Drill Down Summary Tables in Shiny

with Tplyr

Mike Stackhouse
Jessica Higgins, PhD

atorus
Your Complete Data Solution

Pre-reqs for this workshop

- We assume you have:
  - A background in creating clinical tables
  - A working understanding of R
    - Understand the RStudio IDE
    - Understand R objects, such as vectors, lists, dataframes, etc.
    - Basic understanding of user defined objects (i.e. S3)
  - A working understanding of Shiny
    - Reactive elements
    - UI/Server relationship

Agenda

- Workshop Intro and Motivation
- Tplyr Basics
  - Breakout 1
- Configuring Tplyr Tables
  - Breakout 2
- Tplyr Metadata Part 1 – Structure and Concept
  - Breakout 3
- Tplyr Metadata Part 2 – Use in Shiny
  - Breakout 4
  - Breakout 5

What is Tplyr?

- R Package released in 2020
- A grammar of clinical summary tables
- Design by describing the output
- In goes data, out goes presentation ready data.frame

```
tplyr_table(adsl, TRT01P) %>%
   add_layer(
    group_count(AGEGR1, by = "Age Group n (%)") %>%
      set_format_strings(f_str("xx (xx.x%)", n, pct)) %>%
   add_layer(
    group_desc(AGE, by = "Age (years)") %>%
      set_format_strings(
        "n" = f_str("xx", n),
        "Mean (SD)" = f_str("xx.x (xx.xx)", mean, sd)
      ) %>%
  ) %>%
  build()
```

Why'd we make Tplyr?

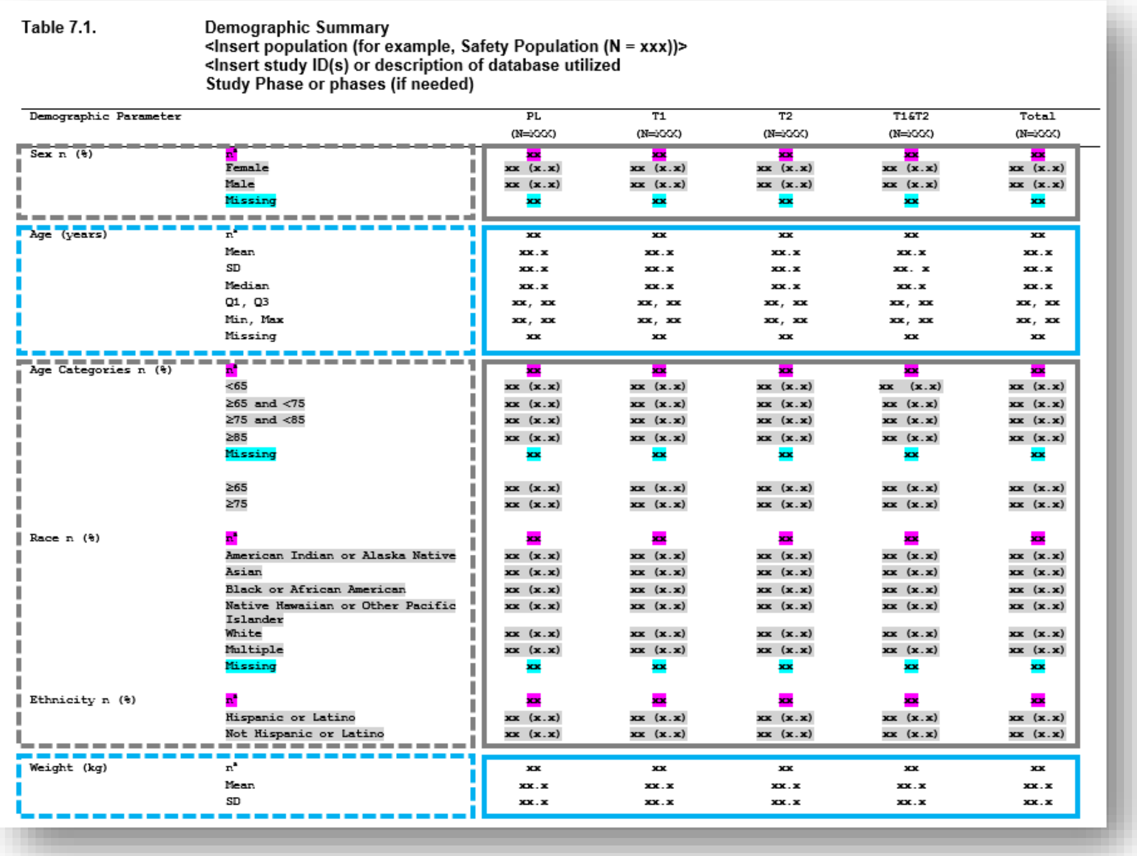


Table 7.1. Demographic Summary
<Insert population (for example, Safety Population (N = xxx))>
<Insert study ID(s) or description of database utilized
Study Phase or phases (if needed)

Demographic Parameter		PL (N=XXX)	T1 (N=XXX)	T2 (N=XXX)	T1&T2 (N=XXX)	Total (N=XXX)
Sex n (%)	n^a	xx	xx	xx	xx	xx
	Female	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Male	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Missing	xx	xx	xx	xx	xx
Age (years)	n^a	xx	xx	xx	xx	xx
	Mean	xx.x	xx.x	xx.x	xx.x	xx.x
	SD	xx.x	xx.x	xx.x	xx. x	xx.x
	Median	xx.x	xx.x	xx.x	xx.x	xx.x
	Q1, Q3	xx, xx	xx, xx	xx, xx	xx, xx	xx, xx
	Min, Max	xx, xx	xx, xx	xx, xx	xx, xx	xx, xx
	Missing	xx	xx	xx	xx	xx
Age Categories n (%)	n^a	xx	xx	xx	xx	xx
	<65	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	≥65 and <75	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	≥75 and <85	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	≥85	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Missing	xx	xx	xx	xx	xx
	≥65	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	≥75	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
Race n (%)	n^a	xx	xx	xx	xx	xx
	American Indian or Alaska Native	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Asian	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Black or African American	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Native Hawaiian or Other Pacific Islander	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	White	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Multiple	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Missing	xx	xx	xx	xx	xx
Ethnicity n (%)	n^a	xx	xx	xx	xx	xx
	Hispanic or Latino	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Not Hispanic or Latino	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
Weight (kg)	n^a	xx	xx	xx	xx	xx
	Mean	xx.x	xx.x	xx.x	xx.x	xx.x
	SD	xx.x	xx.x	xx.x	xx.x	xx.x

Why'd we make Tplyr?

1. Summary tables are highly repetitive, so code should be highly reusable

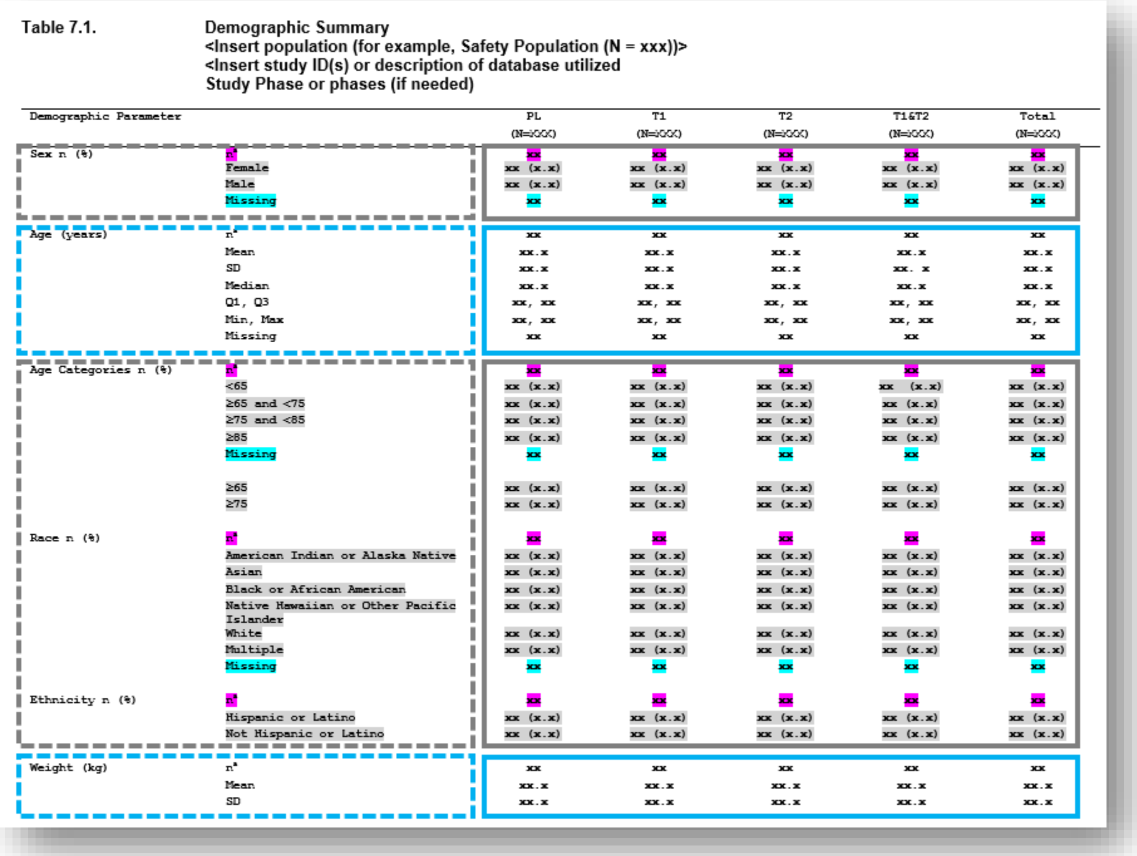


Table 7.1. Demographic Summary
<Insert population (for example, Safety Population (N = xxx))>
<Insert study ID(s) or description of database utilized
Study Phase or phases (if needed)

Demographic Parameter		PL (N=XXX)	T1 (N=XXX)	T2 (N=XXX)	T1&T2 (N=XXX)	Total (N=XXX)
Sex n (%)	n[a]	xx	xx	xx	xx	xx
	Female	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Male	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Missing	xx	xx	xx	xx	xx
Age (years)	n[a]	xx	xx	xx	xx	xx
	Mean	xx.x	xx.x	xx.x	xx.x	xx.x
	SD	xx.x	xx.x	xx.x	xx. x	xx.x
	Median	xx.x	xx.x	xx.x	xx.x	xx.x
	Q1, Q3	xx, xx	xx, xx	xx, xx	xx, xx	xx, xx
	Min, Max	xx, xx	xx, xx	xx, xx	xx, xx	xx, xx
	Missing	xx	xx	xx	xx	xx
Age Categories n (%)	n[a]	xx	xx	xx	xx	xx
	<65	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	≥65 and <75	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	≥75 and <85	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	≥85	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Missing	xx	xx	xx	xx	xx
	≥65	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	≥75	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
Race n (%)	n[a]	xx	xx	xx	xx	xx
	American Indian or Alaska Native	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Asian	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Black or African American	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Native Hawaiian or Other Pacific Islander	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	White	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Multiple	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Missing	xx	xx	xx	xx	xx
Ethnicity n (%)	n[a]	xx	xx	xx	xx	xx
	Hispanic or Latino	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Not Hispanic or Latino	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
Weight (kg)	n[a]	xx	xx	xx	xx	xx
	Mean	xx.x	xx.x	xx.x	xx.x	xx.x
	SD	xx.x	xx.x	xx.x	xx.x	xx.x

Why'd we make Tplyr?

1. Summary tables are highly repetitive, so code should be highly reusable
2. Text formatting of numbers can be very tedious

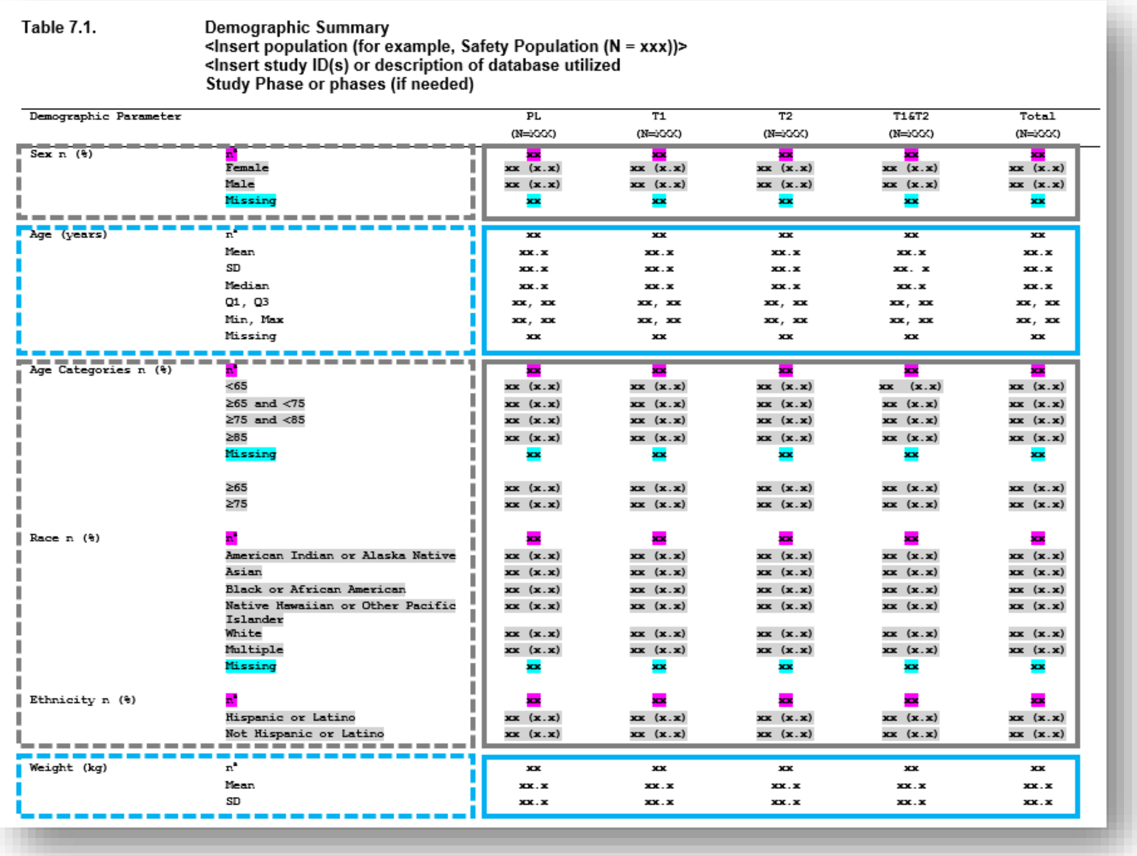


Table 7.1. Demographic Summary
<Insert population (for example, Safety Population (N = xxx))>
<Insert study ID(s) or description of database utilized
Study Phase or phases (if needed)

Demographic Parameter		PL	T1	T2	T1&T2	Total
		(N=XXX)	(N=XXX)	(N=XXX)	(N=XXX)	(N=XXX)
Sex n (%)	n^a	xx	xx	xx	xx	xx
	Female	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Male	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Missing	xx	xx	xx	xx	xx
Age (years)	n^a	xx	xx	xx	xx	xx
	Mean	xx.x	xx.x	xx.x	xx.x	xx.x
	SD	xx.x	xx.x	xx.x	xx. x	xx.x
	Median	xx.x	xx.x	xx.x	xx.x	xx.x
	Q1, Q3	xx, xx	xx, xx	xx, xx	xx, xx	xx, xx
	Min, Max	xx, xx	xx, xx	xx, xx	xx, xx	xx, xx
	Missing	xx	xx	xx	xx	xx
Age Categories n (%)	n^a	xx	xx	xx	xx	xx
	<65	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	≥65 and <75	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	≥75 and <85	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	≥85	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Missing	xx	xx	xx	xx	xx
	≥65	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	≥75	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
Race n (%)	n^a	xx	xx	xx	xx	xx
	American Indian or Alaska Native	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Asian	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Black or African American	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Native Hawaiian or Other Pacific Islander	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	White	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Multiple	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Missing	xx	xx	xx	xx	xx
Ethnicity n (%)	n^a	xx	xx	xx	xx	xx
	Hispanic or Latino	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Not Hispanic or Latino	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
Weight (kg)	n^a	xx	xx	xx	xx	xx
	Mean	xx.x	xx.x	xx.x	xx.x	xx.x
	SD	xx.x	xx.x	xx.x	xx.x	xx.x

Why'd we make Tplyr?

1. Summary tables are highly repetitive, so code should be highly reusable
2. Text formatting of numbers can be very tedious
3. Bridging into interactive tables requires metadata for traceability

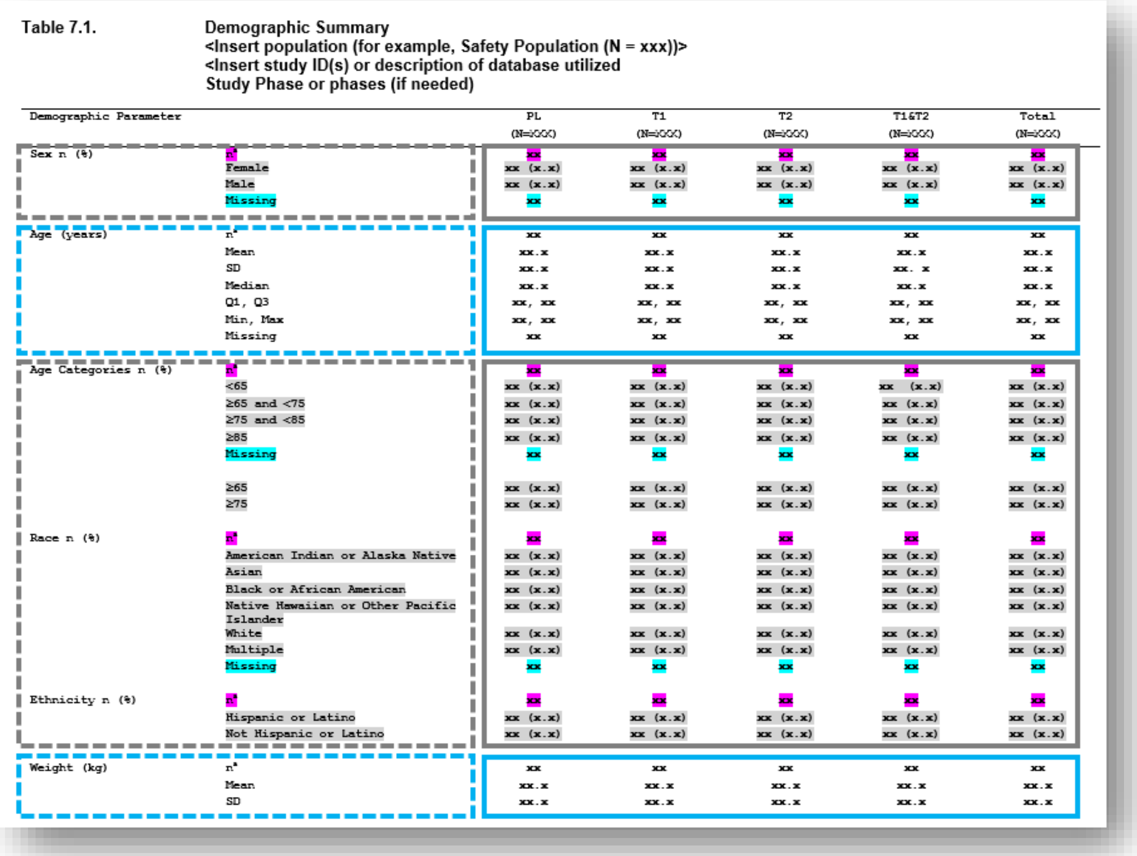


Table 7.1. Demographic Summary
<Insert population (for example, Safety Population (N = xxx))>
<Insert study ID(s) or description of database utilized
Study Phase or phases (if needed)

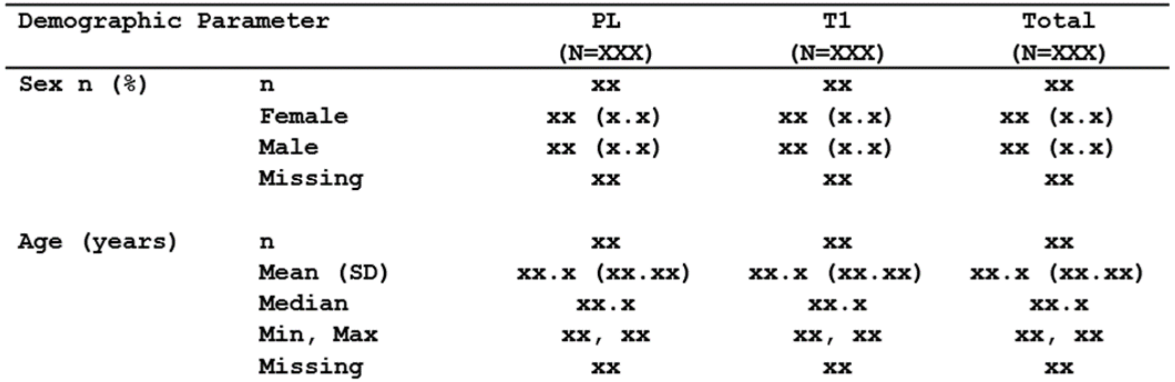
Demographic Parameter		PL (N=XXX)	T1 (N=XXX)	T2 (N=XXX)	T1&T2 (N=XXX)	Total (N=XXX)
Sex n (%)	n[a]	xx	xx	xx	xx	xx
	Female	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Male	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Missing	xx	xx	xx	xx	xx
Age (years)	n[a]	xx	xx	xx	xx	xx
	Mean	xx.x	xx.x	xx.x	xx.x	xx.x
	SD	xx.x	xx.x	xx.x	xx. x	xx.x
	Median	xx.x	xx.x	xx.x	xx.x	xx.x
	Q1, Q3	xx, xx	xx, xx	xx, xx	xx, xx	xx, xx
	Min, Max	xx, xx	xx, xx	xx, xx	xx, xx	xx, xx
	Missing	xx	xx	xx	xx	xx
Age Categories n (%)	n[a]	xx	xx	xx	xx	xx
	<65	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	≥65 and <75	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	≥75 and <85	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	≥85	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Missing	xx	xx	xx	xx	xx
	≥65	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	≥75	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
Race n (%)	n[a]	xx	xx	xx	xx	xx
	American Indian or Alaska Native	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Asian	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Black or African American	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Native Hawaiian or Other Pacific Islander	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	White	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Multiple	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Missing	xx	xx	xx	xx	xx
Ethnicity n (%)	n[a]	xx	xx	xx	xx	xx
	Hispanic or Latino	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
	Not Hispanic or Latino	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)	xx (x.x)
Weight (kg)	n[a]	xx	xx	xx	xx	xx
	Mean	xx.x	xx.x	xx.x	xx.x	xx.x
	SD	xx.x	xx.x	xx.x	xx.x	xx.x

Newest Features in Tplyr 1.0.0

- Traceability metadata framework
- Re-usable layer templates
- External precision data
- Descriptive statistics as columns
- Apply string formatting outside of Tplyr tables
- Hyphenated string wrapping

Tplyr Basics

Introduction to Tplyr



Demographic Parameter		PL (N=XXX)	T1 (N=XXX)	Total (N=XXX)
Sex n (%)	n	xx	xx	xx
	Female	xx (x.x)	xx (x.x)	xx (x.x)
	Male	xx (x.x)	xx (x.x)	xx (x.x)
	Missing	xx	xx	xx
Age (years)	n	xx	xx	xx
	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
	Missing	xx	xx	xx

- Each layer can have 1 of 2 basic “flavors”
  - Counts
  - Descriptive statistics
- Within layers add additional “filling” such as:
  - Distinct counts
  - Statistics to calculate

Table Components

- The tplyr_table() object is the conceptual “table” that contains all the logic necessary to construct and display the data, sort of like a specification.

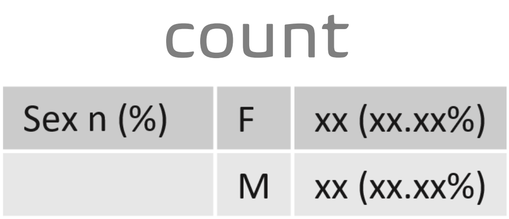
- Tplyr tables are made up of one or more tplyr_layer() objects. Layer objects contain an instruction for a summary to be performed and are stacked in order to create the table object.

Creating Table and Layer Objects

```
tplyr_table(target, treat_var, where=TRUE, cols=vars())
```

Creating Table and Layer Objects

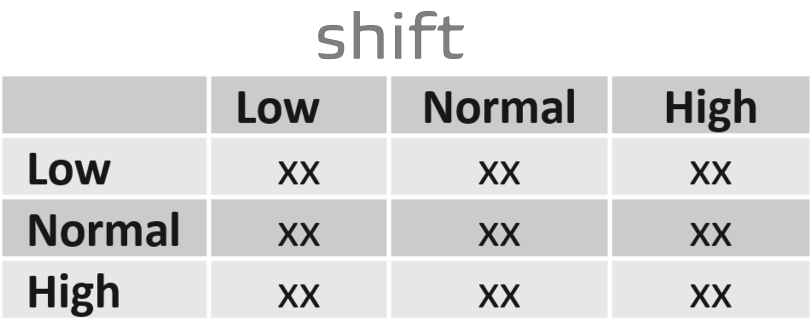
`group_count()`



count

Sex n (%)	F	xx (xx.xx%)
	M	xx (xx.xx%)

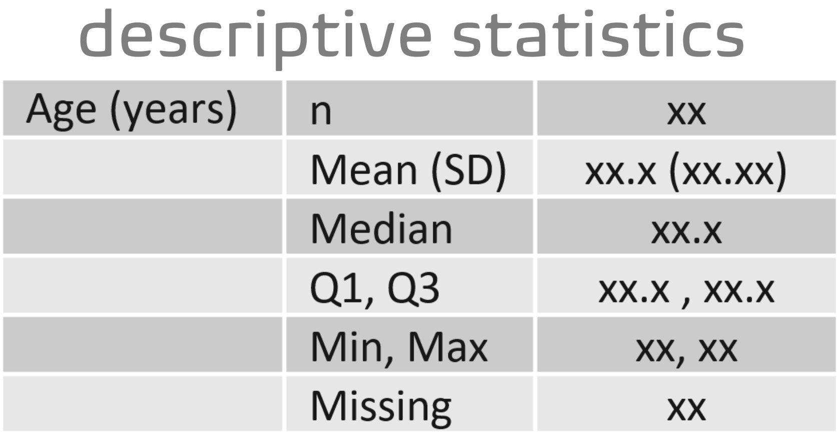
`group_shift()`



shift

	Low	Normal	High
Low	xx	xx	xx
Normal	xx	xx	xx
High	xx	xx	xx

`group_desc()`



descriptive statistics

Age (years)	n	xx
	Mean (SD)	xx.x (xx.xx)
	Median	xx.x
	Q1, Q3	xx.x , xx.x
	Min, Max	xx, xx
	Missing	xx

```
group_<type> (parent, target_var, by=vars(), where=TRUE, ...)
```

Creating Table and Layer Objects

```
add_layer(parent, layer, name=NULL)

add_layers(parent, ...)
```

Creating Table and Layer Objects

```
t <- tplyr_table(adsl, TRT01P, where = SAFFL == "Y") %>%
   add_layer(group_count(AGEGR1, by = "Age categories in n (%)")) %>%
   add_layer(group_desc(AGE, by = "Age (years)"))
t
```

```
*** tplyr_table ***
Target (data.frame):
        Name:  adsl
        Rows:  30
        Columns:  32
treat_var variable (quosure)
        TRT01P
header_n:  header groups
treat_grps groupings (list)
Table Columns (cols):
where: == SAFFL Y
Number of layer(s): 2
layer_output: 0
```

Processing the Data

```
t <- tplyr_table(adsl, TRT01P) %>%
   add_layer(
      group_count(AGEGR1, by = "Age categories n (%)")
   )
```

```
## *** tplyr_table ***
## Target (data.frame):
##  Name:  adsl
##  Rows:  30
##  Columns:  32
## treat_var variable (quosure)
##  TRT01P
## header_n:  header groups
## treat_grps groupings (list)
## Table Columns (cols):
## where: TRUE
## Number of layer(s): 1
## layer_output: 0
```

```
t %>%
  build()
```

```
row_label1            row_label2  var1_Miracle High  var1_Miracle Low  var1_Placebo  ord_layer_index  ord_layer_1  ord_layer_2
                                  Dose               Dose
Age categories n (%)  <65         5 ( 45.5%)         1 ( 10.0%)        2 ( 22.2%)    1                1            1
Age categories n (%)  >80         2 ( 18.2%)         5 ( 50.0%)        4 ( 44.4%)    1                1            2
Age categories n (%)  65-80       4 ( 36.4%)         4 ( 40.0%)        3 ( 33.3%)    1                1            3
```

Breakout 1

Summarize Some Data with Tplyr

Configuring Tplyr Tables

General Table Level Settings

```
tplyr_table(target, treat_var, where=TRUE, cols=vars())
```

```
t <- tplyr_table(adsl, TRT01P, where = SAFFL == "Y", cols = SEX) %>%
   add_layer(
      group_count(RACE, by = "Race")
   ) %>%
   add_layer(
      group_desc(AGE, by = "Age (Years)")
   )

## *** tplyr_table ***
## Target (data.frame):
##  Name:  adsl
##  Rows:  30
##  Columns:  32
## treat_var variable (quosure)
##  TRT01P
## header_n:  header groups
## treat_grps groupings (list)
## Table Columns (cols):
##  SEX
## where: == SAFFL Y
## Number of layer(s): 2
## layer_output: 0
```

General Table Level Settings

```
t <- tplyr_table(adsl, TRT01P, where = SAFFL == "Y", cols = SEX) %>%
   add_layer(
      group_count(RACE, by = "Race")
   ) %>%
   add_layer(
      group_desc(AGE, by = "Age (Years)")
   )
t_df <- t %>%
  build()
```

```
row_label1   row_label2                    var1_Miracle  var1_Miracle  var1_Miracle  var1_Miracle  var1_Placebo_F  var1_Placebo_M
                                           High_Dose_F   High Dose_M   Low_Dose_ F   Low Dose_M
Race         AMERICAN INDIAN OR ALASKA NATIVE 0 (  0.0%)   1 ( 14.3%)    0 (  0.0%)    0 (  0.0%)    0 (  0.0%)      0 (  0.0%)
Race         BLACK OR AFRICAN AMERICAN     0 (  0.0%)    0 (  0.0%)    0 (  0.0%)    0 (  0.0%)    1 ( 25.0%)      0 (  0.0%)
Race         WHITE                         4 (100.0%)    6 ( 85.7%)    5 (100.0%)    5 (100.0%)    3 ( 75.0%)      5 (100.0%)
Age (Years)  n                             4             7             5             5             4               5
Age (Years)  Mean (SD)                     73.0 ( 9.90)  64.9 ( 9.75)  76.6 (10.43)  75.2 ( 6.98)  76.2 ( 9.81)    73.4 (11.97)
Age (Years)  Median                        75.5          61.0          83.0          77.0          79.5            76.0
Age (Years)  Q1, Q3                        70.2, 78.2    58.0, 70.0    75.0, 83.0    70.0, 81.0    74.0, 81.8      69.0, 82.0
Age (Years)  Min, Max                      59, 82        55, 82        59, 83        66, 82        62, 84          55, 85
Age (Years)  Missing                       0             0             0             0             0               0
```

Adding Treatment Groups

```
add_treat_grps(table, ...)
add_total_group(table, group_name="Total")
```

```
t <- tplyr_table(adsl, TRT01P) %>%
   add_treat_grps("Treated" = c("Miracle High Dose", "Miracle Low Dose")) %>%
   add_total_group() %>%
   add_layer(
      group_desc(AGE, by = "Age (Years)")
   )
t_df <- t %>%
  build()
```

```
row_label1   row_label2  var1_Miracle  var1_Miracle  var1_Placebo  var1_Total    var1_Treated
                         High Dose     Low Dose
Age (Years)  n           11            10            9             30            21
Age (Years)  Mean (SD)   67.8 (10.17)  75.9 ( 8.40)  74.7 (10.49)  72.6 (10.08)  71.7 (10.03)
Age (Years)  Median      67.0          79.0          78.0          75.5          74.0
Age (Years)  Q1, Q3      58.5, 75.5    71.2, 82.8    69.0, 82.0    63.0, 82.0    61.0, 82.0
Age (Years)  Min, Max    55, 82        59, 83        55, 85        55, 85        55, 83
Age (Years)  Missing     0             0             0             0             0
```

Adding a Population Dataset

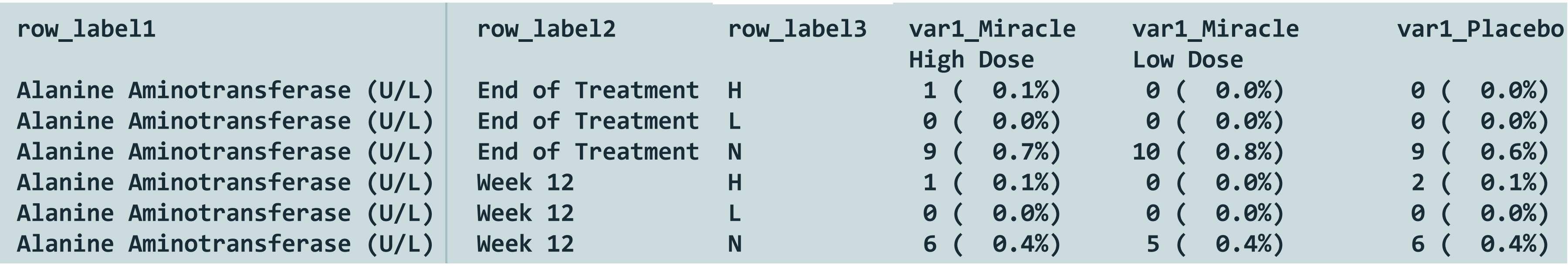
```
set_pop_data(table, pop_data)
set_pop_treat_var(table, pop_treat_var)
set_pop_where(obj, where)
```

```
t <- tplyr_table(adae, TRTA, where = AEREL != "NONE") %>%
  set_pop_data(adsl) %>%
  set_pop_treat_var(TRT01A) %>%
  set_pop_where(TRUE) %>%
  add_layer(
    group_count(AEDECOD)
  )
t_df <- t %>%
  build()
```

- Target dataset does not contain all subjects in study population: `set_pop_data()`
- Population treatment variable is different than treat_var: `set_pop_treat_var()`
- Population subset is different than tplyr_table subset: `set_pop_where()`

Using Grouping Variables

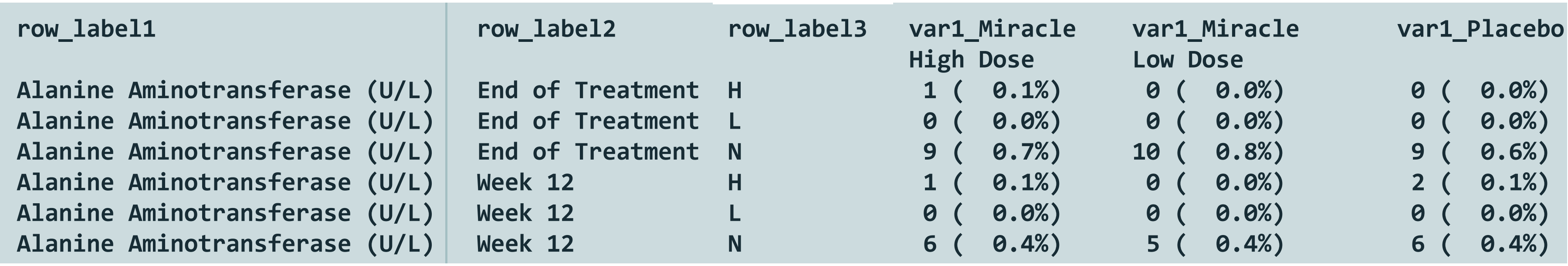
```
t <- tplyr_table(adlb, TRTA, where = SAFFL == "Y" & AVISITN > 0) %>%
  add_layer(
    group_count(ANRIND, by = vars(PARAM,AVISIT))
  )
t_df <- t %>%
  build()
```



```
row_label1                         row_label2          row_label3    var1_Miracle    var1_Miracle       var1_Placebo
                                                                     High Dose       Low Dose
Alanine Aminotransferase (U/L)     End of Treatment    H              1 (  0.1%)      0 (  0.0%)         0 (  0.0%)
Alanine Aminotransferase (U/L)     End of Treatment    L              0 (  0.0%)      0 (  0.0%)         0 (  0.0%)
Alanine Aminotransferase (U/L)     End of Treatment    N              9 (  0.7%)     10 (  0.8%)         9 (  0.6%)
Alanine Aminotransferase (U/L)     Week 12             H              1 (  0.1%)      0 (  0.0%)         2 (  0.1%)
Alanine Aminotransferase (U/L)     Week 12             L              0 (  0.0%)      0 (  0.0%)         0 (  0.0%)
Alanine Aminotransferase (U/L)     Week 12             N              6 (  0.4%)      5 (  0.4%)         6 (  0.4%)
```

Using Grouping Variables

```
t <- tplyr_table(adlb, TRTA, where = SAFFL == "Y" & AVISITN > 0) %>%
  add_layer(
    group_count(ANRIND, by = vars(PARAM,AVISIT))
  )
t_df <- t %>%
  build()
```

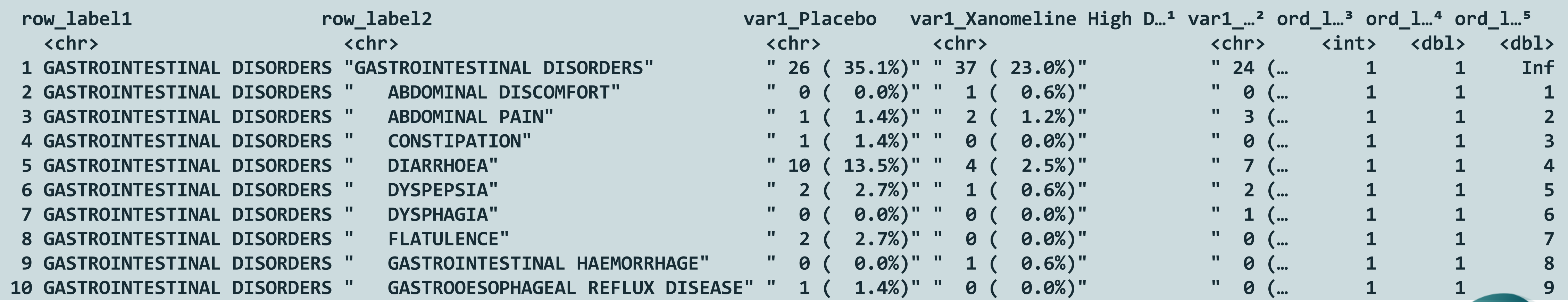


```
row_label1                   row_label2        row_label3  var1_Miracle     var1_Miracle      var1_Placebo
                                                           High Dose        Low Dose
Alanine Aminotransferase (U/L)  End of Treatment  H         1 (  0.1%)       0 (  0.0%)        0 (  0.0%)
Alanine Aminotransferase (U/L)  End of Treatment  L         0 (  0.0%)       0 (  0.0%)        0 (  0.0%)
Alanine Aminotransferase (U/L)  End of Treatment  N         9 (  0.7%)      10 (  0.8%)        9 (  0.6%)
Alanine Aminotransferase (U/L)  Week 12           H         1 (  0.1%)       0 (  0.0%)        2 (  0.1%)
Alanine Aminotransferase (U/L)  Week 12           L         0 (  0.0%)       0 (  0.0%)        0 (  0.0%)
Alanine Aminotransferase (U/L)  Week 12           N         6 (  0.4%)       5 (  0.4%)        6 (  0.4%)
```

Using Grouping Variables – Nested Counts

```
t <- tplyr_table(adae, TRTA) %>%
  add_layer(
    group_count(vars(AEBODSYS, AEDECOD))
  )

t_df <- t %>%  build()
```



```
 row_label1                 row_label2                            var1_Placebo   var1_Xanomeline High D…¹ var1_…² ord_l…³ ord_l…⁴ ord_l…⁵
   <chr>                      <chr>                                 <chr>          <chr>                    <chr>       <int>   <dbl>   <dbl>
 1 GASTROINTESTINAL DISORDERS "GASTROINTESTINAL DISORDERS"          " 26 ( 35.1%)" " 37 ( 23.0%)"           " 24 (…       1       1     Inf
 2 GASTROINTESTINAL DISORDERS "   ABDOMINAL DISCOMFORT"             "  0 (  0.0%)" "  1 (  0.6%)"           "  0 (…       1       1       1
 3 GASTROINTESTINAL DISORDERS "   ABDOMINAL PAIN"                   "  1 (  1.4%)" "  2 (  1.2%)"           "  3 (…       1       1       2
 4 GASTROINTESTINAL DISORDERS "   CONSTIPATION"                     "  1 (  1.4%)" "  0 (  0.0%)"           "  0 (…       1       1       3
 5 GASTROINTESTINAL DISORDERS "   DIARRHOEA"                        " 10 ( 13.5%)" "  4 (  2.5%)"           "  7 (…       1       1       4
 6 GASTROINTESTINAL DISORDERS "   DYSPEPSIA"                        "  2 (  2.7%)" "  1 (  0.6%)"           "  2 (…       1       1       5
 7 GASTROINTESTINAL DISORDERS "   DYSPHAGIA"                        "  0 (  0.0%)" "  0 (  0.0%)"           "  1 (…       1       1       6
 8 GASTROINTESTINAL DISORDERS "   FLATULENCE"                       "  2 (  2.7%)" "  0 (  0.0%)"           "  0 (…       1       1       7
 9 GASTROINTESTINAL DISORDERS "   GASTROINTESTINAL HAEMORRHAGE"     "  0 (  0.0%)" "  1 (  0.6%)"           "  0 (…       1       1       8
10 GASTROINTESTINAL DISORDERS "   GASTROOESOPHAGEAL REFLUX DISEASE" "  1 (  1.4%)" "  0 (  0.0%)"           "  0 (…       1       1       9
```

atorus
Your Complete Data Solution

RESTRICTED

Controlling Display with Format Strings - Counts

```
t <- tplyr_table(adae, TRTA) %>%
  add_layer(
    group_count(AEDECOD) %>%
      set_format_strings(
        f_str("xx (xx.x%) [x]", distinct_n, distinct_pct, n)
      ) %>%
      set_distinct_by(USUBJID)
  )
t_df <- t %>%  build()
```

```
 row_label1                      var1_Placebo     `var1_Xanomeline High Dose` `var1_Xanomeline Low Dose` ord_layer_index ord_layer_1
   <chr>                           <chr>            <chr>                        <chr>                                 <int>       <dbl>
 1 ABDOMINAL DISCOMFORT            " 0 ( 0.0%) [0]" " 1 ( 2.2%) [1]"            " 0 ( 0.0%) [0]"                         1           1
 2 ABDOMINAL PAIN                  " 1 ( 3.1%) [1]" " 1 ( 2.2%) [2]"            " 3 ( 6.0%) [3]"                         1           2
 3 APPLICATION SITE BLEEDING       " 0 ( 0.0%) [0]" " 0 ( 0.0%) [0]"            " 1 ( 2.0%) [1]"                         1           3
 4 APPLICATION SITE DERMATITIS     " 5 (15.6%) [9]" " 7 (15.2%) [12]"           " 9 (18.0%) [15]"                        1           4
 5 APPLICATION SITE DESQUAMATION   " 0 ( 0.0%) [0]" " 0 ( 0.0%) [0]"            " 1 ( 2.0%) [1]"                         1           5
 6 APPLICATION SITE DISCHARGE      " 0 ( 0.0%) [0]" " 1 ( 2.2%) [1]"            " 0 ( 0.0%) [0]"                         1           6
 7 APPLICATION SITE DISCOLOURATION " 0 ( 0.0%) [0]" " 0 ( 0.0%) [0]"            " 1 ( 2.0%) [1]"                         1           7
 8 APPLICATION SITE ERYTHEMA       " 3 ( 9.4%) [3]" "15 (32.6%) [23]"           "12 (24.0%) [20]"                        1           8
 9 APPLICATION SITE INDURATION     " 1 ( 3.1%) [1]" " 0 ( 0.0%) [0]"            " 0 ( 0.0%) [0]"                         1           9
10 APPLICATION SITE IRRITATION     " 3 ( 9.4%) [7]" " 9 (19.6%) [16]"           " 9 (18.0%) [18]"                        1          10
```

Controlling Display with Format Strings - Counts

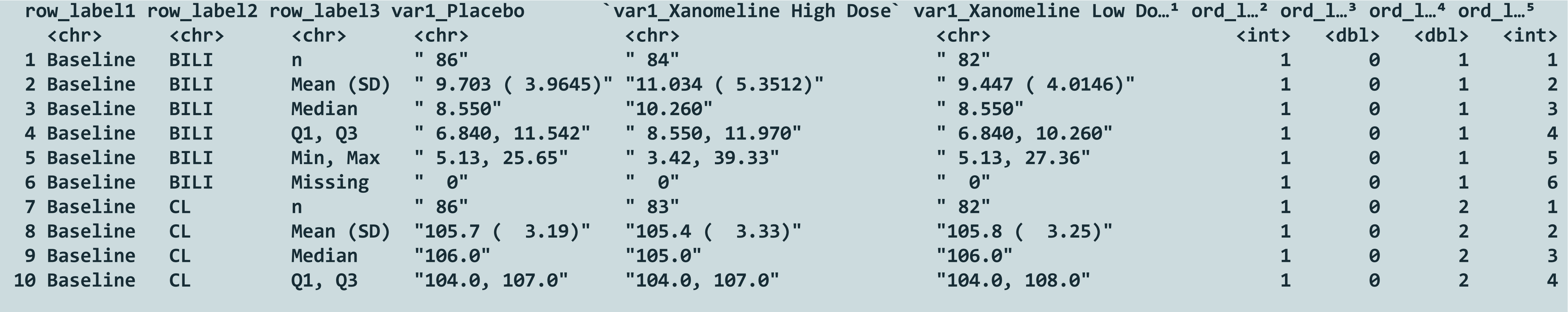
```
t <- tplyr_table(adae, TRTA) %>%
  set_pop_data(adsl) %>%
  set_pop_treat_var(TRT01P) %>%
  add_layer(
    group_count(AEDECOD) %>%
      set_format_strings(
        f_str("xx (xx.x%) [x]", distinct_n, distinct_pct, n)
      ) %>%
      set_distinct_by(USUBJID)
  )
t_df <- t %>%  build()
```

```
 row_label1                      var1_Placebo     `var1_Xanomeline High Dose` `var1_Xanomeline Low Dose` ord_layer_index ord_layer_1
   <chr>                           <chr>            <chr>                       <chr>                                <int>       <dbl>
 1 ABDOMINAL DISCOMFORT            " 0 ( 0.0%) [0]" " 1 ( 1.2%) [1]"            " 0 ( 0.0%) [0]"                         1           1
 2 ABDOMINAL PAIN                  " 1 ( 1.2%) [1]" " 1 ( 1.2%) [2]"            " 3 ( 3.6%) [3]"                         1           2
 3 APPLICATION SITE BLEEDING       " 0 ( 0.0%) [0]" " 0 ( 0.0%) [0]"            " 1 ( 1.2%) [1]"                         1           3
 4 APPLICATION SITE DERMATITIS     " 5 ( 5.8%) [9]" " 7 ( 8.3%) [12]"           " 9 (10.7%) [15]"                        1           4
 5 APPLICATION SITE DESQUAMATION   " 0 ( 0.0%) [0]" " 0 ( 0.0%) [0]"            " 1 ( 1.2%) [1]"                         1           5
 6 APPLICATION SITE DISCHARGE      " 0 ( 0.0%) [0]" " 1 ( 1.2%) [1]"            " 0 ( 0.0%) [0]"                         1           6
 7 APPLICATION SITE DISCOLOURATION " 0 ( 0.0%) [0]" " 0 ( 0.0%) [0]"            " 1 ( 1.2%) [1]"                         1           7
 8 APPLICATION SITE ERYTHEMA       " 3 ( 3.5%) [3]" "15 (17.9%) [23]"           "12 (14.3%) [20]"                        1           8
 9 APPLICATION SITE INDURATION     " 1 ( 1.2%) [1]" " 0 ( 0.0%) [0]"            " 0 ( 0.0%) [0]"                         1           9
10 APPLICATION SITE IRRITATION     " 3 ( 3.5%) [7]" " 9 (10.7%) [16]"           " 9 (10.7%) [18]"                        1          10
```

Controlling Display with Format Strings - Desc

```
t <- tplyr_table(adlb, TRTA) %>%
   add_layer(
    group_desc(AVAL, by = vars(AVISIT, PARAMCD))
  )

t_df <- t %>% build()
```



```
 row_label1 row_label2 row_label3 var1_Placebo       `var1_Xanomeline High Dose` var1_Xanomeline Low Do…¹ ord_l…² ord_l…³ ord_l…⁴ ord_l…⁵
   <chr>      <chr>      <chr>      <chr>              <chr>                       <chr>                      <int>   <dbl>   <dbl>   <int>
 1 Baseline   BILI       n          " 86"              " 84"                       " 82"                          1       0       1       1
 2 Baseline   BILI       Mean (SD)  " 9.703 ( 3.9645)" "11.034 ( 5.3512)"          " 9.447 ( 4.0146)"             1       0       1       2
 3 Baseline   BILI       Median     " 8.550"           "10.260"                    " 8.550"                       1       0       1       3
 4 Baseline   BILI       Q1, Q3     " 6.840, 11.542"   " 8.550, 11.970"            " 6.840, 10.260"               1       0       1       4
 5 Baseline   BILI       Min, Max   " 5.13, 25.65"     " 3.42, 39.33"              " 5.13, 27.36"                 1       0       1       5
 6 Baseline   BILI       Missing    "  0"              "  0"                       "  0"                          1       0       1       6
 7 Baseline   CL         n          " 86"              " 83"                       " 82"                          1       0       2       1
 8 Baseline   CL         Mean (SD)  "105.7 (  3.19)"   "105.4 (  3.33)"            "105.8 (  3.25)"               1       0       2       2
 9 Baseline   CL         Median     "106.0"            "105.0"                     "106.0"                        1       0       2       3
10 Baseline   CL         Q1, Q3     "104.0, 107.0"     "104.0, 107.0"              "104.0, 108.0"                 1       0       2       4
```

Controlling Display with Format Strings - Desc

```
t <- tplyr_table(adlb, TRTA) %>%
   add_layer(
    group_desc(AVAL, by = vars(AVISIT, PARAMCD)) %>%
      set_format_strings(
        'Mean' = f_str('xx.x', mean),
        'SD' = f_str('xx.xx', sd),
        'Q1, Median, Q3' = f_str("xx.x, xx.x, xx.x", q1, median, q3)
      )
  )

t_df <- t %>% build()
```

```
 row_label1 row_label2 row_label3     var1_Placebo          `var1_Xanomeline High Dose` var1_Xanomeline…¹ ord_l…² ord_l…³ ord_l…⁴ ord_l…⁵
   <chr>      <chr>      <chr>          <chr>                 <chr>                       <chr>               <int>   <dbl>   <dbl>   <int>
 1 Baseline   BILI       Mean           " 9.7"                "11.0"                      " 9.4"                  1       0       1       1
 2 Baseline   BILI       SD             " 3.96"               " 5.35"                     " 4.01"                 1       0       1       2
 3 Baseline   BILI       Q1, Median, Q3 " 6.8,  8.5, 11.5"    " 8.5, 10.3, 12.0"          " 6.8,  8.5, 10.…       1       0       1       3
 4 Baseline   CL         Mean           "105.7"               "105.4"                     "105.8"                 1       0       2       1
 5 Baseline   CL         SD             " 3.19"               " 3.33"                     " 3.25"                 1       0       2       2
 6 Baseline   CL         Q1, Median, Q3 "104.0, 106.0, 107.0" "104.0, 105.0, 107.0"       "104.0, 106.0, 1…       1       0       2       3
 7 Baseline   K          Mean           " 4.3"                " 4.3"                      " 4.3"                  1       0       3       1
 8 Baseline   K          SD             " 0.43"               " 0.41"                     " 0.34"                 1       0       3       2
 9 Baseline   K          Q1, Median, Q3 " 4.0,  4.3,  4.5"    " 4.0,  4.3,  4.6"          " 4.1,  4.3,  4.…       1       0       3       3
10 Baseline   SODIUM     Mean           "140.3"               "140.0"                     "140.0"                 1       0       4       1
```

Valid Formatting Variables

Layer	Variable
Count	n
	pct
	total
	distinct_n
	distinct_pct
	distinct_total
Shift	n
	pct
	total

- Distinct values use population data for denominators

Valid Formatting Variables

Layer	Variable
Descriptive statistics	n
	mean
	sd
	median
	var
	min
	max
	iqr
	q1
	q3
	missing

- Custom summaries can be added using the function **set_custom_summaries()**
- **Function definitions in the descriptive statistics vignette**

Breakout 2

Configure Some Tables

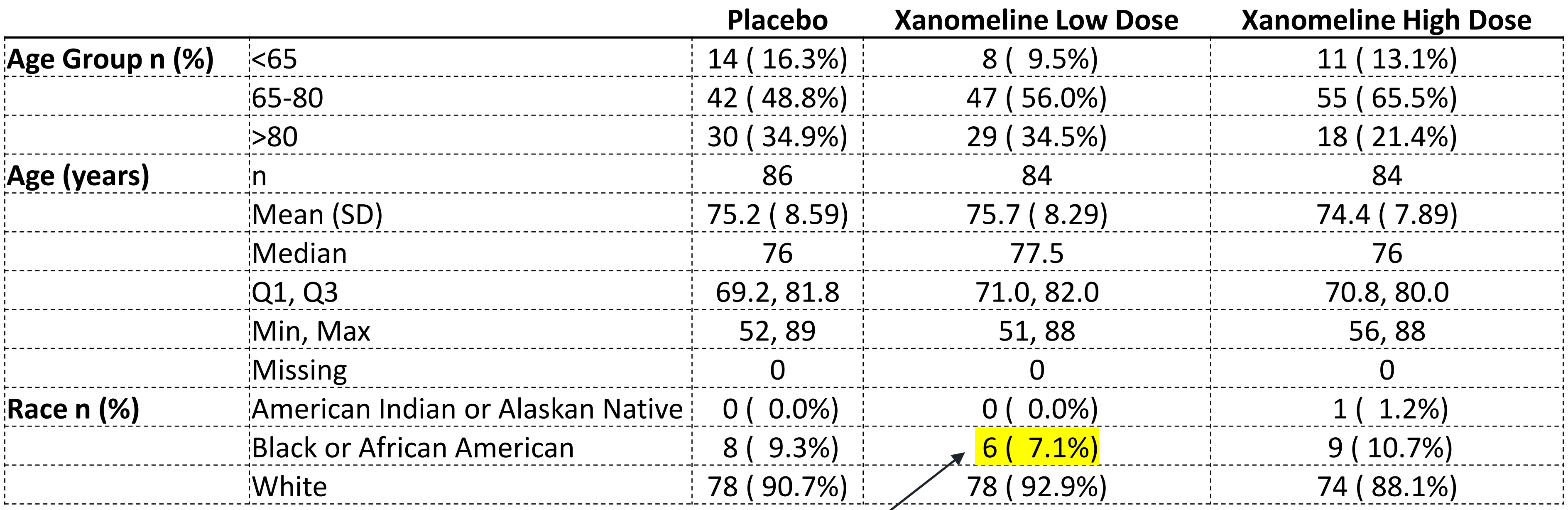
atorus

Your Complete Data Solution

Tplyr Metadata Part 1

Structure and Concept

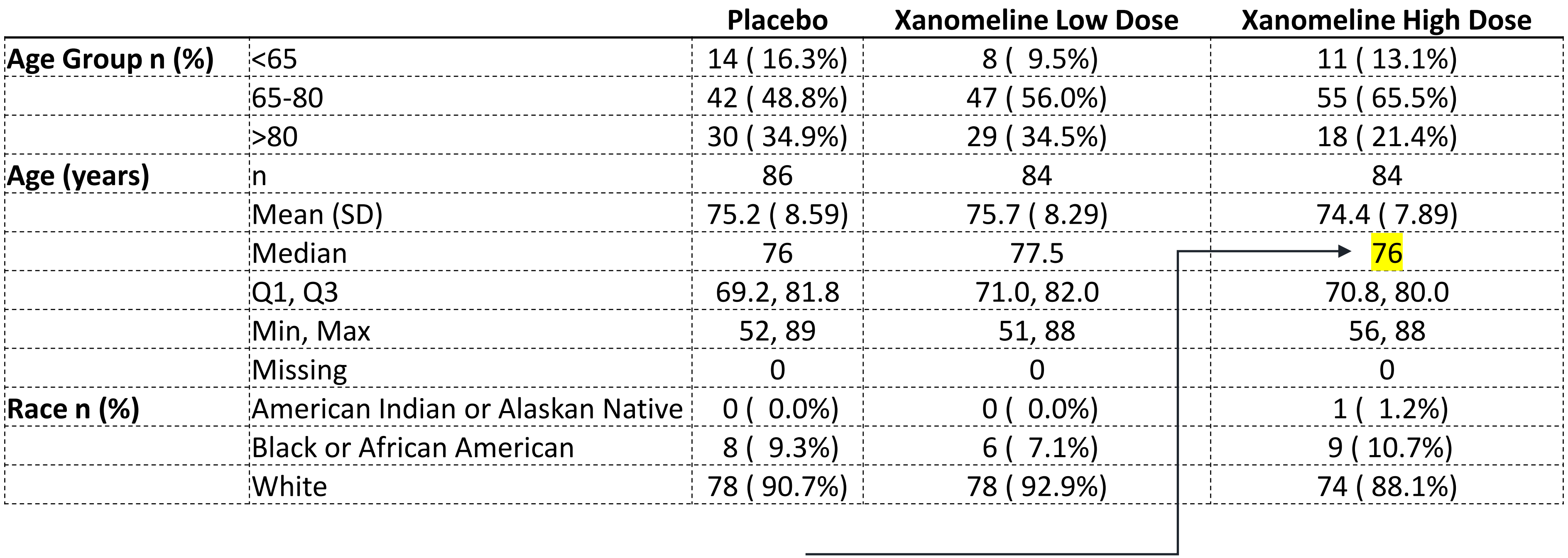
Tplyr's Metadata Concept



		Placebo	Xanomeline Low Dose	Xanomeline High Dose
Age Group n (%)	<65	14 (16.3%)	8 (9.5%)	11 (13.1%)
	65-80	42 (48.8%)	47 (56.0%)	55 (65.5%)
	>80	30 (34.9%)	29 (34.5%)	18 (21.4%)
Age (years)	n	86	84	84
	Mean (SD)	75.2 (8.59)	75.7 (8.29)	74.4 (7.89)
	Median	76	77.5	76
	Q1, Q3	69.2, 81.8	71.0, 82.0	70.8, 80.0
	Min, Max	52, 89	51, 88	56, 88
	Missing	0	0	0
Race n (%)	American Indian or Alaskan Native	0 (0.0%)	0 (0.0%)	1 (1.2%)
	Black or African American	8 (9.3%)	6 (7.1%)	9 (10.7%)
	White	78 (90.7%)	78 (92.9%)	74 (88.1%)

What data produced this result?

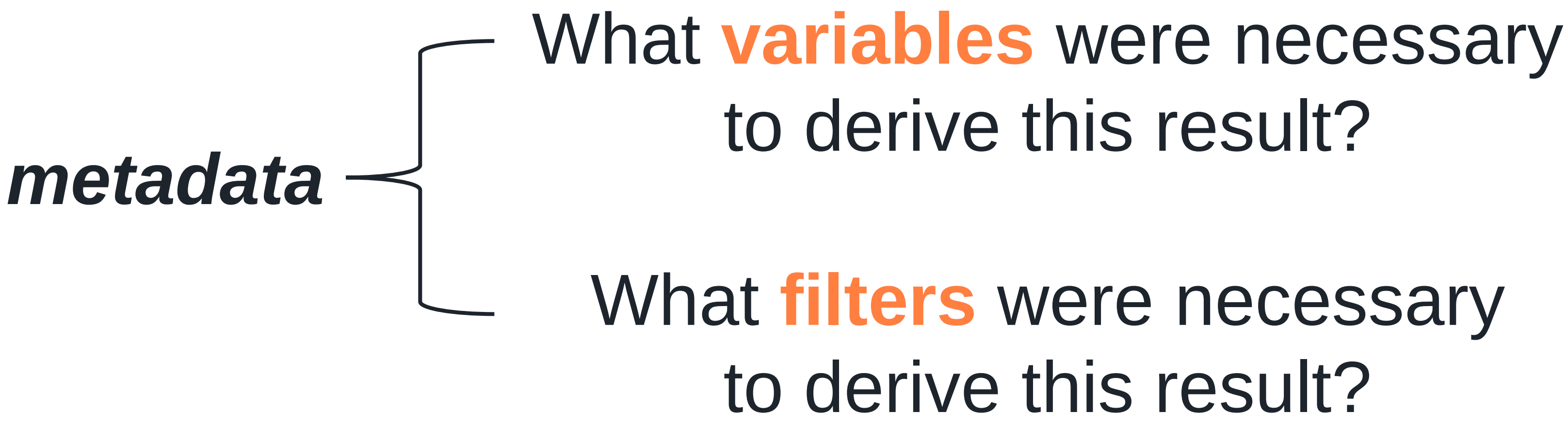
Tplyr's Metadata Concept



		Placebo	Xanomeline Low Dose	Xanomeline High Dose
Age Group n (%)	<65	14 (16.3%)	8 (9.5%)	11 (13.1%)
	65-80	42 (48.8%)	47 (56.0%)	55 (65.5%)
	>80	30 (34.9%)	29 (34.5%)	18 (21.4%)
Age (years)	n	86	84	84
	Mean (SD)	75.2 (8.59)	75.7 (8.29)	74.4 (7.89)
	Median	76	77.5	76
	Q1, Q3	69.2, 81.8	71.0, 82.0	70.8, 80.0
	Min, Max	52, 89	51, 88	56, 88
	Missing	0	0	0
Race n (%)	American Indian or Alaskan Native	0 (0.0%)	0 (0.0%)	1 (1.2%)
	Black or African American	8 (9.3%)	6 (7.1%)	9 (10.7%)
	White	78 (90.7%)	78 (92.9%)	74 (88.1%)

What data produced this result?

What Do We Mean By "Metadata"

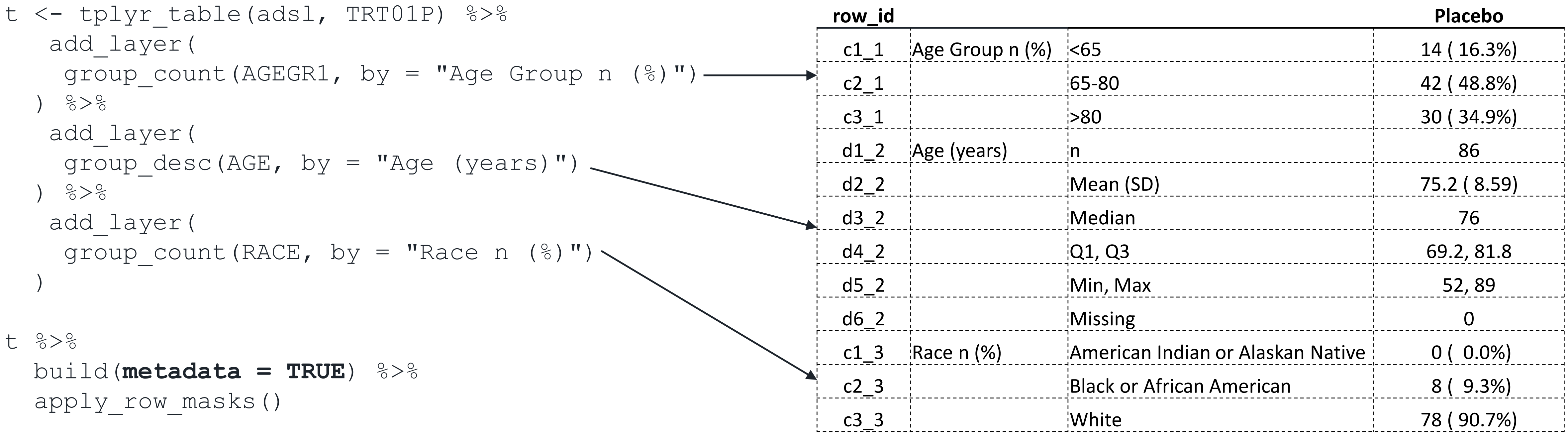


metadata

What **variables** were necessary to derive this result?

What **filters** were necessary to derive this result?

Building Metadata

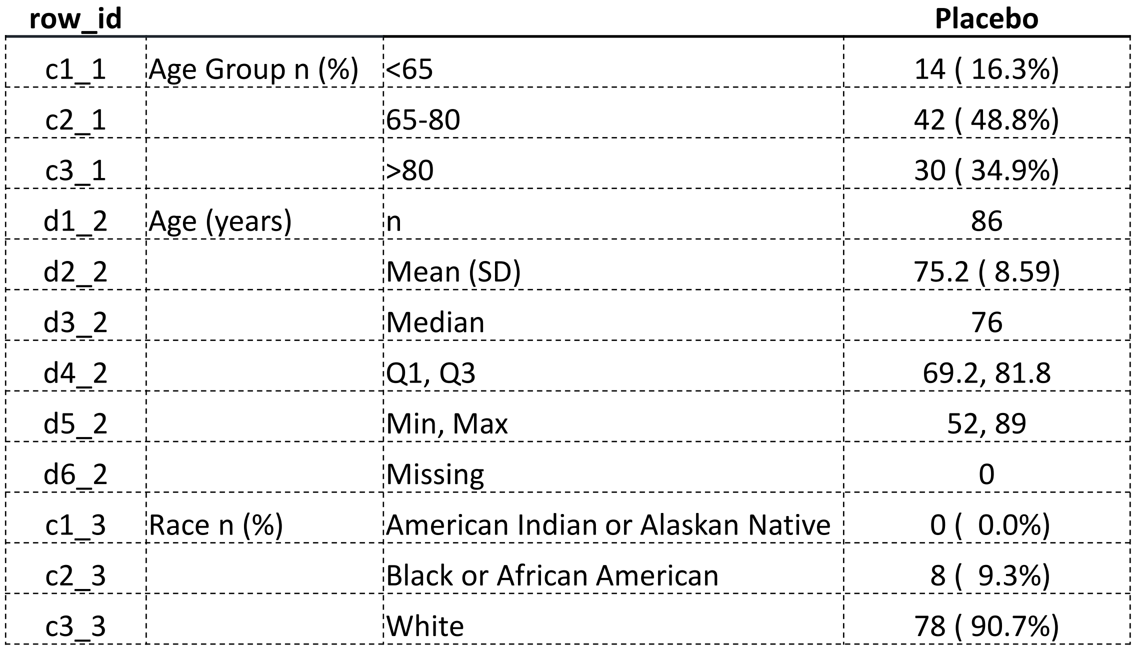


```
t <- tplyr_table(adsl, TRT01P) %>%
   add_layer(
    group_count(AGEGR1, by = "Age Group n (%)")
  ) %>%
   add_layer(
    group_desc(AGE, by = "Age (years)")
  ) %>%
   add_layer(
    group_count(RACE, by = "Race n (%)")
  )

t %>%
  build(metadata = TRUE) %>%
  apply_row_masks()
```

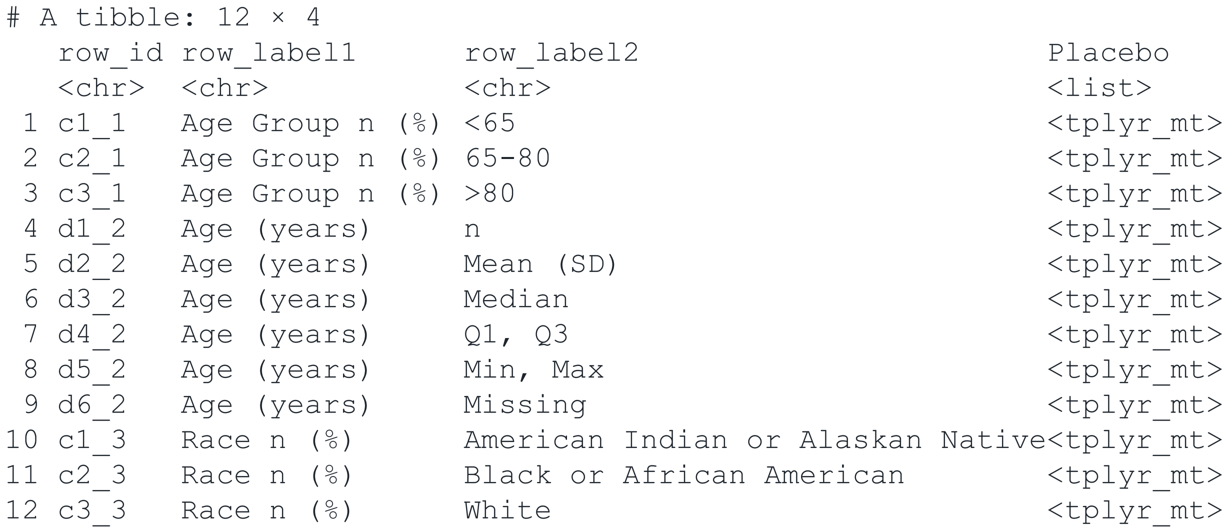
row_id			Placebo
c1_1	Age Group n (%)	<65	14 (16.3%)
c2_1		65-80	42 (48.8%)
c3_1		>80	30 (34.9%)
d1_2	Age (years)	n	86
d2_2		Mean (SD)	75.2 (8.59)
d3_2		Median	76
d4_2		Q1, Q3	69.2, 81.8
d5_2		Min, Max	52, 89
d6_2		Missing	0
c1_3	Race n (%)	American Indian or Alaskan Native	0 (0.0%)
c2_3		Black or African American	8 (9.3%)
c3_3		White	78 (90.7%)

Building Metadata



row_id			Placebo
c1_1	Age Group n (%)	<65	14 (16.3%)
c2_1		65-80	42 (48.8%)
c3_1		>80	30 (34.9%)
d1_2	Age (years)	n	86
d2_2		Mean (SD)	75.2 (8.59)
d3_2		Median	76
d4_2		Q1, Q3	69.2, 81.8
d5_2		Min, Max	52, 89
d6_2		Missing	0
c1_3	Race n (%)	American Indian or Alaskan Native	0 (0.0%)
c2_3		Black or African American	8 (9.3%)
c3_3		White	78 (90.7%)

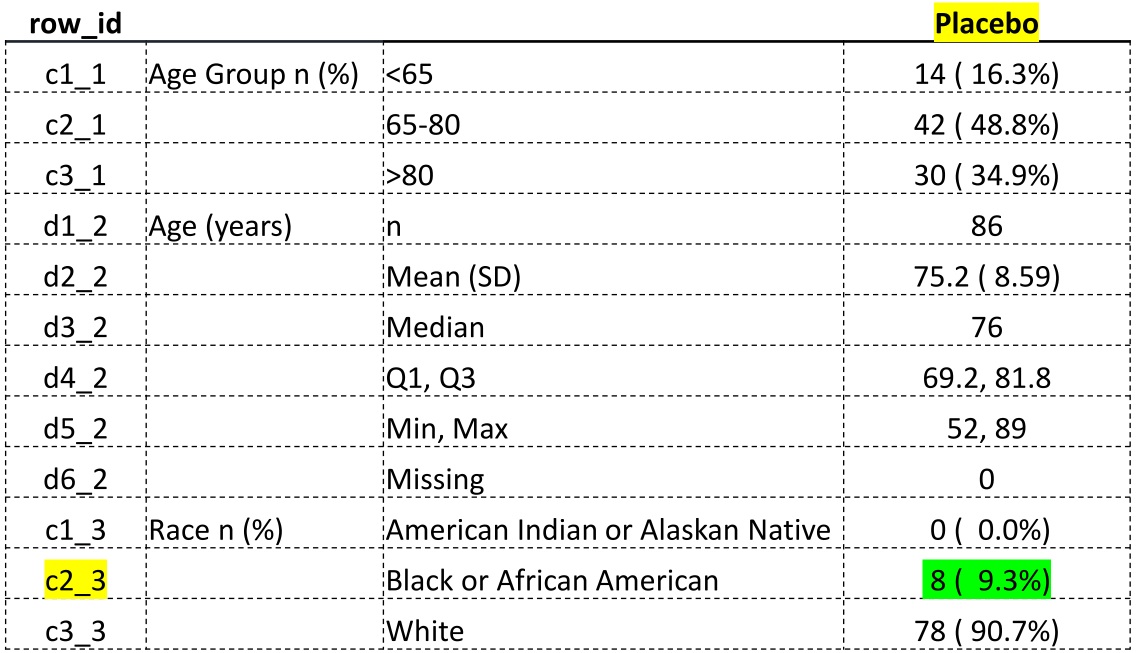
Output Data Frame



```
# A tibble: 12 × 4
   row_id row_label1      row_label2                       Placebo
   <chr>  <chr>           <chr>                            <list>
 1 c1_1   Age Group n (%) <65                              <tplyr_mt>
 2 c2_1   Age Group n (%) 65-80                            <tplyr_mt>
 3 c3_1   Age Group n (%) >80                              <tplyr_mt>
 4 d1_2   Age (years)     n                                <tplyr_mt>
 5 d2_2   Age (years)     Mean (SD)                        <tplyr_mt>
 6 d3_2   Age (years)     Median                           <tplyr_mt>
 7 d4_2   Age (years)     Q1, Q3                           <tplyr_mt>
 8 d5_2   Age (years)     Min, Max                         <tplyr_mt>
 9 d6_2   Age (years)     Missing                          <tplyr_mt>
10 c1_3   Race n (%)      American Indian or Alaskan Native<tplyr_mt>
11 c2_3   Race n (%)      Black or African American        <tplyr_mt>
12 c3_3   Race n (%)      White                            <tplyr_mt>
```

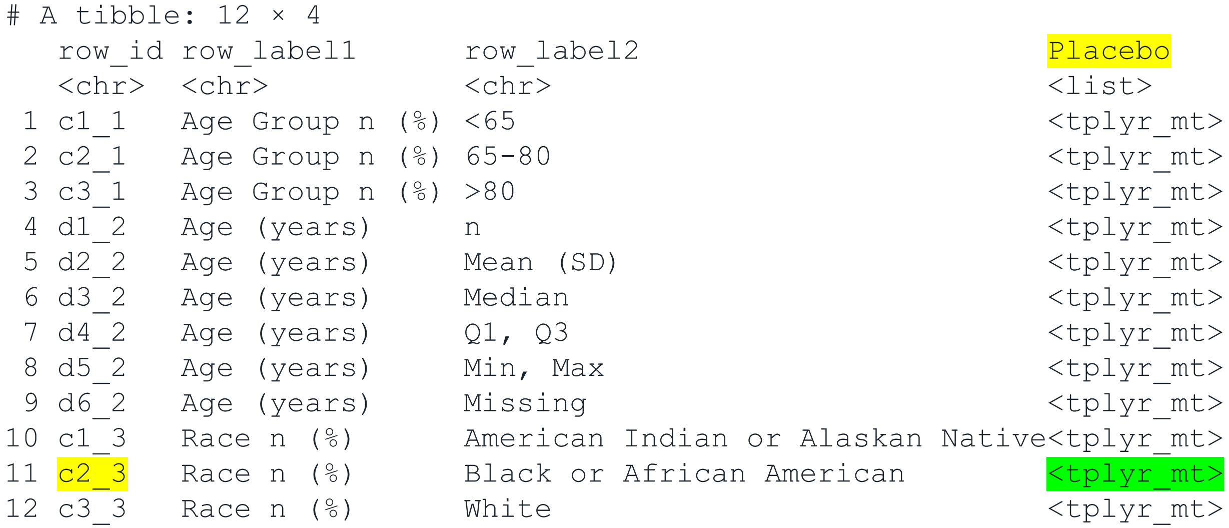
Tplyr Metadata

Building Metadata



row_id			Placebo
c1_1	Age Group n (%)	<65	14 (16.3%)
c2_1		65-80	42 (48.8%)
c3_1		>80	30 (34.9%)
d1_2	Age (years)	n	86
d2_2		Mean (SD)	75.2 (8.59)
d3_2		Median	76
d4_2		Q1, Q3	69.2, 81.8
d5_2		Min, Max	52, 89
d6_2		Missing	0
c1_3	Race n (%)	American Indian or Alaskan Native	0 (0.0%)
c2_3		Black or African American	8 (9.3%)
c3_3		White	78 (90.7%)

Output Data Frame



```
# A tibble: 12 × 4
   row_id row_label1      row_label2                       Placebo
   <chr>  <chr>           <chr>                            <list>
 1 c1_1   Age Group n (%) <65                              <tplyr_mt>
 2 c2_1   Age Group n (%) 65-80                            <tplyr_mt>
 3 c3_1   Age Group n (%) >80                              <tplyr_mt>
 4 d1_2   Age (years)     n                                <tplyr_mt>
 5 d2_2   Age (years)     Mean (SD)                        <tplyr_mt>
 6 d3_2   Age (years)     Median                           <tplyr_mt>
 7 d4_2   Age (years)     Q1, Q3                           <tplyr_mt>
 8 d5_2   Age (years)     Min, Max                         <tplyr_mt>
 9 d6_2   Age (years)     Missing                          <tplyr_mt>
10 c1_3   Race n (%)      American Indian or Alaskan Native<tplyr_mt>
11 c2_3   Race n (%)      Black or African American        <tplyr_mt>
12 c3_3   Race n (%)      White                            <tplyr_mt>
```

Tplyr Metadata

atorus

Your Complete Data Solution

Extracting Metadata

```
get_meta_result(t, 'c2_3', 'Placebo')
```

Tplyr table *row_id* *column name*

```
tplyr_meta: 2 names, 4 filters
Names:
    TRT01P, RACE
Filters:
    TRT01P == c("Placebo"), TRUE, TRUE, RACE == c("BLACK OR AFRICAN AMERICAN")
```

Extracting Metadata

```
get_meta_subset(t, 'c2_3', 'Placebo', add_cols=vars(USUBJID))
```

```
# A tibble: 8 × 3
  USUBJID     TRT01P  RACE
  <chr>       <chr>   <chr>
1 01-701-1203 Placebo BLACK OR AFRICAN AMERICAN
2 01-701-1363 Placebo BLACK OR AFRICAN AMERICAN
3 01-705-1282 Placebo BLACK OR AFRICAN AMERICAN
4 01-706-1041 Placebo BLACK OR AFRICAN AMERICAN
5 01-708-1286 Placebo BLACK OR AFRICAN AMERICAN
6 01-708-1296 Placebo BLACK OR AFRICAN AMERICAN
7 01-708-1378 Placebo BLACK OR AFRICAN AMERICAN
8 01-711-1036 Placebo BLACK OR AFRICAN AMERICAN
```

Expanding Tplyr's Metadata

Function	Description
`tplyr_meta()`	Create a tplyr_meta object
`add_variables()`	Add variables to a tplyr_meta object
`add_filters()`	Add filters to a tplyr_meta object
`append_metadata()`	Append a Tplyr table's metadata dataframe
`get_metadata()`	Extract a Tplyr table's metadata dataframe

Creating Custom Metadata

```
meta <- tplyr_meta(
  names = quos(TRTA, AVAL, PARAMCD, AVISIT),
  filters = quos(SAFFL == "Y", ANL01FL == "Y")
)

meta_hp <- meta %>%
  add_filters(quos(TRTP %in% c("High Dose, "Placebo))

meta_lp <- meta %>%
  add_filters(quos(TRTP %in% c("Low Dose, "Placebo))

meta_hl <- meta %>%
  add_filters(quos(TRTP %in% c("High Dose, "Low Dose))
```

Creating Custom Metadata

```
meta <- tplyr_meta(
  names = quos(TRTA, AVAL, PARAMCD, AVISIT),
  filters = quos(SAFFL == "Y", ANL01FL == "Y")
)
```

```
meta_hp <- meta %>%
  add_filters(quos(TRTP %in% c("High Dose, "Placebo))

meta_lp <- meta %>%
  add_filters(quos(TRTP %in% c("Low Dose, "Placebo))

meta_hl <- meta %>%
  add_filters(quos(TRTP %in% c("High Dose, "Low Dose))
```

Creating Custom Metadata

```
meta <- tplyr_meta(
  names = quos(TRTA, AVAL, PARAMCD, AVISIT),
  filters = quos(SAFFL == "Y", ANL01FL == "Y")
)

meta_hp <- meta %>%
  add_filters(quos(TRTP %in% c("High Dose, "Placebo))

meta_lp <- meta %>%
  add_filters(quos(TRTP %in% c("Low Dose, "Placebo))

meta_hl <- meta %>%
  add_filters(quos(TRTP %in% c("High Dose, "Low Dose))
```

Creating Custom Metadata

```
meta <- tplyr_meta(
  names = quos(TRTA, AVAL, PARAMCD, AVISIT),
  filters = quos(SAFFL == "Y", ANL01FL == "Y")
)

meta_hp <- meta %>%
  add_filters(quos(TRTP %in% c("High Dose, "Placebo))

meta_lp <- meta %>%
  add_filters(quos(TRTP %in% c("Low Dose, "Placebo))

meta_hl <- meta %>%
  add_filters(quos(TRTP %in% c("High Dose, "Low Dose))
```

Creating Custom Metadata

```
meta <- tplyr_meta(
  names = quos(TRTA, AVAL, PARAMCD, AVISIT),
  filters = quos(SAFFL == "Y", ANL01FL == "Y")
)

meta_hp <- meta %>%
  add_filters(quos(TRTA %in% c("High Dose, "Placebo))

meta_lp <- meta %>%
  add_filters(quos(TRTA %in% c("Low Dose, "Placebo))

meta_hl <- meta %>%
  add_filters(quos(TRTA %in% c("High Dose, "Low Dose))
```

Creating Custom Metadata

```
eff_meta <- tibble::tribble(
  ~"row_id", ~"row_label1",                      ~"var1_Xanomeline Low Dose", ~"var1_Xanomeline High Dose",
  "x4_1",    "p-value(Dose Response) [1][2]",      NULL,                        meta,
  "x4_3",    "p-value(Xan - Placebo) [1][3]",      meta_xlp,                    meta_xhp,
  "x4_4",    "   Diff of LS Means (SE)",           meta_xlp,                    meta_xhp,
  "x4_5",    "   95% CI",                          meta_xlp,                    meta_xhp,
  "x4_7",    "p-value(Xan High - Xan Low) [1][3]", NULL,                        meta_xlh,
  "x4_8",    "   Diff of LS Means (SE)",           NULL,                        meta_xlh,
  "x4_9",    "   95% CI",                          NULL,                        meta_xlh
)
```

- Metadata objects held within a "list" column
- Insert metadata objects into dataframe at appropriate locations to match with results
- Row labels aren't strictly necessary – just need row_id to match with the results dataframe

Append the Metadata

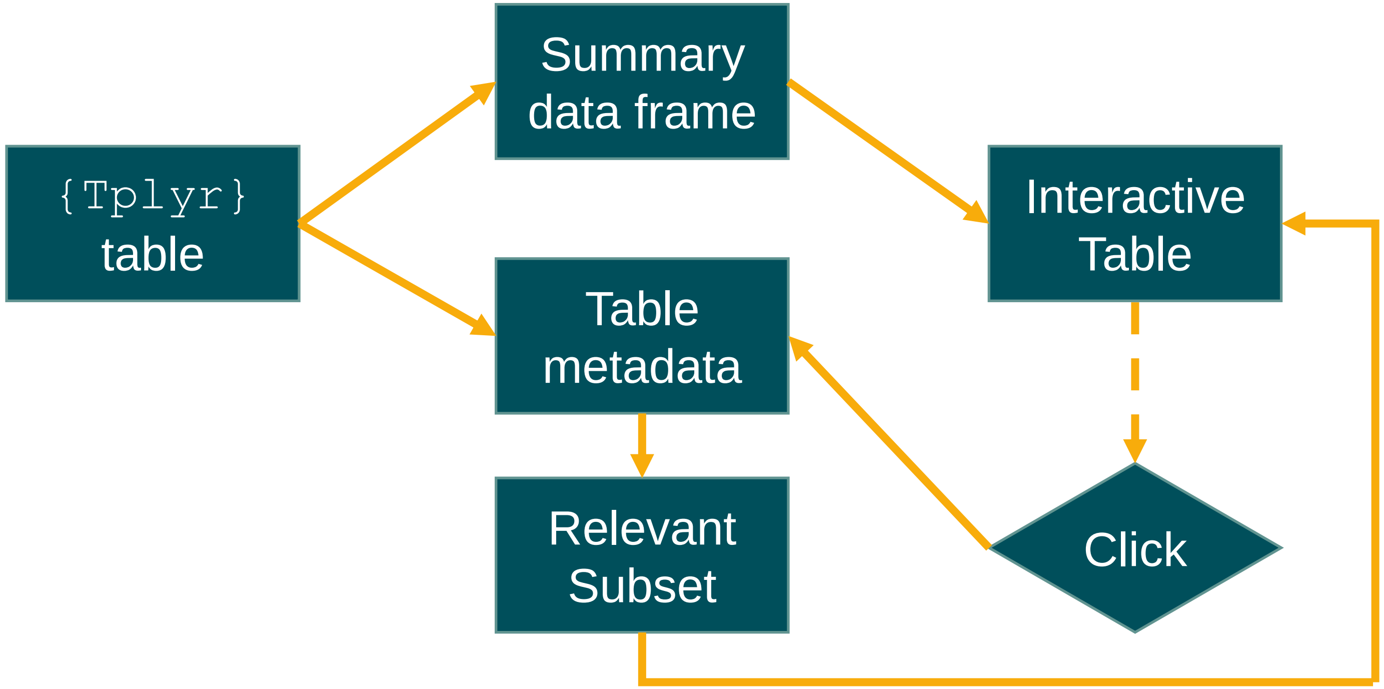
```
t <- append_metadata(t, eff_meta)
```

- `append_metadata()` will attach the metadata to the existing metadata dataframe in the `tplyr_table()`
- Alternatively, extract the metadata dataframe outside of Tplyr and use metadata outside of the Tplyr table

Breakout 3

Generate, Extract, and Extend Tplyr Metadata

Using Tplyr's Metadata



Using the Click Event in reactable

onClick parameter in reactable:

```
onClick = JS("function(rowInfo, colInfo) {
                      if (window.Shiny) {
                        Shiny.setInputValue('row', { index: rowInfo.index + 1 })
                        Shiny.setInputValue('col', { column: colInfo.id })
                        }
                    }")
```

Setting the reactives:

```
row <- reactive(b_tab[input$row$index,1]$row_id)
col <- reactive(input$col$column)
```

Thank you, Matthew Kumar!!!

Using the Click Event

```
sub_data <- reactive({
    req(row(), col())
    tmp <- get_meta_subset(tab, row(), col())
    tmp
})
```

- Call the reactives in get_meta_subset to extract the relevant dataframe
- Resulting dataframe can then be rendered however necessary

Breakout 4

Make a Shiny App with Drill Down Features

Breakout 5

- Replace the table we produced in Breakout 5 with the work we did in Problem 5 and Problem 6 of breakout 3 (including pre-work)

Breakout 5

Make Another App with Extended Metadata

atorus

Your Complete Data Solution

Questions?

atorus

Your Complete Data Solution