PharmaSUG 2021 - Paper SI-203

R Validation: Approaches and Considerations

Phil Bowsher, RStudio PBC;
Sean Lopp, RStudio PBC;

Abstract

Recent R validation efforts in the clinical trials process have brought a voice to pharma that is new and exciting. This paper will highlight some of these efforts and discuss approaches that are arising for clinical workflows.

Introduction

Approaching validation uniformly for any programming language can be challenging as many organizations have different processes and levels of acceptance. This paper will discuss approaches for R and offer considerations for each part. Each approach will build off of the previously detailed section. It is well understood that R is a core language used for clinical reporting and is used by the FDA:

http://washstat.org/presentations/20181024/Schuette.pdf

The software clarifying statement helped bring awareness to the pharma community that any language could be used for submissions:
https://www.fda.gov/files/about%20fda/published/Statistical-Software-Clarifying-Statement-PDF.pdf

You can watch a video on this topic here:

https://channel9.msdn.com/Events/useR-international-R-User-conference/useR2016/Using-R-in-a-regulatory-environment-FDA-experiences

Many organizations have documented their use of R and plans to scale its use in pharma such as the GSK video below:

https://rstudio.com/resources/webinars/using-r-to-drive-agility-in-clinical-reporting/

Step 1 - Defining a baseline of validated external packages using a risk based approach

In 2018, The R Foundation for Statistical Computing released “R: Regulatory Compliance and Validation Issues: A Guidance Document for the Use of R in Regulated Clinical Trial Environments”

https://www.r-project.org/doc/R-FDA.pdf

This is a wonderful document but there was one challenge - it only covered base R:

https://cran.r-project.org/

RStudio in 2020, in collaboration with the community and pharma, released validation guidance documents covering tidyverse, tidymodels, r-lib, gt, shiny and rmarkdown. The main pain and direct links are below:

https://www.rstudio.com/solutions/pharma/

https://www.rstudio.com/assets/img/validation-tidy.pdf

https://www.rstudio.com/assets/img/validation-shiny-rmd.pdf

These documents were meant to supplement the base R document and include guidance for many of the top packages used in clinical workflows such as dplyr, shiny, rmarkdown, etc.

RStudio had previously released the guidance document for the RStudio Team software stack of RStudio Workbench, RStudio Package Manager and RStudio Connect here:

https://rstudio.com/wp-content/uploads/2019/06/rstudio_compliance_validation.pdf

Organizations take the guidance documents above and use them within a risk-based approach. This approach is well documented by the R Validation Hub and documented here:

“A Risk-Based Approach For Assessing R Package Accuracy Within A Validated Infrastructure”

https://www.pharmar.org/white-paper/

These documents, and the implemented risk-based approach, allow organizations to establish a baseline of existing packages that are deemed validated for internal use.

Once a set of packages has been approved through the processes outlined above, many IT organizations will make the packages available to users organizations through an internal repository. Often releases of packages are aligned with IT support for specific versions of R. You can see an example of a validated repository here:

https://colorado.rstudio.com/rspm/client/#/repos/10/packages

R Users will then access packages according to the version of R in use in a project. RStudio Server supports multiple versions of R to allow users to test different versions of R and packages on their code. These packages are automatically used when content like shiny applications are deployed to RStudio Connect.

An alternative approach for using approved packages is through user-managed libraries that take advantage of renv:

https://github.com/rstudio/renv

Step 2 - Validating internal packages or User-Defined Functions

The above process helps identify approved existing packages, validation for base R, and validation guidance for over 100 RStudio affiliated packages. Often the next question is *“What about our own packages and UDFs (user-defined functions)?”* First it is important to treat internal packages and UDFs with the same risk-based approaches as defined above. Understanding the concepts in this paper is helpful:

How Do I Pick a R Package for My Clinical Workflow?

https://github.com/philbowsher/phuse-2019-r-packages/blob/master/How%20Do%20I%20Pick%20a%20R%20Package%20for%20My%20Clinical%20Workflow_.pdf

Internal functions and packages should include documentation, tests, release control, and other processes that are applied to validate external packages. In fact, because internal code is used by a smaller audience than most external open source packages, it is **more important to validate internal UDFs than existing CRAN packages.**

To help formalize this approach, Ellis Hughes is working on a R Package validation framework as highlighted at his 2020 R in Pharma conference 2020 talk here:

https://thebioengineer.github.io/validation_rpharma/

https://www.youtube.com/watch?v=zEH-6Ik-5h8&feature=youtu.be&ab_channel=RinPharma

This process takes advantage of many existing tools for creating robust R packages outside of the validation space such as testthat, rmarkdown and roxygen2.

Many people and organizations have begun to use this process for validating their packages. An example package validated with this process is below:

https://github.com/atorus-research/pharmaRTF

You can see the validated work by going to the tests folder:

https://github.com/atorus-research/pharmaRTF/tree/master/tests

Alternative Approach - Buying Validated Packages

Some organizations prefer to buy validation documentation for existing open source packages. There are various vendors selling validation tests and documents as well as groups selling services to support validation.

Notes - Looking out into the horizon

- The Cloud could have a major impact on this space in many ways, making it easier for organizations to adopt and share standard and explicit infrastructure using tools like Docker.
- Shiny is an exciting tool and many see it as the door into a new horizon for clinical trials. You can see an example here: https://williamnoble.shinyapps.io/the_future_of_clinical_tfls/
  Many groups like Biogen have public repos highlighting their investment in this future: https://github.com/Biogen-Inc/tidyCDISC
- As more and more machine and deep learning applications come to clinical trials, R is well suited for this space as a major data science language.

Contact & Summary

The information above highlights an exciting future for clinical trials being written by the data science and open source community. The approaches above are used by many organizations and finding the right path varies greatly from organization to organization.

Phil Bowsher

phil@rstudio.com

https://github.com/philbowsher