Package ‘TernTables’

May 7, 2026

Type Package

Title Publication-Ready Summary Tables and Statistical Testing for Clinical Research

Version 1.6.4

Description Generates publication-ready summary tables for clinical research, supporting descriptive summaries and comparisons across two or three groups. The package streamlines the analytical workflow by detecting variable types and applying appropriate statistical tests (Welch t-test, Wilcoxon rank-sum, Welch ANOVA, Kruskal-Wallis, Chi-squared, or Fisher's exact test). Results are formatted as 'tibble' objects and can be exported to 'Word' or 'Excel' using the 'officer', 'flextable', and 'writexl' packages. Optional pairwise post-hoc testing for three-group comparisons (Games-Howell and Dunn's test) is available via the 'rstatix' package. Example data are derived from the landmark adjuvant colon cancer trial described in Moertel et al. (1990) <doi:10.1056/NEJM199002083220602>.

License MIT + file LICENSE

URL `https://github.com/jdpreston30/TernTables`,
`https://tern-tables.com/`

BugReports `https://github.com/jdpreston30/TernTables/issues`

Encoding UTF-8

RoxygenNote 7.3.3

Imports cli, dplyr (>= 1.0.0), epitools, flextable (>= 0.9.0), magrittr, officer (>= 0.4.6), rlang, stats, stringr, tibble, withr, writexl

Suggests knitr, multcompView, rmarkdown, rstatix, survival, testthat (>= 3.0.0)

VignetteBuilder knitr

Depends R (>= 4.1.0)

LazyData true

Config/testthat/edition 3

NeedsCompilation no

Author Joshua D. Preston [aut, cre] (ORCID:
<https://orcid.org/0000-0001-9834-3017>),
Helen Abadiotakis [aut] (ORCID:
<https://orcid.org/0009-0002-8268-927X>),
Ailin Tang [aut] (ORCID: <https://orcid.org/0009-0007-8715-1678>),
Clayton J. Rust [aut] (ORCID: <https://orcid.org/0000-0001-5929-0733>),
Joshua L. Chan [aut] (ORCID: <https://orcid.org/0000-0001-7220-561X>)

Maintainer Joshua D. Preston <`joshua.preston@emory.edu`>

Repository CRAN

Date/Publication 2026-03-26 10:00:21 UTC

Contents

`print.ternP_result` *Print method for ternP_result objects*

Description

Re-displays the preprocessing summary for a `ternP_result` object. Note that `ternP` already emits this summary automatically at the time it is called, so this method is most useful for reviewing the summary after the fact (e.g. typing `result` at the console later in a session).

Usage

```
## S3 method for class 'ternP_result'
print(x, ...)
```

Arguments

`x`	A `ternP_result` object returned by `ternP`.
`...`	Currently unused; included for S3-method compatibility.

Value

Invisibly returns `x`.

`ternB` *Combine multiple ternD/ternG tables into a single Word document*

Description

Takes a list of tibbles previously created by `ternD()` or `ternG()` and writes them all into one `.docx` file, one table per page, preserving the exact formatting settings that were used when each table was built.

Usage

```
ternB(
  tables,
  output_docx,
  page_break = TRUE,
  methods_doc = FALSE,
  methods_filename = "TernTables_methods.docx"
)
```

Arguments

`tables`	A **list** of tibbles created by `ternD()` or `ternG()`. Must be constructed with `list()`, not `c()` (e.g. `list(T1, T2, T3)`). Each tibble must have been produced in the *current R session*; the metadata is stored in memory, not in the tibble columns.
`output_docx`	Output file path ending in `.docx`.
`page_break`	Logical; if `TRUE` (default), inserts a page break between each consecutive table.
`methods_doc`	Logical; if `TRUE`, writes a single methods section Word document that covers all tables in the list. Statistical test details are pooled across all tables. Default is `FALSE`.
`methods_filename`	Output file path for the methods document. Defaults to `"TernTables_methods.docx"` in the working directory.

Details

`ternB()` works by replaying the exact `word_export()` call that `ternD()` / `ternG()` would have made – using stored metadata attached as an attribute to each returned tibble – but directing all output into a single combined document instead of separate files.

Table captions (`table_caption`) and footnotes (`table_footnote`) specified in the original `ternD()` / `ternG()` call are reproduced automatically. You can override them by modifying the `"ternB_meta"` attribute before calling `ternB()`, though in practice it is easier to set captions and footnotes when you first build each table.

Value

Invisibly returns the path to the written Word file.

Examples

```
data(tern_colon)

T1 <- ternD(tern_colon,
            exclude_vars  = "ID",
            table_caption = "Table 1. Overall patient characteristics.",
            methods_doc   = FALSE)

T2 <- ternG(tern_colon,
            group_var     = "Recurrence",
            exclude_vars  = "ID",
            table_caption = "Table 2. Characteristics by recurrence status.",
            methods_doc   = FALSE)

ternB(list(T1, T2),
      output_docx = file.path(tempdir(), "combined_tables.docx"))
```

`ternD` *Generate descriptive summary table (optionally normality-aware)*

Description

Creates a descriptive summary table with a single "Total" column format. By default (`consider_normality = "ROBUST"`), continuous variables are shown as mean +/- SD or median [IQR] based on a four-gate decision (n < 3 fail-safe, skewness, CLT, and Shapiro-Wilk). This can be overridden via `consider_normality` and `force_ordinal`.

Usage

```
ternD(
  data,
  vars = NULL,
  exclude_vars = NULL,
  force_ordinal = NULL,
  output_xlsx = NULL,
  output_docx = NULL,
  consider_normality = "ROBUST",
  print_normality = FALSE,
  round_intg = FALSE,
  smart_rename = TRUE,
  insert_subheads = TRUE,
  factor_order = "mixed",
  methods_doc = TRUE,
```

```
  methods_filename = "TernTables_methods.docx",
  category_start = NULL,
  table_font_size = 9,
  manual_italic_indent = NULL,
  manual_underline = NULL,
  table_caption = NULL,
  table_footnote = NULL,
  line_break_header = getOption("TernTables.line_break_header", TRUE)
)
```

Arguments

`data`	Tibble with variables.
`vars`	Character vector of variables to summarize. Defaults to all except `exclude_vars`.
`exclude_vars`	Character vector to exclude from the summary.
`force_ordinal`	Character vector of variables to treat as ordinal (i.e., use median [IQR]) regardless of the `consider_normality` setting. This parameter takes priority over normality testing when `consider_normality` = `"ROBUST"` or `TRUE`.
`output_xlsx`	Optional Excel filename to export the table.
`output_docx`	Optional Word filename to export the table.
`consider_normality`	Character or logical; controls routing of continuous variables to mean ± SD vs median [IQR]. `"ROBUST"` (default) applies a four-gate decision: (1) $n < 3$ → non-parametric (conservative fail-safe); (2) absolute skewness > 2 → non-parametric regardless of n; (3) $n \geq 30$ → parametric via the Central Limit Theorem; (4) otherwise Shapiro-Wilk $p > 0.05$ → parametric. If `TRUE`, uses Shapiro-Wilk alone (can be over-sensitive at large n). If `FALSE`, defaults to mean ± SD for all numeric variables unless specified in `force_ordinal`.
`print_normality`	Logical; if `TRUE`, includes Shapiro-Wilk P values as an additional column in the output. Default is `FALSE`.
`round_intg`	Logical; if `TRUE`, rounds all means, medians, IQRs, and standard deviations to nearest integer (0.5 rounds up). Default is `FALSE`.
`smart_rename`	Logical; if `TRUE`, automatically cleans variable names and subheadings for publication-ready output using built-in rule-based pattern matching for common medical abbreviations and prefixes. Default is `TRUE`.
`insert_subheads`	Logical; if `TRUE` (default), creates a hierarchical structure with a header row and indented sub-category rows for categorical variables with 3 or more levels. Binary variables (Y/N, YES/NO, or numeric 1/0 – which are auto-detected and treated as Y/N) are always displayed as a single row showing the positive/yes count regardless of this setting. Two-level categorical variables whose values are not Y/N, YES/NO, or 1/0 (e.g. Male/Female) use the hierarchical sub-row format, showing both levels as indented rows. If `FALSE`, all categorical variables use a single-row flat format. Default is `TRUE`.

`factor_order`	Character; controls the ordering of factor levels in the output. `"mixed"` (default) applies level-aware ordering for two-level categorical variables and frequency ordering for variables with three or more levels: for any factor, factor level order is always respected regardless of the number of levels; for non-factor two-level variables, levels are sorted alphabetically; for non-factor variables with three or more levels, levels are sorted by decreasing frequency. `"levels"` respects the original factor level ordering for all variables; if the variable is not a factor, falls back to frequency ordering. `"frequency"` orders all levels by decreasing frequency (most common first).
`methods_doc`	Logical; if `TRUE` (default), generates a methods document describing the statistical presentation used. The document contains boilerplate text for all three table types so the relevant section can be copied directly into a manuscript.
`methods_filename`	Character; filename for the methods document. Default is `"TernTables_methods.docx"`.
`category_start`	Named character vector specifying where to insert category headers. Names are the header label text to display; values are the anchor variable – either the original column name (e.g. `"Age_Years"`) or the cleaned display name (e.g. `"Age (yr)"`). Both forms are accepted. Example: `c("Demographics" = "Age_Years", "Clinical Measures" = "bmi")`. Default is `NULL` (no category headers).
`table_font_size`	Numeric; font size for Word document output tables. Default is 9.
`manual_italic_indent`	Character vector of display variable names (post-cleaning) that should be formatted as italicized and indented in Word output – matching the appearance of factor sub-category rows. Has no effect on the returned tibble; only applies when `output_docx` is specified. Default is `NULL`.
`manual_underline`	Character vector of display variable names (post-cleaning) that should be formatted as underlined in Word output – matching the appearance of multi-category variable headers. Has no effect on the returned tibble; only applies when `output_docx` is specified. Default is `NULL`.
`table_caption`	Optional character string for a table caption to display above the table in the Word document. Rendered as size 11 Arial bold, single-spaced with a small gap before the table. Default is `NULL` (no caption). Example: `"Table 1. Patient demographics."`
`table_footnote`	Optional character string for a footnote to display below the table in the Word document. Rendered as size 6 Arial italic with a double-bar border above and below. Default is `NULL` (no footnote).
`line_break_header`	Logical; if `TRUE` (default), column headers are wrapped with `\n` – the first column header includes a category hierarchy label, and the sample size appears on a second line. Set to `FALSE` to suppress all header line breaks. Can also be set package-wide via `options(TernTables.line_break_header = FALSE)`.

Details

The function always returns a tibble with a single `Total (N = n)` column format, regardless of the `consider_normality` setting. The behavior for numeric variables follows this priority:

1. Variables in `force_ordinal`: Always use median [IQR]
2. When `consider_normality = "ROBUST"`: Four-gate decision (n<3 fail-safe, skewness, CLT, Shapiro-Wilk)
3. When `consider_normality = TRUE`: Use Shapiro-Wilk test to choose format
4. When `consider_normality = FALSE`: Default to mean +/- SD

For categorical variables, the function shows frequencies and percentages. When `insert_subheads = TRUE`, categorical variables with 3 or more levels are displayed with hierarchical formatting (main variable as header, levels as indented sub-rows). Binary variables (Y/N, YES/NO, or numeric 1/0 auto-detected as Y/N) always use a single-row format showing only the positive/yes count, regardless of this setting. Two-level categorical variables whose values are not Y/N, YES/NO, or 1/0 (e.g. Male/Female) also use the hierarchical sub-row format.

Value

A tibble with one row per variable (multi-row for factors), containing:

Variable	Variable names with appropriate indentation
Total (N = n)	Summary statistics (mean +/- SD, median [IQR], or n (%) as appropriate)
SW_p	Shapiro-Wilk P values (only if `print_normality = TRUE`)

Examples

```
data(tern_colon)

# Basic descriptive summary
ternD(tern_colon, exclude_vars = c("ID"), methods_doc = FALSE)

# With normality-aware formatting and category section headers
ternD(tern_colon, exclude_vars = c("ID"), methods_doc = FALSE,
      category_start = c("Patient Demographics"  = "Age (yr)",
                         "Tumor Characteristics" = "Positive Lymph Nodes (n)"))

# Force specific variables to ordinal (median [IQR]) display
ternD(tern_colon, exclude_vars = c("ID"), methods_doc = FALSE,
      force_ordinal = c("Positive_Lymph_Nodes_n"))

# Export to Word (writes a file to tempdir)

ternD(tern_colon,
      exclude_vars     = c("ID"),
      methods_doc      = FALSE,
      output_docx      = file.path(tempdir(), "descriptive.docx"),
      category_start   = c("Patient Demographics"  = "Age (yr)",
                           "Surgical Findings"     = "Colonic Obstruction",
                           "Tumor Characteristics" = "Positive Lymph Nodes (n)",
```

```
                        "Outcomes"              = "Recurrence"))
```

---

ternG *Generate grouped summary table with appropriate statistical tests*

Description

Creates a grouped summary table with optional statistical testing for group comparisons. Supports numeric and categorical variables; numeric variables can be treated as ordinal via `force_ordinal`. Includes options to calculate P values and odds ratios. For descriptive (ungrouped) tables, use `ternD`.

Usage

```
ternG(
  data,
  vars = NULL,
  exclude_vars = NULL,
  group_var,
  force_ordinal = NULL,
  group_order = NULL,
  output_xlsx = NULL,
  output_docx = NULL,
  OR_col = FALSE,
  OR_method = "dynamic",
  consider_normality = "ROBUST",
  print_normality = FALSE,
  show_test = FALSE,
  p_digits = 3,
  round_intg = FALSE,
  smart_rename = TRUE,
  insert_subheads = TRUE,
  factor_order = "mixed",
  table_font_size = 9,
  methods_doc = TRUE,
  methods_filename = "TernTables_methods.docx",
  category_start = NULL,
  manual_italic_indent = NULL,
  manual_underline = NULL,
  indent_info_column = FALSE,
  show_total = TRUE,
  table_caption = NULL,
  table_footnote = NULL,
  line_break_header = getOption("TernTables.line_break_header", TRUE),
  post_hoc = FALSE
)
```

Arguments

`data`	Tibble containing all variables.
`vars`	Character vector of variables to summarize. Defaults to all except `group_var` and `exclude_vars`.
`exclude_vars`	Character vector of variable(s) to exclude. `group_var` is automatically excluded.
`group_var`	Character, the grouping variable (factor or character with >=2 levels).
`force_ordinal`	Character vector of variables to treat as ordinal (i.e., use medians/IQR and non-parametric tests).
`group_order`	Optional character vector to specify a custom group level order.
`output_xlsx`	Optional filename to export the table as an Excel file.
`output_docx`	Optional filename to export the table as a Word document.
`OR_col`	Logical; if `TRUE`, adds odds ratios with 95% CI for binary categorical variables (Y/N, YES/NO, or numeric 0/1) and two-level categorical variables (e.g. Male/Female). For two-level categoricals displayed with sub-rows, the reference level (factor level 1, or alphabetical first for non-factors) shows `"1.00 (ref.)"`; the non-reference level shows the computed OR with 95% CI. Variables with three or more levels show `"-"`. Only valid when `group_var` has exactly 2 levels; an error is raised for 3+ group comparisons. Default is `FALSE`.
`OR_method`	Character; controls how odds ratios are calculated when `OR_col = TRUE`. If `"dynamic"` (default), uses Fisher's exact method when any expected cell count is < 5 (Cochran criterion), otherwise uses the Wald method. If `"wald"`, forces the Wald method regardless of expected cell counts.
`consider_normality`	Character or logical; controls how continuous variables are routed to parametric vs. non-parametric tests. `"ROBUST"` (default) applies a four-gate decision consistent with standard biostatistical practice: (1) any group $n < 3$ is a conservative fail-safe to non-parametric; (2) absolute skewness > 2 in any group routes to non-parametric regardless of sample size (catches LOS, counts, etc.); (3) all groups $n \geq 30$ routes to parametric via the Central Limit Theorem; (4) otherwise Shapiro-Wilk $p > 0.05$ in all groups routes to parametric. Normal variables use mean ± SD and Welch t-test (2 groups) or Welch ANOVA (3+ groups); non-normal variables use median [IQR] and Wilcoxon rank-sum (2 groups) or Kruskal-Wallis (3+ groups). If `TRUE`, uses Shapiro-Wilk alone ($p > 0.05$ in all groups = normal). Conservative at large n. If `FALSE`, all numeric variables are treated as normally distributed regardless of distribution. If `"FORCE"`, all numeric variables are treated as non-normal (median [IQR], nonparametric tests).
`print_normality`	Logical; if `TRUE`, includes Shapiro-Wilk P values in the output. Default is `FALSE`.
`show_test`	Logical; if `TRUE`, includes the statistical test name as a column in the output. Default is `FALSE`.
`p_digits`	Integer; number of decimal places for P values (default 3).
`round_intg`	Logical; if `TRUE`, rounds all means, medians, IQRs, and standard deviations to nearest integer (0.5 rounds up). Default is `FALSE`.

`smart_rename`	Logical; if `TRUE`, automatically cleans variable names and subheadings for publication-ready output using built-in rule-based pattern matching for common medical abbreviations and prefixes. Default is `TRUE`.
`insert_subheads`	Logical; if `TRUE` (default), creates a hierarchical structure with a header row and indented sub-category rows for categorical variables with 3 or more levels. Binary variables (Y/N, YES/NO, or numeric 1/0 – which are auto-detected and treated as Y/N) are always displayed as a single row showing the positive/yes count regardless of this setting. Two-level categorical variables whose values are not Y/N, YES/NO, or 1/0 (e.g. Male/Female) use the hierarchical sub-row format, showing both levels as indented rows. If `FALSE`, all categorical variables use a single-row flat format. Default is `TRUE`.
`factor_order`	Character; controls the ordering of factor levels in the output. `"mixed"` (default) applies level-aware ordering for two-level categorical variables and frequency ordering for variables with three or more levels: for any factor, factor level order is always respected regardless of the number of levels; for non-factor two-level variables (e.g. Male/Female), levels are sorted alphabetically; for non-factor variables with three or more levels, levels are sorted by decreasing frequency. `"levels"` respects the original factor level ordering for all variables; if the variable is not a factor, falls back to frequency ordering. `"frequency"` orders all levels by decreasing frequency (most common first).
`table_font_size`	Numeric; font size for Word document output tables. Default is 9.
`methods_doc`	Logical; if `TRUE` (default), generates a methods document describing the statistical tests used.
`methods_filename`	Character; filename for the methods document. Default is `"TernTables_methods.docx"`.
`category_start`	Named character vector specifying where to insert category headers. Names are the header label text to display; values are the anchor variable – either the original column name (e.g. `"Age_Years"`) or the cleaned display name (e.g. `"Age (yr)"`). Both forms are accepted. Example: `c("Demographics" = "Age_Years", "Clinical" = "bmi")`. Default is `NULL` (no category headers).
`manual_italic_indent`	Character vector of display variable names (post-cleaning) that should be formatted as italicized and indented in Word output – matching the appearance of factor sub-category rows. Has no effect on the returned tibble; only applies when `output_docx` is specified or when the tibble is passed to `word_export`.
`manual_underline`	Character vector of display variable names (post-cleaning) that should be formatted as underlined in Word output – matching the appearance of multi-category variable headers. Has no effect on the returned tibble; only applies when `output_docx` is specified or when the tibble is passed to `word_export`.
`indent_info_column`	Logical; if `FALSE` (default), the internal `.indent` helper column is dropped from the returned tibble. Set to `TRUE` to retain it – this is necessary when you intend to post-process the tibble and later pass it to `word_export` directly, as

	`word_export` uses the `.indent` column to apply correct indentation and italic formatting in the Word table.
`show_total`	Logical; if `TRUE`, adds a "Total" column showing the aggregate summary statistic across all groups (e.g., for a publication Table 1 that includes both per-group and overall columns). Default is `TRUE`.
`table_caption`	Optional character string for a table caption to display above the table in the Word document. Rendered as size 11 Arial bold, single-spaced with a small gap before the table. Default is `NULL` (no caption). Example: `"Table 2. Comparison of recurrence vs. no recurrence."`
`table_footnote`	Optional character string for a footnote to display below the table in the Word document. Rendered as size 6 Arial italic with a double-bar border above and below. Default is `NULL` (no footnote).
`line_break_header`	Logical; if `TRUE` (default), column headers are wrapped with `\n` – group names break on spaces, sample size counts move to a second line, and the first column header reads `"Category / Variable"`. Set to `FALSE` to suppress all header line breaks. Can also be set package-wide via `options(TernTables.line_break_header = FALSE)`.
`post_hoc`	Logical; if `TRUE`, runs pairwise post-hoc tests for continuous and ordinal variables in three or more group comparisons and annotates each group column value with a compact letter display (CLD) superscript. Groups sharing a letter are not significantly different at $\alpha = 0.05$. For normally distributed variables (Welch ANOVA path), Games-Howell pairwise tests are used. For non-normal and ordinal variables (Kruskal-Wallis path), Dunn's test with Holm correction is used. Post-hoc testing is never applied to categorical variables. Only valid when `group_var` has three or more levels; silently ignored for two-group comparisons. Requires the `rstatix` package. Default is `FALSE`.

Value

A tibble with one row per variable (multi-row for multi-level factors), showing summary statistics by group, P values, test type, and optionally odds ratios and total summary column.

Examples

```
data(tern_colon)

# 2-group comparison
ternG(tern_colon, exclude_vars = c("ID"), group_var = "Recurrence",
      methods_doc = FALSE)

# 2-group comparison with odds ratios
ternG(tern_colon, exclude_vars = c("ID"), group_var = "Recurrence",
      OR_col = TRUE, methods_doc = FALSE)

# 3-group comparison
ternG(tern_colon, exclude_vars = c("ID"), group_var = "Treatment_Arm",
      group_order = c("Observation", "Levamisole", "Levamisole + 5FU"),
      methods_doc = FALSE)
```

```
# Export to Word (writes a file to tempdir)

ternG(tern_colon,
      exclude_vars   = c("ID"),
      group_var      = "Recurrence",
      OR_col         = TRUE,
      methods_doc    = FALSE,
      output_docx    = file.path(tempdir(), "comparison.docx"),
      category_start = c("Patient Demographics"  = "Age (yr)",
                         "Tumor Characteristics" = "Positive Lymph Nodes (n)"))
```

`ternP` *Preprocess a raw data frame for use with ternG or ternD*

Description

`ternP()` cleans a raw data frame loaded from a CSV or XLSX file, applying a standardized set of transformations and performing validation checks before the data is passed to `ternG` or `ternD`.

Usage

```
ternP(data)
```

Arguments

`data`	A data frame or tibble as loaded from a CSV or XLSX file (e.g. via `readr::read_csv()` or `readxl::read_excel()`). All character columns are processed; numeric and logical columns are passed through unchanged by the string-cleaning steps.

Value

A named list with three elements:

- `clean_data` A tibble containing the fully cleaned dataset, ready to pass to `ternG()` or `ternD()`.
- `sparse_rows` A tibble of rows from `clean_data` where more than 50% of values are `NA`. These rows are *retained* in `clean_data` but extracted here for optional review or download. An empty tibble if no sparse rows exist.
- `feedback` A named list of feedback items. Each element is `NULL` if the corresponding transformation was not triggered, or a value describing what changed:
  - `string_na_converted` A named list with elements `total` (integer count of values converted) and `cols` (character vector of affected column names), or `NULL` if no string NA values were found.
  - `blank_rows_removed` A named list with elements `count` (integer) and `row_indices` (integer vector of original row positions removed), or `NULL` if none.

`sparse_rows_flagged` A named list with elements `count` (integer) and `row_indices` (integer vector of row positions in `clean_data` with >50% missingness), or `NULL` if none.

`case_normalized_vars` A named list with elements `cols` (character vector of affected column names) and `detail` (a named list per column, each with `changed_from` and `changed_to` character vectors showing the exact value changes), or `NULL` if none.

`dropped_empty_cols` Character vector of column names (or `""` for unnamed columns) that were dropped because they were 100% empty, or `NULL` if none.

Cleaning pipeline (in order)

1. String NA values (`"NA"`, `"na"`, `"Na"`, `"unk"`) are converted to `NA`.
2. Leading and trailing whitespace is trimmed from all character columns.
3. Columns that are 100% empty (all `NA`) are silently dropped.
4. Rows where every cell is `NA` are removed.
5. Character columns where values differ only by capitalization (e.g. `"Male"` vs `"MAle"`) are standardized to title case.

Validation hard stops

`ternP()` stops with a descriptive error if:

- Any column name matches a protected health information (PHI) pattern (e.g. `MRN`, `DOB`, `FirstName`). De-identified research identifiers such as `patient_id`, `subject_id`, and `participant_id` are explicitly excluded, as are clinical-event dates (admission date, discharge date, visit date, etc.). Only personal-identity dates such as DOB and DOD are flagged.
- Any column with a blank or whitespace-only header contains data. Completely empty unnamed columns are silently dropped and do not trigger this error.

See Also

`ternG` for grouped comparisons, `ternD` for descriptive statistics.

Examples

```
# Load a messy CSV and preprocess it
path   <- system.file("extdata/csv", "tern_colon_messy.csv",
                      package = "TernTables")
raw    <- read.csv(path, stringsAsFactors = FALSE)
result <- ternP(raw)

# Access cleaned data
result$clean_data

# Review preprocessing feedback
result$feedback

# Sparse rows flagged (>50% missing), retained but not removed
result$sparse_rows
```

`tern_colon` *Colon Cancer Recurrence Data (Example Dataset)*

Description

A processed subset of the colon dataset restricted to the recurrence endpoint (`etype == 1`), providing one row per patient. Variables have been relabelled with clinically descriptive names and factor levels suitable for direct use in TernTables functions. This dataset is provided as a ready-to-use example for demonstrating `ternD()` and `ternG()` functionality.

Usage

```
tern_colon
```

Format

A tibble with 929 rows and 12 variables:

ID Integer patient identifier.

Age_Years Age at study entry (years).

Sex Patient sex: `"Female"` or `"Male"`.

Colonic_Obstruction Colonic obstruction present: `"N"` or `"Y"`.

Bowel_Perforation Bowel perforation present: `"N"` or `"Y"`.

Positive_Lymph_Nodes_n Number of positive lymph nodes detected.

Over_4_Positive_Nodes More than 4 positive lymph nodes: `"N"` or `"Y"`.

Tumor_Adherence Tumour adherence to surrounding organs: `"N"` or `"Y"`.

Tumor_Differentiation Tumour differentiation grade: `"Well"`, `"Moderate"`, or `"Poor"`.

Extent_of_Local_Spread Depth of tumour penetration: `"Submucosa"`, `"Muscle"`, `"Serosa"`, or `"Contiguous Structures"`.

Recurrence Recurrence status: `"No Recurrence"` or `"Recurrence"`.

Treatment_Arm Randomised treatment: `"Levamisole + 5FU"`, `"Levamisole"`, or `"Observation"`.

Source

Derived from `colon` (Laurie et al., 1989). See `colon` for full provenance. Pre-processing script: `data-raw/tern_colon.R`.

Examples

```
data(tern_colon)
head(tern_colon)
```

`val_format` *Format a mean +/- SD string*

Description

Format a mean +/- SD string

Usage

```
val_format(mean, sd)
```

Arguments

`mean`	Numeric mean value. Formatted to 1 decimal place.
`sd`	Numeric standard deviation. Formatted to 1 decimal place.

Value

A character string of the form `"X.X \u00b1 Y.Y"` where both values are rendered to 1 decimal place using fixed-point notation.

`val_p_format` *Format a P value for reporting*

Description

Format a P value for reporting

Usage

```
val_p_format(p, digits = 3)
```

Arguments

`p`	Numeric P value in the range [0, 1]. `NA` values are returned as `NA_character_`. Values >= 1 (or rounding to >= 1) are returned as e.g. `">0.999"`.
`digits`	Integer; number of decimal places for reported P values. Default is 3. Note: for $p < 0.001$, the value is reported in scientific notation with 1 significant figure regardless of `digits` (e.g., `8E-4`).

Value

A character string. Values < 0.001 are formatted in scientific notation with 1 significant figure (e.g., `"8E-4"`). All other values use fixed-point notation rounded to `digits` decimal places.

`word_export` *Export TernTables output to a formatted Word document*

Description

Export TernTables output to a formatted Word document

Usage

```
word_export(
  tbl,
  filename,
  round_intg = FALSE,
  font_size = 9,
  category_start = NULL,
  manual_italic_indent = NULL,
  manual_underline = NULL,
  table_caption = NULL,
  table_footnote = NULL,
  line_break_header = getOption("TernTables.line_break_header", TRUE)
)
```

Arguments

`tbl`	A tibble created by ternG or ternD
`filename`	Output file path ending in .docx
`round_intg`	Logical; if TRUE, adds note about integer rounding. Default is FALSE.
`font_size`	Numeric; font size for table body. Default is 9.
`category_start`	Named character vector specifying category headers. Names are header label text; values are anchor variable names – either the original column name or the cleaned display name (both forms accepted).
`manual_italic_indent`	Character vector of display variable names (post-cleaning) to force into italicized and indented formatting, matching the appearance of factor sub-category rows (e.g., levels of a multi-category variable). Use this for rows that should visually appear as sub-items but are not automatically detected as such.
`manual_underline`	Character vector of display variable names (post-cleaning) to force into underlined formatting, matching the appearance of multi-category variable header rows. Use this for rows that should visually appear as section headers but are not automatically detected as such.
`table_caption`	Optional character string to display as a caption above the table in the Word document. Rendered as size 11 Arial bold, single-spaced with a small gap before the table. Default is `NULL` (no caption).

`table_footnote`	Optional character string to display as a footnote below the table in the Word document. Rendered as size 6 Arial italic. A double-bar border is applied above and below the footnote row. Default is `NULL` (no footnote).
`line_break_header`	Logical; if `TRUE` (default), column headers are wrapped with `\n` – group names break on spaces, sample size counts move to a second line, and the first column header includes a category hierarchy label. Set to `FALSE` to suppress all header line breaks. Can also be set package-wide via `options(TernTables.line_break_header = FALSE)`.

Value

Invisibly returns the path to the written Word file.

Examples

```
data(tern_colon)
tbl <- ternD(tern_colon, exclude_vars = c("ID"), methods_doc = FALSE)
word_export(
  tbl      = tbl,
  filename = file.path(tempdir(), "descriptive.docx"),
  category_start = c(
    "Patient Demographics"  = "Age (yr)",
    "Tumor Characteristics" = "Positive Lymph Nodes (n)"
  )
)
```

`write_cleaning_doc` *Write a cleaning summary document for ternP output*

Description

Generates a Word document summarising the preprocessing transformations applied by `ternP`. Only sections for triggered transformations are written; if the data required no preprocessing, a single sentence stating that is produced instead. The document can be attached to a data-management log or supplemental materials.

Usage

```
write_cleaning_doc(result, filename = "cleaning_summary.docx")
```

Arguments

`result`	A `ternP_result` object returned by `ternP`.
`filename`	Output file path ending in `.docx`. Default is `"cleaning_summary.docx"` in the current working directory.

Value

Invisibly returns the path to the written Word file.

See Also

`ternP`, `write_methods_doc`

Examples

```
path   <- system.file("extdata/csv", "tern_colon_messy.csv",
                      package = "TernTables")
raw    <- read.csv(path, stringsAsFactors = FALSE)
result <- ternP(raw)
write_cleaning_doc(result, filename = file.path(tempdir(), "cleaning_summary.docx"))
```

`write_methods_doc` *Write a methods section document for use with TernTables output*

Description

Generates a Word document containing boilerplate methods text for all three table types produced by TernTables (descriptive, two-group comparison, and three-or-more-group comparison). Each section is headed by a clear label so the user can copy the relevant paragraph directly into a manuscript. When called from `ternG`, the two-group or multi-group section is populated with the statistical tests that were actually used; all other sections use generic boilerplate. When called from `ternD`, all comparison sections use generic boilerplate.

Usage

```
write_methods_doc(
  tbl,
  filename,
  n_levels = 2,
  OR_col = FALSE,
  source = "ternG",
  post_hoc = FALSE
)
```

Arguments

`tbl`	A tibble created by `ternG` or `ternD`, or `NULL` when generating a generic document.
`filename`	Output file path ending in `.docx`.
`n_levels`	Number of group levels used in `ternG` (2 for two-group, 3+ for multi-group). Ignored when called from `ternD`.
`OR_col`	Logical; whether odds ratios were calculated. Default `FALSE`.

`source`	Character; `"ternG"` or `"ternD"`. Controls which section is populated with dynamic test information. Default `"ternG"`.
`post_hoc`	Logical; whether pairwise post-hoc testing was requested (`post_hoc = TRUE` in `ternG`). When `TRUE` and `n_levels >= 3`, the three-group methods paragraph is updated to describe the post-hoc test pairing (Games-Howell or Dunn's + Holm). Default `FALSE`.

Value

Invisibly returns the path to the written Word file.

Examples

```
data(tern_colon)
tbl <- ternG(tern_colon, exclude_vars = c("ID"), group_var = "Recurrence",
            methods_doc = FALSE)
write_methods_doc(tbl, filename = file.path(tempdir(), "methods.docx"))
```

Index