, and batch runs. We address the benefits and detriments of using both R and SAS as opposed to two SAS programs, the challenges that come with implementation, and the work yet to come regarding this project within our organization.

INTRODUCTION

In early June of 2023 PHASTAR, allocated resources towards the R-TFL project. This project was an attempt to introduce the R programming language into the double-programming quality control method for the delivery of tables, figures, and listings (TFLs). Though R programming has been used within the data science department at PHASTAR, it has never before been used for the general programming of SDTM datasets, ADaM datasets, or TFLs. The first portion of the R-TFL project consisted of an in-house, experienced R programmer teaching a group of programmers familiar with the basics of the language methods for importing, manipulating, and exporting datasets, managing workflows with git, and creating various plots. The group then researched different solutions and communicated their findings to a larger test group made up of even more in-house programmers.

R-TFL

The R-TFL project showed the power of the R programming language. The swiftness with which the programmers learned and implemented the language speaks its value. Throughout the project, it was necessary to use third-party libraries and create an autoexec program for ease of programming.

LIBRARIES AND USAGE

One of the key features of R is the easily accessible open-source libraries available to programmers. We used a number of libraries to ease the task of data wrangling, formatting, and interaction with SAS. The majority of the programs are part of the tidyverse – a collection of data science libraries. The table below lists the libraries used, their purpose, and the location of the documentation.

Library	Purpose	Documentation
dplyr	Selecting and manipulating data	https://dplyr.tidyverse.org/
haven	Importing and exporting SAS datasets	https://haven.tidyverse.org/
janitor	Data formatting	https://cran.r-project.org/web/packages/janitor/vignettes/janitor.html
stringr	Easier manipulation of strings	https://stringr.tidyverse.org/
tidyr	Creating tidy* datasets	https://tidyr.tidyverse.org/

* a “tidy” dataset is one in which each variable is a column and vice-versa, each observation is a row and vice-versa, and every value is a cell and vice-versa.

THE AUTOEXEC PROGRAM

An autoexec program consists of functions that assist with programming. When using R as part of a double-programming quality control process, a separate autoexec program must be maintained with R and SAS respectively. Some functions that we used include a function for resizing columns to a length of 200 and renaming them as COL1, COL2, ..., counting rows overall and by groups, formatting, calculating descriptive statistics, and a log check for non-error exceptions.

WORKFLOW

In the workflow diagram in Figure 1, the production program is written and executed in SAS and is represented by the top row, the quality control program is written and executed in R and is represented by the bottom row, and the center row represents the inputs and outputs. The path from collections of input datasets to table, figure, or listing starts with the respective SAS and R autoexec functions/macros along with the input dataset, which is usually an ADaM dataset,

but not necessarily. The autoexecs along with the input dataset are used to create the production program in SAS and the quality control program in R. For the SAS user, importing the ADaM dataset is done simply by traditional SAS methods, usually supported by the autoexec program. For the R user, however, it is necessary to use the `read_sas()` function from the haven library to import the dataset. Once both production and quality control have created the TFL in their respective programs, the R programmer exports the TFL to SAS using the `write_xpt()` function from the haven library. Then, in SAS, the quality control programmer carries out the TFL comparison as normal. If the datasets used to create the TFLs match, then the production programmer uses the dataset to create the output. Otherwise, both the production and quality control programmers return to their respective programs and attempt to reconcile the differences in their datasets. This process repeats until there is a dataset match.

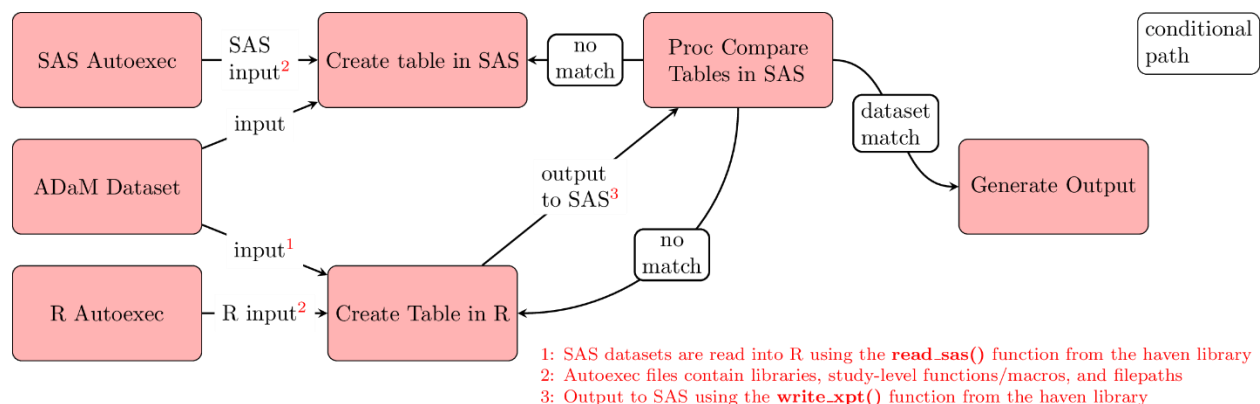


Figure 1

CONCLUSION

Introducing the R programming language into a double-programming work process has many benefits. R is one of the most popular programming languages used by mathematicians, statisticians, data scientists, and other mathematics-adjacent professionals, while SAS is more specific to the statistical programming field. Including R in the double-programming process would allow companies to hire programmers with experience in R and task them part-time with programming responsibilities and part-time with SAS training. Overall, including R casts a wider net when hiring programmers. R and SAS handle datasets differently. Because of this, R programmers have easier access to manipulating datasets by rows and cells. A final example of a benefit lies in the open-source nature of R. Since there is a whole community of talented groups and individuals working constantly in R, there are likely already libraries that solve or optimize any problem you can imagine. Access to this open-source network of solutions is an invaluable boon.

All new additions to a work environment come with complications. The workflow when using R in a double-programming quality control process is slightly more complicated than a SAS-exclusive workflow. In a SAS-exclusive workflow, it is a straight line from input to output. As discussed above, the workflow when including R adds complexity. Resourcing is another concern. Any business's goal is to make a profit, and training costs cut into margins. Teaching staff to use R requires careful consideration on the value of the end result. Finally, the effort required to maintain and update autoexec programs doubles with the addition of a second language. Those in charge of the autoexec programs must always be informed on new versions and developments in the respective languages. In the near future, we hope to pilot the qc in R process on a real study. We will continue to refine the qc process by researching, creating, and implementing new libraries, functions, and techniques. Hiring programmers with previous experience in R will ensure fresh perspectives and lead to new applications for the language.

ACKNOWLEDGMENTS

I would not have the privilege of submitting this poster and paper without the massive amount of work done by my talented colleagues. Thank you to the R Academy project head Kaustav Nandy as well as my fellow project participants: Ben Barnaby-Pass, Matthew Metherell, Grieven Otieno, Nathan Gurzell, Robert Mutua, Rodgers Ngetich, Morgan Dow, John Powers, Michael Garside, Eric Museve, Bronson Carrasco, Amanda Jeh, Andrew Elders, and Elizabeth Zink. I would also like to thank my line manager Jacob Boehmer and programming director James Miller for their ongoing support for my confidence, work, and well-being.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Author Name:

Marcus Trincia

Company:

PHASTAR

Address:
1734 Massachusetts Ave.
Suite 34
Cambridge, MA 02138
Email:
marcus.trincia@phastar.com

Brand and product names are trademarks of their respective companies.