Package 'epitab'

July 4, 2018

Type Package

Title Flexible Contingency Tables for Epidemiology

Version 0.2.2

Author Stuart Lacy

Maintainer Stuart Lacy <stuart.lacy@york.ac.uk>

Description Builds contingency tables that cross-tabulate multiple categorical variables and also calculates various summary measures. Export to a variety of formats is supported, including: 'HTML', 'LaTeX', and 'Excel'.

URL https://github.com/stulacy/epitab

License GPL-2

Imports kableExtra, knitr, MASS, survival, xml2

Encoding UTF-8

LazyData true

RoxygenNote 6.0.1

Suggests dplyr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Repository CRAN

Date/Publication 2018-07-04 20:50:03 UTC

R topics documented:

`contingency_table` *Builds a contingency table*

Description

A contingency table provides cross-tabulated frequencies between an outcome of interest and one or more independent variables. This function extends contingency tables to include summary statistics formed both column-wise and row-wise, looking at outcomes and covariates respectively in isolation. This allows for a large amount of flexibility and tables can be drawn for a variety of situations. By default, the `print` method fits these tables to standard R console output, but publication quality tables can be produced using the `neat_table` function. See the vignette for further guidance.

Usage

```
contingency_table(independents, data, outcomes = NULL,
  crosstab_funcs = NULL, row_funcs = NULL, col_funcs = NULL,
  marginal = TRUE)
```

Arguments

`independents`	A named list of independent variables, which will be distributed down the table's rows. The variables must be specified by strings, with the item name used as the column header.
`data`	The data set that contains the columns specified in `cat_vars` and `outcome`.
`outcomes`	The variables to cross-tabulate by. These will be distributed across the table's columns. Specified as a named list of strings. Must correspond to factor or character variables.
`crosstab_funcs`	A list of functions that are applied to every cross-tabulation permutation of `independents` and `outcomes`. The most common function, the frequency, is provided with the package in function `freq`. See the vignette for further guidance.
`row_funcs`	A list of functions that are applied row-wise to the table, one independent variable at a time, providing a value for each level of the factors specified in `independents`. Two functions: `odds_ratio` and `hazard_ratio` come provided with the package. See the vignette for further guidance.
`col_funcs`	A list of functions that are applied column-wise to the table, to every outcome separate from the independent variables. Examples provided with the package included `summary_mean` and `summary_median`, which calculate the mean and median value of a specified continuous variable for each level of the outcome. See the vignette for further guidance.
`marginal`	Whether to include the counts of each level of `cat_vars`, the marginal frequency.

Value

An S3 object of class `contintab`, that provides the cell contents as a matrix of strings.

Examples

```
# This example uses a dummy data set of whether an individual was treated or not
treat <- data.frame(age=abs(rnorm(100, 60, 20)),
                    sex=factor(sample(c("M", "F"), 100, replace=TRUE)),
                    variant=factor(sample(c("A", "B"), 100, replace=TRUE)),
                    treated=factor(sample(c("Yes", "No"), 100, replace=TRUE),
                                   levels=c("Yes", "No")))
treat$agebin <- cut(treat$age, breaks=c(0, 40, 60, 80, 9999),
                    labels=c("0-40", "41-60", "61-80", "80+"))

# Displays a standard contingency table
contingency_table(list("Age"='agebin', "Sex"='sex'),
                  outcomes=list('Treated'='treated'),
                  crosstab_funcs=list(freq()),
                  data=treat)

# Continuous variables can be summarised with respect to the outcome
# by using col_funcs
contingency_table(list("Age"='agebin', "Sex"='sex'),
                  outcomes=list('Treated'='treated'),
                  crosstab_funcs=list(freq()),
                  col_funcs=list("Mean age"=summary_mean('age')),
                  data=treat)

# Regression coefficients can be added using row_funcs
contingency_table(list("Age"='agebin', "Sex"='sex'),
                   treat,
                   outcomes=list('Treated'='treated'),
                   crosstab_funcs=list(freq()),
                   row_funcs=list("Odds ratio"=odds_ratio('treated'),
                             "Adjusted odds ratio"=odds_ratio('treated', adjusted=TRUE)))
```

`freq` *Builds a function to calculate cross-tabulated frequencies*

Description

Builds a function to calculate cross-tabulated frequencies

Usage

```
freq(proportion = c("column", "row", "none"), display = c("percentage",
  "ratio"), digits = 3, missing = NULL)
```

Arguments

proportion	Whether to display proportions in brackets after the counts as either row-wise or column-wise, or not at all.
display	How to display the proportions if required.
digits	The number of digits to specify proportion to.
missing	Specifies covariates that shouldn't be included in calculating proportions for column-wise summaries.

Value

A function that calculates the frequency of a cell in a contingency table.

---

hazard_ratio *Builds a function used to calculate hazard ratios.*

Description

Builds a function to fit a Cox model to the outcome survival object and returns the hazard ratios for each covariate level.

Usage

```
hazard_ratio(outcome, adjusted = FALSE, relevel_baseline = FALSE,
  digits = 2, ci = TRUE)
```

Arguments

outcome	The dependent variable, specifies a Surv object as a string. For example, hazard_ratio("Surv(time, st
adjusted	Whether to adjust for the other covariates, specified by independents argument to contingency_table.
relevel_baseline	Whether to use the largest level as the baseline.
digits	The number of digits to display.
ci	Whether to include a confidence interval in parentheses after the estimate.

Details

See documentation for contingency_table and vignette for usage.

Value

A function that is used to calculate hazard ratios.

neat_table *Displays a neatly formatted contingency table.*

Description

This function provides a default means of converting a contingency table into HTML or LaTeX for publishing. By default, multiple column and row spanning cells are formed to accentuate the hierarchical nature of the data. The output of this function is a kable object and so can be further manipulated.

Usage

```
neat_table(table, format = c("html", "latex"), ...)
```

Arguments

table	A contintab object, output by contingency_table.
format	A string specifying output format passed to knitr::kable. Currently only 'html' and 'pdf' are supported.
...	Other arguments passed to knitr::kable.

Value

A kable object.

Examples

```
# This example uses a dummy data set of whether an individual was treated or not
treat <- data.frame(age=abs(rnorm(100, 60, 20)),
                    sex=factor(sample(c("M", "F"), 100, replace=TRUE)),
                    variant=factor(sample(c("A", "B"), 100, replace=TRUE)),
                    treated=factor(sample(c("Yes", "No"), 100, replace=TRUE),
                                   levels=c("Yes", "No")))
treat$agebin <- cut(treat$age, breaks=c(0, 40, 60, 80, 9999),
                    labels=c("0-40", "41-60", "61-80", "80+"))

tab <- contingency_table(list("Age"='agebin', "Sex"='sex'),
                         outcomes=list('Treated'='treated'),
                         crosstab_funcs=list(freq()),
                         col_funcs=list("Mean age"=summary_mean('age')),
                         data=treat)

# For use in an Rmarkdown that outputs to HTML
neat_table(tab, 'html')

# When outputting to PDF, the \code{booktabs} option produces well-formatted tables
neat_table(tab, 'latex', booktabs=TRUE)
```

odds_ratio *Builds a function used to calculate odds ratios.*

Description

Builds a function to run logistic regression on the outcome variable of interest and return the odds ratios for each covariate level.

Usage

```
odds_ratio(outcome, adjusted = FALSE, relevel_baseline = FALSE,
  digits = 2, ci = TRUE)
```

Arguments

outcome	The dependent variable as a string.
adjusted	Whether to adjust for the other covariates, specified by independents argument to contingency_table.
relevel_baseline	Whether to use the largest level as the baseline.
digits	The number of digits to display.
ci	Whether to include a confidence interval in parentheses after the estimate.

Details

See documentation for contingency_table and vignette for usage.

Value

A function that is used to calculate odds ratios.

print.contintab *Prints the contingency table as an ASCII table*

Description

Prints the contingency table as an ASCII table

Usage

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

Arguments

x	An object of class contintab return by contingency_table.
...	Options passed to print.

Value

None. Prints a table to standard output.

`summary_mean` *Builds a function to calculate the mean of a continuous variable for each level of an outcome.*

Description

Builds a function to calculate the mean of a continuous variable for each level of an outcome.

Usage

```
summary_mean(var)
```

Arguments

var	A continuous variable name as a string.

Value

A function that calculates the mean value of var for each outcome level.

`summary_median` *Builds a function to calculate the median of a continuous variable for each level of an outcome.*

Description

Builds a function to calculate the median of a continuous variable for each level of an outcome.

Usage

```
summary_median(var)
```

Arguments

var	A continuous variable name as a string.

Value

A function that calculates the median value of var for each outcome level.

Index