# Package 'UncertainInterval'

February 25, 2020

**Version** 0.5.2

**Date** 2020-02-06

**Title** Uncertain Interval Methods for Three-Way Cut-Point Determination in Test Results

**Author** Hans Landsheer [aut, cre]

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

**Depends** R (>= 3.5.0)

**Suggests** MASS, car, carData, knitr, pander, rmarkdown, rticles

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

**License** GPL (>= 2)

**URL** https://github.com/HansLandsheer/UncertainInterval

**BugReports** https://github.com/HansLandsheer/UncertainInterval

**RoxygenNote** 7.0.2

**Imports** stats, utils, reshape2, rootSolve, nloptr, zoo, psych

**LazyLoad** yes

**VignetteBuilder** knitr, pander

**Encoding** UTF-8

## R topics documented:

---

`barplotMD` *Barplot of frequencies, densities or both of the two distributions of patients with and without the targeted condition.*

---

## Description

Barplot of frequencies, densities or both of the two distributions of patients with and without the targeted condition.

## Usage

```
barplotMD(
  ref,
  test,
  name.test = "",
  fixed.range = c(NULL, NULL),
  plot = c("frequencies", "densities", "both", "none"),
  target.condition = "Target Condition",
  position.legend = "top",
  cex.legend = 1
)
```

## Arguments

| | |
|---|---|
| `ref` | Vector of patient status with two ordered values. The first indicates the patients without the targeted condition (for instance 0), the second indicates the patients with the targeted condition (for instance 1). This order is relevant. |
| `test` | Vector of ordinal test scores. The values range from min(test) to max(test) and need to be ordered. Missing values in between are shown in the plots as gaps. |
| `name.test` | Name used for the title of the x axis. |
| `fixed.range` | Default = NULL. If test has numeric values you can set the values to cover a fixed range. This may enable the comparison of different samples with truncated test values. Default: use `min(test):max(test)` as the range of values. |
| `plot` | Default: 'frequencies'. Which plots to create: 'frequencies', 'densities' or 'both'. |
| `target.condition` | Default: 'Target Condition'. Name of target condition. |
| `position.legend` | Default: 'top'. Position of the legend. Most used values: "topleft", "top", "topright". |
| `cex.legend` | Default: 1. Relative size of the legend. |

## Value

named matrix with 2 rows and max(test)-min(test)+1 columns that provide for the position on the x-axis of each of the test values. These values can be used to draw vertical lines to indicate cutoff scores (see example).

## See Also

plotMD

## Examples

```
data(tostbegg2)
barplotMD(ref=tostbegg2$d, test=tostbegg2$y, name='Metastatic Rating', cex=1)
x.axis = barplotMD(ref=tostbegg2$d, test=tostbegg2$y, plot='densities',
name='Metastatic Rating', cex=1)
# Use x.axis to plot vertical line between test score 3 and 4
segments(x0=(x.axis[2,4]+x.axis[1,3])/2, y0=0, y1=.4, col='red')
# include zero score (in this sample empty)
barplotMD(ref=tostbegg2$d, test=tostbegg2$y, fixed.range = c(0, 5),
          plot='densities',name='Metastatic Rating', cex=1)
op = par(mfrow=c(2,1))
barplotMD(ref=tostbegg2$d, test=tostbegg2$y, plot='both',
          name='Metastatic Rating', cex.legend=.6, pos='top')
par(mfrow=op)
```

---

# 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", "none"))
```

## 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. |
| model | The model used for estimation. Default = 'kernel'. When model is kernel or binormal, the test data is checked whether the test has a sufficient number of different values (>= 20). When model is ordinal, the test data are checked whether they are ordinal or not. When model is 'none' the test data are only checked for missing data. |

## 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 internally from every function that requires data. An external 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

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!
```

---

greyzone *Function for the determination of a grey zone for quantitative diagnostic and screening tests*

---

## Description

Function for the determination of a grey zone for quantitative diagnostic and screening tests

## Usage

```
greyzone(
  ref,
  test,
  prevalence = NULL,
  criterion.values = c(0.05, 0.95),
  return.all = F
)
```

## Arguments

| | |
|---|---|
| ref | The reference standard. A column in a data frame or a vector indicating the reference or gold standard. The reference standard must be coded either as 0 (absence of the condition) or 1 (presence of the condition). |
| test | The ordinal test scores under evaluation. Higher scores indicate the presence of the targeted disease. Please use negated values when lower values indicate the presence of the targeted disease. |
| prevalence | The prevalence or pre-test probability to be used. When NULL, the prevalence found in the sample is used. |
| criterion.values | The minimum desired values for respectively the positive and negative post-test probability. |
| return.all | Default = FALSE. When TRUE the full table of all results are returned. |

## Details

This function is proposed by Coste et al. (2003). The current implementation only handles ordinal test values. This functions uses all possible test scores as dichotomous thresholds to calculate Se, Sp, positive and negative likelihood ratios and post-test probabilities. The likelihood ratios are calculated for the cumulated densities of the test scores and indicate the levels of seriousness of the disease for all possible dichotomous thresholds. It uses therefore a cumulative interpretation of the Likelihood Ratios and posttest probabilities. If a test has test scores 1 to 5 (with 5 indicating the largest probability of the disease), Se, positive LR and positive posttest probabilities of the greyzone function concern test results >= 1, >= 2, >= 3, >= 4 and >= 5, while Sp, negative LR and negative posttest probabilities concern test results < 1, < 2, < 3, < 4 and < 5.

Please note that the definition of a grey zone deviates from the definition of an uncertain interval.

The criterion is a required degree of closeness of post-test probabilities to 1 or 0. These post-test probabilities of cumulated test scores may require a value over 0.99 or even 0.999 (or under 0.01 or 0.001) to confirm or exclude the presence of a target disease. The default criterion values are .05 and .95 for respectively a negative and positive classification, which may be sufficient for use by clinicians or Public Health professionals for a first classification whether a target disease may be present or not (Coste et al., 2003).

As such the cumulative likelihood ratios differ from the Interval Likelihood Ratios (see RPV), as proposed by Sonis (1999). These likelihood ratios are calculated for each given interval of test scores separately and uses their densities. In contrast to the greyzone method, Interval Likelihood ratios and interval posttest probabilities concern the separate intervals, that is in this example, the separate score 1 to 5. Interval likelihood ratios assign a specific value to each level of abnormality, and this value is used to calculate the posttest probabilities of disease for each given level of a test (Sonis, 1999). These post-test probabilities differ strongly from the cumulative post-test probabilities and criterion values can be much lower, especially when diseases are life threatening and low-cost treatments are available. See Sonis (1999) for further discussion of the interval interprestation.

## Value

The function returns the lower and upper value of the range of test scores that are considered 'grey' or inconclusive. When return.all = TRUE the full table of the results is returned.

## References

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.

Sonis, J. (1999). How to use and interpret interval likelihood ratios. Family Medicine, 31, 432-437.

## See Also

RPV

## Examples

```
ref=c(rep(0, 250), rep(1, 250))
test = c(rep(1:5, c(90,75,50,35,0)), c(rep(1:5, c(10,25,50,65,100))))
table(ref, test)
greyzone(ref, test, ret=TRUE)
```

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

## Description

Function for the determination of the population thresholds an uncertain and 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 a second intersection where the overlap is negligible).

The function uses an optimization algorithm from the nlopt library (https://nlopt.readthedocs.io/en/latest/NLopt_Algorith the sequential quadratic programming (SQP) algorithm for nonlinearly constrained gradient-based optimization (supporting both inequality and equality constraints), based on the implementation by Dieter Kraft (1988; 1944).

## 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.

## References

Dieter Kraft, "A software package for sequential quadratic programming", Technical Report DFVLR-FB 88-28, Institut für Dynamik der Flugsysteme, Oberpfaffenhofen, July 1988.

Dieter Kraft, "Algorithm 733: TOMP–Fortran modules for optimal control calculations," ACM Transactions on Mathematical Software, vol. 20, no. 3, pp. 262-281 (1994).

## 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 uncertain and inconclusive interval for test scores with a known common distribution.*

## Description

Function for the determination of the population thresholds an uncertain and inconclusive interval for test scores with a known common distribution.

## 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 when the density function is 'dnorm', then the distribution is '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 d0. 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 a second intersection where the overlap is negligible).

The function uses an optimization algorithm from the nlopt library

(https://nlopt.readthedocs.io/en/latest/NLopt_Algorithms/).

It uses the sequential quadratic programming (SQP) algorithm for nonlinearly constrained gradient-based optimization (supporting both inequality and equality constraints), based on the implementation by Dieter Kraft (1988; 1944).

N.B. When a normal distribution is expected, the functions `nlopt.ui` and `ui.binormal` are recommended.

## 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 distribution d0, that is, the values that have been supplied in `parameters.d0`.
- vector of parameter values of distribution d1, that is, the values that have been supplied in `parameters.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.

## References

Dieter Kraft, "A software package for sequential quadratic programming", Technical Report DFVLR-FB 88-28, Institut für Dynamik der Flugsysteme, Oberpfaffenhofen, July 1988.

Dieter Kraft, "Algorithm 733: TOMP–Fortran modules for optimal control calculations," ACM Transactions on Mathematical Software, vol. 20, no. 3, pp. 262-281 (1994).

Landsheer, J. A. (2018). The Clinical Relevance of Methods for Handling Inconclusive Medical Test Results: Quantification of Uncertainty in Medical Decision-Making and Screening. Diagnostics, 8(2), 32. https://doi.org/10.3390/diagnostics8020032

## 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 continuous distribution:
nlopt.ui.general(parameters.d0 = c(mean = 0, sd = 1),
                 parameters.d1 = c(mean = 1.6, sd = 2))
# Function to calculate the Area under the Receiving Operating Characteristics
# Curve (AUC or C-statistic)
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), plot=FALSE))
(nomogram(prob.pre.test = .10, SeSp=c(Se=0.991416309, Sp=0.952789700), plot=FALSE))
(nomogram(prob.pre.test = .10, LR=c(pos=21, neg=0.0090090091), plot=FALSE))
(nomogram(prob.pre.test = .10, SeSp=c(Se=0.99, Sp=0.95), plot=FALSE))
# 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 densities of 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 densities of the two distributions and their overlap. 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.

Landsheer, J. A. (2018). The Clinical Relevance of Methods for Handling Inconclusive Medical Test Results: Quantification of Uncertainty in Medical Decision-Making and Screening. Diagnostics, 8(2), 32. https://doi.org/10.3390/diagnostics8020032

## See Also

barplotMD

## 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 Ranges.

## 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 and only threshold is used for the dichotomization. |
| `model` | The model to use. Default = 'kernel' for continuous data. For discrete data 'ordinal' is the better choice. |

## Details

The Uncertain Interval is generally defined as an interval below and above the intersection, where the densities of the two distributions of patients with and without the targeted impairment are about equal. The various ui-functions for the estimation of the uncertain interval use a sensitivity and specificity below a desired value (default .55). Please refer to the specific function descriptions how the middle section is defined.

The uncertain area is defined as the scores >= threshold and <= threshold.upper. When a single threshold is supplied and no uncertain area is defined, positive classifications (1) are considered for test scores >= threshold. Please note that the indices are calculated for those who receive a decision for or against the targeted disease: the test data in the uncertain interval are ignored. When the lower test scores indicate the targeted condition, please use the negated test values (-test).

The unstandardized predictive values (negative and positive; NPV and PPV) present the comparison of the observed frequencies of the two observed samples, for the evaluated range of test scores. When using a single cut-point the evaluated range of test scores for the NPV are test scores < threshold and for PPV test scores >= threshold. When using trichotomization, the evaluated range is < lower limit (NPV) and > upper limit (PPV). They provide the exact observed proportions of correctly classified patients, given the range of test scores.

The standardized predictive values (SNPV and SPPV) present the comparison of the densities (or relative frequencies) of the two distributions, for the evaluated range of test scores. When using a single cut-point the evaluated range of test scores for the SNPV are test scores < threshold and for SPPV test scores >= threshold. When using trichotomization, the evaluated range is < lower limit (SNPV) and > upper limit (SPPV). These predictive values are called standardized, because the two samples are compared as two independent samples, as if prevalence equals .5.

SNPV and SPPV provide the estimated relative probabilities that a patient is selected from the population of patients without the targeted condition or from the population of patients with the targeted condition, given that the patients test score is in the evaluated range of test scores. Of course, these estimates are better when the sample sizes are larger. N.B. 1 When negative and predictive values would be calculated for the same range of test scores, SNPV = 1 - SPPV and SPPV = 1 - SNPV. N.B. 2 SNPV and SPPV are as independent of prevalence as are specificity and sensitivity and as are negative and positive Likelihood Ratios.

## Value

A list of

**$table** The confusion table of class x ref, where class is the classification based on the test, when applying the threshold(s). The reference standard (ref) has categories 0 and 1, while the classification based on the test scores (class) 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 classifications 0 or 1:

- prevalence: Proportion of classifiable patients with the targeted condition = (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)
- SNPV: standardized negative predictive value = specificity / (1- sensitivity + specificity)
- SPPV: standardized positive predictive value = sensitivity / (sensitivity + 1 - specificity)
- 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. Required.

`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 generally defined as an interval below and above the intersection, where the densities of the two distributions of patients with and without the targeted impairment are about equal. The various functions for the estimation of the uncertain interval use a sensitivity and specificity below a desired value (default .55). This function uses the intersection (the optimal dichotomous threshold) to divide the uncertain interval and provides in this way the indices for the uncertain interval when the optimal threshold would have been applied.

As a result, it may be 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 densities of the test scores.

The patients that have test scores within the Uncertain Interval are prone to be incorrectly 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 classification 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 (that is, the optimal threshold). 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 UI.class x ref for the Uncertain Interval where the scores are expected to be inconclusive. UI.class is the classification of the UI scores divided by the intersection, 0 (UI scores < intersection and 1 (UI scores >= intersection. This shows therefore the results when compared with applying the intersection is used (that is, the optimal dichotomous threshold). Both the reference standard (ref) and the classification based on the test scores (UI.class) have categories 0 and 1. 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: Classifiable patients with the targeted condition / Total 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)
- SNPV: standardized negative predictive value = specificity / (1- sensitivity + specificity)
- SPPV: standardized positive predictive value = sensitivity / (sensitivity + 1 - specificity)
- 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])
```

---

RPV *Trichotomization of ordinal test results using predictive values*

---

## Description

This function calculates Predictive Values, Standardized Predictive Values, Interval Likelihood Ratios and Posttest Probabilities of intervals or individual test scores of discrete ordinal tests. This function can correct for the unreliability of the test. It also trichotomizes the test results, with an uncertain interval where the test scores do not allow for an adequate distinction between the two groups of patients. This function is best applied to large samples with a sufficient number of patients for each test score.

## Usage

```
RPV(
  ref,
  test,
  pretest.prob = NULL,
  reliability,
  roll.length = NULL,
  extend = TRUE,
  decision.odds = 2,
 decision.use = c("standardized.pv", "posttest.probability", "LR", "predictive.value"),
  preselected.thresholds = c(NULL, NULL),
  digits = 3
)
```

## Arguments

| | |
|---|---|
| ref | A vector of two values, ordering 'patients with' > 'patients without', for instance 1, 0. When using a factor, please check whether the correct order of the values is used. |
| test | A vector of ordinal measurement level with a numeric score for every individual. When using a factor, please check whether the correct order of the values is used. Further, a warning message is issued concerning the calculation of the variance of the test when using a factor. |
| pretest.prob | (default = NULL) value to be used as pre-test probability. It is used for the calculation of the post-test probabilities. If pretest.prob = NULL, the sample prevalence is used. |
| reliability | (no default) The known reliability of the test, used to calculate Standard Error of Measurement (SEM). The reliability is expressed as an applicable correlation coefficient, with values between 0 and 1. A Pearson's Product Moment correlation or an Intra-Class Coefficient (ICC) will do. N.B. Setting parameter roll.length to 1 causes the reliability of the test to be ignored. |
| roll.length | (default = NULL) The frame length of the interval of test scores scores for the calculation of the reliable predictive values. When NULL, it is calculated as round(SEM)*2+1 (approximately a 68% confidence interval around the observed test score in which the true score is expected). The roll.length needs to be uneven. When the result is even, you need to choose a value 1 larger or smaller. When roll.length is supplied, this is used instead of the calculated value. Furthermore, roll.length has to be >= 1 and uneven. Applicable values are 1, 3, 5, etc. roll.length = 1 causes the reliability of the test to be ignored. |
| extend | (default = TRUE) The Reliable Predictive Values cannot be calculated for the most extreme scores. As the most extreme scores offer most often least uncertain decisions for or against the disease, the values that can be calculated are extended. When extend = FALSE, NAs (NOT Availables) are produced. |
| decision.odds | (default = 2). The minimum odds for and against the targeted impairment for a positive or negative classification. For the uncertain range of test scores both the odds for and the odds against are therefore between 2 and 1/2. A decision.odds of 1 causes all test scores to be used for either positive or negative decisions. The limit for the Predictive Value = decision.odds / (decision.odds+1); for a decision, the Predictive Values needs to be larger than this limit. NB 1 Decison.odds can be a broken number, such as .55/(1-.55), which defines the decision limit for |

predictive values as .55. The default is therefore (2/3) / (1 - (2/3)) = 2. NB 2 When a test is more reliable and valid, a higher value for decision.odds can be applied. NB 3 For serious diseases with relatively uncomplicated cures, decision odds can be smaller than one. In that case, a large number of false positives is unavoidable and positive decisions are inherently uncertain. See Sonis(1999) for a discussion.

`decision.use` (default = 'standardized.pv'). The probability to be used for decisions. When 'standardized.pv' is chosen, the standardized positive predictive value is used for positive decisions and the standardized negative predictive value is used for negative decisions. When 'posttest.probability' is chosen, pt.prob is used for positive decisions and (1 \- pt.prob) is used for negative decisions. When 'predictive.value' is chosen the rnpv are used for negative decisions and the rppv for positive decisions. N.B. These parameters can be abbreviated as 'stand', 'post' and 'pred'. N.B. The posttest probability is equal to the positive predictive value when pre-test probability = sample prevalence. N.B. The posttest probability is equal to the standardized positive predictive value when pre-test probability = .5.

`preselected.thresholds` (default = c(NULL, NULL)). For use in comparisons, when preselected.thresholds has valid values these values are used for the determination of the cut-points. The two cut-points indicate the limits of the uncertain area. Parameter decision.use is ignored. When preselected.thresholds[2] > preselected.thresholds[1], the higher scores are used for positive decisions. When preselected.thresholds[2] < preselected.thresholds[1], the lower scores are used for positive decisions. An uncertain Interval of a single test score can be determined with in-between values. For instance c(1.5, 0.8) defines an uncertain interval of test score 1 for a descending ordinal test.

`digits` the number of digits used in the output.

## Details

This function can be applied to ordinal data. Uncertain test scores are scores that have about the same density in the two distributions of patients with and without the targeted condition. This range is typically found around the optimal cut-point, that is, the point of intersection or Youden index (Schisterman et al., 2005). This function uses as a default the decision odds of ordinal test scores near 1 (default < 2). This results in a limit for the Predictive Values = decision.odds / (decision.odds+1).

N.B. 1: Sp = Negative Decisions | true.neg.status; Se = Positive Decisions | true.pos.status. Please note that the values for Se and Sp are underestimated, as the uncertain test scores are considered here as errors, which they are not. (Se and Sp are dichotomous indices.). Use `quality.threshold` and `quality.threshold.uncertain` for obtaining respectively quality indices for the test scores when ignoring test scores in the uncertain interval and the quality indices of the test scores within the uncertain interval.

N.B. 2: For the category Uncertain the odds are for the targeted condition (sum of patients with a positive.status)/(sum of patients with negative.status).

N.B. 3: Set roll.length to 1 to ignore the test reliability and obtain raw predictive values, likelihood ratios, etc., that are not corrected for the unreliability of the test.

Raw predictive values compare the frequencies and provide exact sample values and are most suitable for evaluating the sample results. When prevalence is low, Positive Predictive Values can be disappointingly low, even for tests with high Se values. When prevalence is high, Negative Predictive Values can be low. Reliable Standardized Predictive Values compare the densities (relative

frequencies) and are most suitable for comparing the two distributions of the scores for patients with and without the targeted condition.

The predictive values are calculated from the observed frequencies in the two samples of patients with and without the targeted disease. For a range of test scores x, if f0(x) and f1(x) are the frequencies of respectively patients without and with the targeted disease, then the negative predictive value (NPV) can be defined as: NPV(x) = f0(x) / (f0(x) + f1(x)) and the positive predictive value (PPV) as: PPV(x) = f1(x) / (f0(x) + f1(x)). The densities for a range of test scores x can be defined d0(x) = f0(x) / n0 and d1(x) = f1(x) / n1, where n0 and n1 are the number of observed patients in the two samples. The standardized negative predictive value (SNPV) is defined as SNPV(x) = d0(x) / (d0(x) + d1(x)) and the standardized positive predictive value (SPPV) as SPPV(x) = d1(x) / (d0(x) + d1(x)). The two distributions are weighed equally, or in other words, the prevalence is standardized to .5. N.B. The posttest probability is equal to the positive predictive value when the pretest probability is set to the sample prevalence, while the standardized positive predictive value is equal to the posttest probability when the pretest probability is set to .5.

Reliable estimates of the predictive probabilities correct to a certain degree for random variations. In test theory this random effect is estimated with the Standard Error of Measurement (SEM), which is directly dependent on the reliability of the test: `SEM = s * sqrt(1 -r)`, where s is the standard deviation of the test scores and r the estimated reliability of the test (Crocker & Algina, 1986; Harvill, 1991). The true score of a patient lies with some probability (roughly 68 of +- 1 SEM around the acquired test score. This provides information about the range of test scores that can be expected due to all kinds of random circumstances where no real changing agent has effect.

The results show the obtained values for the sample and are not corrected in any way. The classification 'Uncertain' shows the scores that lead to odds (d1(x) / d0(x)) that are lower than limit. This indicates that it is difficult to base classifications on that range of scores. The positive classifications are less error prone, with realized odds (d1(x) / d0(x)). These odds are close to 1 and smaller than the decision.odds. The negative classifications are less error prone than 'Uncertain' (odds = d0(x) / d1(x)).

The accuracy indices are shown as percentages. Sp = negative classifications given a true negative status. Se = positive classifications given a true positive status. NPV = proportion of Negative Classifications that are correct. PPV = proportion of Positive Classifications that are correct.

## Value

A list of:

**$parameters:** A named vector:

- pretest.prob: provided or calculated pre-test probability. Default, the calculated sample prevalence is used.
- sample.prevalence: the calculated sample prevalence.
- reliability: must be provided; ignored when roll.length = 0.
- SEM: the calculated Standard Error of Measurement (SEM).
- roll.length: the total length of the range around the test score (2 * SEM + 1).
- rel.conf.level: the confidence level of the range, given the reliability.
- limit: the limit applied to the values for calculating the decision result.

**$messages:** Two messages: 1. The test scores are reported for which reliable predictive values could not be calculated and have been extended from the nearest calculated value, 2. the kind of values (probabilities or LR) that are used for decisions.

**$rel.pred.values:** A table the test scores as columns and with rows: N.B. When roll.length is set to 1, the test reliability is ignored and the outcomes are not corrected for unreliability.

- rnpv: (more) reliable negative predictive value. Fitting for reporting sample results.

- rppv: (more) reliable positive predictive value.
- rsnpv: (more) reliable standardized negative predictive value.
- rsppv: (more) reliable standardized positive predictive value.
- rilr: (more) reliable interval likelihood ratio.
- rpt.odds: (more) reliable posttest odds.
- rpt.prob: (more) reliable posttest probabilities.

**$result:** Table of results for the current sample, calculated with the provided parameters.

- columns: Negative Classifications, Uncertain, Positive Classifications.
- row scores: range of test scores for the three categories.
- row total.sample: percentage of the total sample.
- row correct.decisions: percentages of correct negative and positive decisions (NPV and PPV).
- row true.neg.status: percentage of patients with a true negative status for the 3 categories.
- row true.pos.status: percentage of patients with a true positive status for the 3 categories.
- row realized.odds: The odds that are realized in the sample for each of the three categories. NB The odds of the uncertain range of test scores concerns the odds for the targeted condition.

## References

Sonis, J. (1999). How to use and interpret interval likelihood ratios. Family Medicine, 31, 432–437.

Crocker, L., & Algina, J. (1986). Introduction to classical and modern test theory. Holt, Rinehart and Winston, 6277 Sea Harbor Drive, Orlando, FL 32887 ($44.75).

Harvill, L. M. (1991). Standard error of measurement. Educational Measurement: Issues and Practice, 10(2), 33–41.

Landsheer, J. A. (In press). Impact of the Prevalence of Cognitive Impairment on the Accuracy of the Montreal Cognitive Assessment: The advantage of using two MoCA thresholds to identify error-prone test scores. Alzheimer Disease and Associated Disorders. https://doi.org/10.1097/WAD.0000000000000365

## See Also

synthdata_NACC for an example

## Examples

```
set.seed(1)
# example of a validation sample
ref=c(rep(0,1000), rep(1, 1000))
test=round(c(rnorm(1000, 5, 1), rnorm(1000, 8, 2)))
# calculated roll.length is invalid. Set to 3. Post test probability equals
# Positive Predictive Values. Parameter pretest.prob is set to sample prevalence.
RPV(ref, test, reliability = .9, roll.length = 3)
# Set roll.length = 1 to ignore test reliability (value of parameter
# reliability is ignored, but must be set to some value.)
RPV(ref, test, reliability = 0, roll.length = 1)
# When pretest.prob is set to .5, the Post-test Probabilities are equal to
# the Standardized Positive Predictive Values.
RPV(ref, test, pretest.prob = .5, reliability = .9, roll.length = 3)
```

synthdata_NACC *synthdata NACC*

## Description

NACC MoCA synthetic example data (2433 observations of patients with no clinical assessment of cognitive impairment and 2644 observations with a clinical assessment of some form of cognitive impairment.

- ID patient id; sequential, randomly assigned
- center Alphanumeric id of the clinical center where the data has been collected (30 centers)
- ref.1 Gold standard (true status) at the first measurement. 0: no cognitive impairment 1: cognitive impairment
- MOCATOTS.1 Total MoCA score at the first measurwment (0 .. 30)
- vdate.1 Date of the first measurement
- ref.2 Gold standard (true status) at the second measurement. 0: no cognitive impairment 1: cognitive impairment
- MOCATOTS.2 Total MoCA score at the second measurwment (0 .. 30)
- vdate.2 Date of the second measurement

## Details

For use as an example, a single data set of 6670 observations is generated based on the NACC dataset, from 30 different clinical centers. To generate the artificial data, the R package synthpop (Nowok B, Raab GM, Dibben C, 2016) is used to create the synthetic data, based on the original data from the Uniform Data Set (UDS), collected by the University of Washington's National Alzheimer's Coordinating Center (NACC). The syntetic data provide similar statistical results, but differ for each individual and each clinical center. These data are provided as data for the replication of the examples. Results of the real data are presented in Landsheer (In Press).

Researchers who want to use these data for other purposes than replication of the results presented here, are kindly requested to submit a new request for the original data to the NACC. The user of the data may either get a new file or request a file using the specifications of the original data file (https://www.alz.washington.edu/).

## References

Nowok B, Raab GM, Dibben C (2016). "synthpop: Bespoke Creation of Synthetic Data in R." Journal of Statistical Software, 74(11), 1–26. doi:10.18637/jss.v074.i11.

Landsheer, J. A. (In press). Impact of the Prevalence of Cognitive Impairment on the Accuracy of the Montreal Cognitive Assessment: The advantage of using two MoCA thresholds to identify error-prone test scores. Alzheimer Disease and Associated Disorders. https://doi.org/10.1097/WAD.0000000000000365

## Examples

```
data(synthdata_NACC) # needs R version 3.5 or later
head(synthdata_NACC) # Show head of the dataset
nrow(synthdata_NACC) # total number of observations
# select part of data for the first measurement
# N.B. ref is not available when it is inconclusive
```

```
m1 = synthdata_NACC[!is.na(synthdata_NACC$MOCATOTS.1)
                    & !is.na(synthdata_NACC$ref.1), ]
# preliminary check data for possible missing values
addmargins(table(m1$ref.1, m1$MOCATOTS.1, useNA = 'always'))
# Show the data
barplotMD(m1$ref.1, m1$MOCATOTS.1)

# calculate the difference between the two measurements in days
ddiff = (m1$vdate.2 - m1$vdate.1)
# There is a wide variety !!!
summary(ddiff)
# Estimate the test-retest reliability
library(psych)
ICC(na.omit(cbind(m1$MOCATOTS.1, m1$MOCATOTS.2)))
# Reducing the variety of time between measurements:
timesel = (ddiff >= 335) & (ddiff <= 395)
ICC(na.omit(cbind(m1$MOCATOTS.1[timesel], m1$MOCATOTS.2[timesel])))

# error when using default calculated value for roll.length
# RPV(m1$ref.1, m1$MOCATOTS.1, reliability = .86)
RPV(m1$ref.1, m1$MOCATOTS.1, reliability = .86, roll.length = 5)
```

---

TG.ROC *Two-Graphs Receiving Operating Characteristics.*

---

## Description

The function supports the determination and plot of the sensitivity and specificity against the possible thresholds and shows an intermediate range of test results that is considered as less accurate.

## Usage

```
TG.ROC(
  ref,
  test,
  Se.criterion = 0.9,
  Sp.criterion = 0.9,
  model = c("none", "binormal"),
  plot = FALSE,
  position.legend = "left",
  cex.legend = 1
)
```

## 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 numeric test scores under evaluation. Higher scores indicate the presence of the targeted disease. Please use negated values when lower values indicate the presence of the targeted disease. |

| | |
|---|---|
| `Se.criterion` | Default = .95. Minimum desired value of Se. |
| `Sp.criterion` | Default = .95. Minimum desired value of Sp. |
| `model` | Default = 'none'. Model to use, either binormal or none (non-parametric) |
| `plot` | Defaults= FALSE. Whether a plot is shown for Se and Sp against the thresholds. |
| `position.legend` | Default: 'left'. Position of the legend. Most used values: "left", "right". |
| `cex.legend` | Default: 1. Relative size of the legend. |

## Details

This function implements a non-parametric and a bi-normal model. See Landsheer(2018) for an evaluative description.

Warning: Although the test scores <= the lower limit and the test scores >= the upper limit are interpreted for respectively negative and positive classifications, the range of test values >= lower limit provides the desired positive accuracy (Se.citerion), while the range of test values <= upper limit provides the desired negative accuracy (Sp.citerion). This is problematic for its double count: the values in the intermediate zone are needed both for the desired Se and for the desired Sp value.

Please note that the definition of the intermediate interval deviates from the definition of an uncertain interval.

The TG-ROC (Two Graphs Receiver Operating Characteristics) plot shows the diminishing values of Se and increasing values of Sp against the possible thresholds.

## Value

Thresholds for the intermediate zone. Lower threshold < Test scores < Upper threshold is the intermediate range. The range of test values >= lower limit provides the desired positive accuracy (Se.citerion), while the range of test values <= upper limit provides the desired negative accuracy (Sp.citerion).

## References

Greiner, M. (1995). Two-graph receiver operating characteristic (TG-ROC): A Microsoft-EXCEL template for the selection of cut-off values in diagnostic tests. Journal of Immunological Methods, 185(1), 145-146.

Greiner, M. (1996). Two-graph receiver operating characteristic (TG-ROC): Update version supports optimisation of cut-off values that minimise overall misclassification costs. Journal of Immunological Methods, 191(1), 93-94.

Landsheer, J. A. (2018). The Clinical Relevance of Methods for Handling Inconclusive Medical Test Results: Quantification of Uncertainty in Medical Decision-Making and Screening. Diagnostics, 8(2), 32. https://doi.org/10.3390/diagnostics8020032

## Examples

```
ref = c(rep(0,100), rep(1,100))
test = c(rnorm(100, 0, 1), rnorm(100, 1, 1))
TG.ROC(ref, test, model='binormal', plot=TRUE)
TG.ROC(ref, test, model='none', plot=TRUE)
```

## 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.
- y rating 1 to 5: 5 definite metastatic disease to the liver; 4 probable metastatic disease to the liver; 3 possible metastatic disease to the liver; 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, 204-215 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 thresholds of an uncertain interval for bi-normal distributed test scores that are considered as inconclusive.*

### Description

Function for the determination of the thresholds of an uncertain interval for bi-normal distributed test scores that are considered as inconclusive.

### 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 using the function get.intersection. |
| 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 generally defined as an interval below and above the intersection, where the densities of the two distributions of patients with and without the targeted condition are about equal. These test scores are considered as inconclusive for the decsion for or against the targeted condition. This function uses for the definition of the uncertain interval a sensitivity and specificity of the uncertain test scores below a desired value (default .55).

Only a single intersection is assumed (or a 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.

The function uses an optimization algorithm from the nlopt library (https://nlopt.readthedocs.io/en/latest/NLopt_Algorith the sequential quadratic programming (SQP) algorithm for nonlinearly constrained gradient-based optimization (supporting both inequality and equality constraints), based on the implementation by Dieter Kraft (1988; 1944).

## 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.

## References

Dieter Kraft, "A software package for sequential quadratic programming", Technical Report DFVLR-FB 88-28, Institut für Dynamik der Flugsysteme, Oberpfaffenhofen, July 1988.

Dieter Kraft, "Algorithm 733: TOMP–Fortran modules for optimal control calculations," ACM Transactions on Mathematical Software, vol. 20, no. 3, pp. 262-281 (1994).

Landsheer, J. A. (2018). The Clinical Relevance of Methods for Handling Inconclusive Medical Test Results: Quantification of Uncertainty in Medical Decision-Making and Screening. Diagnostics, 8(2), 32. https://doi.org/10.3390/diagnostics8020032

## 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 Uncertain Interval is generally defined as an interval below and above the intersection, where the densities of the two distributions of patients with and without the targeted condition are about equal. These test scores are considered as inconclusive for the decision for or against 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 function can be used for a test without a defined distribution of the continuous test scores. The Uncertain Interval is generally defined as an interval below and above the intersection, where the densities of the two distributions of patients with and without the targeted condition are about equal. This function uses for the definition of the uncertain interval a sensitivity and specificity of the uncertain test scores below a desired value (default .55).

This essentially non-parametric function finds the best possible solution for a sample. This function can be used for test with continuous scores or for test with about twenty or more ordered 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 a second intersection where the overlap is negligible). 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 ordered tests with distinguishable 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.

## 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.

Landsheer, J. A. (2018). The Clinical Relevance of Methods for Handling Inconclusive Medical Test Results: Quantification of Uncertainty in Medical Decision-Making and Screening. Diagnostics, 8(2), 32. https://doi.org/10.3390/diagnostics8020032

## 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 explore possible uncertain intervals of ordinal test results of individuals with (1) and without (0) the targeted condition.*

## Description

This function explores possible uncertain intervals (UI) of the test results of the two groups. This functions allows for considerable fine-tuning of the characteristics of the interval of uncertain test scores, in comparison to other functions for the determination of the uncertain interval and is intended for tests with a limited number of ordered values and/or small samples.

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. When a sufficiently large dataset is available, the function `RPV` may be preferred for the analysis of ordered data.

## 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 or predictor under evaluation. A column in a dataset or vector indicating the test results on an ordinal 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 (AUC), 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 (AUC), Acc (accuracy), 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 (AUC) 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 the values of these constraints, 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,model='ordinal').`

`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. This problem is aggrevated when samples are small. For any threshold determination, one needs a large representative sample (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 selected on various properties of the uncertain interval. The defaults are C (AUC) 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 densities function `plotMD` 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). Dichotomization loss L2 can be defined as the sum of false negatives and false positives. The Youden threshold minimizes these. The Loss formula L3 for trichotomization of ordinal test scores is (created by

https://www.codecogs.com/latex/eqneditor.php):

$$L_3 = \frac{\left(\sum_{i=l}^{u} |d0_i - d1_i| + \sum_{i=u+1}^{h} d1_i + \sum_{i=1}^{l-1} d0_i\right)}{N}$$

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 AUROCC (Area Under Receiving Operating Characteristics Curve or AUC)

**$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 plus minus 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 trichotomization. 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

Youden, W. J. (1950). Index for rating diagnostic tests. Cancer, 3(1), 32-35. https://doi.org/10.1002/1097-0142(1950)3:1<32::AID-CNCR2820030106>3.0.CO;2-3

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.

Landsheer, J. A. (2018). The Clinical Relevance of Methods for Handling Inconclusive Medical Test Results: Quantification of Uncertainty in Medical Decision-Making and Screening. Diagnostics, 8(2), 32. https://doi.org/10.3390/diagnostics8020032

## See Also

plotMD or barplotMDfor plotting the mixed densities of the test values. density for the parameters of the density function. ui.nonpar or ui.binormal can be used when more than 20 values can be distinguished on the ordinal test scale. When a large data set for an ordinal test is available, one might consider RPV.

## 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

Uncertain test scores are scores that have about the same density in the two distributions of patients with and without the targeted condition. This range is typically found around the optimal cut-point, that is the point of intersection or Youden index (Schisterman et al., 2005). Most functions use a specified low value for Se and Sp to find this uncertain interval (default Se = Sp = .55). The most recent added function `RPV` uses the odds of Standardized Predictive Values of ordinal test scores near 1 (default < 2).

This library also contains two alternative definitions. Coste et al. (2003) defined a grey zone in between positive and negative conclusions (see `greyzone`), and Greiner (1995) defined a middle inconclusive zone of intermediate values (see `TG.ROC`). See Index for all available functions and plot possibilities.

## 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.

Landsheer, J. A. (2018). The Clinical Relevance of Methods for Handling Inconclusive Medical Test Results: Quantification of Uncertainty in Medical Decision-Making and Screening. Diagnostics, 8(2), 32. https://doi.org/10.3390/diagnostics8020032

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.

Greiner, M. (1995). Two-graph receiver operating characteristic (TG-ROC): A Microsoft-EXCEL template for the selection of cut-off values in diagnostic tests. Journal of Immunological Methods, 185(1), 145-146.

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

## See Also

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

# Index