

Package ‘tabbitR’

February 9, 2026

Title Weighted Cross-Tabulations Exported to 'Excel'

Version 0.1.3

Description Produces weighted cross-tabulation tables for one or more outcome variables across one or more breakdown variables, and exports them directly to 'Excel'. For each outcome-by-breakdown combination, the package creates a weighted percentage table and a corresponding unweighted count table, with transparent handling of missing values and light, readable formatting. Designed to support social survey analysis workflows that require large sets of consistent, publication-ready tables.

Language en-GB

License MIT + file LICENSE

Encoding UTF-8

Imports haven, openxlsx, stats

RoxygenNote 7.3.3

URL <https://github.com/smmcandrew/tabbitR>

BugReports <https://github.com/smmcandrew/tabbitR/issues>

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Author Siobhan McAndrew [aut, cre]

Maintainer Siobhan McAndrew <siobhan.mcandrew@gmail.com>

Repository CRAN

Date/Publication 2026-02-09 13:50:02 UTC

Contents

tabbit_excel	2
--------------	---

Index	5
--------------	---

tabbit_excel

*Write Weighted Crosstab Tables To An Excel Workbook***Description**

`tabbit_excel()` produces weighted percentage tables and unweighted counts for one or more outcome variables, optionally including missing responses and/or row percentages. Output is written to an Excel file with one sheet per breakdown variable by default (or all results in a single sheet if `by_breakdown = FALSE`).

Usage

```
tabbit_excel(
  data,
  vars,
  breakdown,
  file,
  wtvar,
  row_pct = FALSE,
  decimals = 1L,
  nooverall = FALSE,
  nototal = FALSE,
  missingasrow = FALSE,
  nomissing = FALSE,
  by_breakdown = TRUE,
  sheet_base = "Frequencies",
  ...
)
```

Arguments

<code>data</code>	A data frame.
<code>vars</code>	Character vector of outcome variable names.
<code>breakdown</code>	Character vector of breakdown variables.
<code>file</code>	Path to the Excel file to create.
<code>wtvar</code>	Name of the weight variable (string). Must be present in <code>data</code> .
<code>row_pct</code>	Logical. If FALSE (default), tables show column percentages. If TRUE, tables show row percentages.
<code>decimals</code>	Integer. Number of decimal places for percentages (0-6; default 1).
<code>nooverall</code>	Logical. If TRUE, suppress the "Overall" column (or "Overall column percentage (valid responses)" table).
<code>nototal</code>	Logical. If TRUE, suppress the "Total percent" row in the percentage table (column mode only).
<code>missingasrow</code>	Logical. If TRUE, include "Response missing" as an explicit row in the main percentage table.

nomissing	Logical. If TRUE, drop missing responses from the unweighted N table (and from the percentage table unless <code>missingasrow</code> = TRUE).
by_breakdown	Logical. If TRUE (default), create one sheet per breakdown variable. If FALSE, stack all results into a single sheet called <code>sheet_base</code> .
sheet_base	Sheet name to use when <code>by_breakdown</code> = FALSE.
...	For future extension.

Details

For each outcome variable in `vars` and each breakdown variable in `breakdown`, `tabbit_excel()`:

- computes weighted percentages (by column or by row),
- optionally adds an overall distribution across breakdowns,
- optionally adds a "Total percent" row (column mode only),
- handles missing outcomes either as a separate "Response missing" row or by excluding them, with or without a separate "Missing percent" line, and
- writes a corresponding unweighted N table including (or excluding) missing responses.

A light formatting layer is applied using the `openxlsx` package:

- table headers: bold, with top and bottom borders,
- row labels: bold,
- total rows ("Total percent" and "Column totals"): bold, with top and bottom borders, and
- missing-percentage row: italic.

Value

Invisibly, the file path of the created workbook (a character string).

Examples

```
out_file <- tempfile(fileext = ".xlsx")

df <- data.frame(
  courteous = factor(c("Definitely true", "Mostly true", NA, "Mostly false")),
  listener = factor(c("Often", "Sometimes", "Never", NA)),
  sex = factor(c("Male", "Female", "Female", "Male")),
  agegrp1 = factor(c("18-34", "35-54", "18-34", "55+")),
  weight = c(1, 1.5, 0.8, 1.2)
)

tabbit_excel(
  data = df,
  vars = c("courteous", "listener"),
  breakdown = c("sex", "agegrp1"),
  file = out_file,
  wtvar = "weight",
  row_pct = FALSE,
```

```
decimals      = 1L,  
nooverall    = FALSE,  
nototal      = FALSE,  
missingasrow  = FALSE,  
nomissing    = FALSE  
)
```

Index

[tabbit_excel](#), 2