Paper SM02

Same Same but Different: Leveraging R for Comparing Clinical Study TLF Versions

Tiina Kirsilä, Novartis, Basel, Switzerland
Anna Wiksten, Bristol Myers Squibb, Boudry, Switzerland

ABSTRACT

Large amounts of outputs are generated during a clinical trial. Whenever there are changes in the underlying data, expected or unexpected, the outputs need to be re-generated.

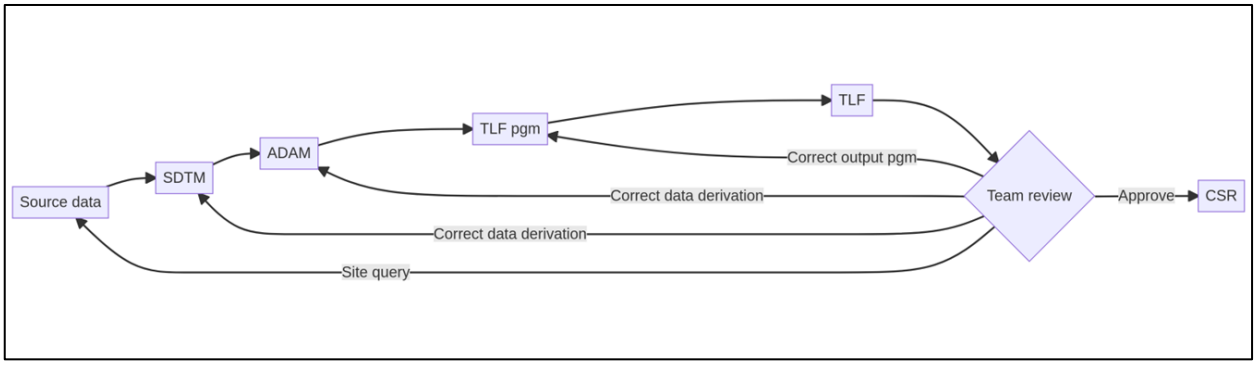
In this paper we present a set of R functions that offers a powerful and flexible solution for comparing and reviewing clinical study output versions. Using these functions one can speed up the review of the outputs by quickly identifying the outputs that have and haven't changed. Hence, no time is wasted for reviewing the outputs which have not been affected by data changes.

The functions can be used via R-shiny App providing a user-friendly interface for running the functions and discussing the findings with different stakeholders from clinical trial and medical writing teams.

INTRODUCTION

Tables, listings and figures are a mandated part of clinical study report from a clinical trial [1]. The golden standard of double programming ensures that the gathered & derived data, statistical analysis plan and the mock output layouts are correctly translated into programmed outputs. In addition, visual inspection of the TLFs is needed, by the QC programmer and the clinical team. During the review change requests and findings may lead to changes at any level; all the way from source data to the TLF program as illustrated in Figure 1.It is normal to have hundreds of outputs that will go thru this same process with re-iterations.

Figure 1: Clinical trial data and TLF flow



For various reasons, especially due to changes in the underlying data (source or derived), all the TLFs may need to be re-run. At this stage doing the visual inspection of every output is very time consuming and identifying the outputs with changes is error prone. Many companies have implemented SAS®-macros for running `proc compare` to identify impacted outputs and providing static reports about changes. These are excellent for individual reviews or for recording the changes for audit trail, however they may not be optimal for interactive review. Opportunity to do an interactive review is especially helpful when discussing with other stake holders like medical writers and clinical scientists. Other challenge is that these comparison macros often require the results to be saved in a separate SAS data set. We provide a solution where this step is not needed, and the outputs can be compared directly.

In the pharma industry the TLFs are often produced in the RTF format. RTF files are just text; hence the files are machine readable. This allows direct comparisons of old and new RTF file versions programmatically instead of visual comparison. With the proposed method the RTF files are first read in and then comparisons are performed.

This paper introduces a set of R functions for reading and comparing the RTF files. The functions are available in an experimental state R-package `verifyr`. It is available at: https://github.com/Novartis/verifyr

FUNCTIONS

There are 3 functions within the R-package `verifyr` at the moment.

Function	Output
`list_files`	returns the list of file names that exist in given folders. (Optional: consider only files with defined name pattern)
`initial_comparison`	returns basic info about the differences of two files.
`full_comparison`	returns side by side comparison of two RTF files highlighting the differences.

R SHINY APP

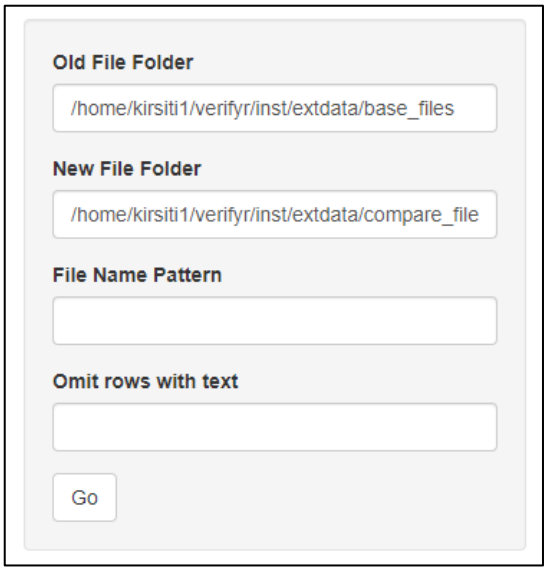
Although the functions can be used as standalone function, the interactive features of R-Shiny can really make the comparison of the RTF files quick and efficient.
The `verifyr` package contains a Shiny App example that can be called after installing the package with the following command:

```
verifyr::runExample()
```

This will open a Shiny App that demonstrates the use of the functions. As first step the user has to give two folder locations; the full path to the old version outputs and new version outputs. Optionally the user can:

1. Read only a subset of outputs by defining the pattern of the file name (e.g. 14-1)
2. Omit certain rows with a given pattern (e.g. Date:)

INPUTS



After the folder locations are given, the files names are listed (using `verifyr::list_files`) and the initial comparison is executed for every file separately (using `verifyr::initial_comparison`). The comparison column gives the high-level info of the comparison.

LIST FILES AND INITIAL COMPARISON

Show 10 entries Search:

	file	old_path	new_path	omitted	comparison
1	14-1.01.rtf	/home/kirsiti1/verifyr/inst/extdata/base_files/14-1.01.rtf	/home/kirsiti1/verifyr/inst/extdata/compare_files/14-1.01.rtf	null	Output has changes in 3 places
2	14-1.02.rtf	/home/kirsiti1/verifyr/inst/extdata/base_files/14-1.02.rtf	/home/kirsiti1/verifyr/inst/extdata/compare_files/14-1.02.rtf	null	No changes
3	14-1.03.rtf	/home/kirsiti1/verifyr/inst/extdata/base_files/14-1.03.rtf		null	Unable to compare
4	14-2.01.rtf	/home/kirsiti1/verifyr/inst/extdata/base_files/14-2.01.rtf	/home/kirsiti1/verifyr/inst/extdata/compare_files/14-2.01.rtf	null	Only one row has changed
5	Table 10-1.1.rtf	/home/kirsiti1/verifyr/inst/extdata/base_files/Table 10-1.1.rtf	/home/kirsiti1/verifyr/inst/extdata/compare_files/Table 10-1.1.rtf	null	Output size has changes

Showing 1 to 5 of 5 entries Previous 1 Next

In the above table the user can click one of the rows and this will open a side-by-side comparison of the selected output (using `verifyr::full_compare`) where the changed rows are highlighted.

FULL COMPARISON

CONCLUSION

Hundreds of outputs are generated for a clinical study report and sometimes re-run of them is needed. To avoid the need of reviewing all the outputs again by the study team the method presented in this paper quickly and accurately identifies the outputs that have and haven't changed. The presented codes have proven to be accurate and quick. It can read and run the initial comparison of all outputs from a large Phase III study in less than 10 minutes, for reference sometimes even opening two versions of a large listing can take longer.

Although the codes are tailored towards a scenario where minimal impact is expected the same method can be applied to reviewing IB, PSUR etc. outputs that are periodically updated. Similarly, the method can be used to verify original and qc tables if they follow same layout.

REFERENCES

1. ICH Topic E 3 Structure and Content of Clinical Study Reports https://www.ema.europa.eu/en/documents/scientific-guideline/ich-e-3-structure-content-clinical-study-reports-step-5_en.pdf

ACKNOWLEDGMENTS

We would like to thank Pekka Heikkilä for igniting our passion for standardizing things and teaching us that RTF is just text. We would also like to thank Rashad Rasul and Karma Tarap for encouraging us to publish our work.

CONTACT INFORMATION

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

Tiina Kirsilä
Email: tiina.kirsilae(at)novartis.com