Package 'zebu'

October 24, 2017

Type Package

Title Local Association Measures

Version 0.1.2

Date 2017-10-21

Author Olivier M. F. Martin [aut, cre],
Michel Ducher [aut]

Maintainer Olivier M. F. Martin <oliviermfmartin@gmail.com>

Description Implements the estimation of local (and global) association measures: Ducher's Z, pointwise mutual information and normalized pointwise mutual information. The significance of local (and global) association is accessed using p-values estimated by permutations. Finally, using local association subgroup analysis, it identifies if the association between variables is dependent on the value of another variable.

URL http://github.com/oliviermfmartin/zebu

BugReports https://github.com/oliviermfmartin/zebu/issues

Depends R (>= 2.10)

License GPL-3

LazyData true

Imports ggplot2, foreach, iterators, reshape2, plyr, utils

Suggests knitr, rmarkdown, devtools

VignetteBuilder knitr

RoxygenNote 6.0.1

NeedsCompilation no

Repository CRAN

Date/Publication 2017-10-24 14:05:08 UTC

R topics documented:

`estimate_prob` *Estimate marginal and multivariate probabilities*

Description

Maximum-likelihood estimation of marginal and multivariate observed and expected independence probabilities. Marginal probability refers to probability of each factor per individual column. Multivariate probability refer to cross-classifying factors for all columns.

Usage

```
estimate_prob(x)
```

Arguments

x	data.frame or matrix.

Value

List containing the following values:

- margins: a list of marginal probabilities. Names correspond to colnames(x).
- observed: observed multivariate probability array.
- expected: expected multivariate probability array

Examples

```
# This is what happens behind the curtains in the 'lassie' function
# Here we compute the association between the 'Girth' and 'Height' variables
# of the 'trees' dataset

# 'select' and 'continuous' take column numbers or names
select <- c('Girth', 'Height') # select subset of trees
continuous <-c(1, 2) # both 'Girth' and 'Height' are continuous
```

```
# equal-width discretization with 3 bins
breaks <- 3

# Preprocess data: subset, discretize and remove missing data
pre <- preprocess(trees, select, continuous, breaks)

# Estimates marginal and multivariate probabilities from preprocessed data.frame
prob <- estimate_prob(pre$pp)

# Computes local and global association using Ducher's Z
lam <- local_association(prob, measure = 'z')
```

`format.lassie` *Format a lassie object*

Description

Formats a `lassie` object for printing to console (see `print.lassie`) and for writing to a file (see `write.lassie`). Melts probability or local association measure arrays into a data.frame.

Usage

```
## S3 method for class 'lassie'
format(x, what_x, range, what_range, what_sort, decreasing,
  na.rm, ...)
```

Arguments

`x`	`lassie` S3 object.
`what_x`	vector specifying values to be returned: • 'local': local association measure values (default). • 'obs': observed probabilities. • 'exp': expected probabilities. • 'local_p': p-value of local association (after running `permtest`).
`range`	range of values to be retained (vector of two numeric values).
`what_range`	character specifying what value `range` refers to (same options as `what_x`). By default, takes the first value in `what_x`.
`what_sort`	character specifying according to which values should `x` be sorted (same options as `what_x`). By default, takes the first value in `what_x`.
`decreasing`	logical value specifying sort order.
`na.rm`	logical value indicating whether NA values should be stripped.
`...`	other arguments passed on to methods. Not currently used.

See Also

`lassie`

lassie *Local Association Measures*

Description

Estimates local (and global) association measures: Ducher's Z and pointwise mutual information and normalized pointwise mutual information.

Usage

```
lassie(x, select, continuous, breaks, measure = "z", default_breaks = 4)
```

Arguments

x	data.frame or matrix.
select	optional vector of column numbers or column names specifying a subset of data to be used. By default, uses all columns.
continuous	optional vector of column numbers or column names specifying continuous variables that should be discretized. By default, assumes that every variable is categorical.
breaks	numeric vector or list passed on to cut to discretize continuous variables. When a numeric vector is specified, break points are applied to all continuous variables. In order to specify variable-specific breaks, lists are used. List names identify variables and list values identify breaks. List names are column names (not numbers). If a continuous variable has no specified breaks, then default_breaks will be applied.
measure	name of measure to be used: • 'z': Ducher's 'z'. • 'pmi': Pointwise mutual information (in bits). • 'npmi': Normalized pointwise mutual information.
default_breaks	default break points for discretizations. Same syntax as in cut.

Value

An instance of S3 class lassie with the following objects:

- data: raw and preprocessed data.frames (see preprocess).
- prob probability arrays (see estimate_prob).
- global global association (see local_association).
- local local association arrays (see local_association).
- lassie_params parameters used in lassie.

See Also

Results can be visualized using `plot.lassie` and `print.lassie` methods. `plot.lassie` is only available in the bivariate case and returns a tile plot representing the probability or local association measure matrix. `print.lassie` shows an array or a data.frame.

Results can be saved using `write.lassie`.

The `permtest` function accesses the significance of local and global association values using p-values estimated by permutations.

The `subgroups` function identifies if the association between variables is dependent on the value of another variable.

Examples

```
# In this example, we will use the 'mtcars' dataset

# Selecting a subset of mtcars.
# Takes column names or numbers.
# If nothing was specified, all variables would have been used.
select <- c('mpg', 'cyl') # or select <- c(1, 2)

# Specifying 'mpg' as a continuous variables using column numbers
# Takes column names or numbers.
# If nothing was specified, all variables would have been used.
continuous <- 'mpg' # or continuous <- 1

# How should breaks be specified?
# Specifying equal-width discretization with 5 bins for all continuous variables ('mpg')
# breaks <- 5

# Specifying user-defined breakpoints for all continuous variables.
# breaks <- c(10, 15, 25, 30)

# Same thing but only for 'mpg'.
# Here both notations are equivalent because 'mpg' is the only continuous variable.
# This notation is useful if you wish to specify different break points for different variables
# breaks <- list('mpg' = 5)
# breaks <- list('mpg' = c(10, 15, 25, 30))

# Calling lassie
# Not specifying breaks means that the value in default_breaks (4) will be used.
las <- lassie(mtcars, select = c(1, 2), continuous = 1)

# Print local association to console as an array
print(las)

# Print local association and probabilities
# Here only rows having a positive local association are printed
# The data.frame is also sorted by observed probability
print(las, type = 'df', range = c(0, 1), what_sort = 'obs')

# Plot results as heatmap
```

```
plot(las)

# Plot observed probabilities using different colours
plot(las, what_x = 'obs', low = 'white', mid = 'grey', high = 'black', text_colour = 'red')

# Write results to text file
write.lassie(las, file = 'test.csv')

# Retrieve results
lassie_df <- read.table('test.csv', sep = ',', header = TRUE)
```

`lassie_get` *Return the value of 'lassie' object*

Description

Subroutine for `lassie` methods. Tries to retrieve a value from a `lassie` object and gives an error if value does not exist.

Usage

```
lassie_get(x, what_x)
```

Arguments

`x`	`lassie` S3 object.
`what_x`	vector specifying values to be returned: • 'local': local association measure values (default). • 'obs': observed probabilities. • 'exp': expected probabilities. • 'local_p': p-value of local association (after running `permtest`).

Value

Corresponding array contained in `lassie` object.

Examples

```
las <- lassie(trees)
las_array <- lassie_get(las, 'local')
```

`local_association` *Local Association Measures*

Description

Subroutines called by `lassie` to compute local and global association measures from a list of probabilities.

Usage

```
local_association(x, measure)

duchers_z(x)

pmi(x, normalize = FALSE)

npmi(x)
```

Arguments

`x`	list of probabilities as outputted by `estimate_prob`.
`measure`	name of measure to be used: • 'z': Ducher's 'z'. • 'pmi': Pointwise mutual information (in bits). • 'npmi': Normalized pointwise mutual information.
`normalize`	Normalizes pointwise mutual information when calling `pmi`

Details

- `local_association(x, measure = 'z')` is equivalent to `duchers_z(x)`.
- `local_association(x, measure = 'pmi')` is equivalent to `pmi(x)`.
- `local_association(x, measure = 'npmi')` is equivalent to `npmi(x)` and `pmi(x, normalize = TRUE)`.

Value

List containing the following values:

- local: local association array (may contain NA, NaN and Inf values).
- global: global association numeric value.

See Also

`lassie`

Examples

```
# This is what happens behind the curtains in the 'lassie' function
# Here we compute the association between the 'Girth' and 'Height' variables
# of the 'trees' dataset

# 'select' and 'continuous' take column numbers or names
select <- c('Girth', 'Height') # select subset of trees
continuous <-c(1, 2) # both 'Girth' and 'Height' are continuous

# equal-width discretization with 3 bins
breaks <- 3

# Preprocess data: subset, discretize and remove missing data
pre <- preprocess(trees, select, continuous, breaks)

# Estimates marginal and multivariate probabilities from preprocessed data.frame
prob <- estimate_prob(pre$pp)

# Computes local and global association using Ducher's Z
lam <- local_association(prob, measure = 'z')
```

`permtest` *Permutation test for local and global association measures*

Description

Permutation test: statistical significance of local and global association measures

Usage

```
permtest(x, group = as.list(colnames(x$data$pp)), nb = 1000L,
  p_adjust = "BH", progress_bar = FALSE)
```

Arguments

`x`	`lassie` S3 object.
`group`	list of column names specifying which columns should be permuted together. This is useful for the multivariate case, for example, when there is many dependant variables and one independant variable. By default, permutes all columns separetely.
`nb`	number of resampling iterations.
`p_adjust`	multiple testing correction method. (see `p.adjust.methods` for a list of methods).
`progress_bar`	logical specifying if progress bar should be displayed.

Value

permtest returns an S3 object of class lassie and permtest. Adds the following to the lassie object x:

- global_p: global association p-value.
- local_p: array of local association p-values.
- global_perm: numeric global association values obtained with permutations.
- local_perm: matrix local association values obtained with permutations. Column number correspond to positions in local association array after converting to numeric (e.g. local_perm[, 1] corresponds to local[1]).
- perm_params: parameters used when calling permtest (nb and p_adjust).

See Also

lassie

Examples

```
# Calling lassie on cars dataset
las <- lassie(cars)

# Permutation test using default settings
permtest(las)
```

---

plot.lassie *Plot a lassie object*

Description

Plots a lassie object as a tile plot using the ggplot2 package. Only available for bivariate association.

Usage

```
## S3 method for class 'lassie'
plot(x, what_x = "local", digits = 3, low = "blue",
  mid = "white", high = "red", na = "purple", text_colour = "black",
  text_size, limits, midpoint, ...)
```

Arguments

x	lassie S3 object.
what_x	vector specifying values to be returned: • 'local': local association measure values (default). • 'obs': observed probabilities.

- 'exp': expected probabilities.
- 'local_p': p-value of local association (after running permtest).

digits	integer indicating the number of decimal places.
low	colour for low end of the gradient.
mid	colour for midpoint of the gradient.
high	colour for high end of the gradient.
na	colour for NA values.
text_colour	colour of text inside cells.
text_size	integer indicating text size inside cells.
limits	limits of gradient.
midpoint	midpoint of gradient.
...	other arguments passed on to methods. Not currently used.

See Also

lassie

---

preprocess *Preprocess data*

Description

Subroutine called by lassie. Discretizes, subsets and remove missing data from a data.frame.

Usage

```
preprocess(x, select, continuous, breaks, default_breaks = 4)
```

Arguments

x	data.frame or matrix.
select	optional vector of column numbers or column names specifying a subset of data to be used. By default, uses all columns.
continuous	optional vector of column numbers or column names specifying continuous variables that should be discretized. By default, assumes that every variable is categorical.
breaks	numeric vector or list passed on to cut to discretize continuous variables. When a numeric vector is specified, break points are applied to all continuous variables. In order to specify variable-specific breaks, lists are used. List names identify variables and list values identify breaks. List names are column names (not numbers). If a continuous variable has no specified breaks, then default_breaks will be applied.
default_breaks	default break points for discretizations. Same syntax as in cut.

Value

List containing the following values:

- raw: raw subsetted data.frame
- pp: discretized, subsetted and complete data.frame
- select
- continuous
- breaks
- default_breaks

Examples

```
# This is what happens behind the curtains in the 'lassie' function
# Here we compute the association between the 'Girth' and 'Height' variables
# of the 'trees' dataset

# 'select' and 'continuous' take column numbers or names
select <- c('Girth', 'Height') # select subset of trees
continuous <-c(1, 2) # both 'Girth' and 'Height' are continuous

# equal-width discretization with 3 bins
breaks <- 3

# Preprocess data: subset, discretize and remove missing data
pre <- preprocess(trees, select, continuous, breaks)

# Estimates marginal and multivariate probabilities from preprocessed data.frame
prob <- estimate_prob(pre$pp)

# Computes local and global association using Ducher's Z
lam <- local_association(prob, measure = 'z')
```

---

print.lassie *Print a lassie object*

Description

Print a lassie object as an array or a data.frame.

Usage

```
## S3 method for class 'lassie'
print(x, type, what_x, range, what_range, what_sort,
  decreasing, na.rm, ...)
```

Arguments

x	lassie S3 object.
type	print style: 'array' for array or 'df' for data.frame.
what_x	vector specifying values to be returned: • 'local': local association measure values (default). • 'obs': observed probabilities. • 'exp': expected probabilities. • 'local_p': p-value of local association (after running permtest).
range	range of values to be retained (vector of two numeric values).
what_range	character specifying what value range refers to (same options as what_x). By default, takes the first value in what_x.
what_sort	character specifying according to which values should x be sorted (same options as what_x). By default, takes the first value in what_x.
decreasing	logical value specifying sort order.
na.rm	logical value indicating whether NA values should be stripped.
...	other arguments passed on to methods. Not currently used.

See Also

lassie, permtest

---

subgroups *Local Association Subgroup Analysis*

Description

Identifies if the local association between variables (named associated variables) is dependent on the value of an another variable (named interacting variable). Associated variables are specified by las. Interacting variable(s) values are specified by x.

Usage

```
subgroups(las, x, select, continuous, breaks, default_breaks = 4,
  thresholds = c(-0.05, 0.05), significance, alpha = 0.01)
```

Arguments

las	lassie S3 object. Corresponds to associated variables.
x	data.frame or matrix. Corresponds to interacting variable(s) specified by select.
select	optional vector of column numbers or column names specifying a subset of data to be used. By default, uses all colnames in x except those in las object.

continuous	optional vector of column numbers or column names specifying continuous variables that should be discretized. By default, assumes that every variable is categorical.
breaks	numeric vector or list passed on to cut to discretize continuous variables. When a numeric vector is specified, break points are applied to all continuous variables. In order to specify variable-specific breaks, lists are used. List names identify variables and list values identify breaks. List names are column names (not numbers). If a continuous variable has no specified breaks, then default_breaks will be applied.
default_breaks	default break points for discretizations. Same syntax as in cut.
thresholds	vector specifying respectively the negative and the positive association threshold. Local association values between these thresholds are considered independent. Values not contained in this range are classified as independent.
significance	optional logical value specifying if only non-significant local association values should be considered as independent. Only available if las is also a permtest object.
alpha	alpha error level. Local association with p-values above this value are considered as independent. Only available if las is also a permtest object.

Details

Associated variables events are recoded into a subgroup variable according to local association values (and eventually significance) into 'positive', 'negative' and 'independent'. This is specified by the thresholds, significance and alpha arguments. The local (and global) association between the new subgroup variable and the interacting variable is then estimated using lassie.

Value

An instance of S3 class lassie.

See Also

Significance can be accessed using a permutation test: permtest.

Examples

```
# In this example, we will use the zebu 'trial' dataset.
# See vignette example for more detailed explanation

# 'trial' corresponds to a simulated clinical trial where patient recovery
# is dependent on drug intake ('drug') and resistance status ('resistance').
# Patient recovery is monitored by a biomarker (continuous variable from 0 to 1)
# Patients with post-treatment biomarker ('postbiom') above 0.7 is have recovered.

# Load 'trial' dataset
data(trial)

# Compute the association between drug intake and patient recovery
las <- lassie(trial,
```

```
                select = c("drug", "postbiom"),
                continuous = c("postbiom"),
                breaks = c(0, 0.7, 1))

# Permuation test
# Access significance of global and local association
las <- permtest(las)

# Global association between drug intake and recovery but not for all patients
# Being in the drug group is locally independent of having not recovered
print(las)

# Local association subgroup analysis
sub <- subgroups(las, trial, select = "resistance", alpha = 0.01)

# Variable 'resistance' explains differences between sensitive and resistance patients
print(sub)
```

`trial` *Resistance to drug treatment*

Description

Simulated clinical trial where patient recovery is dependent on drug intake and resistance status.

Usage

```
trial
```

Format

A data frame with 100 rows and 3 variables:

drug binary variable (placebo, drug), did patient receive drug

resistance binary variable (sensitive, resistant), is patient resistance to drug

prebiom continuous variable between 0 and 1, biomarker that represents health status of patient before treatment; healthy patients have values around 0.6

postbiom continuous variable between 0 and 1, biomarker that represents health status of patient after treatment; healthy patients have values above 0.6

`write.lassie` *Write a lassie object*

Description

Writes `lassie` object to a file in a table structured format.

Usage

```
write.lassie(x, file, sep = ",", dec = ".", col.names = TRUE,
  row.names = FALSE, quote = TRUE, ...)
```

Arguments

`x`	`lassie` S3 object.
`file`	character string naming a file.
`sep`	the field separator string. Values within each row of `x` are separated by this string.
`dec`	the string to use for decimal points in numeric or complex columns: must be a single character.
`col.names`	either a logical value indicating whether the column names of `x` are to be written along with `x`, or a character vector of column names to be written. See the section on 'CSV files' for the meaning of `col.names = NA`.
`row.names`	either a logical value indicating whether the row names of `x` are to be written along with `x`, or a character vector of row names to be written.
`quote`	a logical value (`TRUE` or `FALSE`) or a numeric vector. If `TRUE`, any character or factor columns will be surrounded by double quotes. If a numeric vector, its elements are taken as the indices of columns to quote. In both cases, row and column names are quoted if they are written. If `FALSE`, nothing is quoted.
`...`	other arguments passed on to write.table.

See Also

`lassie`, `permtest`

zebu *zebu: Local Association Measures*

Description

The zebu package implements the estimation of local (and global) association measures: Ducher's Z, pointwise mutual information and normalized pointwise mutual information. The significance of local (and global) association is accessed using p-values estimated by permutations. Finally, using local association subgroup analysis, it identifies if the association between variables is dependent on the value of another variable.

Functions

`lassie` estimates local (and global) association measures: Ducher's Z, pointwise mutual information and normalized pointwise mutual information.

`permtest` accesses the significance of local (and global) association values using p-values estimated by permutations.

`subgroups` identifies if the association between variables is dependent on the value of another variable.

Index