

A Novel Way of Creating Tables in R

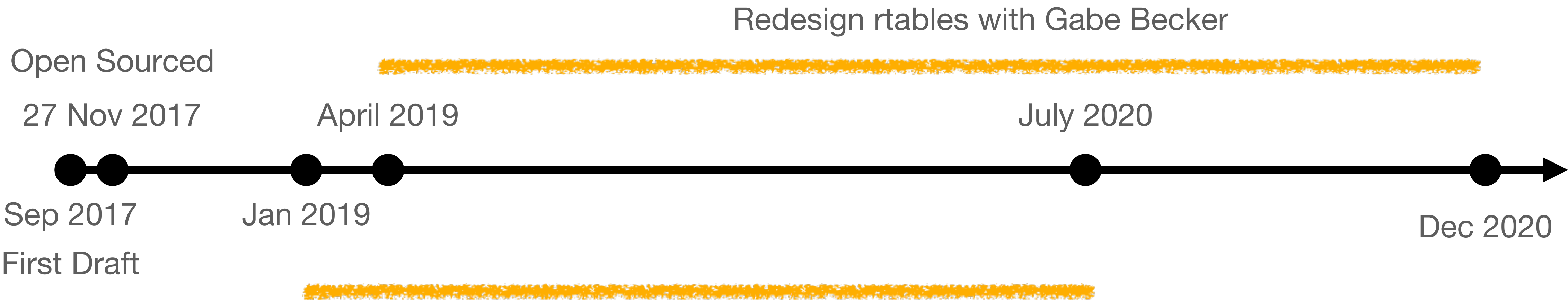
using the refactored rtables R package

SD05 - Software Demonstration Stream
Phuse EU Connect 2020

Adrian Waddell, Roche Basel, September 18, 2020



rtables roadmap



NEST Subject Matter Expert Team
Implementation of ~ 120 standard tables



Reimplementation of 120 tables
with refactored rtables
Add more standard tables

Why Did We Refactor rtables



- Improved Data Structure
 - accessors and modifiers
 - pagination
 - titles, footnotes, referencing cell/row/column content
- More Powerful Tabulation Framework
 - focus on analysis vs. tabulation
 - speed

The rtables version used for this presentation is tagged v0.3.2.4 on github.com/Roche/rtables

```
devtools::install_github("Roche/rtables", ref = "v0.3.2.4")
```



```
> library(rtables)
> ex_adsl
# A tibble: 400 x 31
  STUDYID USUBJID SUBJID SITEID  AGE SEX  RACE  COUNTRY INVID ARM  ARMCD
<chr> <chr> <chr> <chr> <int> <fct> <fct> <fct> <chr> <fct> <fct>
1 AB12345 AB1234... id-128 CHN-3  32 M   ASIAN CHN    CHN-3 A: D... ARM A
2 AB12345 AB1234... id-262 CHN-15  35 M   BLAC... CHN    CHN-... C: C... ARM C
3 AB12345 AB1234... id-378 RUS-3  30 F   ASIAN RUS    RUS-3 C: C... ARM C
4 AB12345 AB1234... id-220 CHN-11  26 F   ASIAN CHN    CHN-... B: P... ARM B
5 AB12345 AB1234... id-267 CHN-7  40 M   ASIAN CHN    CHN-7 B: P... ARM B
6 AB12345 AB1234... id-201 CHN-15  49 M   ASIAN CHN    CHN-... C: C... ARM C
7 AB12345 AB1234... id-45  USA-1  34 F   ASIAN USA    USA-1 A: D... ARM A
8 AB12345 AB1234... id-261 USA-1  32 F   ASIAN USA    USA-1 B: P... ARM B
9 AB12345 AB1234... id-173 NGA-11  24 F   BLAC... NGA    NGA-... C: C... ARM C
10 AB12345 AB1234... id-307 CHN-1  24 M   ASIAN CHN    CHN-1 B: P... ARM B
# ... with 390 more rows, and 20 more variables: ACTARM <fct>,
# ACTARMCD <fct>, STRATA1 <fct>, STRATA2 <fct>, BMRKR1 <dbl>,
# BMRKR2 <fct>, ITTFL <fct>, SAFFL <fct>, BMEASIFL <fct>, BEP01FL <fct>,
# RANDDT <date>, TRTSDTM <dtm>, TRTEDTM <dtm>, EOSSTT <fct>,
# EOSDT <date>, EOSDY <int>, DCSREAS <fct>, DTHDT <date>,
# LSTALVDT <date>, study_duration_secs <dbl>
```

```
> library(rtables)
```

```
> ex_adsl
```

```
# A tibble: 400 x 31
```

```
  STUDYID USUBJID SUBJID SITEID  AGE SEX  RACE  COUNTRY INVID ARM  ARMCD
```

```
  <chr>   <chr>   <chr>   <chr>   <int> <fct> <fct> <fct>   <chr> <fct> <fct>
```

```
1 AB1234
```

```
2 AB1234
```

```
3 AB1234
```

```
4 AB1234
```

```
5 AB1234
```

```
6 AB1234
```

```
7 AB1234
```

```
8 AB1234
```

```
9 AB1234
```

```
10 AB1234
```

```
# ... with
```

```
# ACTAR1
```

```
# BMRKR2 <tct>, ITTFL <tct>, SAFFL <tct>, BMEASIFL <tct>, BEPU1FL <tct>,
```

```
# RANDDT <date>, TRTSDTM <dtm>, TRTEDTM <dtm>, EOSSTT <fct>,
```

```
# EOSDT <date>, EOSDY <int>, DCSREAS <fct>, DTHDT <date>,
```

```
# LSTALVDT <date>, study_duration_secs <dbl>
```

This data was created with the random.cdisc.data R package (to be open sourced in the near future) and is currently available in the rtables R package.

This data is completely synthetic.

```
> basic_table() %>% split_cols_by("ARMCD") %>% # new rtables
  analyze("AGE", mean) %>% build_table(ex_adsl)
```

ARM A

ARM B

ARM C

```
-----
mean  33.7686567164179  35.4328358208955  35.4318181818182
```

```
> basic_table() %>% split_cols_by("ARMCD") %>% # new rtables
  analyze("AGE", mean) %>% build_table(ex_adsl)
```

	ARM A	ARM B	ARM C
--	-------	-------	-------

mean	33.7686567164179	35.4328358208955	35.4318181818182
------	------------------	------------------	------------------

```
> rtabulate(ex_adsl$AGE, ex_adsl$ARMCD, mean) # old rtables
```

	ARM A	ARM B	ARM C
--	-------	-------	-------

	33.7686567164179	35.4328358208955	35.4318181818182
--	------------------	------------------	------------------

```
> basic_table() %>% split_cols_by("ARMCD") %>% # new rtables
  analyze("AGE", mean) %>% build_table(ex_adsl)
```

	ARM A	ARM B	ARM C
--	-------	-------	-------

mean	33.7686567164179	35.4328358208955	35.4318181818182
------	------------------	------------------	------------------

```
> rtabulate(ex_adsl$AGE, ex_adsl$ARMCD, mean) # old rtables
```

	ARM A	ARM B	ARM C
--	-------	-------	-------

	33.7686567164179	35.4328358208955	35.4318181818182
--	------------------	------------------	------------------

```
> tapply(ex_adsl$AGE, ex_adsl$ARMCD, mean) # base R
```

	ARM A	ARM B	ARM C
--	-------	-------	-------

	33.76866	35.43284	35.43182
--	----------	----------	----------

```
> basic_table() %>% split_cols_by("ARMCD") %>%
  analyze("AGE", mean) %>% build_table(ex_adsl)
```

	ARM A	ARM B	ARM C
--	-------	-------	-------

mean	33.7686567164179	35.4328358208955	35.4318181818182
------	------------------	------------------	------------------

```
> rtabulate(ex_adsl$AGE, ex_adsl$ARMCD, mean)
```

	ARM A	ARM B	ARM C
--	-------	-------	-------

	33.7686567164179	35.4328358208955	35.4318181818182
--	------------------	------------------	------------------

```
> tapply(ex_adsl$AGE, ex_adsl$ARMCD, mean)
```

ARM A	ARM B	ARM C
-------	-------	-------

33.76866	35.43284	35.43182
----------	----------	----------

Data to analyze

```
> basic_table() %>% split_cols_by("ARMCD") %>%  
  analyze("AGE", mean) %>% build_table(ex_adsl)
```

	ARM A	ARM B	ARM C
--	-------	-------	-------

mean	33.7686567164179	35.4328358208955	35.4318181818182
------	------------------	------------------	------------------

```
> rtabulate(ex_adsl$AGE, ex_adsl$ARMCD, mean)
```

	ARM A	ARM B	ARM C
--	-------	-------	-------

	33.7686567164179	35.4328358208955	35.4318181818182
--	------------------	------------------	------------------

```
> tapply(ex_adsl$AGE, ex_adsl$ARMCD, mean)
```

ARM A	ARM B	ARM C
-------	-------	-------

33.76866	35.43284	35.43182
----------	----------	----------

Data to analyze

Compare between groups

```
> basic_table() %>% split_cols_by("ARMCD") %>%
  analyze("AGE", mean) %>% build_table(ex_adsl)
```

	ARM A	ARM B	ARM C
--	-------	-------	-------

mean	33.7686567164179	35.4328358208955	35.4318181818182
------	------------------	------------------	------------------

```
> rtabulate(ex_adsl$AGE, ex_adsl$ARMCD, mean)
```

	ARM A	ARM B	ARM C
--	-------	-------	-------

	33.7686567164179	35.4328358208955	35.4318181818182
--	------------------	------------------	------------------

```
> tapply(ex_adsl$AGE, ex_adsl$ARMCD, mean)
```

ARM A	ARM B	ARM C
-------	-------	-------

33.76866	35.43284	35.43182
----------	----------	----------

Data to analyze

Compare between groups

Analysis function

Tabulation



- What data should be analysed
- How should the data be analysed
- Between which groups should the analysis be compared

Tabulation / Tables



- What data should be analysed
- How should the data be analysed
- Between which groups should the analysis be compared
- Constructing tables manually
- Cell & Table Format
- Output in ASCII, HTML, PDF, XML, etc.
- etc...

Tabulation

```
> basic_table() %>%
+ split_cols_by("ARMCD") %>%
+ analyze("AGE", mean) %>%
+ build_table(ex_adsl)
```

	ARM A	ARM B	ARM C
mean	33.7686567164179	35.4328358208955	35.4318181818182

```
> basic_table() %>%
+ split_cols_by("ARMCD") %>%
+ analyze("AGE", mean) %>%
+ build_table(ex_adsl)
```

Define tabulation “pre-data”: layout

Perform tabulation on data: table

	ARM A	ARM B	ARM C
mean	33.7686567164179	35.4328358208955	35.4318181818182

```
> lyt <- basic_table() %>%  
+ split_cols_by("ARMCD") %>%  
+ analyze("AGE", mean)  
> lyt
```

A Pre-data Table Layout

Column-Split Structure:

ARMCD (lvls)

Row-Split Structure:

AGE (** analysis **)

```
> lyt <- basic_table() %>%  
+ split_cols_by("ARMCD") %>%  
+ analyze("AGE", mean)  
> lyt
```

A Pre-data Table Layout

Column-Split Structure:

ARMCD (lvls)

Row-Split Structure:

AGE (** analysis **)

Layouts can be reused

```
> tbl1 <- lyt %>%  
+ build_table(ex_adsl)  
  
> tbl2 <- lyt %>%  
+ build_table(ex_adsl %>% filter(SEX == "M"))  
  
> tbl3 <- lyt %>%  
+ build_table(ex_adsl %>% filter(ITTFL == "Y"))
```

Building Layouts



split_cols_by()
split_cols_by_multivar()
split_cols_by_cuts()
split_cols_by_cutfun()
split_cols_by_quartiles()

split_rows_by()
split_rows_by_cuts()
split_rows_by_cutfun()
split_rows_by_quartiles()

analyze()
analyze_colvars()
analyze_against_ref_group()
summarize_row_groups()

add_colcounts()
add_overall_level()

Analysis Functions



```
s_summary <- function(x) {  
  if (is.numeric(x)) {  
    in_rows(  
      "n" = rcell(sum(!is.na(x)), format = "xx"),  
      "Mean (sd)" = rcell(c(mean(x, na.rm = TRUE), sd(x, na.rm = TRUE)),  
        format = "xx.xx (xx.xx)"),  
      "IQR" = rcell(IQR(x, na.rm = TRUE), format = "xx.xx"),  
      "min - max" = rcell(range(x, na.rm = TRUE), format = "xx.xx - xx.xx")  
    )  
  } else if (is.factor(x)) {  
  
    vs <- as.list(table(x))  
    do.call(in_rows, lapply(vs, rcell, format = "xx"))  
  
  } else (  
    stop("type not supported")  
  )  
}
```

Analysis

```
s_summary <- function(x) {
  if (is.numeric(x)) {
    in_rows(
      "n" = rcell(sum(!is.na(x)), format = "xx"),
      "Mean (sd)" = rcell(c(mean(x, na.rm = TRUE), sd(x, na.rm = TRUE)),
        format = "xx.xx (xx.xx)"),
      "IQR" = rcell(IQR(x, na.rm = TRUE), format = "xx.xx"),
      "min - max" = rcell(range(x, na.rm = TRUE), format = "xx.xx - xx.xx")
    )
  } else if (is.factor(x)) {

    vs <- as.list(table(x))
    do.call(in_rows, lapply(vs, rcell, format = "xx"))

  } else (
    stop("type not supported")
  )
}
```

Formatting

```
s_summary <- function(x) {  
  if (is.numeric(x)) {  
    in_rows(  
      "n" = rcell(sum(!is.na(x)), format = "xx"),  
      "Mean (sd)" = rcell(c(mean(x, na.rm = TRUE), sd(x, na.rm = TRUE)),  
        format = "xx.xx (xx.xx)"),  
      "IQR" = rcell(IQR(x, na.rm = TRUE), format = "xx.xx"),  
      "min - max" = rcell(range(x, na.rm = TRUE), format = "xx.xx - xx.xx")  
    )  
  } else if (is.factor(x)) {  
  
    vs <- as.list(table(x))  
    do.call(in_rows, lapply(vs, rcell, format = "xx"))  
  
  } else (  
    stop("type not supported")  
  )  
}
```

Table Construction



```
s_summary <- function(x) {  
  if (is.numeric(x)) {  
    in_rows(  
      "n" = rcell(sum(!is.na(x)), format = "xx"),  
      "Mean (sd)" = rcell(c(mean(x, na.rm = TRUE), sd(x, na.rm = TRUE)),  
                          format = "xx.xx (xx.xx)"),  
      "IQR" = rcell(IQR(x, na.rm = TRUE), format = "xx.xx"),  
      "min - max" = rcell(range(x, na.rm = TRUE), format = "xx.xx - xx.xx")  
    )  
  } else if (is.factor(x)) {  
  
    vs <- as.list(table(x))  
    do.call(in_rows, lapply(vs, rcell, format = "xx"))  
  
  } else (  
    stop("type not supported")  
  )  
}
```

```
> basic_table() %>%
+   split_cols_by("ARMCD") %>%
+   analyze("AGE", s_summary) %>%
+   build_table(ex_adsl)
```

	ARM A	ARM B	ARM C
n	134	134	132
Mean (sd)	33.77 (6.55)	35.43 (7.9)	35.43 (7.72)
IQR	11	10	10
min - max	21 - 50	21 - 62	20 - 69

```
> basic_table() %>%
+   split_cols_by("ARMCD") %>%
+   analyze(c("AGE", "SEX"), s_summary) %>%
+   build_table(ex_adsl)
```

	ARM A	ARM B	ARM C

AGE			
n	134	134	132
Mean (sd)	33.77 (6.55)	35.43 (7.9)	35.43 (7.72)
IQR	11	10	10
min - max	21 - 50	21 - 62	20 - 69
SEX			
F	79	77	66
M	51	55	60
U	3	2	4
UNDIFFERENTIATED	1	0	2

```
> basic_table() %>%
+   split_cols_by("ARMCD") %>%
+   analyze(c("AGE", "SEX"), s_summary) %>%
+   build_table(ex_adsl)
```

	ARM A			

AGE	s_summary can l			
n	134	demo		
Mean (sd)	33.7			
IQR	11			
min - max	21 - 50	21 - 62	20 - 69	
SEX				
F	79	77	66	
M	51	55	60	
U	3	2	4	
UNDIFFERENTIATED		1	0	2

s_summary can be used to build standard demographic tables

```
s_events_patients <- function(x, .N_col, lblstr = NULL) {
  in_rows(
    "Total number of patients with at least one event" =
      rcell(length(unique(x)) * c(1, 1/.N_col), format = "xx (xx.xx%)"),

    "Total number of events" = rcell(length(x), format = "xx")
  )
}
```

```
basic_table() %>%  
  split_cols_by("ARM") %>%  
  add_colcounts() %>%  
  analyze("USUBJID", s_events_patients) %>%  
  split_rows_by("AEBODSYS", child_labels = "visible", nested = FALSE) %>%  
  summarize_row_groups("USUBJID", cfun = s_events_patients) %>%  
  analyze("AEDECOD", show_labels = "hidden", indent_mod = -1) %>%  
  build_table(ADAE2, col_counts = table(ADSL2$ARM))  
  
# ADSL2 is defined in the clinical trials vignette in rtables
```



	ARM A (N=146)	ARM B (N=154)
Total number of patients with at least one event		113 (77.4%) 150 (97.4%)
Total number of events		2044 1060
GASTROINTESTINAL DISORDERS		
Total number of patients with at least one event		113 (77.4%) 129 (83.77%)
Total number of events		757 373
ABDOMINAL DISCOMFORT		113 65
ABDOMINAL FULLNESS DUE TO GAS		119 64
BACK PAIN	0	0
DIARRHEA	107	53
FAECES SOFT	121	58
GINGIVAL BLEEDING		146 71
HEADACHE	0	0
HYPOTENSION	0	0
NAUSEA (INTERMITTENT)		151 62
ORTHOSTATIC HYPOTENSION		0 0
WEAKNESS	0	0
MUSCULOSKELETAL AND CONNECTIVE TISSUE DISORDERS		
Total number of patients with at least one event		98 (67.12%) 81 (52.6%)
Total number of events		272 142
ABDOMINAL DISCOMFORT		0 0
ABDOMINAL FULLNESS DUE TO GAS		0 0
BACK PAIN	134	75



	ARM A (N=146)	ARM B (N=154)

Total number of patients with at least one event	113 (77.4%)	150 (97.4%)
Total number of events	2044	1060
GASTROINTESTINAL DISORDERS		
Total number of patients with at least one event	113 (77.4%)	129 (83.77%)
Total number of events	757	373
ABDOMINAL DISCOMFORT	113	65
ABDOMINAL FULLNESS DUE TO GAS	0	0
BACK PAIN	0	0
DIARRHEA	0	0
FAECES SOFT	0	0
GINGIVAL BLEEDING	0	0
HEADACHE	0	0
HYPOTENSION	0	0
NAUSEA (INTERMITTENT)	151	62
ORTHOSTATIC HYPOTENSION	0	0
WEAKNESS	0	0
MUSCULOSKELETAL AND CONNECTIVE TISSUE DISORDERS		
Total number of patients with at least one event	98 (67.12%)	81 (52.6%)
Total number of events	272	142
ABDOMINAL DISCOMFORT	0	0
ABDOMINAL FULLNESS DUE TO GAS	0	0
BACK PAIN	134	75

s_events_patients can be used to build adverse events tables

Demographic Table:

```
basic_table() %>%  
  split_cols_by("ARMCD") %>%  
  analyze(c("AGE", "SEX"), s_summary) %>%  
  build_table(ex_adsl)
```

Adverse Events Table:

```
basic_table() %>%  
  split_cols_by("ARM") %>%  
  add_colcounts() %>%  
  analyze("USUBJID", s_events_patients) %>%  
  split_rows_by("AEBODSYS", child_labels = "visible", nested = FALSE) %>%  
  summarize_row_groups("USUBJID", cfun = s_events_patients) %>%  
  analyze("AEDECOD", show_labels = "hidden", indent_mod = -1) %>%  
  build_table(ADAE2, col_counts = table(ADSL2$ARM))
```

Demographic Table:

```
basic_table() %>%  
  split_cols_by("ARMCD") %>%  
  analyze(c("AGE", "SEX"), s_summary) %>%  
  build_table(ex_adsl)
```

Adverse Events Table:

```
basic_table() %>%  
  split_cols_by("ARM") %>%  
  add_colcounts() %>%  
  analyze("USUBJID", s_events_patients) %>%  
  split_rows_by("AEBODSYS", child_labels = "visible", nested = FALSE) %>%  
  summarize_row_groups("USUBJID", cfun = s_events_patients) %>%  
  analyze("AEDECOD", show_labels = "hidden", indent_mod = -1) %>%  
  build_table(ADAE2, col_counts = table(ADSL2$ARM))
```

Tables

How can we access 6.55?

```
> tbl <- basic_table() %>%
+   split_cols_by("ARMCD") %>%
+   analyze(c("AGE", "SEX"), s_summary) %>%
+   build_table(ex_adsl)
> tbl
```

	ARM A	ARM B	ARM C
AGE			
n	134	134	132
Mean (sd)	33.77 (6.55)	35.43 (7.9)	
IQR	11	10	10
min - max	21 - 50	21 - 62	20
SEX			
F	79	77	66
M	51	55	60
U	3	2	4
UNDIFFERENTIATED	1	0	

```
> tbl[3, 1, drop = TRUE][2]
```

```
6.553326
```

```
> cell_value(tbl, c("AGE", "mean_sd"), "ARM A")$sd
```

```
6.553326
```

How can we access 6.55?

```
> tbl <- basic_table() %>%
+   split_cols_by("ARMCD") %>%
+   analyze(c("AGE", "SEX"), s_summary) %>%
+   build_table(ex_adsl)
> tbl
```

	ARM A	ARM B	ARM C

AGE			
n	134	134	132
Mean (sd)	33.77 (6.55)	35.43 (7.9)	
IQR	11	10	10
min - max	21 - 50	21 - 62	20
SEX			
F	79	77	66
M	51	55	60
U	3	2	4
UNDIFFERENTIATED	1	0	

```
> tbl[3, 1, drop = TRUE][2]
```

6.553326

```
> cell_value(tbl, c("AGE", "mean_sd"), "ARM A")$sd
```

6.553326

- Unrounded numbers
- Pathing will be added soon
 - tabulation defines table data structure

```
basic_table() %>%  
  split_cols_by("ARM") %>%  
  add_colcounts() %>%  
  analyze("USUBJID", afun = s_events_patients) %>%  
  split_rows_by("AEBODSYS", child_labels = "visible", nested = FALSE) %>%  
  summarize_row_groups("USUBJID", cfun = s_events_patients) %>%  
  analyze("AEDECOD", show_labels = "hidden", indent_mod = -1) %>%  
  build_table(ADAE2, col_counts = table(ADSL2$ARM))
```

ARM A
(N=146)

ARM B
(N=154)



Total number of patients with at least one event 113 (77.4%) 150 (97.4%)

Total number of events 2044 1060

GASTROINTESTINAL DISORDERS

Total number of patients with at least one event 113 (77.4%) 129 (83.77%)

Total number of events 757 373

ABDOMINAL DISCOMFORT		113	65
ABDOMINAL FULLNESS DUE TO GAS		119	64
BACK PAIN	0	0	
DIARRHEA	107	53	
FAECES SOFT	121	58	
GINGIVAL BLEEDING		146	71
HEADACHE	0	0	
HYPOTENSION	0	0	
NAUSEA (INTERMITTENT)		151	62
ORTHOSTATIC HYPOTENSION		0	0
WEAKNESS	0	0	

MUSCULOSKELETAL AND CONNECTIVE TISSUE DISORDERS

Total number of patients with at least one event 98 (67.12%) 81 (52.6%)

Total number of events 272 142

ABDOMINAL DISCOMFORT		0	0
ABDOMINAL FULLNESS DUE TO GAS		0	0
BACK PAIN	134	75	

Label Rows

Content Rows

Analysis Rows



sorting/pruning

outputs

pdf, html, rtf, xml, etc...

table post
processing

pagination

listings

What More?

coercions
e.g. to gt

title/footnotes

(forest) plots

cell/row/column
references

comparison
against baseline

subsetting

...and more...

Acknowledgements



- Roche for financing and supporting the development of rtables
- Main rtables developer: Gabe Becker
- NEST business lead: Tadeusz Lewandowski
- SME NEST team
 - Nick Paszty, Jana Stoilova, Heng Wang, Francois Collin, Daniel Sabanes Bove, Nina Qi
- Everyone else who provided feedback on rtables

Thank You!

github.com/Roche/rtables

