gtsummary: Clinical Tables That Survive Peer Review

Table 1, regression, survival and export recipes in R

Rverse Analytics

Build the statistical object first; style it last. A `gtsummary` table preserves labels, tests and estimates as structured data, so one pipeline can feed HTML, Word and PDF outputs.

1 · Baseline characteristics (Table 1)

```
library(gtsummary)
library(dplyr)

tab1 <- trial |>
  select(trt, age, marker, stage, grade, response) |>
  tbl_summary(
    by = trt,
    label = list(age ~ "Age, years", marker ~ "Marker, ng/mL"),
    type = list(response ~ "dichotomous"),
    statistic = list(
      all_continuous() ~ "{median} ({p25}, {p75})",
      all_categorical() ~ "{n} ({p}%)"
    ),
    digits = all_continuous() ~ 1,
    missing = "ifany",
    missing_text = "Missing"
  ) |>
  add_n() |>
  add_overall(last = TRUE) |>
  add_p(
    test = list(
      all_continuous() ~ "wilcox.test",
      all_categorical() ~ "fisher.test"
    ),
    pvalue_fun = label_style_pvalue(digits = 2)
  ) |>
  add_q(method = "BH") |>
  bold_labels() |>
  bold_p(t = 0.05) |>
  modify_caption("**Table 1. Baseline characteristics**")
```

Need	Add or change
Mean (SD)	`statistic = all_continuous() ~ "{mean} ({sd})"`
Median (min, max)	`"{median} ({min}, {max})"`
Separate missing row	`missing = "always"`
No missing row	`missing = "no"`
Standardized difference	`add_difference(test = everything() ~ "smd")`
Custom test	`add_p(test = age ~ "t.test", group = center)`
Sort binary variables	`sort = all_dichotomous() ~ "frequency"`

2 · Regression tables

```
# Logistic model: exponentiate log-odds to odds ratios
fit_logit <- glm(response ~ age + stage + grade,
                 data = trial, family = binomial)

tab_or <- tbl_regression(
  fit_logit,
  exponentiate = TRUE,
  label = list(age ~ "Age, per year"),
  estimate_fun = label_style_ratio(digits = 2),
  pvalue_fun = label_style_pvalue(digits = 2)
) |>
  add_global_p() |>
  add_n(location = "level") |>
  bold_labels() |>
```

```r
  bold_p()

# Univariable screen: one model per predictor
tab_uni <- trial |>
  tbl_uvregression(
    method = glm,
    y = response,
    method.args = list(family = binomial),
    exponentiate = TRUE
  )
```

Model	Function	`exponentiate`	Report
Linear	`lm(y ~ x, data)`	`FALSE`	β, 95% CI
Logistic	`glm(y ~ x, family = binomial)`	`TRUE`	OR, 95% CI
Poisson log-link	`glm(count ~ x, family = poisson)`	`TRUE`	rate ratio
Cox PH	`coxph(Surv(time, event) ~ x, data)`	`TRUE`	HR, 95% CI
Mixed model	`lme4::glmer(...)`	usually `TRUE`	conditional OR/RR

3 · Combine, annotate and export

```r
tbl_merge(
  tbls = list(tab_uni, tab_or),
  tab_spanner = c("**Unadjusted**", "**Adjusted**")
) |>
  modify_header(label = "**Characteristic**") |>
  modify_footnote(all_stat_cols() ~ "Estimate (95% CI)")

# Stack compatible tables vertically
tbl_stack(list(table_a, table_b), group_header = c("Cohort A", "Cohort B"))

# Inspect hidden columns before a modify_* call
show_header_names(tab_or)

# Output adapters
as_gt(tab1) |> gt::gtsave("table1.html")
as_flex_table(tab1) |> flextable::save_as_docx(path = "table1.docx")
```

Review checklist

- Define the analysis population and denominator before making the table.
- Choose summaries and tests from the estimand and design, not from the observed *p*-value.
- For randomized trials, prefer standardized differences over baseline significance tests.
- State units, reference levels, missing-data handling and multiplicity adjustment.
- Do not call an odds ratio a risk ratio; exponentiation changes the scale, not the estimand.

Rverse Analytics · rverseanalytics.com · Original reference sheet · gtsummary 2.x workflow