phuse·virtual

Single Day Event

How not to ‘err’ with R?

Nandini Rajendhiran
and Anmol Gureja,
Novartis

Content

- Introduction
- Risks
- Challenges and solutions
- Packages and functions
- Future scope

Disclaimer

- The views presented are the views of the presenter, not necessarily those of Novartis.
- These slides are intended for educational purposes only and for the personal use of the audience. These slides are not intended for wider distribution outside the intended purpose without presenter approval.
- The content of this slide deck is accurate to the best of the presenter's knowledge at the time of production.

Introduction

From the perspective of **Statistical Programming**, use of open-source software such as R is the next evolutionary step in Clinical trials. Implementation of **R programming** for the **validation** of all **CSR related Tables, Listings, and Figures** of a **Full Development Study** has shown how to venture into this.

Why?	Evolving with current trends in the industry, TFLs validation is a good starting point to explore open-source software
Who?	Identified a few SAS programmers who are academically/ externally trained in R
How?	Using R Programming, exploring new packages, with focus on meeting industry-level standards

Risks

During the process of validation, few risks were identified that we could minimize at initial stage.

- Validation of packages
- Function deprecation
- Function masking

Validation of Packages

With the great capabilities and cost-effectiveness of R software, there is a concern of risk and compliance of it being open-source.

Will I get accurate results?

What if the package is updated?

Can I use any package?

I should validate the package.

Steps to follow

Investigating

Testing

Validating

Approval

Function related issues

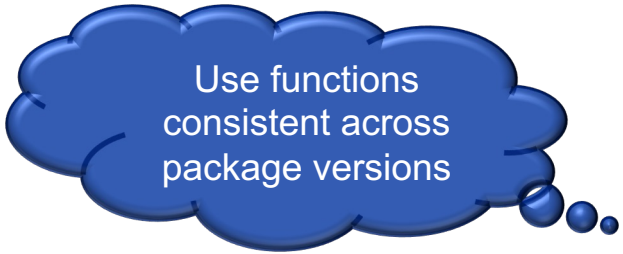
- **Function deprecation**: As functions change in different versions of their packages, warnings may occur.

```
Warning message:
funs() is soft deprecated as of dplyr 0.8.0
Please use a list of either functions or lambdas:

  # Simple named list:
  list(mean = mean, median = median)

  # Auto named with `tibble::lst()`:
  tibble::lst(mean, median)

  # Using lambdas
  list(~ mean(., trim = .2), ~ median(., na.rm = TRUE))
This warning is displayed once per session.
```



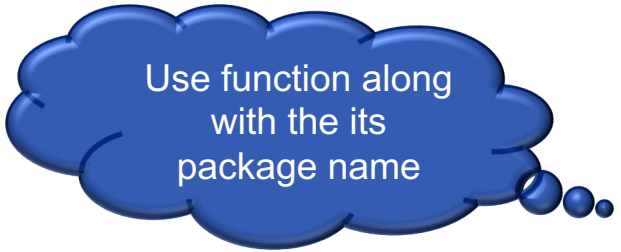
- **Function masking**: Different functions with the same name being present in different packages, the actual function being used may not be the one desired.

```
R 3.6.1> library(data.table)
R 3.6.1> library(reshape2)

Attaching package: 'reshape2'

The following objects are masked from 'package:data.table':

    dcast, melt
```



Ex.: data.table::dcast

Challenges faced

- Warnings and errors
- Decimal issues
- Parameter issues
- SAS vs R differences

Warnings and errors

- Warning message:
  attributes are not identical across measure variables;
  they will be dropped

- Warning message:
  In if (a > 0) { :
    the condition has length > 1 and only the first element
  will be used

- Error in .checkTypos(e, names_x) :
    Object 'usubjid' not found. Perhaps you intended USUBJID

Remove redundant attributes (or) use alternate functions

Use appropriate functions depending on the data structure

Ensure variable in right case is present

Decimal issues

- **Confidence Interval calculations:** Mismatches due to formula issue

SAS values	R values
(-415.6,-223.8)	(-415.7,-223.8)
(-597.3,-128.2)	(-597.3,-128.1)
(-510.3,31.5)	(-510.4,31.5)
(-89.3,72.0)	(-89.3,71.9)

- **Rounding off values:** Mismatches due to internal calculation rule

orig	round
2.55	2.6
2.65	2.7
2.75	2.8

Rounding in SAS

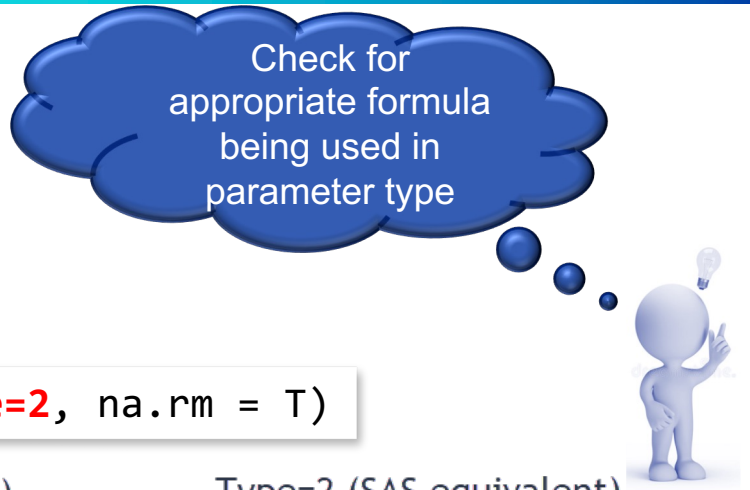
orig	round
2.55	2.5
2.65	2.6
2.75	2.8

Rounding in R

Parameter issues

The default parameter used in R gives different values compared to SAS.

1. **Quantile function-** Default parameter in R is **type=7**, but its equivalent in SAS is **type=2**




```
q1 = quantile(x = var, probs = 0.25, type=2, na.rm = T)
```

variable	TrtA	TrtB	Total
25-75th percentiles	85.5 - 113.9	92.9 - 113.3	89.1 - 113.7

Values of quantiles in SAS

Type=7 (Default)

variable	TrtA	TrtB	Total
q1_q3	86.2-113.8	92.9-113.3	89.1-113.7

Type=2 (SAS equivalent)

variable	TrtA	TrtB	Total
q1_q3	85.5-113.9	92.9-113.3	89.1-113.7

Values of quantiles in R

Parameter issues

2. **Survfit function for CI -** Default parameter in SAS is **log-log**, but its default in R is **log.**
   So this parameter has to be changed from log to log-log in R.

Check for appropriate formula being used in parameter type

```
CI = survfit(Surv(time,event)~grp, data, conf.type="log-log", conf.int=0.95)
```

Type= log (Default)

est	lower_CI	upper_CI
20.3	0	42.6
20.3	0	42.6

Type=log-log (SAS equivalent)

est	lower_CI	upper_CI
20.3	5.7	60.9
20.3	5.7	60.9

Values of CI in R

est	lower_CI	upper_CI
20.3	5.7	60.9
20.3	5.7	60.9

Values of CI in SAS

SAS vs R differences

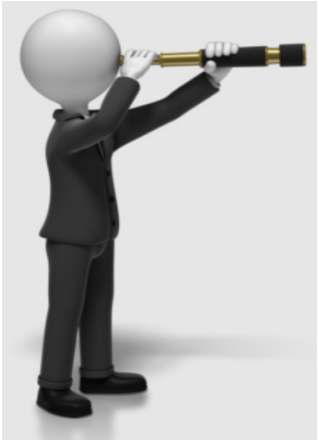
Difference	SAS	R	How to handle it
Data type	2 types- Numeric and character	Multiple types- Integer, Complex, Character, Logical, etc.	Make sure **type conversion** is done where required
Data structure	Data tables	Vector, Matrix, data frame, list , etc.	**Optimize** use of data structures as required
Missing values	. and “”	NA, NA_numeric, NA_character, etc.	Ensure **proper conversion** of NA values

Packages and functions

Along with the base and stats packages, we relied on a few others such as:

- **pacman** – efficient loading and unloading of packages.
- **haven** – importing SAS datasets.
- **dplyr, tidyr** and **data.table** – more efficient handling of datasets.
- **sqldf** – functions to use SQL syntax in R.
- **stringr** and **lubridate**– functions for data types handling.
- **striprtf** and **arsenal** – import rtf files and compare datasets.

Future scope



- Use of R for development of datasets and TFLs.
- Create pharma related packages.
- Automate the process of developing TFLs using Shiny app.
- Utilization of validated and updated packages across organizations.

"It always seems impossible until it's done" – Nelson Mandela

Thank You

phuse

The Global Healthcare Data Science Community

Contact Channels

UK +44 1843 609600
USA +1 609 514 5105
office@phuse.global
phuse.global

Social Media

@phusetwitta
/phusebook
/phusetube
/company/phuse