Package ‘ezfun’

April 7, 2017

Title Emily C. Zabor's functions

Version 0.1.3

Date 2016-12-10

Description This package contains a number of functions that generate and format results of common procedures for clincial projects into table form for printing in R Markdown Word documents. A few basic utility functions for common procedures are also included.

Depends R (>= 3.1.0)

License GPL-2

LazyData TRUE

Imports survival, aod, cmprsk, clinfun, shiny

RoxygenNote 5.0.1

Author Emily Zabor [aut, cre]

Maintainer Emily Zabor <zabore@mskcc.org>

R topics documented:

bycont *Table of one or more categorical variables by a single continuous variable*

Description

`bycont` takes a list of categorical variabls and returns median(min, max) of the single continuous variable within each level of each categorical variable. Computes Kruskall-Wallis p-values.

Usage

```
bycont(catvars, contvar, dat, pval = TRUE)
```

Arguments

`catvars`	is a list of the categorical variables for the rows of the table e.g. list('Gene1', 'Gene2')
`contvar`	is the continuous variable you want summarized by each categorical variable. Must be in quotes.
`dat`	is the dataset to use for analysis
`pval`	takes the value TRUE or FALSE indicating whether p-values should be computed. Defaults to TRUE. When TRUE, Kruskal-Wallis p-values are produced.

Value

Returns a dataframe

Author(s)

Emily C Zabor <`zabore@mskcc.org`>

lowerchar *Convert to lowercase*

Description

`lowerchar` converts the levels of character variables from upper or mixed case to lower case

Usage

```
lowerchar(dfname)
```

Arguments

`dfname`	is the name of the dataframe on which to perform the action

Value

Nothing is returned from `lowerchar`, the action is simply perfomed on the columns of dataframe `dfname`

Author(s)

Emily C Zabor <`zabore@mskcc.org`>

`mvcoxres` *Format resuls from multivariable Cox regression model*

Description

`mvcoxres` takes a multivariable Cox regression object and formats the resulting HR (95% CI) and p-values into a table

Usage

```
mvcoxres(mod)
```

Arguments

`mod`	is a multivariable Cox regression object

Value

Returns a dataframe

Author(s)

Emily C Zabor <`zabore@mskcc.org`>

`mvcrrres` *Format resuls from multivariable competing risks regression model*

Description

`mvcrrres` takes a multivariable competing risks regression object and puts the resulting HR (95% CI) and p-values into a table

Usage

```
mvcrrres(mod)
```

Arguments

`mod`	is a multivariable Cox regression object

Value

Returns a dataframe

Author(s)

Emily C Zabor <`zabore@mskcc.org`>

mvlogitres *Format resuls from multivariable Cox regression model*

Description

`mvlogitres` takes a multivariable logistic regression object and formats the resulting OR (95% CI) and p-values into a table

Usage

```
mvlogitres(mod)
```

Arguments

`mod`	is a multivariable logistic regression object from `glm`

Value

Returns a dataframe

Author(s)

Emily C Zabor <`zabore@mskcc.org`>

ph2simonApp *Interactive Simon's 2-stage Shiny app*

Description

`ph2simonApp` is simply a Shiny interface for the `ph2simon` function from the `clinfun` package. No arguments need to be passed to the function.

Usage

```
ph2simonApp()
```

Value

The output includes 1) fields to enter the design parameters for the Simon 2-stage Phase II design, 2) R output with `ph2simon` results, 3) a paragraph interpreting the results, and 4) a plot of maximum versus expected number of patients indicating the optimal and minimax results.

sdp *Get p-value from survdiff()*

Description

sdp returns the p-value from the survdiff function

Usage

```
sdp(sd)
```

Arguments

sd	is a survdiff object

Value

Returns a p-value rounded to 3 digits or "<.001" if the p-value is <.001

Author(s)

Emily C Zabor <zabore@mskcc.org>

tab1 *Table 1*

Description

tab1 takes lists of continuous and/or categorical variables and returns Median (spread) for continuous variables and N (%) for categorical variables. Produces a table with both an overall column and columns by another variable.

Usage

```
tab1(contvars, catvars, byvar, dat, col = TRUE, spread = "range",
  pval = TRUE, fisher = TRUE)
```

Arguments

contvars	is a list of the continuous variables you want in the rows e.g. list('Age'). Can be NULL.
catvars	is a list of the categorical variables you want in the rows e.g. list('Gender','Race'). Can be NULL.
byvar	is the categorical variable you want to tabulate by across the columns (needs to be in quotes). Can be NULL.
dat	is the dataset to use for analysis
col	takes the value TRUE or FALSE indicating whether you want column percent (TRUE, default) or row percent (FALSE)

`spread`	takes the value "range" or "iqr" indicating whether you want (min, max) or (Q1, Q3) in summaries of continuous variables. Defaults to "range".
`pval`	takes the value TRUE or FALSE indicating whether p-values should be included. Defaults to TRUE. If TRUE, `kruskal.test` p-values are produced for continuous variables and either `fisher.test` or `chisq.test` p-values are produced for categorical variables. See `param` for testing details for categorical variables.
`fisher`	takes the value TRUE or FALSE. If TRUE, `fisher.test` p-values are produced. If FALSE, `chisq.test` p-values are produced.

Value

Returns a dataframe. If there are warnings or errors from `kruskal.test`, `fisher.test`, or `chisq.test` then NA is returned in place of the p-value.

Author(s)

Emily C Zabor <`zabore@mskcc.org`>

`tabna` *Cross-tabulation with* `useNA = "ifany"`

Description

`tabna` is an implementation of `table` with argument `useNA = "ifany"`

Usage

```
tabna(...)
```

Arguments

`...`	the function takes any standard arguments to `table`

Details

See the help file for `table` for detailed information about possible arguments to the function

Value

`tabna` returns a contingency table with NAs included, if any

Author(s)

Emily C Zabor <`zabore@mskcc.org`>

todate *Convert to date format*

Description

todate converts any POSIXct format variables in the dataframe to date format

Usage

```
todate(dfname)
```

Arguments

dfname	is the name of the dataframe on which to perform the action

Details

Note that this function will mainly apply to dataframes imported using the read_excel function from the readxl package. Dataframes imported using, for example, read.csv instead will have dates in character format and therefore todate will not apply.

Value

Nothing is returned from todate, the action is simply perfomed on the columns of dataframe dfname

Author(s)

Emily C Zabor <zabore@mskcc.org>

uvcoxph *Table of univariable Cox regression results*

Description

uvcoxph takes lists of continuous and/or categorical variables, runs a univariable coxph model for each, and puts the resulting HR (95% CI) and p-value into a table suitable for printing in a Word R Markdown file.

Usage

```
uvcoxph(contvars, catvars, event, time, dat)
```

Arguments

contvars	is a list of the continuous variables you want in the rows e.g. list('Age')
catvars	is a list of the categorical variables you want in the rows e.g. list('Gender','Race')
event	is the event indicator (needs to be in quotes)
time	is the survival time variables (needs to be in quotes)
dat	is the dataset for analysis

Value

Returns a dataframe. If there are warnings or errors from coxph then blank rows are returned.

Author(s)

Emily C Zabor <zabore@mskcc.org>

---

uvcrr *Table of univariable competing risks regression results*

Description

uvcrr takes lists of continuous and/or categorical variables, runs a univariable crr model for each, and puts the resulting HR (95% CI) and p-value into a table suitable for printing in a Word R Markdown file.

Usage

```
uvcrr(contvars, catvars, event, time, dat)
```

Arguments

contvars	is a list of the continuous variables you want in the rows e.g. list('Age')
catvars	is a list of the categorical variables you want in the rows e.g. list('Gender','Race')
event	is the event indicator (needs to be in quotes)
time	is the survival time variables (needs to be in quotes)
dat	is the dataset for analysis

Details

uvcrr uses all function defaults to crr. For example, the failure code is set to 1. See the help file for crr for additional details.

Value

Returns a dataframe. If there are warnings or errors from crr then blank rows are returned.

Author(s)

Emily C Zabor <zabore@mskcc.org>

`uvlm` *Table of univariable linear regression results*

Description

`uvlm` takes lists of continuous and/or categorical variables, calls `lm` to run a linear regression model for each, and returns a table with Est (SE) and p-value for each variable that is suitable for printing in a Word `R Markdown` file.

Usage

```
uvlm(contvars, catvars, out, dat)
```

Arguments

`contvars`	is a list of the continuous variables you want in the rows e.g. list('Age')
`catvars`	is a list of the categorical variables you want in the rows e.g. list('Gender','Race')
`out`	is the continuous outcome variable (needs to be in quotes)
`dat`	is the dataset for analysis

Value

Returns a dataframe. If there are warnings or errors from `lm` then blank rows are returned.

Author(s)

Emily C Zabor <`zabore@mskcc.org`>

`uvlogit` *Table of univariable logistic regression results*

Description

`uvlogit` takes lists of continuous and/or categorical variables, calls `glm` to run a logistic regression model for each, and returns a table with OR (95 Word `R Markdown` file.

Usage

```
uvlogit(contvars, catvars, out, dat)
```

Arguments

`contvars`	is a list of the continuous variables you want in the rows e.g. list('Age')
`catvars`	is a list of the categorical variables you want in the rows e.g. list('Gender','Race')
`out`	is the binary outcome variable. Must be coded 0/1. (needs to be in quotes)
`dat`	is the dataset for analysis

Value

Returns a dataframe. If there are warnings or errors from `glm` then blank rows are returned.

Author(s)

Emily C Zabor <zabore@mskcc.org>

`uvsurv` *Table of univariable survival analysis results*

Description

`uvsurv` takes lists of continuous and/or categorical variables. For continuous variables, `coxph` returns HR (95% CI) and log-rank p-values. For categorical variables, `coxph` returns HR (95% CI) and log-rank p-values and `survfit` produces median survival (95% CI) and a survival estimate at a specified time. Results are put into a table suitable for printing in a Word `R Markdown` file.

Usage

```
uvsurv(contvars, catvars, event, time, test, dat)
```

Arguments

`contvars`	is a list of the continuous variables you want in the rows e.g. list('Age')
`catvars`	is a list of the categorical variables you want in the rows e.g. list('Gender','Race')
`event`	is the survival event indicator (needs to be in quotes)
`time`	is the survival time variable (needs to be in quotes)
`test`	is the timepoint you would like to estimate, in whatever units the survival time is in
`dat`	is the dataset to use for analysis

Value

Returns a dataframe

Author(s)

Emily C Zabor <zabore@mskcc.org>

Index