

Adopting gtsummary at Scale

How Roche Built a Companion to gtsummary to
Standardise and Simplify ARD-Based Reporting

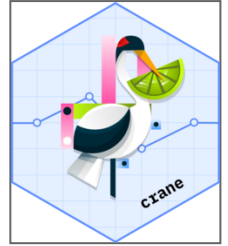
Daniel D. Sjoberg

PHUSE US Connect 2026



What is {crane} ?

- {crane} is an R package extending {gtsummary} for Roche's reporting requirements
- {crane} exports a {gtsummary} theme
- {crane} exports functions to bespoke summary tables



But First, What is {gtsummary} ?

How IT STARTED

- Began to address reproducibility issues while working in academia
- Goal to build a package to summarize study results with code that was **both simple and customizable**

How IT'S GOING

- The stats
 - **2,000,000 installations** from CRAN
 - 1,200 GitHub stars
 - 1,200 citations in peer-reviewed articles
 - 50 code contributors
- Won the 2025 Brian Bole Award of Excellence from R in Pharma
- Won the 2021 American Statistical Association (ASA) Innovation in Programming Award
- Won the 2024 Posit Pharma Table Contest





Monthly {gtsummary} CRAN Downloads

Figure contains 30-day downloads for all summary table packages on CRAN.



{gtsummary} + LLMs

- Since {gtsummary} is widely adopted, our LLMs besties work wonderfully out of the box. No additional training needed!
- Recently added an AI assistant to the package site and it's AMAZING! (Thanks [kapa.ai](#))



gtsummary 2.4.0.9003 Reference Articles ▾ News

gtsummary

The {gtsummary} package provides a framework to create publication-ready analytic reports in R programming language. The {gtsummary} package works with data sets, regression models, and other objects with highly customizable capabilities.

- **Summarize data frames or models** (e.g. presenting descriptive statistics, creating **demographics** (e.g. creating continuous, categorical, and other descriptive statistics, and all other summary statistics).
- **Summarize regression models** (e.g. presenting regression coefficients, odds ratios, hazard ratios, and all other summary statistics).

gtsummary Docs AI

How do I create a demographics table?

To create a demographics table with the gtsummary package in R, you have two main approaches: using `tbl_summary()` directly on your data frame, or using the ARD-first workflow with `tbl_ard_summary()` for more complex or regulatory-style tables.

1. Using `tbl_summary()`

This is the most common and straightforward way. Here's a basic example using the built-in `trial` dataset:

```
library(gtsummary)

# Basic demographics table
table1 <- trial |>
  tbl_summary(
    include = c(age, grade, response)
  )
```



This Talk is Not about {gtsummary}

But, I want to touch on two items

- .. {gtsummary} creates beautiful tables that are easy to customize
- ?. {gtsummary} supports themes that allow users to change defaults and other details of summary tables



A Little Data Preparation

```
1 library(gtsummary)
2 library(tidyverse)
3
4 adsl <- pharmaverseadam::adsl |>
5   filter(SAFFL == "Y") |>
6   mutate(ARM2 = word(ARM), FEMALE = SEX == "F") |>
7   labelled::set_variable_labels(FEMALE = "Female")
8
9 adae <- pharmaverseadam::adae |>
10  filter(
11    USUBJID %in% adsl$USUBJID,
12    AESOC %in% c("CARDIAC DISORDERS", "EYE DISORDERS"),
13    AEDECOD %in% c("ATRIAL FLUTTER", "MYOCARDIAL INFARCTION", "E
14  ) |>
15  mutate(ARM2 = word(ARM))
16
17 adtte <- pharmaverseadam::adtte_onco |>
18  dplyr::filter(PARAM == "Progression Free Survival") |>
19  mutate(ARM2 = word(ARM))
```



{gtsummary} Tables

We will review briefly just one summary table function.



- `tbl_summary()`

Other functions helpful functions we're not covering:

- `tbl_hierarchical()`: Summarize AE, Con Meds, and other similar rates
- `tbl_hierarchical_count()`: similar to `tbl_hierarchical()` for counts instead of rates
- `tbl_cross()`: cross tabulations
- `tbl_continuous()`: summarizing continuous variables by 2 categorical variables
- `tbl_wide_summary()`: similar to `tbl_summary()` but statistics are presented in separate columns
- many more!



Basic tbl_summary()

```
1 library(gtsummary)
2
3 adsl |>
4   tbl_summary(
5     include = c(AGE, ETHNIC,
6   )
```

Characteristic	N = 254 ¹
Age	77 (70, 81)
Ethnicity	
HISPANIC OR LATINO	12 (4.7%)
NOT HISPANIC OR LATINO	242 (95%)
Female	143 (56%)
¹ Median (Q1, Q3); n (%)	

- Four types of summaries: `continuous`, `continuous2`, `categorical`, and `dichotomous`
- Statistics are `median (IQR)` for continuous, `n (%)` for categorical/dichotomous



- Variables coded 0/1, TRUE/FALSE, Yes/No treated as dichotomous by default
- Label attributes are printed automatically



Customize tbl_summary() output

```
l |>
tbl_summary(
  include = c(AGE, ETHNIC, FEMALE),
  by = ARM2,
```

Characteristic	Placebo N = 86 ¹	Xanomeline N = 168 ¹
Age	76 (69, 82)	77 (71, 81)
Ethnicity		
HISPANIC OR LATINO	3 (3.5%)	9 (5.4%)
NOT HISPANIC OR LATINO	83 (97%)	159 (95%)
Female	53 (62%)	90 (54%)
¹ Median (Q1, Q3); n (%)		

- **by**: specify a column variable for cross-tabulation



Customize tbl_summary() output

```
l |>
tbl_summary(
  include = c(AGE, ETHNIC, FEMALE),
  by = ARM2,
  type = AGE ~ "continuous2",
```

Characteristic	Placebo N = 86 ¹	Xanomeline N = 168 ¹
Age		
Median (Q1, Q3)	76 (69, 82)	77 (71, 81)
Ethnicity		
HISPANIC OR LATINO	3 (3.5%)	9 (5.4%)
NOT HISPANIC OR LATINO	83 (97%)	159 (95%)
Female	53 (62%)	90 (54%)
¹ n (%)		

- **by**: specify a column variable for cross-tabulation
- **type**: specify the summary type



Customize tbl_summary() output

```
l |>
tbl_summary(
  include = c(AGE, ETHNIC, FEMALE),
  by = ARM2,
  type = AGE ~ "continuous2",
  statistic =
    list(
      AGE ~ c("{mean} ({sd})",
              "{min}, {max}"),
      FEMALE ~ "{n} / {N} ({p}%)"
    ),

```

Characteristic	Placebo N = 86 ¹	Xanomeline N = 168 ¹
Age		
Mean (SD)	75 (9)	75 (8)
Min, Max	52, 89	51, 88
Ethnicity		
HISPANIC OR LATINO	3 (3.5%)	9 (5.4%)
NOT HISPANIC OR LATINO	83 (97%)	159 (95%)
Female	53 / 86 (62%)	90 / 168 (54%)
¹ n (%); n / N (%)		



- **by**: specify a column variable for cross-tabulation
- **type**: specify the summary type
- **statistic**: customize the reported statistics



Customize tbl_summary() output

```
l |>
tbl_summary(
  include = c(AGE, ETHNIC, FEMALE),
  by = ARM2,
  type = AGE ~ "continuous2",
  statistic =
    list(
      AGE ~ c("{mean} ({sd})",
              "{min}, {max}"),
      FEMALE ~ "{n} / {N} ({p}%)"
    ),
  label =
    AGE ~ "Age, years",
```

Characteristic	Placebo N = 86 ¹	Xanomeline N = 168 ¹
Age, years		
Mean (SD)	75 (9)	75 (8)
Min, Max	52, 89	51, 88
Ethnicity		
HISPANIC OR LATINO	3 (3.5%)	9 (5.4%)
NOT HISPANIC OR LATINO	83 (97%)	159 (95%)
Female	53 / 86 (62%)	90 / 168 (54%)

¹ n (%); n / N (%)



- **by**: specify a column variable for cross-tabulation
- **type**: specify the summary type
- **statistic**: customize the reported statistics
- **label**: change or customize variable labels



Customize tbl_summary() output

```
l |>
tbl_summary(
  include = c(AGE, ETHNIC, FEMALE),
  by = ARM2,
  type = AGE ~ "continuous2",
  statistic =
    list(
      AGE ~ c("{mean} ({sd})",
              "{min}, {max}"),
      FEMALE ~ "{n} / {N} ({p}%)"
    ),
  label =
    AGE ~ "Age, years",
  digits = AGE ~ list(sd = 1) # report SD(age) to
    e decimal place
```

Characteristic	Placebo N = 86 ¹	Xanomeline N = 168 ¹
Age, years		
Mean (SD)	75 (8.6)	75 (8.1)
Min, Max	52, 89	51, 88
Ethnicity		
HISPANIC OR LATINO	3 (3.5%)	9 (5.4%)
NOT HISPANIC OR LATINO	83 (97%)	159 (95%)
¹ n (%); n / N (%)		



Characteristic	Placebo N = 86 ¹	Xanomeline N = 168 ¹
Female	53 / 86 (62%)	90 / 168 (54%)
¹ n (%); n / N (%)		

- **by**: specify a column variable for cross-tabulation
- **type**: specify the summary type
- **statistic**: customize the reported statistics
- **label**: change or customize variable labels
- **digits**: specify the number of decimal places for rounding



Add-on functions in {gtsummary}

`tbl_summary()` objects can also be updated using related functions.

- `add_*()` add **additional column** of statistics or information, e.g. p-values, q-values, overall statistics, treatment differences, N obs., and more
- `modify_*()` **modify** table headers, spanning headers, footnotes, and more



Update tbl_summary() with add_*

```
l |>
tbl_summary(
  by = ARM2,
  include = c(AGE, ETHNIC, FEMALE)
)|>
add_overall(last = TRUE)
```

Characteristic	Placebo N = 86 ¹	Xanomeline N = 168 ¹	Overall N = 254 ¹
Age	76 (69, 82)	77 (71, 81)	77 (70, 81)
Ethnicity			
HISPANIC OR LATINO	3 (3.5%)	9 (5.4%)	12 (4.7%)
NOT HISPANIC OR LATINO	83 (97%)	159 (95%)	242 (95%)
Female	53 (62%)	90 (54%)	143 (56%)

¹ Median (Q1, Q3); n (%)

- `add_overall()`: adds a column of overall statistics



Add-on functions in {gtsummary}

And many more!

See the documentation at

<http://www.danielsjoberg.com/gtsummary/reference/index>.

And a detailed `tbl_summary()` vignette at

[http://www.danielsjoberg.com/gtsummary/articles/tbl_sumi](http://www.danielsjoberg.com/gtsummary/articles/tbl_summary)



Cobbling Table with {gtsummary}

Two or more {gtsummary} tables can be combined by either **merging** or **stacking**.

- `tbl_merge()` for horizontal combining
- `tbl_stack()` for vertical combining

But more on this later in the {crane} section



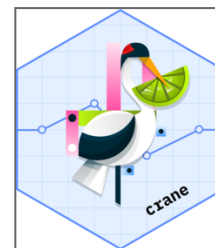
Finally, Introducing {crane}

-
-



Wrapping Functions

The first function we added to {crane} was `tbl_roche_summary()`: a *very thin* wrapper for `gtsummary::tbl_summary()`.



- Continuous variables default to `continuous2`.
- `tbl_summary(missing*)` arguments have been changed to `tbl_roche_summary(nonmissing*)`.
- We highlight non-missing counts over missing counts, which are the default in {gtsummary}
- Counts represented by `0 (0%)` print as `0`.

```
1 library(crane)
2
3 adsl |>
4   dplyr::mutate(ETHNIC = forcats::fct_expand(ETHNIC, "REFUSED"))
5   tbl_roche_summary(
6     by = ARM2,
7     include = c(AGE, ETHNIC),
8     nonmissing = "always"
9   )
```



Wrapping Functions

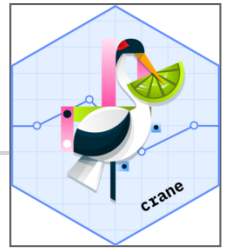


Table 1

	Placebo (N = 86)	Xanomeline (N = 168)
Age		
n	86	168
Mean (SD)	75.2 (8.6)	75.0 (8.1)
Median	76.0	77.0
Min - Max	52 - 89	51 - 88
ETHNIC		
n	86	168
HISPANIC OR LATINO	3 (3.5%)	9 (5.4%)
NOT HISPANIC OR LATINO	83 (96.5%)	159 (94.6%)
REFUSED	0	0

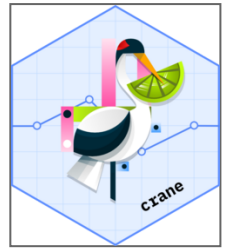


Extending with New Functions

Lab values are summarized by visit and include the change from baseline.

This is a simple table that is just a `tbl_merge()` of the **AVAL** summary and the **CHG** summary.

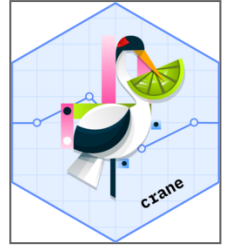
But the general structure appears enough times in our catalog, we make it simple for our programmers to create.



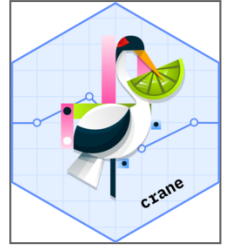
```
1 library(crane)
2
3 adlb |>
4   dplyr::filter(PARAM == "Albumin (g/L)") |>
5   tbl_baseline_chg(
6     by = "ARM",
7     baseline_level = "Baseline",
8     denominator = adsl
9   )
```



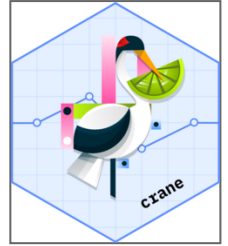
Extending with New Functions



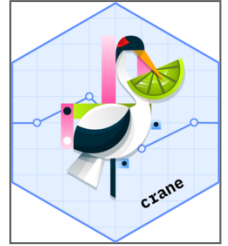
Extending with New Functions



Extending with New Functions



Extending with New Functions



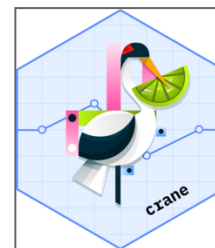
Create a Company Theme

Our theme is implemented in

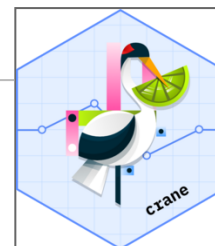
```
crane::theme_gtsummary_roche()
```

Primary changes include:

- Sets a custom function for **rounding percentages**.
- Round all **p-values** to four decimal places.
- **Headers** default to include the N in parenthesis *without* bold, e.g. 'Placebo \n (N = 184)'.
- All tables are printed with **{flextable}** and we add Roche-specific styling to the table.
- Update the default **font, font size, table borders, cell padding, etc.** to meet our guidelines.



Create a Company Theme



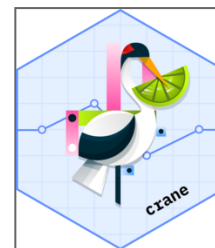
```
1 theme_gtsummary_roche()
2
3 adsl |>
4   dplyr::mutate(ETHNIC = forcats::fct_expand(ETHNIC, "REFUSED"))
5   tbl_roche_summary(
6     by = ARM2,
7     include = c(AGE, ETHNIC),
8     nonmissing = "always"
9   )
```

	Placebo (N = 86)	Xanomeline (N = 168)
Age		
n	86	168
Mean (SD)	75.2 (8.6)	75.0 (8.1)
Median	76.0	77.0
Min - Max	52 - 89	51 - 88
ETHNIC		
n	86	168
HISPANIC OR LATINO	3 (3.5%)	9 (5.4%)
NOT HISPANIC OR LATINO	83 (96.5%)	159 (94.6%)
REFUSED	0	0

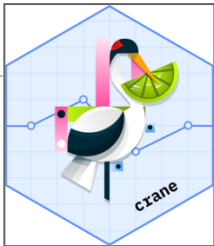


Extend with ARD-first Functionality

- We don't have time to cover in detail, but there is another wonderful way to create bespoke tables and functions.
- The {gtsummary} package supports creating tables using ARDs (Analysis Results Datasets).
- Data ➡ ARD ➡ Table
- This method is particularly useful for efficacy tables, as they contain statistics that are not our standard rates, counts, and univariate descriptor statistics.
- Review the **ARD-first Vignette** for a detailed walk through.



Extend with ARD-first Functionality



```
1 tbl_survfit_times(  
2   data = addtte,  
3   times = 12,  
4   by = "ARM2",  
5   label = "Month {time}"  
6 )
```

	Placebo (N = 86)	Xanomeline (N = 168)
Month 12		
Patients remaining at risk	5	6
Event Free Rate (%)	80.0	100.0

If you prefer not to make an ARD first, you can also just create a data frame of a table and convert it to {gtsummary} and style it from there.



A “coding partner” that strictly adheres to your domain-specific standards without constant reminders.

SKILLS.md

```

1 # Skills: Clinical Trial Summary Table Development
2
3 Standardized workflow for R-based clinical reporting using `crane`
4
5 ## Configuration & Principles
6 * **Setup:** Always run `crane::theme_gtsummary_roche()` after l
7 * **Data Prep:** Use **tidyverse** for data preparation before c
8
9 ## Table Workflows
10
11 ### Path A: Data-First (Primary)
12 Use for standard summaries.
13 1. **Function:** `crane::tbl_*()` (if applicable), else `gtsummary
14 2. **Traceability:** Extract ARD immediately using `gtsummary::
15
16 ### Path B: ARD-First (Fallback)
17 Use for complex/custom nesting.
18 1. **Calculate:** Generate ARD via `cards::ard_stack()` or `card
19 2. **Render:** Pass ARD to `gtsummary::tbl_ard_*()` functions.
20
21 ## Core Libraries
22 `library(tidyverse)`, `library(crane)`, `library(gtsummary)`, `l

```



When it comes time to build your custom tables, use the {crane} package as a blueprint.





www.danielsjoberg.com/phuse-crane-2026/

Speaker notes

