

# Descriptive tables with

`{gtsummary}`

Make an easy Table 1

# What is `{gtsummary}`?

- Create tables that are publication-ready
- Highly customizable
- Descriptive tables, regression tables, etc.



# gtsummary::tbl\_summary()

```
1 library(gtsummary)
2
3 tbl_summary(
4   nlsy,
5   by = sex_cat,
6   include = c(sex_cat,
7               eyesight
```

	Male N = 6,403 <sub>1</sub>	Female N = 6,283 <sub>1</sub>
Characteristic		
race_eth_cat		
Hispanic	1,000 (16%)	1,002 (16%)
Black	1,613 (25%)	1,561 (25%)
Non-Black, Non-Hispanic	3,790 (59%)	3,720 (59%)
region_cat		
Northeast	1,296 (21%)	1,254 (20%)
North Central	1,488 (24%)	1,446 (23%)
South	2,251 (36%)	2,317 (38%)
West	1,253 (20%)	1,142 (19%)
Unknown	115	124
eyesight_cat		
Excellent	1,582 (38%)	1,334 (31%)
Very good	1,470 (35%)	1,500 (35%)
Good	792 (19%)	1,002 (23%)
<sub>1</sub> n (%); Median (Q1, Q3)		

	Male	Female
	N = 6,403 <sub>1</sub>	N = 6,283 <sub>1</sub>
Characteristic		
Fair	267 (6.4%)	365 (8.5%)
Poor	47 (1.1%)	85 (2.0%)
Unknown	2,245	1,997
glasses	1,566 (38%)	2,328 (54%)
Unknown	2,241	1,995
age_bir	25 (21, 29)	22 (19, 27)
Unknown	3,652	3,091
<sub>1</sub>		
n (%); Median (Q1, Q3)		

# You can also refer to variables using helper functions

```
1 library(gtsummary)
2
3 tbl_summary(
4   nlsy,
5   by = sex_cat,
6   include = c(ends_wit
```

	Male N = 6,403 <sup>1</sup>	Female N = 6,283 <sup>1</sup>
Characteristic		
region_cat		
Northeast	1,296 (21%)	1,254 (20%)
North Central	1,488 (24%)	1,446 (23%)
South	2,251 (36%)	2,317 (38%)
West	1,253 (20%)	1,142 (19%)
Unknown	115	124
race_eth_cat		
Hispanic	1,000 (16%)	1,002 (16%)
Black	1,613 (25%)	1,561 (25%)
Non-Black, Non-Hispanic	3,790 (59%)	3,720 (59%)
eyesight_cat		
Excellent	1,582 (38%)	1,334 (31%)

<sup>1</sup>  
n (%); Median (Q1, Q3)

	Male	Female
	N = 6,403 <sub>1</sub>	N = 6,283 <sub>1</sub>
Characteristic		
Very good	1,470 (35%)	1,500 (35%)
Good	792 (19%)	1,002 (23%)
Fair	267 (6.4%)	365 (8.5%)
Poor	47 (1.1%)	85 (2.0%)
Unknown	2,245	1,997
glasses	1,566 (38%)	2,328 (54%)
Unknown	2,241	1,995
age_bir	25 (21, 29)	22 (19, 27)
Unknown	3,652	3,091
<sub>1</sub>		
n (%); Median (Q1, Q3)		

# We probably want to name the variables

```
1 tbl_summary(  
2   nlsy,  
3   by = sex_cat,  
4   include = c(sex_cat, race_eth_cat,  
5               eyesight_cat, glasses,  
6   label = list(  
7     race_eth_cat ~ "Race/ethnicity",  
8     region_cat ~ "Region",  
9     eyesight_cat ~ "Eyesight",  
10    glasses ~ "Wears glasses",  
11    age_bir ~ "Age at first birth"  
12  ),  
13  missing_text = "Missing")
```

	Male N = 6,403	Female N = 6,283
Characteristic		
Race/ethnicity		
Hispanic	1,000 (16%)	1,002 (16%)
Black	1,613 (25%)	1,561 (25%)
Non-Black, Non-Hispanic	3,790 (59%)	3,720 (59%)
Region		
Northeast	1,296 (21%)	1,254 (20%)
North Central	1,488 (24%)	1,446 (23%)
South	2,251 (36%)	2,317 (38%)
West	1,253 (20%)	1,142 (19%)
Missing	115	124
Eyesight		
Excellent	1,582 (38%)	1,334 (31%)
Very good	1,470 (35%)	1,500 (35%)
Good	792 (19%)	1,002 (23%)

1  
n (%); Median (Q1, Q3)

Characteristic	Male	Female
	N = 6,403 <sub>1</sub>	N = 6,283 <sub>1</sub>
Fair	267 (6.4%)	365 (8.5%)
Poor	47 (1.1%)	85 (2.0%)
Missing	2,245	1,997
Wears glasses	1,566 (38%)	2,328 (54%)
Missing	2,241	1,995
Age at first birth	25 (21, 29)	22 (19, 27)
Missing	3,652	3,091
<sup>1</sup>		
n (%); Median (Q1, Q3)		



# And do a million other things

```
1 tbl_summary(  
2   nlsy,  
3   by = sex_cat,  
4   include = c(sex_cat, race_eth_cat,  
5               eyesight_cat, glasses,  
6   label = list(  
7     race_eth_cat ~ "Race/ethnicity",  
8     eyesight_cat ~ "Eyesight",  
9     glasses ~ "Wears glasses",  
10    age_bir ~ "Age at first birth"  
11  ),  
12  missing_text = "Missing") |>  
13  add_p(test = list(all_continuous()  
14                  all_categorical()  
15  add_overall(col_label = "**Total**"  
16  bold_labels() |>  
17  modify_footnote(update = everythin  
18  modify_header(label = "**Variable*
```

Variable	Total	Male N = 6,403	Female N = 6,283	P
<b>Race/ethnicity</b>				
Hispanic	2,002 (16%)	1,000 (16%)	1,002 (16%)	
Black	3,174 (25%)	1,613 (25%)	1,561 (25%)	
Non-Black, Non-Hispanic	7,510 (59%)	3,790 (59%)	3,720 (59%)	
<b>Eyesight</b>				
Excellent	2,916 (35%)	1,582 (38%)	1,334 (31%)	
Very good	2,970 (35%)	1,470 (35%)	1,500 (35%)	
Good	1,794 (21%)	792 (19%)	1,002 (23%)	
Fair	632 (7.5%)	267 (6.4%)	365 (8.5%)	
Poor	132 (1.6%)	47 (1.1%)	85 (2.0%)	
Missing	4,242	2,245	1,997	
<b>Wears glasses</b>				
Missing	4,236	2,241	1,995	

# Additional arguments

We saw `include =`, `by =`, `label =`, `missing_text =` in the example

`statistic =:`

- The default is `list(all_continuous() ~ "{median} ({p25}, {p75})", all_categorical() ~ "{n} ({p}%)"`
- For categorical variables, you can use `{n}` (frequency), `{N}` (denominator), `{p}` formatted percentage
- For continuous variables, you can use `{median}`, `{mean}`, `{sd}`, `{var}`, `{min}`, `{max}`, `{sum}`, `{p##}` (any percentile), or any function `{foo}`
- You can refer to individual variables with their names:  
`list(age ~ "min = {min}; max = {max}")`

# Additional arguments

`digits =:`

- It will do its best to guess the appropriate number of digits
- Otherwise, you can pass a function:
  - `digits = everything() ~ style_sigfig`
- Or a value for each statistic shown
  - `statistic = list(age ~ "min = {min}; max = {max}",  
year_of_birth = "{median} ({p25}, {p75})") :`
  - `digits = list(age ~ c(1, 1)  
year_of_birth ~ c(0, 0, 0))`

# Additional arguments

`type =:`

- One of “continuous”, “continuous2”, “categorical”, “dichotomous”
  - If a variable only has 0/1, TRUE/FALSE, or yes/no values, it will be treated as dichotomous
    - You can override this with `type = list(`varname` ~ "categorical")`
    - Dichotomous variables only show one row (i.e., the percentage of 1's) unless you change to categorical
      - You can change which level to show with `value = list(varname ~ "level to show")`
  - “continuous2” variables can have multiple rows of statistics

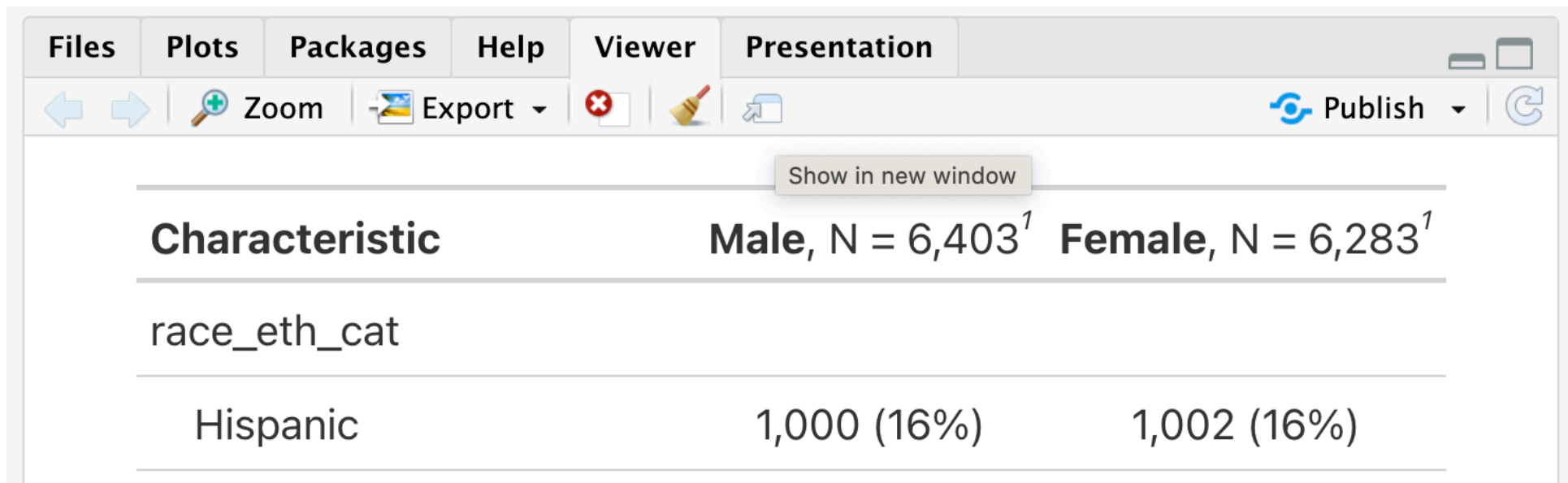
`missing =:`

# Additional functions

- `add_overall()`: In a stratified table, add a column for all strata combined
- `bold_labels()`: Bold the variable names (also `bold_levels()`)
- `add_p()`: Add a p-value (required by some journals 🙋)
- `modify_footnote(update = everything() ~ NA)`: Remove the footnotes (can also add footnotes!)
- `modify_header()`: Change the header column

```
tbl_summary()
```

- Incredibly customizable
  - So many options can be overwhelming
  - The [FAQ/gallery](#) is an incredible resource
- To save, I often just view in the web browser and copy and paste into a Word document
  - Can also be used within quarto/R Markdown



The screenshot shows the Quarto Viewer interface. The top navigation bar includes tabs for Files, Plots, Packages, Help, Viewer, and Presentation. Below the tabs is a toolbar with navigation arrows, a Zoom icon, an Export dropdown, a close button, a refresh button, and a Publish button. A 'Show in new window' button is also visible above the table.

Characteristic	Male, N = 6,403 <sup>1</sup>	Female, N = 6,283 <sup>1</sup>
race_eth_cat		
Hispanic	1,000 (16%)	1,002 (16%)



# Exercises

1. Open the script with some examples.
  2. Install `{gtsummary}` and run the examples.
  - 3-7. You're on your own! Work with your neighbors, and we'll come back together to go over these.
- Extra time? Start a table using the data you downloaded for your final project! Make sure you switch to that R project!

15 : 00