Package ‘cases’

May 18, 2023

Type Package

Title Stratified Evaluation of Subgroup Classification Accuracy

Version 0.1.1

Description Enables simultaneous statistical inference for the accuracy of multiple classifiers in multiple subgroups (strata). For instance, allows to perform multiple comparisons in diagnostic accuracy studies with co-primary endpoints sensitivity and specificity. (Westphal, Max, and Antonia Zapf. (2021). ``Statistical Inference for Diagnostic Test Accuracy Studies with Multiple Comparisons.'' <arXiv:2105.13469>.)

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Imports bindata, boot, copula, corrplot, dplyr, extraDistr, magrittr, Matrix, multcomp, mvtnorm, ggplot2

Suggests testthat (>= 3.0.0), knitr, readr, rmarkdown, covr, badger, glmnet, splitstackshape

RoxygenNote 7.2.0

VignetteBuilder knitr, rmarkdown

Config/testthat/edition 3

URL `https://github.com/maxwestphal/cases`

BugReports `https://github.com/maxwestphal/cases/issues`

Depends R (>= 2.10)

NeedsCompilation no

Author Max Westphal [aut, cre] (<`https://orcid.org/0000-0002-8488-758X`>)

Maintainer Max Westphal <`max.westphal@steady.ai`>

Repository CRAN

Date/Publication 2023-05-18 08:30:02 UTC

R topics documented:

`cases` `cases` *package*

Description

Enables simultaneous statistical inference for the accuracy of multiple classifiers in multiple subgroups (strata). For instance, allows to perform multiple comparisons in diagnostic accuracy studies with co-primary endpoints sensitivity and specificity. (Westphal, Max, and Antonia Zapf. "Statistical Inference for Diagnostic Test Accuracy Studies with Multiple Comparisons." arXiv:2105.13469 (2021).)

Details

See the vignettes vignette()

`categorize` *Categorize continuous values*

Description

This function allows to split continuous values, e.g. (risk) scores or (bio)markers, into two or more categories by specifying one or more cutoff values.

Usage

```
categorize(
  values,
  cutoffs = rep(0, ncol(values)),
  map = 1:ncol(values),
  labels = NULL
)
```

Arguments

`values`	numeric matrix of continuous values to be categorized. Assume an (n x r) matrix with n observations (subjects) of r continuous values.
`cutoffs`	numeric matrix of dimension m x k. Each row of cutoffs defines a split into k+1 distinct categories. Each row must contain distinct values. In the simplest case, cutoffs is a single column matrix whereby is row defines a binary split (<=t vs. >t). In this case (k=1), cutoffs can also be a numeric vector.
`map`	integer vector of length k with values in 1:r, whereby r = ncol(values). map_l gives the value which column of values should be categorized by ...
`labels`	character of length m (= number of prediction r)

Value

numeric (n x k) matrix with categorical outcomes after categorizing.

Examples

```
set.seed(123)
M <- as.data.frame(mvtnorm::rmvnorm(20, mean=rep(0, 3), sigma=2*diag(3)))
M
categorize(M)
C <- matrix(rep(c(-1, 0, 1, -2, 0, 2), 3), ncol=3, byrow = TRUE)
C
w <- c(1, 1, 2, 2, 3, 3)
categorize(M, C, w)
```

`compare` *Compare predictions and labels*

Description

Compare predictions and labels

Usage

```
compare(
  predictions,
  labels,
  partition = TRUE,
  names = c(specificity = 0, sensitivity = 1)
)
```

Arguments

`predictions`	integer, predicted class
`labels`	integer, true class state (reference standard)
`partition`	logical, should result be split into one matrix per class (TRUE; default) or not (FALSE)
`names`	integer (named), values give data values, names give class names

Value

data matrix with values 1 (correct prediction) and 0 (false prediction)

Examples

```
pred <- matrix(c(1,1,0), 5, 3)
labels <- c(1, 1, 0, 0, 1)
compare(pred, labels, FALSE)
compare(pred, labels, TRUE)
```

`complete_results` *Complete evaluation results*

Description

Complete evaluation results

Usage

```
complete_results(results, benchmark, alpha, analysis)
```

Arguments

`results`	"cases_results" object, i.e. result of evaluate
`benchmark`	numeric, vector of benchmark values
`alpha`	numeric, significance level
`analysis`	character, either "co-primary" or "full"

Details

Not exported, but applied at the end of evaluate by default

Value

"cases_results" object

`cormat_ar1` *Create an AR(1) correlation matrix*

Description

Create an AR(1) correlation matrix

Usage

```
cormat_ar1(m, rho, d = TRUE)
```

Arguments

`m`	integer, dimension
`rho`	numeric, correlation parameter in (0,1)
`d`	binary vector of length m, whereby TRUE/FALSE (alternatively 1/0) indicate active/inactive components of underlying random vector.

Value

$R_{ij} = \rho^{|i-j|}$

cormat_equi *Create an equicorrelation matrix*

Description

Create an equicorrelation matrix

Usage

```
cormat_equi(m, rho, d = TRUE)
```

Arguments

`m`	integer, dimension
`rho`	numeric, correlation parameter in (0,1)
`d`	binary vector of length m, whereby TRUE/FALSE (alternatively 1/0) indicate active/inactive components of underlying random vector.

Value

$R_{ij} = \rho, i \neq j$

data_wdbc *Breast Cancer Wisconsin (Diagnostic) Data Set*

Description

Dataset documentation can be found at the source website and references below.

Usage

```
data_wdbc
```

Format

`data_wdbc`**:**

A data frame with 569 rows (patients) and 31 columns (1 target, 30 features).

Details

The ID variable was removed. Diagnosis (1= malignant, 0 = benign). Feature variables have been renamed.

Source

https://archive.ics.uci.edu/ml/datasets/breast+cancer+wisconsin+(diagnostic)

References

- W.N. Street, W.H. Wolberg and O.L. Mangasarian. Nuclear feature extraction for breast tumor diagnosis. IS&T/SPIE 1993 International Symposium on Electronic Imaging: Science and Technology, volume 1905, pages 861-870, San Jose, CA, 1993.
- O.L. Mangasarian, W.N. Street and W.H. Wolberg. Breast cancer diagnosis and prognosis via linear programming. Operations Research, 43(4), pages 570-577, July-August 1995.

`define_contrast` *Define a contrast (matrix) to specify exact hypothesis system*

Description

Define a contrast (matrix) to specify exact hypothesis system

Usage

```
define_contrast(type = c("raw", "dunnett", "tukey"), comparator = NA)
```

Arguments

`type`	character, either "Raw", "dunnett" or "tukey")
`comparator`	either integer (index of comparator) or character (name of comparator)

Details

"raw" contrast: compare all candidates against specified benchmark values

"dunnett" (all vs. one) contrast: compare all candidates to a single comparator.

"tukey" (all vs. all) contrast: compare all candidates against each other.

Value

`cases_contrast` object to be passed to `evaluate`

Examples

```
define_contrast("dunnett", 1)
```

draw_data *Generate binary data*

Description

Generate binary data

Usage

```
draw_data(
  n = 200,
  prev = c(0.5, 0.5),
  random = FALSE,
  m = 10,
  method = c("roc", "lfc", "pr"),
  pars = list(),
  ...
)
```

Arguments

n	integer, overall sample size
prev	numeric, vector of class prevalences (adding up to 1)
random	logical, random sampling (TRUE) or fixed group sample sizes
m	integer, number of models
method	character, either "roc", "lfc" (multiple subgroups) or "prob" (no subgroups)
pars	list, containing further named parameters passed to draw_data_roc, draw_data_lfc
...	further named parameters passed

Value

generated binary data (possibly stratified for subgroups)

Examples

```
draw_data()
```

`draw_data_lfc` *Generate binary data (LFC model)*

Description

Generate binary data (LFC model)

Usage

```
draw_data_lfc(
  n = 100,
  prev = c(0.5, 0.5),
  random = FALSE,
  m = 10,
  se = 0.8,
  sp = 0.8,
  B = round(m/2),
  L = 1,
  Rse = diag(rep(1, m)),
  Rsp = diag(rep(1, m)),
  modnames = paste0("model", 1:m),
  ...
)
```

Arguments

`n`	integer, total sample size
`prev`	numeric, disease and healthy prevalence (adds up to 1)
`random`	logical, random sampling (TRUE) or fixed prevalence (FALSE)
`m`	integer, number of models
`se`	numeric, sensitivity (length 1)
`sp`	numeric, specificity (length 1)
`B`	integer, between 1 and m, specifies how many sensitivity values are projected to 1
`L`	numeric, worst alternative is computed under side condition Acc <= L (default value L=1 corresponds to true LFC where values are projected to 1)
`Rse`	matrix, correlation matrix for empirical sensitivities (m x m)
`Rsp`	matrix, correlation matrix for empirical specificities (m x m)
`modnames`	character, model names (length m)
`...`	further arguments (currently unused)

Value

Generated binary dataset

Examples

```
data <- draw_data_lfc()
head(data)
```

draw_data_prb *Sample binary data (single sample)*

Description

This function is wrapper for rmvbin.

Usage

```
draw_data_prb(n = 100, pr = c(0.8, 0.8), R = diag(length(pr)))
```

Arguments

n	integer, sample size
pr	numeric, vector with marginal success probabilities
R	matrix, square correlation matrix

Value

a matrix with n rows and length(pr) columns of randomly generated binary (0, 1) data

draw_data_roc *Generate binary data (ROC model)*

Description

Generate binary data (ROC model)

Usage

```
draw_data_roc(
  n = 100,
  prev = c(0.5, 0.5),
  random = FALSE,
  m = 10,
  auc = seq(0.85, 0.95, length.out = 5),
  rho = c(0.25, 0.25),
  dist = c(”normal”, ”exponential”),
  e = 10,
  k = 100,
  delta = 0,
  modnames = paste0("model", 1:m),
  corrplot = FALSE,
  ...
)
```

Arguments

`n` integer, total sample size

`prev` numeric, disease and healthy prevalence (adds up to 1)

`random` logical, random sampling (TRUE) or fixed prevalence (FALSE)

`m` integer, number of models

`auc` numeric, vector of AUCs of biomarkers

`rho` numeric, vector (length 2) of correlations between biomarkers

`dist` character, either "normal" or "exponential" specifying the subgroup biomarker distributions

`e` numeric, emulates better (worse) model selection quality with higher (lower) values of e

`k` integer, technical parameter which adjusts grid size

`delta` numeric, specify importance of sensitivity and specificity (default 0)

`modnames` character, model names (length m)

`corrplot` logical (default: FALSE), if TRUE do not return data but instead plot correlation matrices for final binary data

`...` further arguments (currently unused)

Value

Generated binary dataset

Examples

```
data <- draw_data_roc()
head(data)
```

`evaluate` *Evaluate the accuracy of multiple (candidate) classifiers in several subgroups*

Description

Assess classification accuracy of multiple classifcation rules stratified by subgroups, e.g. in diseased (sensitivity) and healthy (specificity) individuals.

Usage

```
evaluate(
  data,
  contrast = define_contrast("raw"),
  benchmark = 0.75,
  alpha = 0.05,
  alternative = c("two.sided", "greater", "less"),
  adjustment = c("none", "bonferroni", "maxt", "bootstrap", "mbeta"),
  transformation = c("none", "logit"),
  analysis = c("co-primary", "full"),
  regu = FALSE,
  pars = list(),
  ...
)
```

Arguments

`data` list of n_g x m binary matrix or data.frame (n_g observations of m binary decisions), g is the index of subgroups/classes, usually created via `compare`.

`contrast` `cases_contrast` object, specified via `define_contrast`

`benchmark` value to compare against (RHS), should have same length as data.

`alpha` numeric, significance level (default: 0.05)

`alternative` character, specify alternative hypothesis

`adjustment` character, specify type of statistical adjustment taken to address multiplicity

`transformation` character, define transformation to ensure results (e.g. point estimates, confidence limits) lie in unit interval ("none" (default) or "logit")

`analysis` character, "co-primary" or "full"

`regu` numeric vector of length 3, specify type of shrinkage. Alternatively, logical of length one (TRUE := c(2, 1, 1/2), FALSE := c(0, 0, 0))

`pars` further parameters given as named list list(type="pairs", nboot=10000)

`...` additional named parameters, can be used instead of (in in conjunction with) `pars`

Details

Adjustment methods (`adjustment`) and additional parameters (`pars` or `...`):

"none" (default): no adjustment for multiplicity

"bonferroni": Bonferroni adjustment

"maxt": maxT adjustment

"bootstrap": Bootstrap approach

- type: "pairs" (default) or "wild" = type (for adjustment="bootstrap)
- nboot: number of bootstrap draws (default: 5000)
- res_tra: = 0,1,2 or 3 = type of residual transformation of wild boostrap (default = 0: no transformation) (see https://www.math.kth.se/matstat/gru/sf2930/papers/wild.bootstrap.pdf)

"mbeta": A heuristic Bayesian approach which is based on a multivariate beta-binomial model.

- nrep: number of posterior draws (default: 5000)
- lfc_pr: prior probability of 'least-favorable parameter configuration' (default: 1).

Value

cases_results object, which is a list of analysis results

Examples

```
#
data <- draw_data_roc()
evaluate(data)
```

generate_instance_lfc *Generate data sets under least favorable parameter configurations*

Description

Generates a (simulation) instance, a list of multiple datasets to be processed (analyzed) with process_instance. Ground truth parameters (Sensitvity & Specificity) are least-favorable in the sense that the type-I error rate of the subsequently applied multiple test procedures is maximized.

Usage

```
generate_instance_lfc(
  nrep = 10,
  n = 100,
  prev = 0.5,
  random = FALSE,
  m = 10,
  se = 0.8,
  sp = 0.8,
  L = 1,
  rhose = 0,
  rhosp = 0,
  cortype = "equi",
  ...,
  data = NULL,
  job = NULL
)
```

Arguments

nrep	integer, number of instances
n	integer, total sample size
prev	numeric, disease prevalence
random	logical, fixed prevalence (FALSE) or simple random sampling (TRUE)
m	integer, number of candidates
se	numeric
sp	numeric
L	numeric
rhose	numeric
rhosp	numeric
cortype	character, "equi" or "ak1"
...	further arguments
data	ignored (for batchtools compatibility)
job	ignored (for batchtools compatibility)

Details

Utilizes same arguments as draw_data_lfc unless mentioned above.

Value

a list, a single (LFC) simulation instance

generate_instance_roc *Generate data sets under realistic parameter configurations*

Description

Generates a (simulation) instance, a list of multiple datasets to be processed (analyzed) with process_instance. Ground truth parameters (Sensitvity & Specificity) are initially generated according to a generative model whereby multiple decision rules (with different parameter values) are derived by thresholding multiple biomarkers.

Usage

```
generate_instance_roc(
  nrep = 10,
  n = 100,
  prev = 0.5,
  random = FALSE,
  m = 10,
  auc = "seq(0.85, 0.95, length.out = 5)",
  rhose = 0.5,
  rhosp = 0.5,
  dist = "normal",
  e = 10,
  k = 100,
  delta = 0,
  ...,
  data = NULL,
  job = NULL
)
```

Arguments

nrep	integer, number of instances
n	integer, total sample size
prev	numeric, disease prevalence
random	logical, fixed prevalence (FALSE) or simple random sampling (TRUE)
m	integer, number of candidates
auc	numeric
rhose	numeric
rhosp	numeric
dist	character
e	numeric
k	numeric
delta	numeric
...	further arguments
data	ignored (for batchtools compatibility)
job	ignored (for batchtools compatibility)

Details

Utilizes same arguments as draw_data_roc unless mentioned above.

Value

a list, a single (ROC) simulation instance

`process_instance` *Analyze simulated synthetic datasets.*

Description

Process data instances, a list of multiple datasets generated via generate_instance_lfc or generate_instance_roc. This function applies evaluate to all datasets.

Usage

```
process_instance(
  instance = NULL,
  contrast = "cases::define_contrast('raw', NA)",
  benchmark = 0.5,
  alpha = 0.05,
  alternative = "greater",
  adjustment = "none",
  transformation = "none",
  analysis = "co-primary",
  regu = "c(1,1/2,1/4)",
  pars = "list()",
  ...,
  data = NULL,
  job = list(id = NA)
)
```

Arguments

`instance`	generated via generate_instance_lfc or generate_instance_roc.
`contrast`	`cases_contrast` object, specified via `define_contrast`
`benchmark`	value to compare against (RHS), should have same length as data.
`alpha`	numeric, significance level (default: 0.05)
`alternative`	character, specify alternative hypothesis
`adjustment`	character, specify type of statistical adjustment taken to address multiplicity
`transformation`	character, define transformation to ensure results (e.g. point estimates, confidence limits) lie in unit interval ("none" (default) or "logit")
`analysis`	character, "co-primary" (default; only option currently)
`regu`	numeric vector of length 3, specify type of shrinkage. Alternatively, logical of length one (TRUE := c(2, 1, 1/2), FALSE := c(0, 0, 0))
`pars`	further parameters given as named list
`...`	additional named parameters
`data`	ignored (for batchtools compatibility)
`job`	for batchtools compatibility, do not change

Details

Utilizes same arguments as evaluate unless mentioned above.

Value

standardized evaluation results

`visualize` *Visualize evaluation results*

Description

Visualize evaluation results

Usage

```
visualize(x, ...)
```

Arguments

`x,`	a `cases_results` object, see `evaluate`
`...`	further arguments (currently ignored)

Details

+++ early development version (only alternative = "greater" is supported) +++

Value

a ggplot

Index