gt summary

Publication-ready analytical and summary tables with R

Cheat Sheet

Core Table Functions

tbl_summary()
Calculates descriptive stats for continuous, categorical, and dichotomous variables.

tbl_regression()
Turns a regression model object into a customized, formatted table.

tbl_survfit()
Turns a survfit object into a customized table with time-to-event estimates.

tbl_summary() using tidyverse syntax to summarize specific columns of a dataset with flexible customization options (See vignette!)

Basic code

```
trial %>% select(trt,age,grade,response) %>% tbl_summary()
```

Basic table

Characteristic	N = 200[1]
Chemotherapy Treatment	
Drug A	98 (49%)
Drug B	102 (51%)
Age, yrs	47 (38, 57)
Unknown	11
Grade	
I	68 (34%)
II	68 (34%)
III	64 (32%)
Tumor Response	61 (32%)
Unknown	7

[1] n (%); Median (IQR)

tbl_svysummary() for survey objects

Same functionality as tbl_summary(), but takes a survey object as input, and accounts for survey weights and design.

More info at:
http://www.danieldsjoberg.com/gtsummary/reference/tbl_svysummary.html

Customized code

```
trial %>%
 select(trt, age, grade, response) %>%
 tbl_summary(
  by = trt,
  label = list(age ~ "Age (years)",
               grade ~ "Tumor grade"),
  percent = "row"
  digits = list(age ~ 2),
  statistic = list(age ~ "{mean} ({sd})",
                   response ~ "{n}/{N} ({p}%)"),
  type = list(response ~ "categorical"),
  missing = "always",
  missing_text = "Missing",
  )
```

Customized table

Characteristic	Drug A, N = 98[1]	Drug B, N = 102[1]
Age (years)	47.01 (14.71)	47.45 (14.01)
Missing	7	4
Tumor grade		
I	35 (51%)	33 (49%)
II	32 (47%)	36 (53%)
III	31 (48%)	33 (52%)
Missing	0	0
Tumor Response		
0	67/132 (51%)	65/132 (49%)
1	28/61 (46%)	33/61 (54%)
Missing	3	4

[1] Mean (SD); n (%); n/N (%)

For more info on customization arguments and options, visit http://www.danieldsjoberg.com/gtsummary/reference/tbl_summary.html

Customization options

Argument	Input	Effect on table
by	Column to crosstabulate by	Summary statistics will be calculated separately for each level of the variable
label	Formula list of variable labels	Changes text of variable name in table
statistic	Formula list of summary statistic type for each variable	Changes summary statistics displayed for specified variables in table
digits	Formula list of number of decimal places to display	Changes number of rounded decimal places in table for specified continuous variables
type	Formula list specifying variable types	Changes variable type for specified variables, affecting which summary statistics are displayed
value	Formula list of value to display for dichotomous variables	Changes the value displayed for dichotomous type variables
missing	“no”, “ifany”, “always”	Changes whether missing observations are reported
missing_text	String to display for count of missing observations	Changes the name of the missing data level for appropriate variables
sort	Formula list of type of sorting to perform (“frequency” or “alphanumeric”)	Changes the type of sorting for categorical variables
percent	“column”, “row”, or “cell”	Changes how percentage statistics are calculated and displayed

Helper functions useful extensions to tbl_summary()

Extended code

```
trial %>%
  select(trt, age,
         response) %>%
  tbl_summary(
    by = trt,
    missing= "no“
    ) %>%
  add_n() %>%
  add_overall() %>%
  add_p()
```

Extended table

Characteristic	N	Overall, N = 200[1]	Drug A, N = 98[1]	Drug B, N = 102[1]	p-value[2]
Age, yrs	189	47 (38, 57)	46 (37, 59)	48 (39, 56)	0.7
Tumor Response	193	61 (32%)	28 (29%)	33 (34%)	0.5

[1] Median (IQR); n (%)
[2] Wilcoxon rank sum test; Pearson's Chi-squared test

add_n()
Adds a column with the total number of non-missing observations

add_overall()
Adds a column with overall summary statistics

See also:
add_q()
bold_p()
bold_labels()
add_stat()

add_p() Adds column of p-values generated by testing for differences between groups. Takes arguments below.

Argument	Default	Input	Effect on table
test	Continuous: “Kruskal test”; Categorical, expected cell counts ≥5: “chisq.test.no.correct“; Categorical, expected cell counts < 5: “fisher.test”	Formula list specifying statistical test to perform for each variable: "t.test“, “aov“, "wilcox.test“, "kruskal.test“, "chisq.test“ and "lme4“; custom tests possible too	Changes p-value in table based on specified statistical test
pvalue_fun	style_pvalue()	Function to round and format p-values	Changes format of p-values in table

For more info, visit http://www.danieldsjoberg.com/gtsummary/reference/index.html

tbl_regression() Present regression model object in publication-ready table

cox model: basic code

```
library(survival)
cox1 = coxph(Surv(ttdeath, death) ~ age + marker,
             data = trial)
tbl_regression(cox1, exponentiate = TRUE)
```

cox model: basic table

Characteristic	HR[1]	95% CI[1]	p-value
Age, yrs	1.01	0.99, 1.02	0.4
Marker Level, ng/mL	0.96	0.76, 1.21	0.7

[1] HR = Hazard Ratio, CI = Confidence Interval

1. **Build model of interest**
2. **Use tbl_regression() to present results**

tbl_regression() supports most commonly used regression models, and uses `broom::tidy(x)` to perform initial tidying.

For more info:
http://www.danieldsjoberg.com/gtsummary/reference/tbl_regression.html

glm model: basic code

```
m1 <- glm(response ~ age + stage, data = trial,
          family = binomial)
tbl_regression(m1, exponentiate = TRUE)
```

glm model: basic table

Characteristic	OR[1]	95% CI[1]	p-value
Age, yrs	1.02	1.00, 1.04	0.091
T Stage			
T1	—	—	
T2	0.58	0.24, 1.37	0.2
T3	0.94	0.39, 2.28	0.9
T4	0.79	0.33, 1.90	0.6

[1] OR = Odds Ratio, CI = Confidence Interval

glm model: custom code using helper functions

```
m1 %>%
  tbl_regression(exponentiate = TRUE) %>%
  add_global_p() %>%
  bold_p(t = 0.10) %>%
  bold_labels() %>%
  italicize_levels()
```

Helper functions

glm model: custom table

Characteristic	OR[1]	95% CI[1]	p-value
Age, yrs	1.02	1.00, 1.04	**0.087**
T Stage			0.6
T1	—	—	
T2	0.58	0.24, 1.37	
T3	0.94	0.39, 2.28	
T4	0.79	0.33, 1.90	

[1] OR = Odds Ratio, CI = Confidence Interval

tbl_uvregression() displays multiple univariate regression models at once

cox univariate models: code

```
library(survival)
tbl_uvregression(
    trial %>% select(ttdeath, death, age,
                     grade, response),
    method = coxph,
    y = Surv(ttdeath, death),
    exponentiate = TRUE,
    label = list(grade ~ "Tumor grade")
    ) %>%
  add_global_p()
  add_n(location = "level") %>%
  add_nevent(location = "level")
```

cox univariate models: table

Characteristic	N	Event N	HR[1]	95% CI[1]	p-value
Age, yrs	189	103	1.01	0.99, 1.02	0.3
Tumor grade					0.075
I	68	33	—	—	
II	68	36	1.28	0.80, 2.05	
III	64	43	1.69	1.07, 2.66	
Tumor Response	193	107	0.50	0.31, 0.78	0.001

[1] HR = Hazard Ratio, CI = Confidence Interval

Requires "method" parameter specifying model type. Can estimate univariate regression models holding either outcome ("y") or covariate ("x") constant, or both (see "formula" parameter). For more info about "formula" and other parameters, see:
https://www.danieldsjoberg.com/gtsummary/reference/tbl_uvregression.html

tbl_survfit() Present survfit object with custom estimates In publication-ready table

```
library(survival)
tbl_survfit(
 list(
    survfit(Surv(ttdeath, death) ~ 1, trial),
    survfit(Surv(ttdeath, death) ~ trt, trial)
 ),
  times = c(12, 24),
  label_header = "**{time} Month**"
)
```

Options for defining model:
1. x=explicit survfit model (or list of models) from dataframe
2. x=dataframe and designate y=Surv object and include=covariates in model

Options for output:
1. time-specific survival estimates using times=times of interest
2. quantile (e.g. median) survival times using probs=quantile of interest

For more info:
http://www.danieldsjoberg.com/gtsummary/reference/tbl_survfit.html

Characteristic	12 Month	24 Month
Overall	88% (84%, 93%)	44% (38%, 51%)
Chemotherapy Treatment		
Drug A	91% (85%, 97%)	47% (38%, 58%)
Drug B	86% (80%, 93%)	41% (33%, 52%)

```
library(survival)
tbl_survfit(
  trial,
  y = Surv(ttdeath, death),
  include = c(trt, grade),
  probs = 0.5,
  label_header = "**Median Survival**"
) %>% add_p()
```

Characteristic	Median Survival	p-value[1]
Chemotherapy Treatment		0.2
Drug A	24 (21, —)	
Drug B	21 (18, —)	
Grade		0.072
I	— (22, —)	
II	22 (18, —)	
III	20 (18, 23)	

[1] Log-rank test

tbl_merge(), tbl_stack() combine tables by row or column

```
t1 = tbl_survfit(
 list(survfit(Surv(ttdeath, death) ~ trt +
grade, trial),
  times = c(12, 24),
  label_header = "**{time} Month**"
)
```

```
t2 = tbl_survfit(
  trial,
  y = Surv(ttdeath, death),
  include = c(trt, grade),
  probs = 0.5,
  label_header = "**Median Survival**"
) %>% add_p()
```

```
tbl_merge(list(t1,t2), tab_spanner = FALSE)
```

Characteristic	12 Month	24 Month	Median Survival	p-value[1]
Chemotherapy Treatment				0.2
Drug A	91% (85%, 97%)	47% (38%, 58%)	24 (21, —)	
Drug B	86% (80%, 93%)	41% (33%, 52%)	21 (18, —)	
Grade				0.072
I	97% (93%, 100%)	51% (41%, 65%)	— (22, —)	
II	82% (74%, 92%)	47% (37%, 61%)	22 (18, —)	
III	86% (78%, 95%)	33% (23%, 47%)	20 (18, 23)	

[1] Log-rank test

tbl_merge combines columns, tbl_stack() combines rows. For more info, see
https://www.danieldsjoberg.com/gtsummary/reference/