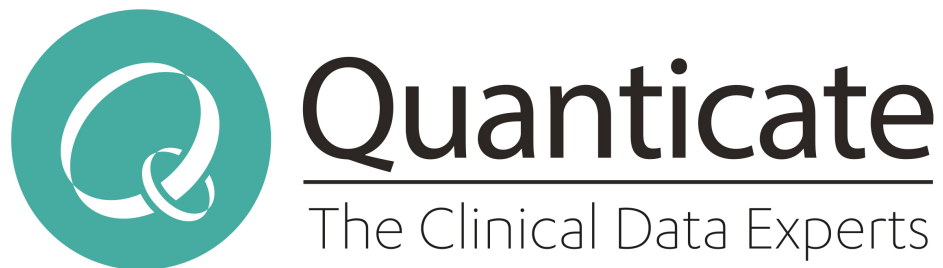


Speaker's ID card

- Statistician (who enjoys programming a lot 😊)
- Used R for 8+ years in academic and research institutions
- Re-learned SAS when re-joining the industry 5 years ago
- Current full-time SAS user
- Current full-time R wannabe-user



TT13

Are there any better options for TLFs than SAS? Yes, there R!

Niccolo' Bassani

Agenda

- What to expect from this presentation
- The current process in SAS
- How to do it in R
- Comparison: pros and cons
- Does R rival SAS, in the end?

What to expect?

- A practical example to create a summary/analysis Table
- R code and illustrations of its flexibility
- Some hints and tips to create great looking outputs
- Comparison with SAS main reporting features (PROC REPORT)
- A statistician perspective on ‘Why R?’

Our target is...

	Drug A	Drug D	Drug F
Baseline			
Mean (SD)	xx.x (xx.xx)	xx.x (xx.xx)	xx.x (xx.xx)
Median	xx.x	xx.x	xx.x
Min ; Max	xx ; xx	xx ; xx	xx ; xx
End of Treatment			
Mean (SD)	xx.x (xx.xx)	xx.x (xx.xx)	xx.x (xx.xx)
Median	xx.x	xx.x	xx.x
Min ; Max	xx ; xx	xx ; xx	xx ; xx
ANCOVA			
LS Means (95%CI)	xx.x (xx.x – xx.x)	xx.x (xx.x – xx.x)	xx.x (xx.x – xx.x)
LS Means Difference (95% CI)	xx.x (xx.x – xx.x)	xx.x (xx.x – xx.x)	
p-value	0.xxx	0.xxx	

The current process in SAS?

- PROC MEANS/SUMMARY/UNIVARIATE to get the summary statistics
- PROC GENMOD/GLM/MIXED to estimate LS Means, their difference and 95%CI
- A mixture of data steps and PROC SQL to manipulate datasets
- A PROC REPORT call to generate the final output

The result is nice!

Protocol: Example Study

Page 1 of 1

Table 1. Summary and Analysis of Covariance (ANCOVA) analysis - leprosy treatment data

	Drug A	Drug D	Drug F
Baseline			
N	10	10	10
Mean (SD)	9.3 (4.76)	10.0 (5.25)	12.9 (3.96)
Median	9.0	8.0	12.0
Min ; Max	3.0 ; 19.0	5.0 ; 19.0	7.0 ; 21.0
End of Treatment			
N	10	10	10
Mean (SD)	5.3 (4.64)	6.1 (6.15)	12.3 (7.15)
Median	5.0	3.5	12.5
Min ; Max	0.0 ; 13.0	0.0 ; 18.0	1.0 ; 23.0
ANCOVA			
LS Means	6.71	6.82	10.16
(95%CI)	(4.07, 9.36)	(4.21, 9.44)	(7.46, 12.87)
Difference in LS means	-3.45	-3.34	
(95%CI)	(-7.32, 0.43)	(-7.15, 0.47)	
p-value	0.0793	0.0835	

Generated on 30JAN2020 (9:29)

by J:\Statistics\Private\Niccolo\TT13 - programs and data\SAS output\table1.sas

How do you translate this in R?

- Use functionalities from the `ddply` package to calculate summary statistics
- Base R coupled with the `emmeans` package to estimate LS Means and their differences
- The `sqldf` package allows using SQL-like syntax much like in SAS
- Use `officer` and `flextable` packages for reporting!

How do you translate this in R?

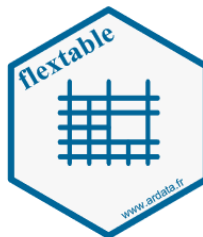
- Use functionalities from the `ddply` package to calculate summary statistics
- Base R coupled with the `emmeans` package to estimate LS Means and their differences
- The `sqldf` package allows using SQL-like syntax much like in SAS
- **Use `officer` and `flextable` packages for reporting!**

The flexible package

flextable **0.5.8** Overview Selectors Layout Format visual properties Format Content Render as image

flextable R package

build passing  BUILD PASSING CRAN 0.5.8 downloads 16K/month repo status Active



The flextable package provides a framework for easily create tables for reporting and publications. Tables can be embedded within:

- R Markdown documents with support for HTML, Word and PowerPoint documents.
- Microsoft Word or PowerPoint documents.
- PDF documents with package `pagedown` (it's only HTML)

Tables can also be rendered as R plots or graphic files (png, pdf and jpeg).

How it works

- The key part is the `flextable` object class
- Once a data frame is converted styling functions can be applied to define header, body, titles/footnotes, etc
- The new object can still be accessed as a data frame (i.e. indexes)
- The ordering of functions can matter!

Using the flextable package

```
summary_ft <- flextable(summary07) %>%
```

Define
column/rows
features

```
width(j = 2:4, width = 2) %>%  
width(j = 1, width = 2.5) %>%  
height(height = 0.15, part = "body") %>%  
align(j = 2:4, align = "center", part = "all") %>%  
align(j = 1, align = "left", part = "all") %>%
```

Add titles
and
foonotes

```
add_header_lines(values = paste("Table 1. Summary and Analysis of covariance  
  (ANCOVA) analysis - leprosy data", strrep(" ", 50), sep =  
  "")) %>%  
add_header_lines(values = paste(prot, strrep(" ", 93), "Page 1 of 1", sep =  
  "")) %>%  
add_footer_lines(values = paste("Generated on ", Sys.time(), " by ", work,  
  "Table1.R", sep = "")) %>%
```

Fix
horizontal
borders

```
hline_bottom(border = fp_border(color = "black"), part = "body") %>%  
hline_bottom(border = fp_border(color = "black"), part = "header") %>%  
border_inner_h(border = fp_border("black"), part = "header") %>%
```

Font
features

```
hline(i = 1, border = fp_border("white"), part = "header") %>%  
font(font = "CourierNewPSMT", part = "all") %>%  
fontsize(size = 8, part = "all")
```

Exporting: the `officer` package



Quanticate
The Clinical Data Experts

- Create a “base” empty document (with margins and orientation consistent)
- Open the connection with the `read_docx` function
- Add the flextable object with the `body_add_flextable` function
- Print the final object to its destination

```
doc <- read_docx("base.docx") %>%  
  body_add_flextable(value =  
    summary_ft)  
print(doc, target = "Table1.docx")
```

How does it look?

Protocol: Example Study

Page 1 of 1

Table 1. Summary and Analysis of Covariance (ANCOVA) analysis - leprosy data

	Drug A	Drug D	Drug F
Baseline			
N	10	10	10
Mean (SD)	9.3 (4.76)	10.0 (5.25)	12.9 (3.96)
Median	9.0	8.0	12.0
Min ; Max	3.0 ; 19.0	5.0 ; 19.0	7.0 ; 21.0
End of Treatment			
N	10	10	10
Mean (SD)	5.3 (4.64)	6.1 (6.15)	12.3 (7.15)
Median	5.0	3.5	12.5
Min ; Max	0.0 ; 13.0	0.0 ; 18.0	1.0 ; 23.0
ANCOVA			
LS Means	6.71	6.82	10.16
95% CI	(4.07 , 9.36)	(4.21 , 9.44)	(7.46 , 12.87)
Difference in LS Means	-3.45	-3.34	
95% CI	(-7.32 , 0.43)	(-7.15 , 0.47)	
p-value	0.0793	0.0835	

Generated on 2020-01-30 10:26:14 by J:/Statistics/Private/Niccolo/TT13 - programs and data/Table1.R

flextable vs PROC REPORT

Scope	PROC REPORT option/statement	FLEXTABLE function
Aligning columns	<code>center/left/right</code> options in the <code>define</code> statement	<code>align</code> function
Width of columns	<code>width</code> option in the <code>define</code> statement	<code>width</code> function
Line above column names	<code>'___'</code> in the <code>column</code> statement	<code>border_inner_h</code> function
Line below column names	<code>headline</code> option in <code>proc report</code> statement	<code>hline_bottom</code> function
Line at the bottom of the table	<code>compute after _page_ /<page-variable></code> block	
Add footnotes/titles	<code>footnoten/titlen</code> statement <code>/line</code> statement in a <code>compute after</code> block (footnotes only)	<code>add_header_lines</code> (titles) <code>add_footer_lines</code> (footnotes)
Multicolumn header	Embed columns in brackets: <code>("header" col1 col2)</code>	<code>add_header_row</code> using the <code>colwidths</code> option

How about multiple pages?

- There is not a definite function to achieve *exactly* what we usually intend...
- ...but R is flexible: you can write your own function that does what you need!
- Separately print pages to the output destination and add page breaks

Multiple pages (code)

```
if (i == 1) {  
  doc <- read_docx("base.docx") %>%  
    body_add_flextable(value = summary_ft)  
  fileout <- "Table1.docx"  
  print(doc, target = fileout)  
} else {  
  doc <- read_docx("Table1.docx") %>%  
    body_add_break() %>%  
    body_add_flextable(value = summary_ft)  
  fileout <- "Table1.docx"  
  print(doc, target = fileout)  
}  
}
```

When pages after the first are printed they're appended to the document where the first one was printed, but in a new page, as we wanted

Does R rival SAS, in the end?

- Short answer: yes, it does 😊
- Great flexibility: existing methods can be tweaked and adapted, SAS is not always that straightforward
- Data ‘wrangling’ (via `tidyverse`) and SQL-like features exists
- R is more efficient and intuitive to program computationally intensive methods (e.g. permutations, resampling)
- Breadth of statistical methods available
- Most differences are not ‘real’ differences

Final remarks

- R can create nice looking TLFs
- The process is similar and as/more efficient as/then with SAS
- After all, we know very well how to efficiently use SAS...
- ...but not quite the same can be said for R.
- As R, hopefully, spreads, experience will be gained and its capabilities will be more 'objectively' assessed

Know your tools!



If statistics programs/languages were cars...





Questions?

Paper TT13

“Is there any better option than SAS for
TLFs? Yes, there R!”

Email to:

niccolo.bassani@quanticate.com