Package 'UncertainInterval'

April 3, 2018

Version 0.4.7

Date 2018-4-3

Title Uncertain Interval Methods for Cut-Point Determination in Tests

Author Hans Landsheer [aut, cre]

Maintainer Hans Landsheer <j.a.landsheer@uu.nl>

Depends R (>= 3.0.0)

Suggests MASS, car (>= 2.1-2), knitr, pander, rmarkdown

Description Functions for the determination of an uncertain interval, i.e., a range of test scores that are inconclusive and do not allow a diagnosis, other than 'Uncertain' (Reference: J.A. Landsheer (2016) <doi:10.1371/journal.pone.0166007>).

License GPL (>= 2)

URL http://r-forge.r-project.org/scm/?group_id=258

BugReports http://r-forge.r-project.org/tracker/?group_id=258

RoxygenNote 6.0.1

Imports stats, utils, reshape2, rootSolve, nloptr

LazyLoad yes

VignetteBuilder knitr, pander

NeedsCompilation no

Repository CRAN

Date/Publication 2018-04-03 12:25:29 UTC

R topics documented:

`check.data` *Function to check the dataset of individuals with (1) and without (0) the targeted condition.*

Description

Function to check the dataset of individuals with (1) and without (0) the targeted condition.

Usage

```
check.data(ref, test, model = c("kernel", "binormal", "ordinal"))
```

Arguments

`ref`	The reference standard. A column in a data frame or a vector indicating the classification by the reference test. The reference standard must be coded either as 0 (absence of the condition) or 1 (presence of the condition)
`test`	The index test or test under evaluation. A column in a dataset or vector indicating the test results in a continuous scale.
`model`	The model used for estimation. Default = 'kernel'. When model != ordinal, the dataset is checked whether the test has a sufficient number of different values (>= 20).

Details

The first check is whether ref and test have equal length. If not, checkdata is aborted with an error message. The second check is whether ref is coded solely with 0 and 1. If not, check.data is aborted and an error message is shown. The third check is whether ref and test have missing values. If true, list wise deletion is applied and a warning message is shown. The fourth check is whether test is continuous or not. If test has less than 20 different values, a warning message is shown. This test is omitted when ordinal = TRUE.

This function is called from every function that requires data. A call is only useful to check warnings and errors.

Value

Either a valid dataset as data.frame with two variables ref and test or an error message.

Examples

```
#' set.seed(1)
ref=c(rep(0,500), rep(1,500))
test=c(rnorm(500,0,1), rnorm(500,1,1.2))
check.data(ref, test) # model = 'kernel'
```

`get.intersection` *get.intersection Obtain the intersection of two distributions using the kernel method*

Description

`get.intersection` Obtain the intersection of two distributions using the kernel method. Warning: This function does not check the parameters ref and test.

Usage

```
get.intersection(ref, test, model = c("kernel", "binormal", "ordinal"), ...)
```

Arguments

`ref`	The reference standard. A column in a data frame or a vector indicating the classification by the reference test. The reference standard must be coded either as 0 (absence of the condition) or 1 (presence of the condition)
`test`	The index test or test under evaluation. A column in a dataset or vector indicating the test results on a continuous scale.
`model`	The model used for estimating the intersection(s). Default = 'kernel'.
`...`	passing arguments to the kernel density function, other than kernel='gaussian' (default).

Value

A vector of points of intersection, ordered on their density. The tail has the highest density.

References

Landsheer, J. A. (2016). Interval of Uncertainty: An Alternative Approach for the Determination of Decision Thresholds, with an Illustrative Application for the Prediction of Prostate Cancer. PloS One, 11(11), e0166007.

See Also

`density`

Examples

```
ref=c(rep(0,500), rep(1,500))
test=c(rnorm(500,0,1), rnorm(500,1,2))
(get.intersection(ref, test)) # two intersections! Generates warning in other functions!
```

nlopt.ui *Function for the determination of the population thresholds an inconclusive interval for bi-normal distributed test scores.*

Description

Function for the determination of the population thresholds an inconclusive interval for bi-normal distributed test scores.

Usage

```
nlopt.ui(Se = 0.55, Sp = 0.55, mu0 = 0, sd0 = 1, mu1 = 1, sd1 = 1,
  intersection = NULL, start = NULL, print.level = 0)
```

Arguments

Se	(default = .55). Desired sensitivity of the test scores within the uncertain interval. A value <= .5 is not allowed.
Sp	(default = .55). Desired specificity of the test scores within the uncertain interval. A value <= .5 is not allowed.
mu0	Population value or estimate of the mean of the test scores of the persons without the targeted condition.
sd0	Population value or estimate of the standard deviation of the test scores of the persons without the targeted condition.
mu1	Population value or estimate of the mean of the test scores of the persons with the targeted condition.
sd1	Population value or estimate of the standard deviation of the test scores of the persons with the targeted condition.
intersection	Default NULL. If not null, the supplied value is used as the estimate of the intersection of the two bi-normal distributions. Otherwise, it is calculated.
start	Default NULL. If not null, the first two values of the supplied vector are used as the starting values for the nloptr optimization function.
print.level	Default is 0. The option print_level controls how much output is shown during the optimization process. Possible values: 0 (default) no output; 1 show iteration number and value of objective function; 2 1 + show value of (in)equalities; 3 2 + show value of controls.

Details

The function can be used to determinate the uncertain interval of two bi-normal distributions. The Uncertain Interval is defined as an interval below and above the intersection of the two distributions, with a sensitivity and specificity below a desired value (default .55).

Only a single intersection is assumed (or an second intersection where the overlap is negligible).

Value

List of values:

$status: Integer value with the status of the optimization (0 is success).

$message: More informative message with the status of the optimization

$results: Vector with the following values:

- exp.Sp.ui: The population value of the specificity in the Uncertain Interval, given mu0, sd0, mu1 and sd1. This value should be very near the supplied value of Sp.
- exp.Sp.ui: The population value of the sensitivity in the Uncertain Interval, given mu0, sd0, mu1 and sd1. This value should be very near the supplied value of Se.
- mu0: The value that has been supplied for mu0.
- sd0: The value that has been supplied for sd0.
- mu1: The value that has been supplied for mu1.
- sd1: The value that has been supplied for sd1.

$solution: Vector with the following values:

- L: The population value of the lower threshold of the Uncertain Interval.
- U: The population value of the upper threshold of the Uncertain Interval.

Examples

```
# A simple test model:
nlopt.ui()
# Using another bi-normal distribution:
nlopt.ui(mu0=0, sd0=1, mu1=1.6, sd1=2)
```

`nlopt.ui.general` *Function for the determination of the population thresholds an inconclusive interval for bi-normal distributed test scores.*

Description

Function for the determination of the population thresholds an inconclusive interval for bi-normal distributed test scores.

Usage

```
nlopt.ui.general(Se = 0.55, Sp = 0.55, distribution = "norm",
  parameters.d0 = c(mean = 0, sd = 1), parameters.d1 = c(mean = 1, sd = 1),
  overlap.interval = NULL, intersection = NULL, start = NULL,
  print.level = 0)
```

Arguments

`Se`	(default = .55). Desired sensitivity of the test scores within the uncertain interval. A value <= .5 is not allowed, while a value larger than .6 is not recommended.
`Sp`	(default = .55). Desired specificity of the test scores within the uncertain interval. A value <= .5 is not allowed, while a value larger than .6 is not recommended.
`distribution`	Name of the continuous distribution, exact as used in R package stats. Equal to density function minus d. For instance 'norm'.
`parameters.d0`	Named vector of population values or estimates of the parameters of the distribution of the test scores of the persons without the targeted condition. For instance `c(mean = 0, sd = 1)`. This distribution should have the lower values.
`parameters.d1`	Named vector of population values or estimates of the parameters of the distribution of the test scores of the persons with the targeted condition. For instance `c(mean = 1, sd = 1)`. The test scores of d1 should have higher values than d1. If not, use -(test scores). This distribution should have the higher values.
`overlap.interval`	A vector with a raw estimate of the lower and upper relevant of the overlap of the two distributions. If NULL, set to quantile .001 of the distribution of persons with the targeted condition and quantile .999 of the distribution of persons without the condition. Please check whether this is a good estimate of the relevant overlap.
`intersection`	Default NULL. If not null, the supplied value is used as the estimate of the intersection of the two bi-normal distributions. Otherwise, it is calculated.
`start`	Default NULL. If not null, the first two values of the supplied vector are used as the starting values for the `nloptr` optimization function.
`print.level`	Default is 0. The option print_level controls how much output is shown during the optimization process. Possible values: 0 (default) no output; 1 show iteration number and value of objective function; 2 1 + show value of (in)equalities; 3 2 + show value of controls.

Details

The function can be used to determinate the uncertain interval of the two continuous distributions. The Uncertain Interval is defined as an interval below and above the intersection of the two distributions, with a sensitivity and specificity below a desired value (default .55).

Only a single intersection is assumed (or an second intersection where the overlap is negligible).

Value

List of values:

$status: Integer value with the status of the optimization (0 is success).

$message: More informative message with the status of the optimization

$results: Vector with the following values:

- exp.Sp.ui: The population value of the specificity in the Uncertain Interval, given mu0, sd0, mu1 and sd1. This value should be very near the supplied value of Sp.
- exp.Sp.ui: The population value of the sensitivity in the Uncertain Interval, given mu0, sd0, mu1 and sd1. This value should be very near the supplied value of Se.
- vector of parameter values of d0The values that have been supplied for d0.
- vector of parameter values of d1The values that have been supplied for d1.

$solution: Vector with the following values:

- L: The population value of the lower threshold of the uncertain interval.
- U: The population value of the upper threshold of the uncertain interval.

Examples

```
# A simple test model:
nlopt.ui.general(Se = .55, Sp = .55,
                 distribution = 'norm',
                 parameters.d0 = c(mean = 0, sd = 1),
                 parameters.d1 = c(mean = 1, sd = 1),
                 overlap.interval=c(-2,3))
# Standard procedure when using a continous distribution:
nlopt.ui.general(parameters.d0 = c(mean = 0, sd = 1),
                 parameters.d1 = c(mean = 1.6, sd = 2))
emp.AUC <- function(norm, abnorm) {
  o = outer(abnorm, norm, "-")
  mean((o > 0) + .5 * (o == 0))
}

library(MASS)
library(car)
# gamma distributed data
set.seed(4)
d0 = rgamma(100, shape=2, rate=.5)
d1 = rgamma(100, shape=7.5, rate=1)
# 1. obtain parameters
parameters.d0=fitdistr(d0, 'gamma')$estimate
parameters.d1=fitdistr(d1, 'gamma')$estimate
# 2. test if supposed distributions (gamma) is fitting
qqPlot(d0, distribution='gamma', shape=parameters.d0['shape'])
qqPlot(d1, distribution='gamma', shape=parameters.d1['shape'])
# 3. draw curves and determine overlap
curve(dgamma(x, shape=parameters.d0['shape'], rate=parameters.d0['rate']), from=0, to=16)
curve(dgamma(x, shape=parameters.d1['shape'], rate=parameters.d1['rate']), from=0, to=16, add=TRUE)
overlap.interval=c(1, 15) # ignore intersection at 0; observe large overlap
# 4. get empirical AUC
emp.AUC(d0, d1)
# about .65 --> Poor
# .90-1 = excellent (A)
# .80-.90 = good (B)
# .70-.80 = fair (C)
# .60-.70 = poor (D)
# .50-.60 = fail (F)
# 5. Get uncertain interval
```

```
(res=nlopt.ui.general (Se = .57,
                       Sp = .57,
                       distribution = 'gamma',
                       parameters.d0 = parameters.d0,
                       parameters.d1 = parameters.d1,
                       overlap.interval,
                       intersection = NULL,
                       start = NULL,
                       print.level = 0))
abline(v=c(res$intersection, res$solution))
# 6. Assess improvement when diagnosing outside the uncertain interval
sel.d0 = d0 < res$solution[1] |  d0 > res$solution[2]
sel.d1 = d1 < res$solution[1] |  d1 > res$solution[2]
(percentage.selected.d0 = sum(sel.d0) / length(d0))
(percentage.selected.d1 = sum(sel.d1) / length(d1))
emp.AUC(d0[sel.d0], d1[sel.d1])
# AUC for selected scores outside the uncertain interval
emp.AUC(d0[!sel.d0], d1[!sel.d1])
# AUC for deselected scores; worst are deselected
# weibull distributed data
set.seed(4)
d0 = rweibull(100, shape=3, scale=50)
d1 = rweibull(100, shape=3, scale=70)
# 1. obtain parameters
parameters.d0=fitdistr(d0, 'weibull')$estimate
parameters.d1=fitdistr(d1, 'weibull')$estimate
# 2. test if supposed distributions (gamma) is fitting
qqPlot(d0, distribution='weibull', shape=parameters.d0['shape'])
qqPlot(d1, distribution='weibull', shape=parameters.d1['shape'])
# 3. draw curves and determine overlap
curve(dweibull(x, shape=parameters.d0['shape'],
      scale=parameters.d0['scale']), from=0, to=150)
curve(dweibull(x, shape=parameters.d1['shape'],
      scale=parameters.d1['scale']), from=0, to=150, add=TRUE)
overlap.interval=c(1, 100) # ignore intersection at 0; observe overlap
# 4. get empirical AUC
emp.AUC(d0, d1)
# about .65 --> Poor
# .90-1 = excellent (A)
# .80-.90 = good (B)
# .70-.80 = fair (C)
# .60-.70 = poor (D)
# .50-.60 = fail (F)
# 5. Get uncertain interval
(res=nlopt.ui.general (Se = .55,
                       Sp = .55,
                       distribution = 'weibull',
                       parameters.d0 = parameters.d0,
                       parameters.d1 = parameters.d1,
                       overlap.interval,
                       intersection = NULL,
                       start = NULL,
                       print.level = 0))
```

```
abline(v=c(res$intersection, res$solution))
# 6. Assess improvement when diagnosing outside the uncertain interval
sel.d0 = d0 < res$solution[1] |  d0 > res$solution[2]
sel.d1 = d1 < res$solution[1] |  d1 > res$solution[2]
(percentage.selected.d0 = sum(sel.d0) / length(d0))
(percentage.selected.d1 = sum(sel.d1) / length(d1))
emp.AUC(d0[sel.d0], d1[sel.d1])
# AUC for selected scores outside the uncertain interval
emp.AUC(d0[!sel.d0], d1[!sel.d1])
# AUC for deselected scores; these scores are almost indistinguishable
```

`nomogram` *Fagan's nomogram to show the relationships between the prior probability, the likelihood ratios, sensitivity and specificity, and the posterior probability.*

Description

Next to plotting Fagan's nomogram, this function also calculates the minimally needed values for specificity and sensitivity to reach desired posttest probabilities (or likelihood ratios) for a grey zone (Coste et al., 2003, 2006).

Usage

```
nomogram(prob.pre.test = 0.5, probs.post.test = c(pos = NULL, neg = NULL),
  SeSp = c(Se = NULL, Sp = NULL), LR = c(PLR = NULL, NLR = NULL),
  plot = T)
```

Arguments

`prob.pre.test`	The prior test probability, with a default value of .5. Often, (local) prevalence is used.
`probs.post.test`	A vector of two values that give the desired posttest probabilities of observing the event in the case of a positive test result (positive posttest probability: pos), and the posttest probability of observing the event in the case of a negative test result (negative posttest probability: neg). When not given, these probabilities are calculated using the likelihood ratios (LR).
`SeSp`	A vector of two values that give the desired sensitivity and specificity. When not given, the Se and Sp values are calculated from the desired posttest probabilities.
`LR`	A vector of two values that give the positive likelihood ratio (sensitivity / (1-specificity)): PLR of observing the event, and the negative likelihood ratio ((1 - sensitivity) / specificity): NLR of not observing the event. PLR is a value > 1, NLR is a value between 0 and 1. When not given, the LR values are calculated from the desired posttest probabilities.
`plot`	A Boolean that indicates whether a plot is desired.

Details

Parameter probs.post.test or SeSp or LR must be supplied, the other two values are calculated. When more than one parameter is given the other two are ignored. The basis of this function is adapted from package TeachingDemos.

Value

Vector of values:

$pre: The given pre-test probability.

$min.LRpos: The given or calculated minimally required positive likelihood ratio. If no value is provided, it is calculated.

$max.LRneg: The given or calculated maximally required negative likelihood ratio. If no value is provided, it is calculated.

$post.pos: The given or calculated positive posttest probability.

$minSp: The minimum value for the specificity, needed to reach the desired posttest probabilities.

$minSe: The minimum value for the sensitivity, needed to reach the desired posttest probabilities.

References

Fagan, T. J. (1975). Nomogram for Bayes theorem. The New England Journal of Medicine, 293(5), 257???257.

Coste, J., Jourdain, P., & Pouchot, J. (2006). A gray zone assigned to inconclusive results of quantitative diagnostic tests: application to the use of brain natriuretic peptide for diagnosis of heart failure in acute dyspneic patients. Clinical Chemistry, 52(12), 2229???2235.

Coste, J., & Pouchot, J. (2003). A grey zone for quantitative diagnostic and screening tests. International Journal of Epidemiology, 32(2), 304???313.

Examples

```
# Show calculated results (first 3 times about the same)
(nomogram(prob.pre.test = .10, probs.post.test=c(pos=.70, neg=.001)))
(nomogram(prob.pre.test = .10, SeSp=c(Se=0.991416309, Sp=0.952789700)))
(nomogram(prob.pre.test = .10, LR=c(pos=21, neg=0.0090090091)))
(nomogram(prob.pre.test = .10, SeSp=c(Se=0.99, Sp=0.95)))
# plot only
nomogram(prob.pre.test = .10, LR=c(pos=21, neg=0.0090090091))
# plot and display precise results
(nomogram(prob.pre.test = .10, probs.post.test=c(pos=.70, neg=.001)))

# check the influence of different values of prevalence
i=1
out=matrix(0,nrow = 9, ncol= 7)
for (prev in (seq(.1, .9, by=.1))) {
  out[i,]=nomogram(prob.pre.test=prev, probs.post.test=c(.95, .05), plot=FALSE)
  i=i+1
}
colnames(out) = names(nomogram(prob.pre.test=prev, probs.post.test=c(.95, .05), plot=FALSE))
```

out

---

plotMD *Function to plot the mixed densities of distributions of individuals with (1) and without (0) the targeted condition.*

Description

This plot function shows the two distributions and their overlap in a single graph.

Usage

```
plotMD(ref, test, breaks = 20, subtitle = "",
  position.legend = "topright", colspace = c("color", "grayscale", "BW"),
  model = c("kernel", "binormal", "ordinal"), ...)
```

Arguments

ref	The reference standard. A column in a data frame or a vector indicating the classification by the reference test. The reference standard must be coded either as 0 (absence of the condition) or 1 (presence of the condition)
test	The index test or test under evaluation. A column in a dataset or vector indicating the test results in a continuous scale.
breaks	Breaks used to construct the histograms. Either a single integer number or a vector containing the actual breaks. In the case of a vector, the number should cover all available test values. In the case of a single integer number, this number has to be equal or lower than the discernable values in the test. For short ordinal scales a vector should be uses covering all possible test values.
subtitle	Optional subtitle
position.legend	The location can be specified by a single keyword from the list "topright", "topleft", "top", "right", "bottomright", "bottom", "bottomleft", "left" and "center". Default is "top.right".
colspace	Use colors, grayscale or only black and white as plot colors. Default = color.
model	The model used for estimation. Default = 'kernel'. Adapts also breaks and the call to the density function (parameter adjust). When the model is obviously wrong, warnings are produced.
...	passing arguments to the kernel density function, other than kernel='gaussian' (default).

Details

The graph shows the two distributions, their overlap and the Dicrimination Curve. Many tests of intermediate quality have a considerable overlap. Also, the distributions as estimated by the `density` function, using the gaussian kernel is shown. The intersection is indicated by a vertical line. This graph allows the visual inspection of the two distributions, as well a visual inspection of the approximation of the `density`, based on the gaussian kernel. When the density estimation is way off, the standard estimation of the intersection will be incorrect, and another estimation has to be supplied.

The function `plotMD` can also be used for visual inspection of the Uncertain Interval (see examples). Please note that the sensitivity and specificity values > .5 (including the default of .55) allows for some positive bias.

Value

No Value returned.

References

Landsheer, J. A. (2016). Interval of Uncertainty: An Alternative Approach for the Determination of Decision Thresholds, with an Illustrative Application for the Prediction of Prostate Cancer. PloS One, 11(11), e0166007.

Examples

```
# A test of intermediate quality
set.seed(1)
ref=c(rep(0,500), rep(1,500))
test=c(rnorm(500,0,1), rnorm(500,1,1.2))
plotMD(ref, test)
ua = ui.nonpar(ref, test) # with warning message!
# Add lines to indicate Uncertain Interval
abline(v=ua[1:2])
select=(test <= ua[2] & test >= ua[1])
# plot the mixed densities for the Uncertain Interval
plotMD(ref[select], test[select])
plotMD(ref[select], test[select], colspace='gray')
plotMD(ref[select], test[select], colspace='BW')

# An ordinal test
norm     = rep(1:5, times=c(33,6,6,11,2))
abnorm   = rep(1:5, times=c(3,2,2,11,33))
testres  = c(abnorm,norm)
truestat = c(rep(1,length(abnorm)), rep(0,length(norm)))
plotMD(ref=truestat, test=testres, model='ordinal')

# ordinal test: weak test
set.seed(2)
nobs=1000
Z0 <- rnorm(nobs, mean=0)
b0=seq(-5, 5, length.out=31) # range sufficient to cover both z0 and z1
```

```
f0=cut(Z0, breaks = b0, labels = c(1:30))
x0=as.numeric(levels(f0))[f0]
Z1 <- rnorm(nobs, mean=.5) # very weak test, not recommended for practical use
f1=cut(Z1, breaks = b0, labels = c(1:30))
x1=as.numeric(levels(f1))[f1]
test=c(x0, x1)
ref =c(rep(0, length(x0)), rep(1, length(x1)))
(pr=prop.table(table(ref, test)))
breaks=c(min(test)-.5, seq(min(test), max(test), by=1)+.5)
plotMD(ref, test, model='ordinal')
# when model = 'binormal' or 'kernel', default breaks do not work well for
# ordinal data, and have to be set by hand
plotMD(ref, test, breaks=c(min(test)-.5, seq(min(test), max(test), by=1)+.5),
       model='binormal')
plotMD(ref, test, breaks=c(min(test)-.5, seq(min(test), max(test), by=1)+.5),
       model='kernel')
```

---

psa2b *CARET PSA Biomarker data - Etzioni substudy (454 control patients; 229 patients with prostate cancer)*

Description

- id patient id; sequential, randomly assigned
- d Prostate Ca (0 no 1 yes). Non-cancer patients are controls matched to cases on age and # sample.
- t time (years) relative to prostate Ca Dx
- fpsa free PSA
- tpsa total PSA
- age patient age at blood draw

References

Etzioni R, Pepe M, Longton G, Hu C, Goodman G (1999). Incorporating the time dimension in receiver operating characteristic curves: A case study of prostate cancer. Medical Decision Making 19:242-51. https://research.fhcrc.org/diagnostic-biomarkers-center/en/datasets.html http://mdm.sagepub.com/content/19/3/242.abstract

`quality.threshold` *Function for the description of the qualities of one or two decision thresholds or threshold.*

Description

This function can be used for both dichotomization (single threshold or cut-point) methods and for trichotomization (two thresholds or cut-points) methods. In the case of the Uncertain Interval trichotomization method, it provides descriptive statistics for the test scores outside the Uncertain Interval. For the TG-ROC trichotomization method it provides the descriptive statistics for TG-ROC's Valid Range.

Usage

```
quality.threshold(ref, test, threshold, threshold.upper = NULL,
  model = c("kernel", "binormal", "ordinal"))
```

Arguments

`ref`	The reference standard. A column in a data frame or a vector indicating the classification by the reference test. The reference standard must be coded either as 0 (absence of the condition) or 1 (presence of the condition)
`test`	The index test or test under evaluation. A column in a dataset or vector indicating the test results in a continuous scale.
`threshold`	The decision threshold of a dichotomization method, or the lower decision threshold of a trichotomization method.
`threshold.upper`	(default = NULL). The upper decision threshold of a trichotomization method. When NULL, the test scores are dichotomized.
`model`	The model to use. Default = 'kernel'.

Value

A list of

table The confusion table of diag x ref, where diag is the diagnosis based on the test, when applying the threshold(s). The reference standard (ref) has categories 0 and 1, while the diagnosis based on the test scores (diag) has categories 0 and 1 in the case of applying a single threshold (dichotomization), and the categories 0, NA and 1 in the case of trichotomization. In the case of the Uncertain Interval trichotomization method, the row NA shows the count of test scores within the Uncertain Interval. When applying the trichotomization method TG-ROC, the row NA shows the count of the test scores within the Intermediate Range. Table cell 0, 0 shows the True Negatives (TN), cell 0, 1 shows the False Negatives (FN), cell 1, 0 shows the False Positives (FP), and cell 1, 1 shows the True Positives (TP).

cut The values of the threshold(s).

indices A named vector, with the following statistics for the test-scores with diagnosis 0 or 1:

- prevalence: Diagnosable patients with the targeted condition / Total diagnosable sample = (TP+FN)/(TN+FP+FN+TP)
- correct.classification.rate (or Accuracy): (TP+TN)/(TN+FP+FN+TP)
- balance.correct.incorrect : (TP+TN)/(FP+FN)
- specificity: TN/(TN+FN)
- sensitivity: TP/(TP+FN)
- negative.predictive.value: TN/(TN+FN)
- positive.predictive.value: TP/(TN+FN)
- neg.likelihood.ratio: (1-sensitivity)/specificity
- pos.likelihood.ratio: sensitivity/(1-specificity)
- concordance: The probability that a random chosen patient with the condition is correctly ranked higher than a randomly chosen patient without the condition. Equal to AUC, with for the more certain interval a higher outcome than the overall concordance.

Examples

```
# A simple test
ref=c(rep(0,500), rep(1,500))
test=c(rnorm(500,0,1), rnorm(500,1,1))
ua = ui.nonpar(ref, test)
quality.threshold(ref, test, threshold=ua[1], threshold.upper=ua[2])
```

`quality.threshold.uncertain`

Function for the description of the qualities of the Uncertain Interval.

Description

This function can be used only for trichotomization (double thresholds or cut-points) methods. In the case of the Uncertain Interval trichotomization method, it provides descriptive statistics for the test scores within the Uncertain Interval. For the TG-ROC trichotomization method it provides the descriptive statistics for TG-ROC's Intermediate Range.

Usage

```
quality.threshold.uncertain(ref, test, threshold, threshold.upper,
  intersection = NULL, model = c("kernel", "binormal", "ordinal"))
```

Arguments

`ref`	The reference standard. A column in a data frame or a vector indicating the classification by the reference test. The reference standard must be coded either as 0 (absence of the condition) or 1 (presence of the condition)
`test`	The index test or test under evaluation. A column in a dataset or vector indicating the test results in a continuous scale.
`threshold`	The lower decision threshold of a trichotomization method.

`threshold.upper`
: The upper decision threshold of a trichotomization method. When NULL, the test scores are dichotomized.

`intersection`
: (default = NULL). When NULL, the intersection is calculated with `get.intersection`, which uses the kernel density method to obtain the intersection. When another value is assigned to this parameter, this value is used instead.

`model`
: (default = 'kernel'). The model used defines the intersection. Default the kernel densities are used with adjust = 1, for ordinal models adjust = 2 is used. For binormal models the binormal estimate of the intersection is used. The model defines the intersection, which defines the output of this function.

Details

The Uncertain Interval is defined as an interval below and above the intersection, with a sensitivity and specificity below a desired value (default .55). As a result, it is expected that Chi-square tests are not significant, provided that the count of individuals within the Uncertain Interval is not too large. Most often, the t-test is also non-significant, but as the power of the t-test is considerably larger than the power of the Chi-square test, this is less often the case. It is recommended to look at the difference of the means of the two sub-samples and to visually inspect the inter-mixedness of the test scores.

The patients that have test scores within the Uncertain Interval cannot be correctly classified on the basis of their test result. The results within the Uncertain Interval differ only slightly for patients with and without the targeted condition. Patients with slightly lower or higher test scores too often have the opposite status. They receive the diagnostic result 'Uncertain'; it is better to apply additional tests or to await further developments.

When applying the method to the results of a logistic regression, one should be aware of possible problems concerning the determination of the intersection. Somewhere in the middle, logistic predictions can have a range where the distributions have similar densities or have multiple intersections near to each other. Often, this problem can be approached effectively by using the linear predictions instead of the logistic predictions. The linear predictions offer often a far more clear point of intersection. The solution can then be applied to the prediction values using the inverse logit of the intersection and the two cut-points. The logistic predictions and the linear predictions have the same rank ordering.

Value

A list of

intersection The value used as estimate of the intersection. NOTE: The trichotomization method TG-ROC has no defined position for its Intermediate Range, but usage of the point where Sensitivity=Specificity seems a reasonable choice.

table The confusion table of diag x ref for the Uncertain Interval where the scores are expected to be inconclusive, where diag is the diagnosis based on the test, when applying the thresholds. Both the reference standard (ref) and the diagnosis based on the test scores (diag) have categories 0 and 1. In the case of the Uncertain Interval trichotomization method, the row NA shows the count of test scores within the Uncertain Interval. When applying the trichotomization method TG-ROC, the row NA shows the count of the test scores within the Intermediate Range. Table

cell 0, 0 shows the True Negatives (TN), cell 0, 1 shows the False Negatives (FN), cell 1, 0 shows the False Positives (FP), and cell 1, 1 shows the True Positives (TP).

cut The values of the thresholds.

X2 Table with the outcomes of three Chi-square tests of the confusion table:

- TN.FP: Chi-square test of the comparison of TN versus FP.
- FN.TP: Chi-square test of the comparison of FN versus TP.
- overall: Chi-square test of all four cells of the table.

t.test Table with t-test results for the comparison of the means. Within the Uncertain Interval, the test scores are compared of individuals without the targeted condition (ref = 0) and individuals with the targeted condition (ref = 1).

indices A named vector, with the following statistics for the test-scores within the Uncertain Interval:

- prevalence: Diagnosable patients with the targeted condition / Total diagnosable sample = (TP+FN)/(TN+FP+FN+TP)
- correct.classification.rate (or Accuracy): (TP+TN)/(TN+FP+FN+TP)
- balance.correct.incorrect: (TP+TN)/(FP+FN)
- specificity: TN/(TN+FN)
- sensitivity: TP/(TP+FN)
- negative.predictive.value: TN/(TN+FN)
- positive.predictive.value: TP/(TN+FN)
- neg.likelihood.ratio: (1-sensitivity)/specificity
- pos.likelihood.ratio: sensitivity/(1-specificity)
- concordance: The probability that a random chosen patient with the condition is correctly ranked higher than a randomly chosen patient without the condition. Equal to AUC, with for the uncertain interval an expected outcome < .60. (Not equal to a partial AUC.)

Examples

```
# A simple test model
ref=c(rep(0,500), rep(1,500))
test=c(rnorm(500,0,1), rnorm(500,1,sd=1))
ua = ui.nonpar(ref, test)
quality.threshold.uncertain(ref, test, ua[1], ua[2])
```

---

tostbegg2 *Hepatic metastasis ultrasound study - Tosteston & Begg study (96 patients)*

Description

- type primary cancer type: 0 Colon 1 Breast
- d hepatatic metastasis (0 no 1 yes). Non-cancer patients are controls matched to cases on age and # sample.
- y rating 1 to 5: 5 definite metastatic disease to the liver; 4 probable metastatic disease to the liven; 3 possible metastatic disease to the liven; 2 probably normal; and 1 definitely normal.

Details

A liver metastasis is a malignant tumor in the liver that has spread from another organ that has been affected by cancer.

References

Tosteson AN, & CB Begg (1988). A general regression methodology for ROC curve estimation. Medical Decision Making 8, 204???15 https://research.fhcrc.org/diagnostic-biomarkers-center/en/datasets.html http://journals.sagepub.com/doi/abs/10.1177/0272989X8800800309

`ui.binormal` *Function for the determination of the sample thresholds an inconclusive interval for bi-normal distributed test scores.*

Description

Function for the determination of the sample thresholds an inconclusive interval for bi-normal distributed test scores.

Usage

```
ui.binormal(ref, test, Se = 0.55, Sp = 0.55, intersection = NULL,
  start = NULL, print.level = 0)
```

Arguments

`ref`	The reference standard. A column in a data frame or a vector indicating the classification by the reference test. The reference standard must be coded either as 0 (absence of the condition) or 1 (presence of the condition).
`test`	The index test or test under evaluation. A column in a dataset or vector indicating the test results in a continuous scale.
`Se`	(default = .55). Desired sensitivity of the test scores within the uncertain interval. A value <= .5 is not allowed.
`Sp`	(default = .55). Desired specificity of the test scores within the uncertain interval. A value <= .5 is not allowed.
`intersection`	Default NULL. If not null, the supplied value is used as the estimate of the intersection of the two bi-normal distributions. Otherwise, it is calculated.
`start`	Default NULL. If not null, the first two values of the supplied vector are used as the starting values for the `nloptr` optimization function.
`print.level`	Default is 0. The option print_level controls how much output is shown during the optimization process. Possible values: 0 (default) no output; 1 show iteration number and value of objective function; 2 1 + show value of (in)equalities; 3 2 + show value of controls.

Details

This function can be used for a test with bi-normal distributed scores. The Uncertain Interval is defined as an interval below and above the intersection, with a sensitivity and specificity below a desired value (default .55).

Only a single intersection is assumed (or an second intersection where the overlap is negligible). If another intersection exists and the overlap around this intersection is considerable, the test with such a non-negligible overlap is problematic and difficult to apply and interpret.

In general, when estimating decision thresholds, a sample of sufficient size should be used. It is recommended to use at least a sample of 100 patients with the targeted condition, and a 'healthy' sample (without the targeted condition) of the same size or larger.

Lastly, it should be noted that the Uncertain interval method has been developed recently, and future research may provide more satisfactory answers.

Value

List of values:

$status: Integer value with the status of the optimization (0 is success).

$message: More informative message with the status of the optimization

$results: Vector with the following values:

- exp.Sp.ui: The population value of the specificity in the Uncertain Interval, given mu0, sd0, mu1 and sd1. This value should be very near the supplied value of Sp.
- exp.Sp.ui: The population value of the sensitivity in the Uncertain Interval, given mu0, sd0, mu1 and sd1. This value should be very near the supplied value of Se.
- mu0: The value that has been supplied for mu0.
- sd0: The value that has been supplied for sd0.
- mu1: The value that has been supplied for mu1.
- sd1: The value that has been supplied for sd1.

$solution: Vector with the following values:

- L: The population value of the lower threshold of the Uncertain Interval.
- U: The population value of the upper threshold of the Uncertain Interval.

Examples

```
# A simple test model
ref=c(rep(0,500), rep(1,500))
test=c(rnorm(500,0,1), rnorm(500,1,1))
ui.binormal(ref, test)
```

`ui.nonpar` *Function for the determination of an inconclusive interval for continuous test scores*

Description

This function uses a non-parametric approach to determine an interval around the intersection of the two distributions of individuals without (0) and with (1) the targeted condition. The interval is restricted both by a maximum sensitivity of the test scores within the uncertain interval (sens.ui) and by a maximum specificity of the test scores within the uncertain interval (spec.ui).

Usage

```
ui.nonpar(ref, test, sens.ui = 0.55, spec.ui = 0.55, intersection = NULL,
  return.first = T, select = c("nearest", "limited"))
```

Arguments

`ref`	The reference standard. A column in a data frame or a vector indicating the classification by the reference test. The reference standard must be coded either as 0 (absence of the condition) or 1 (presence of the condition)
`test`	The index test or test under evaluation. A column in a dataset or vector indicating the test results in a continuous scale.
`sens.ui`	(default = .55). The sensitivity of the test scores within the uncertain interval is either limited to this value or is the nearest to this value. A value <= .5 is useless.
`spec.ui`	(default = .55). The specificity of the test scores within the uncertain interval is either limited to this value or is the nearest to this value. A value <= .5 is useless.
`intersection`	(default = NULL) When NULL, the intersection is calculated with `get.intersection`, which uses the kernel density method to obtain the intersection. When another value is assigned to this parameter, this value is used instead.
`return.first`	(default = TRUE) Return only the widest possible interval, given the restrictions. When FALSE all calculated intervals with their sensitivity and specificity are returned. NOTE: This function does not always find a suitable interval and can return a vector of NULL values.
`select`	(default = 'nearest') If 'nearest', sensitivity and specificity of the uncertain interval are nearest sens.ui and spec.ui respectively. When 'limited' the solutions have an uncertain interval with a sensitivity and specificity limited by sens.ui and spec.ui respectively.

Details

This essentially non-parametric function finds the best possible solution for the sample. This function can be used for test with continuous scores or for test with about twenty or more ordened test scores. The Uncertain interval is defined as an interval below and above the intersection, with a sensitivity and specificity nearby or below a desired value (default .55).

In its core, the `ui.nonpar` function is non-parametric, but it uses the gaussian kernel for estimating the intersection between the two distributions. Always check whether your results are within reason. If the results are unsatisfactory, first check on the intersection. The `density` function allows for other approximations than gaussian. Another estimate can be obtained by using a more suitable kernel in the `density` function. The parameter `intersection` can be used to assign the new estimate to the `uncertain.interval` method.

Furthermore, only a single intersection is assumed (or an second intersection where the overlap is negligible). If another intersection exists and the overlap around this intersection is considerable, a second uncertain interval may be determined by using the parameter `intersection`. It should be noted that in most cases, a test with more than one intersection with non-negligible overlap is problematic and difficult to apply.

The Uncertain interval method is developed for continuous distributions, although it can be applied to tests with distinguishable categorical distributions. When a test is used with less than 20 discernible values, a warning is issued. The method may work satisfactorily, but results should always be checked carefully.

In general, when estimating decision thresholds, a sample of sufficient size should be used. It is recommended to use at least a sample of 100 patients with the targeted condition, and a 'healthy' sample (without the targeted condition) of the same size or larger.

The Uncertain interval method is not always capable to deliver results, especially when select == 'limited'. Clearly, when there is no overlap between the two distributions, there cannot be an uncertain interval. A very small interval of overlap can also limit the possibilities to find a solution. When there is no solution found, a vector of NA values is returned.

Lastly, it should be noted that the Uncertain interval method has been developed recently, and future research may provide more satisfactory answers.

Value

A `data.frame` of

cp.l Lower bound of the Uncertain interval.

cp.h Upper bound of the Uncertain interval.

FN Count of false negatives within the Uncertain interval.

TP Count of true positives within the Uncertain interval.

TN Count of true negatives within the Uncertain interval.

FP Count of false positives within the Uncertain interval.

sensitivity Sensitivity of the test scores within the Uncertain interval.

specificity Specificity of the test scores within the Uncertain interval.

Only a single row is returned when parameter `return.first` = TRUE (default).

References

Landsheer, J. A. (2016). Interval of Uncertainty: An Alternative Approach for the Determination of Decision Thresholds, with an Illustrative Application for the Prediction of Prostate Cancer. PloS One, 11(11), e0166007.

Examples

```
# A simple test model
set.seed(1)
ref=c(rep(0,500), rep(1,500))
test=c(rnorm(500,0,1), rnorm(500,1,1))
ui.nonpar(ref, test, select='limited')

ref = c(rep(0,20), rep(1,20))
test= c(rnorm(20), rnorm(20, mean=1))
ui.nonpar(ref, test)
```

`ui.ordinal` *Function to calculate possible uncertain intervals of ordinal test results of individuals with (1) and without (0) the targeted condition.*

Description

This function calculates the uncertain interval (UI) of the test results of the two groups. This function is intended to be used for tests with 20 or less ordered test values. The lower range of test scores identifies patients without the targeted condition (lower More Certain Interval (MCI)), the upper interval of test scores above the uncertain interval identifies the patients with the condition (upper MCI). Due to the limited number of distinguishable scores, the estimations are course. When more than 20 values can be distinguished, `ui.nonpar` or `ui.binormal` may be preferred.

Usage

```
ui.ordinal(ref, test, select.max = c("MCI.Sp+MCI.Se", "MCI.C", "MCI.Acc",
  "MCI.Se", "MCI.Sp", "MCI.n", "All"), constraints = c(C = 0.57, Acc = 0.6,
  lower.ratio = 0.8, upper.ratio = 1.25), weights = c(1, 1, 1),
  intersection = NULL, return.all = FALSE, ...)
```

Arguments

`ref`	The reference standard. A column in a data frame or a vector indicating the classification by the reference test. The reference standard must be coded either as 0 (absence of the condition) or 1 (presence of the condition)
`test`	The test pr predictor under evaluation. A column in a dataset or vector indicating the test results in a continuous scale. It is expected that true patients have higher scores than non-patients. If this is not the case, the test scores should be negated (test = -(test scores)).
`select.max`	Selects the candidate thresholds on basis of a desired property of the More Certain Intervals (MCI). The criteria are: maximum Se+Sp (default), maximum C, maximum Accuracy, maximum Sp, maximum Se, maximum size of MCI. The last alternative 'All' is to choose all possible details.

`constraints` Sets upper constraints for various properties of the uncertain interval: C-statistic, Acc, lower and upper limit of the ratio of the proportions with and without the targeted condition. The default values are C = .57, Acc = .6, lower.ratio = .8, upper.ratio = 1.25. These values implement the desired uncertainty of the uncertain interval. The value of C is considered the most important and has the most restrictive default value. For Acc and C, the values closest to the desired value are found and then all smaller values are considered. The other two constraints are straightforward lower and upper limits of the ratio between the number of patients with and without the targeted disease. If you want to change these values, it is necessary to name all values. C = 1 or Acc = 1 excludes C respectively accuracy as selection criterion. If no solution is found, the best is showed together with a warning message.

`weights` (Default = c(1, 1, 1). Vector with weights for the loss function. weights[1] is the weight of false negatives, weights[2] is the weight for loss in the uncertain interval (deviations from equal chances to belong to either distribution), and weights[3] is the weight for false positives. When a weight is set to a larger value, thresholds are selected that make the corresponding error smaller while the area grows smaller.

`intersection` (Default = NULL). Optional value to de used as value for the intersection. If no value is supplied, the intersection is calculated using `get.intersection(ref = ref, test = test, mod`

`return.all` (Default = FALSE). When TRUE $data.table and $uncertain.interval are included in the output.

`...` Further parameters that can be transferred to the density function.

Details

Due to the limited possibilities of short scales, it is more difficult to determine a suitable uncertain interval when compared to longer scales. For any threshold determination, one needs a representative sample of sufficient size (200 or larger). If there are no test scores below the intersection in the candidate uncertain area, Sp of the Uncertain Interval (UI.Sp) is not available, while UI.Se equals 1. The essential question is always whether the patients with the test scores inside the uncertain interval can be sufficiently distinguished. The candidate intervals are first selected on the properties of the uncertain interval. The defaults are C lower than .6, Acc lower than .6, and the ratio of proportions of persons with / without the targeted condition between .8 and 1.25. These criteria ensure that all candidates for the uncertain interval have insufficient accuracy. The second criterion is the desired property of the More Certain Intervals (see select.max parameter). The model used is 'ordinal'. This model default for the adjust parameter send to the density function is 2, but you can enter another value such as adjust = 1.

Discussion of the first example (please run the code first): Visual inspection of the Mixed Barplot shows that distinguishing patients with and without the targeted condition is almost impossible for test scores 2, 3 and 4. Sensitivity and Specificity of the uncertain interval should be not too far from .5. In the first example, the first interval (3:3) has no lower scores than the intersection (3), and therefore Ui.Sp is not available and UI.Se = 1. The UI.ratio indicates whether the number of patients with and without the condition is equal in this interval. For these 110 patients, a diagnosis of uncertainty is probably the best choice. The second interval (3:4) has an UI.Sp of .22, which is a large deviation from .5. In this slightly larger interval, the patients with a test score of 3 have a slightly larger probability to belong to the group without the condition. UI.Se is .8. UI.ratio is

close to 1, which makes it a feasible candidate. The third interval (2:4) has an UI.Sp of .35 and an UI.Se of .70 and an UI.ratio still close to one. The other intervals show either Se or Sp that deviate strongly from .5, which makes them unsuitable choices. Probably the easiest way to determine the uncertain interval is the interval with minimum loss. This is interval (2:4). The Loss formula is (created by https://www.codecogs.com/latex/eqneditor.php):

$$L = \frac{\left(\sum_{i=l}^{u} |d0_i - d1_i| + \sum_{i=u+1}^{h} d1_i + \sum_{i=1}^{l-1} d0_i\right)}{N}$$

where *d0* represents the test scores of the norm group, *d1* represents the test scores of the targeted patient group, *l* is the lower limit of the uncertain interval, *u* the upper limit, the first test score is enumerated 1 and the last test score is enumerated *h*. *N* is the total number of all persons with test scores.

- $\sum_{i=l}^{u} |d0_i - d1_i|$ is the loss in the uncertain interval, that is, the total deviation from equality.
- $\sum_{i=u+1}^{h} d1_i$ is the loss in the lower More Certain Interval, that is, the total of False Negatives, the number of patients with the targeted condition with a test score lower than *l*, and
- $\sum_{i=u+1}^{h} d0_i$ is the loss in the upper More Certain Interval, that is, the total of False Positives, the number of patients without the targeted condition with a test score higher than *u*.

  Loss L is higher when the deviation from equality is higher in the uncertain area, higher when the number of False Negatives is higher, and higher when the number of False Positives is higher. The loss of a single threshold method equals 1 - its Accuracy. In this example, the minimum Loss is found with interval (2:4). As this agrees with values for UI.C and UI.ratio that sufficiently indicates the uncertainty of these test scores, this seems the most suitable choice: the number of patients with test scores 2 to 4 are almost as likely to come from either population. The remaining cases outside the uncertain interval (2:4) show high C, Accuracy, Specificity and Sensitivity.

Value

List of values:

$Youden A vector of statistics concerning the maximized Youden index:

- max.Youden: The value of the Maximized Youden Index (= max(tpr - fpr)).
- threshold: The threshold associated with the Maximized Youden Index. Test values >= threshold indicate the targeted condition.
- Sp: The Specificity of the test when this threshold is applied.
- Se: The Sensitivity of the test when this threshold is applied.
- Acc: The Accuracy of the test when this threshold is applied.
- Loss: min(fnr + fpr) = min(1 - (Se + Sp -1)) = 1 - max(tpr - fpr) lower range (< threshold): the summed number of false positives for each test score, divided by the number of persons that have received that test score. upper range (>= threshold): the summed number of false negatives, divided by the number of persons that have received that test score. The Youden Loss is equal to 1-Youden.index.
- C: Concordance; equals AUROC (Area Under Receiving Operating Curve)

$data.table A data.frame with the following columns:

- test: The test scores.

- d0: The frequencies of the test scores of the norm group.
- d1: The frequencies of the test scores of the group with the targeted condition.
- tot: The total frequency of each test scores.
- TP: The number of True Positives when this test score is used as threshold.
- FP: The number of False Positives when this test score is used as threshold.
- tpr: The true positive rate when this test score is used as threshold.
- fpr: The false positive rate when this test score is used as threshold.
- Y: The Youden Index (= tpr - fpr) when this test score is used as threshold.

$intersection The (rounded) intersection for the distributions of the two groups. Most often, these distributions have no true point of intersection and the rounded intersection is an approximation. Often, this equals the Maximized Youden threshold (see Schisterman 2005). Warning: When a limited range of scores is available, it is more difficult to estimate the intersection. Different estimates can easily differ plusminus 1. When using a non-rounded value (for example 16.1), the effective threshold for the uncertain area is round(intersection+.5), in the mentioned example: 16.1 becomes 17.

$uncertain.interval Data frame with the statistics of all possible bounds of the uncertain interval. The columns are the following:

- lowerbound: Lower bound of the possible uncertain interval.
- upperbound: Upper bound of the possible uncertain interval.
- UI.Sp: Specificity of the test scores between and including the lower and upper boundary. Closer to .5 is 'better', that is, more uncertain. This estimate is rough and dependent on the intersection and cannot be recommended as a criterion for a short, ordinal scale.
- UI.Se: Sensitivity of the test scores between and including the lower and upper boundary. Closer to .5 is 'better', that is, more uncertain. This estimate is rough and dependent on the intersection and cannot be recommended as a criterion for a short, ordinal scale.
- UI.Acc: Accuracy of the test scores between and including the lower and upper boundary. Closer to .5 is 'better', that is, more uncertain. This estimate is rough and dependent on the intersection and cannot be recommended as a criterion for a short, ordinal scale.
- UI.C: Concordance (AUROC) of the test scores between and including the lower and upper boundary. Closer to .5 is 'better', that is, more uncertain. Rule of thumb: <= .6
- UI.ratio: The ratio between the proportion of patients in the uncertain area with and without the condition. Closer to one is 'better', that is, more uncertain; 0.8 < UI.ratio < 1.25 as a rule of fist.
- UI.n: Number of patients with test scores between and including the lower and upper boundary.
- MCI.Sp: Specificity of the more certain interval, i.e., the test scores lower than the lower boundary and higher than the upper boundary.
- MCI.Se: Sensitivity of the test scores lower than the lower boundary and higher than the upper boundary.
- MCI.C: Concordance (AUROC) of the test scores outside the uncertain interval. Closer to .5 is 'better', that is, more uncertain. Rule of thumb: <= .6
- MCI.Acc: Accuracy of the test scores lower than the lower boundary and higher than the upper boundary.
- MCI.n: Number of patients with test scores lower than the lower boundary and higher than the upper boundary.

- Loss: Loss of the test scores lower than the lower boundary and higher than the upper boundary. The total loss is the sum of the loss of the three areas: lower MCI: the summed number of false positives for each test score, divided by the number of persons that have received that test score. uncertain interval: the sum of the absolute differences in the number of people in the norm group d0 and the number of persons in the group with the targeted condition (d1) per test score, divided by the total number of persons. upper MCI: the summed number of false negatives, divided by the number of persons that have received that test score. The Loss can be compared to the loss of the Youden threshold, provided that the intersection is equal to the Youden threshold. If necessary, this can be forced by attributing the value of the Youden threshold to the intersection parameter.

$candidates: Candidates with a loss lower than the Youden loss which might be considered for the Uncertain Interval. The candidates are selected based on the constraints parameter, that defines the desired constraints of the uncertain area, and the select.max parameter, that selects the desired properties of the lower and upper More Certain Interval.

References

Schisterman, E. F., Perkins, N. J., Liu, A., & Bondell, H. (2005). Optimal cut-point and its corresponding Youden Index to discriminate individuals using pooled blood samples. Epidemiology, 73-81.

Landsheer, J. A. (2016). Interval of Uncertainty: An alternative approach for the determination of decision thresholds, with an illustrative application for the prediction of prostate cancer. PLOS One.

See Also

plotMD for plotting the mixed densities of the test values. density for the parameters of the density function. ui.nonpar or ui.binormal may be preferred when more than 20 values can be distinguished on the ordinal test scale.

Examples

```
# A short test with 5 ordinal values
test0     = rep(1:5, times=c(165,14,16,55, 10)) # test results norm group
test1     = rep(1:5, times=c( 15,11,13,55,164)) # test results of patients
ref = c(rep(0, length(test0)), rep(1, length(test1)))
test = c(test0, test1)
table(ref, test)
plotMD(ref, test, model='ordinal') # visual inspection
ui.ordinal(ref, test, select.max='All')
# Same solution, but other layout of the results:
ui.ordinal(ref, test, select.max=c('MCI.Sp+MCI.Se', 'MCI.C', 'MCI.Acc',
                                   'MCI.Se', 'MCI.Sp', 'MCI.n'))
# forcing the Youden threshold as intersection gives the same best result.
# However, the estimates for ui.Se, ui.Sp and ui.Acc differ:
ui.ordinal(ref, test, intersection='Youden', select.max='All')

nobs=1000
set.seed(6)
Z0 <- rnorm(nobs, mean=0)
b0=seq(-5, 8, length.out=31)
```

```
f0=cut(Z0, breaks = b0, labels = c(1:30))
x0=as.numeric(levels(f0))[f0]
Z1 <- rnorm(nobs, mean=1, sd=1.5)
f1=cut(Z1, breaks = b0, labels = c(1:30))
x1=as.numeric(levels(f1))[f1]
ref=c(rep(0,nobs), rep(1,nobs))
test=c(x0,x1)
plotMD(ref, test, model='ordinal') # looks like binormal
# looks less binormal, but in fact it is a useful approximation:
plotMD(ref, test, model='binormal')
ui.ordinal(ref, test)
ui.binormal(ref, test) # compare application of the bi-normal model
```

`UncertainInterval` *Set of functions for the determination of an Uncertain Interval of test scores*

Description

A collection of functions to determine a range of test scores that are inconclusive and do not allow a diagnosis (other than Uncertain) and to access its qualities.

Details

See Index

References

Landsheer, J. A. (2016). Interval of Uncertainty: An Alternative Approach for the Determination of Decision Thresholds, with an Illustrative Application for the Prediction of Prostate Cancer. PloS One, 11(11), e0166007.

See Also

`ui.nonpar`, `plotMD`, `get.intersection`, `quality.threshold`, `quality.threshold.uncertain`

Index