

Package ‘DiagnosisMed’

February 14, 2012

Version 0.2.3

Date 2010-03-04

Author Pedro Brasil <pedro.brasil@ipeec.fiocruz.br>

Maintainer Pedro Brasil <pedro.brasil@ipeec.fiocruz.br>

Depends R (>= 2.7.2),epitools, TeachingDemos, tcltk, AMORE,utils

Title Diagnostic test accuracy evaluation for medical professionals.

Description DiagnosisMed is a package to analyze data from diagnostic test accuracy evaluating health conditions. It is being built to be used by health professionals. This package is able to estimate sensitivity and specificity from categorical and continuous test results including some evaluations of indeterminate results, or compare different categorical tests, and estimate reasonable cut-offs of tests and display it in a way commonly used by health professionals. No graphical interface is available yet. Partners are most welcome.

License GPL (>= 2)

URL <http://r-forge.r-project.org/projects/diagnosismed/>

Repository CRAN

Repository/R-Forge/Project diagnosismed

Repository/R-Forge/Revision 19

Date/Publication 2010-03-09 19:59:44

R topics documented:

diagnosis	2
interact.ROC	5
LRgraph	7
ROC	9
rocdata	13
TGROC	14
tutorial	19

diagnosis	<i>Diagnostic test accuracy evaluation</i>
-----------	--

Description

`diagnosis` estimate sensitivity, specificity, predictive values, likelihood ratios, area under ROC curve and other validity measures for binary diagnostic test evaluation. It accepts as input either columns from a dataset or vectors, a 2 x 2 table or numbers representing true positives, false negatives, false positives and true negatives. `plot` for `diagnosis` draw a simple nomogram and `summary` for `diagnosis` creates an output in a table format that allows the output to be easily exported to a spreadsheet.

Usage

```
diagnosis(a,b,c,d, CL = 0.95, print = TRUE, plot = FALSE)
## S3 method for class 'diag'
plot(x, print = FALSE, ...)
## S3 method for class 'diag'
print(x, ...)
## S3 method for class 'diag'
summary(object, ...)
```

Arguments

<code>a</code>	A number representing True Positives from a 2x2 table. Also, TP could either a 2x2 table (see below) or a column in a dataset representing the reference standard.
<code>b</code>	A number representing False Negatives from a 2x2 table. If TP is a column in a dataset FN should also be a columns in a dataset, however representing the index test.
<code>c</code>	A number representing False Positives from a 2x2 table
<code>d</code>	A number representing True Negatives from a 2x2 table
<code>print</code>	If TRUE, <code>diagnosis</code> will print in the output window the statistics resulted from the 2x2 table. For <code>plot</code> , this will print in the output window a table with all pre-test and its corresponding post-test probabilities.
<code>plot</code>	There are two options of <code>plot</code> . If <code>plot</code> is called in <code>diagnosis</code> , then a ROC curve of the test under evaluation plotted. If <code>plot</code> is called from an object storing <code>diagnosis</code> output (see example) than a nomogram is plotted. These plots may later be edited, as any other plot, with title, legends etc. Default is FALSE.
<code>x</code>	For <code>plot</code> and <code>print</code> functions, <code>x</code> is an object assigned with <code>diagnosis</code> output.
<code>CL</code>	Confidence limits for confidence intervals. Must be a numeric value between 0 and 1. Default is 0.95.
<code>object</code>	For <code>summary</code> function, 'object' is an object assigned with <code>diagnosis</code> output.
<code>...</code>	Other options passed to <code>print</code> , <code>plot.default</code> or <code>summary</code> .

Details

In diagnosis, the values entered must be either two variables in a data frame, a 2 x 2 table or numbers corresponding to 2 x 2 table cells. If vectors or columns from a dataset, the first one should be the gold standard and the second should be the index test. These two variables must be coded either as numeric - 0 for negative and 1 for a positive test - or with the words "positive" and "negative" or "presence" and "absence". In an older version, there was diagnosisI function that was replaced by diagnosis function. The values of a 2 x 2 table can be inputted as: TP is true positive; TN is true negative; FP is false positive and FN is false negative. Sensitivity, Specificity, Predictive values and Accuracy confidence limits rely on binomial distribution, which does not give result outside [0:1] such as normal distribution or asymptotic theory. DOR, Likelihood ratios and Youden J index confidence limits rely on normal approximation (Wald method for likelihoods). The AUC (area under the ROC curve) is estimated by trapezoidal method (see below). Also, these functions have a summary function which creates a matrix as a result (identical to the default print option) which allows to easily export the results to a spreadsheet or to a odt file (with OdfWeave) in a table format (see example). If the input is a 2 x 2 table it should be formatted as:

TN	FN
FP	TP

plot.diag will draw a very simple nomogram as many examples from wikipedia <http://en.wikipedia.org/wiki/Nomogram>. This is not a generic nomogram as shown in many evidenced based medicine texts, because this one shows only pre-test and post-test variations with a fixed positive likelihood ratio. This likelihood is a statistic from an object created by diagnosis function. Its usage is the same as applying the Bayes theorem where the pre-test odds times positive likelihood ratio equals the post-test odd (transforming the odds to probabilities). To use it, draw, with a ruler, a vertical line from a desired pre-test probability, and to find the corresponding post-test probability, draw a horizontal line from the intersection of the curve and the vertical line toward the vertical axis.

Value

A 2x2 table from which the validity measures are calculated.

Sample size	The number of subjects analyzed.
Prevalence	The proportion classified as with the target condition by the reference standard
Sensitivity	the probability of the test to correctly classify subjects with the target condition (TP/(TP+FN))
Specificity	the probability of the test to correctly classify subjects without the target condition (TN/(TN+FP))
Predictive values	the probabilities of being with (positive predictive value) (TP/(TP+FP)) or without (negative predictive value) the target condition given a test result (TN/(TN+FN)).
Likelihood ratios	the probability of test a result in people with the target condition, divided by the probability of the same test result in people without the target condition (PLR = Se/(1-Sp); NLR = (1-Sp)/Se)

Diagnostic odds ratio	represents the overall discrimination of a dichotomous test, and is equivalent to the ratio of PLR and NLR.
Error trade off	Is the amount of false positives traded with false negatives for each decision threshold; here expressed as an odd - for binary results there is only one threshold
Error rate	Expresses how many errors we make when we diagnose patients with an abnormal test result as diseased, and those with a normal test result as non-diseased ((FP+FN)/sample size).
Accuracy	overall measure that express the capacity of the test to correctly classify subjects with and without the target condition ((TP+TN)/(sample size))
Area under ROC curve	overall measure of accuracy - here the method is the trapezoidal. It gives identical results as (Se+SP)/2.

Note

Bug reports, malfunctioning, or suggestions for further improvements or contributions can be sent, preferentially, through the DiagnosisMed email list, or R-Forge website <https://r-forge.r-project.org/projects/diagnosismed/>.

Author(s)

Pedro Brasil - <diagnosismed-list@lists.r-forge.r-project.org>

References

- Knotterus. The Evidence Based Clinical Diagnosis; BMJBooks, 2002.
- Xiou-Hua Zhou, Nancy A Obuchowsky, Donna McClish. Statistical Methods in diagnostic Medicine; Wiley, 2002.
- Simel D, Samsa G, Matchar D (1991). Likelihood ratios with confidence: Sample size estimation for diagnostic test studies. Journal of Clinical Epidemiology 44: 763 - 770

See Also

[LRgraph](#), [ROC](#), [LRgraph](#), [binom.conf.int](#), [sensSpec](#), [epi.tests](#), [nomogram](#), [epi.nomogram](#)

Examples

```
# Simulating a dataset
mydata <- as.data.frame(rbind(
  cbind(rep(c("positive"),18),rep(c("negative"),18)),
  cbind(rep(c("positive"),72),rep(c("positive"),72)),
  cbind(rep(c("negative"),25),rep(c("positive"),25)),
  cbind(rep(c("negative"),149),rep(c("negative"),149))
))
colnames(mydata) <- c('culture', 'serology')
# A little description of the data set to check if it is ok!
```

```

str(mydata)
# Attaching the data set and checking
attach(mydata)

# Running the diagnosis analysis
diagnosis(culture,serology)

#Simulating a table
mytable <- matrix(c(149,18,25,72), nrow = 2, ncol=2, byrow=TRUE,
                  dimnames = list(enzyme=c('absent','present'),
                                   citology=c('absent','present')))
# Running analysis from a 2 x 2 table
diagnosis(mytable)

#Inserting values as isolated numbers
diagnosis(72,18,25,149)

#-----
# Export results to a spreadsheet:
#-----

# Assing diagnosis to an object
mytest <- diagnosis(364,22,17,211,print=FALSE)
# Assign the summary to an object
mt.sum <- summary(mytest)
# Export to a spreadsheet using csv format - could also work to text with OdfWeave export.
# write.csv(mt.sum,'MytestResults.csv',quote = F,na='')

# Draw a nomogram from a test
plot(mytest)

rm(mydata,mytable,mytest,mt.sum)

```

interact.ROC

Interactively draw a ROC curve with your data

Description

Draw a ROC curve with the user data, interactively, sliding a button, and watch how changes at the cutoff, correlate with the changes of Sensitivity and Specificity while building the ROC curve itself.

Usage

```
interact.ROC(gold, test)
```

Arguments

<code>gold</code>	A column in a data frame or a vector indicating the classification by the reference test. Must be coded either as 0 - without target condition - or 1 - with the target condition.
<code>test</code>	A column in a data frame or a vector indicating the test under study (index test) results. Must be numeric .

Details

`interact.ROC` is a call from `roc.demo` function in `TeachingDemo` package. The difference is that `interact.ROC` allow the input data be displayed as usually is in diagnostic studies, a column with the test result and the other with the reference standard results. Inside this function the "ask" option - which controls the "next" button to see the next graph - is turned off - `par(ask=FALSE)`. Also, other options must be set to this function to work fine. Type "`options()`". The option `ask` and `device.ask.default` should be set as `FALSE`, as they usually are by default. To turn it on again later, type `par(ask=TRUE)`. The test must have a rationale that higher values of the index test belong to those with the target disease and those with lower values belong to those without the target disease. If this is not the case, the suggestion is to transform the tests results by multiplying it by -1 before running `interact.ROC`.

Value

`interact.ROC` generates two graphs in the same window: the upper graph is a ROC graph (Sensitivity on the vertical axis and 1-Specificity on the horizontal axis); the lower graph is a density plot (the density on the vertical axis and the test cut-off (or threshold) on the horizontal axis). With a sliding button is possible to interact and see how the Sensitivity and Specificity changes while changing the cut-off. In the upper graph the cut-off is represented by the different dots and the purple line represents the distance to the "optimal" threshold. At the lower graph, the red line and dashes represent the density and the test result from those without the target condition respectively. While the blue ones represent those with the target condition. If the dashes are at the bottom of the lower graph then they are classified as without the target condition, if at the top, with the target condition. The green vertical line represents the cut-off and changes with the sliding button. The cut-off itself can be seen right above the sliding button and the respective sensitivity and specificity at the bottom of the graph window.

Note

Bug reports, malfunctioning, or suggestions for further improvements or contributions can be sent, preferentially, through the `DiagnosisMed` email list, or R-Forge website <https://r-forge.r-project.org/projects/diagnosismed/>.

Author(s)

Pedro Brasil - <diagnosismed-list@lists.r-forge.r-project.org>

References

JA Knotterus, *The Evidence Based Clinical Diagnosis*; BMJBooks, 2002

See Also

[diagnosis](#), [ROC](#), [TGRROC](#), [performance](#), [roc.demo](#), [contROC](#)

Examples

```
data(rocdata)
attach(rocdata)
interact.ROC(Gold, test2)
rm(rocdata)
```

LRgraph

Comparing diagnostic tests: a simple graphic using likelihood ratios.

Description

LRgraph graphically compares two or more (all of them with the first test) diagnostic tests with binary results through their likelihood ratios, based on the rationale that the predictive ability of a test is a more interesting characteristic than sensitivity and/or specificity. It is possible to see through the graph that if the tests with smaller sensitivity or specificity may have superior predictive ability, that is, increases the prediction ability with small sensitivity/specificity trade-off.

Usage

```
LRgraph(tests, lwd = 2, lty = 1, cex = 1, leg.cex = 1.5, pt.cex = 2, ...)
```

Arguments

tests	a is a object composed by two or more tests. This object should be created binding two objects created by diagnosis functions as 'cbind(mytest1,mytest2)'. The user may insert as many tests as one wishes. See below.
lwd	Line width. See par , points , legend
lty	Line type. See par
cex	Symbols and text size. See par , points
leg.cex	Legend text size, this will replace the cex option in the legend. See legend
pt.cex	Size of the symbols in the legend. See legend
...	Other graph parameters. See plot.default

Details

When a diagnostic test has both sensitivity and specificity higher than a competing test is easy to see that the former is superior than the later. However, sometimes a test may have superior sensitivity and inferior specificity (or the other way around). In this case, a good decision may be toward the test that have a better prediction ability. The graph visually helps the user to see and compare these abilities. The graph is very similar to the ROC graph. The vertical and horizontal axis have the same length as the ROC graph. However, the diagnostic tests are represented as dots instead of

curves. The solid line passing through (0,0) is the likelihood ratio positive-line and the solid line passing through (1,1) is the likelihood ratio negative-line. Both negative and positive likelihood are numerically equivalent to the slopes of the solid lines. The solid lines split the graph into four areas (run the example). Also, there are dashed lines representing the sensitivity and specificity of the first test plotted. One may see that there are areas that a test may have superior sensitivity (or specificity) and yet the dot may be below the likelihood solid line. That is because the sensitivity / specificity trade-off is not reasonable, making the test with less predictive ability.

Value

Returns only a graph which is divided in four areas by the black solid lines (the likelihood ratios of the first test). The interpretation of the comparisons will depend on which area the competing tests will fall in. See and run the example to have the idea on how interpretation must be done.

Note

Bug reports, malfunctioning, or suggestions for further improvements or contributions can be sent, preferentially, through the DiagnosisMed email list, or R-Forge website <https://r-forge.r-project.org/projects/diagnosismed/>.

Author(s)

Pedro Brasil - <diagnosismed-list@lists.r-forge.r-project.org>

References

Biggerstaff, B.J. Comparing diagnostic tests: a simple graphic using likelihood ratios. *Statistics in Medicine*. 2000; 19(5):649-663

See Also

[diagnosis](#)

Examples

```
# Making tests with diagnosis function with different performances for comparison.
# mytest5 is the one which all others will be compared with.
mytest5<-diagnosis(80,20,20,80,print=FALSE)

# mytest1 has higher sensitivity and specificity.
# mytest1 is overall superior compared to mytest5.
mytest1<-diagnosis(90,10,10,90,print=FALSE)

# mytest2 has lower sensitivity but higher specificity.
# mytest2 is better to identify the presence of the target condition compared to mytest5.
mytest2<-diagnosis(72,28,3,97,print=FALSE)

# mytest3 has higher sensitivity but lower specificity.
# mytest3 is better to identify the absence of the target condition compared to mytest5.
mytest3<-diagnosis(92,8,37,63,print=FALSE)
```

```

# mytest41 has lower sensitivity and specificity.
# mytest41 is overall inferior compared to mytest5.
mytest41<-diagnosis(72,28,35,65,print=FALSE)

# mytest42 has lower specificity but higher sensitivity.
# Nevertheless, mytest42 still is overall inferior compared to mytest5.
mytest42<-diagnosis(82,18,42,58,print=FALSE)

# But that becomes clear only after plotting the tests.
LRgraph(cbind(mytest5,mytest1,mytest2,mytest3,mytest41,mytest42),cex=2.5)

# The texts below are not part of the function but helps to understand the areas
text(x=.5, y =.5, labels ="Area 4: Overall inferior", col="lightgray",cex=.8)
text(x=.5, y =1, labels ="Area 2: Absence", col="lightgray",cex=.8)
text(x=.07, y =.68, labels ="Area 3: Presence", col="lightgray",cex=.8)
text(x=.1, y =1, labels ="Area 1: Overall superior", col="lightgray",cex=.8)

rm(mytest1)
rm(mytest2)
rm(mytest3)
rm(mytest41)
rm(mytest42)
rm(mytest5)

```

ROC

Draw a ROC curve, estimate good cut-offs and compute validity measures for each cut-off

Description

Draw a non-parametric (empirical) ROC - receiver operating characteristic - curve and compute test sensitivity, specificity, predictive values and likelihood ratios (and respective confidence limits) for each decision threshold. Estimate good decision threshold by a variety of methods.

Usage

```

ROC(gold,
    test,
    CL = 0.95,
    Cost = 1,
    Prevalence = 0,
    Plot = TRUE,
    Plot.point = "Min.ROC.Dist",
    p.cex=1,
    Full = FALSE,
    Print = TRUE)
## S3 method for class 'ROC'
plot(x,
    Plot.point = "Min.ROC.Dist",

```

```

    cex.sub=.85,
    p.cex=1,
    ...)
## S3 method for class 'ROC'
print(x,Full=FALSE,...)

```

Arguments

gold	The reference standard. A column in a data frame indicating the classification by the reference test. The reference standard must have two levels: must be coded either as 0 - without target condition - or 1 - with the target condition; or could be coded as.factor with the words "negative" - without target condition - and "positive" - with the target condition.
test	The index test or test under evaluation. A column in a dataframe or vector indicating the test results in a continuous scale. Might also work with discrete or ordinal scale.
CL	Confidence limit. The limits of the confidence interval. Must be coded as number in a range from 0 to 1. Default value is 0.95
Cost	Cost = cost(FN)/cost(FP). MCT will be used to estimate a good cut-off. It is a value in a range from 0 to infinite. Could be financial cost or a health outcome with the perception that FN are more undesirable than FP (or the other way around). This item will run into MCT (misclassification cost term) - $(1 - \text{prevalence}) * (1 - \text{Sp}) + \text{Cost} * \text{prevalence} * (1 - \text{Se})$. Cost=1 means FN and FP have even cost. Cost = 0.9 means FP are 10 percent more costly. Cost = 0.769 means that FP are 30 percent more costly. Cost = 0.555 means that FP are 80 percent more costly. Cost = 0.3 means that FP are 3 times more costly. Cost = 0.2 means that FP are 5 times more costly. Also, it can be more easily inserted as any ratio such as 1/2.5 or 1/4.
Prevalence	Prevalence of the disease in the population who the test will be performed. It must be a value from 0 to 1. If left 0 (the default value), this will be replaced by the disease prevalence in the sample. This value will be used in the MCT and Efficiency formulas to estimate good cut-offs.
Plot	If FALSE, the ROC curve plot will not be displayed. Default is TRUE. If default options of graphics parameters from par are not satisfactory, then the suggestion is to assign the output into a object and call options from <code>plot</code>
Plot.point	The method of best cut-off estimation which will be displayed at ROC curve as a dot. Default is "Min.ROC.Dist". Possible options are: "None" - only the AUC in the legend will appear; "Max.Accuracy" - the cut-off which maximize the accuracy; "Max.DOR" - the cut-off which maximize the diagnostic odds ratio; "Error.rate" - the cut-off which minimizes the error rate; "Max.Accuracy.area" - the cut-off which maximize the accuracy area; "Max.Sens+Spec" - the cut-off which maximize the sum of sensitivity with specificity; "Max.Youden" - the cut-off which maximize the Youden index; "Se=Sp" - the cut-off which Sensitivity is equal to Specificity;

	"Min.ROC.Distance" - the cut-off which minimize the distance between the curve and the upper left corner of the graph;
	"Max.Efficiency" - the cut-off which maximize the efficiency;
	"Min.MCT" - the cut-off which minimize the misclassification cost term.
cex.sub	The magnification to be used for sub-titles relative to the current setting of cex. See par
p.cex	Symbol expansion - a numerical vector - passed to points. See points
Print	If FALSE, no results (detailed below in values section) will be displayed on the output window. Default is TRUE
Full	If TRUE, a table with sensitivity, specificity, predictive values and likelihood ratios (and respective confidence limits) for each decision threshold will be displayed. Default is FALSE.
x	For the plot and print functions, x is an object storing ROC function output.
...	Other plot or print parameters form plot.default

Details

Tests results matching the cut-off values will be considered a positive test. ROC assumes that subjects with higher values of the test are with the target condition and those with lower values are without the target condition. Tests that behave like glucose (middle values are supposed to be normal and extreme values are supposed to be abnormal) and immunofluorescence (lower values - higher dilutions - are supposed to be abnormal) will not be correctly analyzed. In the latter, multiplying the test results by -1 or other transformation before analysis could make it work. The AUC (area under the ROC curve) is estimated by the trapezoidal method (also known as Mann-Whitney statistic), its confidence interval is estimated by DeLong method. The AUC confidence limits should be used only to compare the AUC with the null value for AUC which is 0.5 and not to compare the AUC from different tests. The validity measures such as Sensitivity, Specificity and Likelihood ratios and its confidence limits are estimated as in [diagnosis](#) function. If ROC output is assign to an object (see example), tests results could be easily exported to a spreadsheet and other graphics that might be of interest could be easily done.

Diagnostic odds ratio: $DOR = (TP * TN) / (FN * FP)$; *thesameas* : $DOR = PLR / NLR$

Accuracy area: $AA = (TP * TN) / ((TP + FN) * (FP + TN))$

Youden index: $Y = Se + Sp - 1$; *thesameas* : $Y = Se - FPR$

Minimum ROC distance: $mROCDis = (Sp - 1)^2 + (1 - Se)^2$

Efficiency: $Ef = Se * prevalence + (1 - prevalence) * Sp$

Misclassification Cost Term: $MCT = (1 - prevalence) * (1 - Sp) + (cost(FN) / cost(FP)) * prevalence * (1 - Se)$

Value

pop.prevalence	The disease prevalence informed by the user. If not informed, it will be the same as the sample prevalence.
sample.prevalence	The disease prevalence in the sample

<code>sample.size</code>	The number of subjects analyzed
<code>test.summary</code>	A table showing the quintiles, mean and standard deviation of overall test results, test results from those with the target condition and without the target condition
<code>AUC.summary</code>	A table showing the AUC estimated by DeLong method (trapezoidal) and its confidence limits.
<code>test.best.cutoff</code>	A table showing the best cut-offs estimated by methods described above, its corresponding sensitivity, specificity and positive likelihood ratio (and their confidence limits)

Note

Bug reports, malfunctioning, or suggestions for further improvements or contributions can be sent, preferentially, through the DiagnosisMed email list, or R-Forge website <https://r-forge.r-project.org/projects/diagnosismed/>.

Author(s)

Pedro Brasil; Beranrdo Rangel Tura - <diagnosismed-list@lists.r-forge.r-project.org>

References

- Knotterus. The Evidence Based Clinical Diagnosis; BMJBooks, 2002.
- Xiou-Hua Zhou, Nancy A Obuchowsky, Donna McClish. Statistical Methods in diagnostic Medicine; Wiley, 2002.
- Simel D, Samsa G, Matchar D (1991). Likelihood ratios with confidence: Sample size estimation for diagnostic test studies. *Journal of Clinical Epidemiology* 44: 763 - 770
- S.B. Cantor, C.C. Sun, G. Tortolero-Luna, R. Richards-Kortum, and M. Follen. (1999) A comparison of C/B ratios from studies using receiver operating characteristic curve analysis. *Journal of Clinical Epidemiology*, 52(9):885-892.
- Greiner, M. (1996) Two-graph receiver operating characteristic (TG-ROC): update version supports optimisation of cut-off values that minimize overall misclassification costs. *J.Immunol.Methods* 191:93-94.
- Gengsheng Qin, Lejla Hotilovac. Comparison of non-parametric confidence intervals for the area under the ROC curve of a continuous-scale diagnostic test. *Statistical Methods in Medical Research* 2008; 17:207-221.

See Also

[binom.conf.int](#),[diagnosis](#),[interact.ROC](#),[TGROC](#),[performance](#),[contROC](#)

Examples

```
# loading a dataset
data(rocdata)
# Attaching the data set.
attach(rocdata)
# A little description of the data set to check if it is ok!
```

```

str(rocdata)
# Running ROC analysis with the full table option
# and storing ROC objects into 'x' from which there are tables to draw the graphs below.
x<-DiagnosisMed::ROC(Gold,test2,Full=TRUE)
# There is no need to stick the package name before the function if it is loaded as first in search path!
# Adding a title to the graph.
title(main="ROC graph")
# Some graphs that may be of interest. Validity measures at each test value.
# Setting the plot window to get nine graphs
# Some graphs showing some validity measures and some indexes variations used to choose good cut-offs
par(mfrow=c(3,3))
plot(x$test.diag.table$test.values,x$test.diag.table$DOR,type="l",ylab="DOR",xlab="Test values")
title(main="Test values x DOR")
plot(x$test.diag.table$test.values,x$test.diag.table$MCT,type="l",ylab="MCT",xlab="Test values")
title(main="Test values x MCT")
plot(x$test.diag.table$test.values,x$test.diag.table$Efficiency,type="l",ylab="Efficiency",xlab="Test values")
title(main="Test values x Efficiency")
plot(x$test.diag.table$test.values,x$test.diag.table$Youden,type="l",ylab="Youden index",xlab="Test values")
title(main="Test values x Youden index")
plot(x$test.diag.table$test.values,x$test.diag.table$PLR,type="l",ylim=c(0,49),ylab="Likelihood ratios",xlab="Test values")
lines(x$test.diag.table$test.values,x$test.diag.table$NLR,type="l",lty=2)
legend("right",lty=c(1,2),legend=c("PLR","NLR"),bty = 'n')
title(main="Test values x Likelihood ratios")
plot(x$test.diag.table$test.values,x$test.diag.table$PPV,type="l",ylab="Predictive values",xlab="Test values")
lines(x$test.diag.table$test.values,x$test.diag.table$NPV,type="l",lty=2)
legend("bottomright",lty=c(1,2),legend=c("PPV","NPV"),bty = 'n')
title(main="Test values x Predictive values")
plot(x$test.diag.table$test.values,x$test.diag.table$Accuracy.area,type="l",ylab="Accuracy area",xlab="Test values")
title(main="Test values x Accuracy area")
plot(x$test.diag.table$test.values,x$test.diag.table$MinRocDist,type="l",ylab="ROC distance",xlab="Test values")
title(main="Test values x ROC distance")
plot(x$test.diag.table$test.values,x$test.diag.table$Accuracy,type="l",ylab="Error rate & Accuracy",xlab="Test values")
lines(x$test.diag.table$test.values,x$test.diag.table$error.rate,type="l",lty=2)#,xlim=c(0,2.5))
legend("bottomright",lty=c(1,2),legend=c("Accuracy","Error rate"),bty = 'n')
par(mfrow=c(1,1))

# Also, results from ROC analysis could easily exported to a spreadsheet file or to a odt file by OdfWeave.
# Exporting the full table:
# write.csv(x$test.diag.table[,-c(2:5,24:34)],'MytestFulltable.csv')
# Exporting AUC summary:
# write.csv(x$AUC.summary,'MytestAUC.csv')
# Exporting Test summary:
# write.csv(x$test.summary,'MytestSummary.csv')
# Exporting Test best-cut-offs table:
# write.csv(x$test.best.cutoff,'MytestBestcutoff.csv')
rm(rocdata,x)

```

Description

This data set gives results of two serological tests - continuous scale - and its corresponding reference test classification - 0 for negative and 1 for positive.

Usage

```
rocddata
```

Format

A data.frame containing 3 variables and 148 observations.

Source

Not informed

References

Not informed

TGROC

TG-ROC - Two Graphic Receiver Operating Characteristic

Description

TGROC draws a graph of sensitivity and specificity with the variations of a diagnostic test scale. Also, it demonstrates which cut-offs (or decision thresholds) may trichotomize the test results into a range where the test is good to identify those with the target condition, a inconclusive range and a range where the test is good to identify those without the target condition according with the researcher tolerance. Also, it estimates and graphically demonstrates good cut-offs by different methods. TGROC estimates non-parametric statistics and uses the AMORE package to simulate the parametric curve and values with a neural network.

Usage

```
TGROC(gold,
      test,
      Cost=1,
      CL=0.95,
      Inconclusive=0.95,
      Prevalence=0,
      t.max=NULL,
      t.min=NULL,
      precision=.0001,
      n.neurons=c(1,5,1),
      learning.rate.global=1e-2,
      momentum.global=0.3,
```

```

    error.criterium="LMS",
    Stao=NA,
    hidden.layer="sigmoid",
    output.layer="sigmoid",
    method="ADAPTgdwm",
    report=FALSE,
    show.step=5000,
    n.shows=1,
    Plot="Both",
    Plot.inc.range=TRUE,
    Plot.Cl=FALSE,
    Plot.cutoff="None",
    cex=0.5,
    cex.sub=0.85,
    Print=TRUE)
## S3 method for class 'TGROC'
plot(x,...,
      Plot="Both",
      Plot.inc.range=TRUE,
      Plot.Cl=FALSE,
      Plot.cutoff="None",
      cex=0.5,
      cex.sub=0.85)

```

Arguments

gold	The reference standard. A column in a data frame or a vector indicating the classification by the reference test. The reference standard must have two levels: must be coded either as 0 - without target condition - or 1 - with the target condition; or could be coded as.factor with the words "negative" - without target condition - and "positive" - with the target condition.
test	The index test or test under evaluation. A column in a dataset or vector indicating the test results in a continuous scale. It may also work with discrete ordinal scale.
Cost	Cost = cost(FN)/cost(FP). MCT (misclassification cost term) will be used to estimate a good cut-off. It is a value in a range from 0 to infinite. Could be financial cost or a health outcome with the perception that FN are more undesirable than FP (or the other way around). This item will run into $MCT - (1 - prevalence) * (1 - Sp) + Cost * prevalence(1 - Se)$. Cost=1 means FN and FP have even cost. Cost = 0.9 means FP are 10 percent more costly. Cost = 0.769 means that FP are 30 percent more costly. Cost = 0.555 means that FP are 80 percent more costly. Cost = 0.3 means that FP are 3 times more costly. Cost = 0.2 means that FP are 5 times more costly. Also, it can be inserted as any ratio such as 1/2.5 or 1/4.
CL	Confidence limit. The limits of the confidence interval. Must be coded as number in range from 0 to 1. Default value is 0.95
Inconclusive	Inconclusive is a value that ranges from 0 to 1 that will identify the test range where the performance of the test is not acceptable and thus considered inconclusive. It represents the researcher tolerance of how good the test should be. If

	it is set to 0.95 (which is the default value), test results that have less than 0.95 sensitivity and specificity will be in the inconclusive range.
Prevalence	Prevalence of the disease in the population who the test will be performed. It must be a value from 0 to 1. If left 0 (the default value), this will be replaced by the disease prevalence in the sample. This value will be used in the MCT and Efficiency formulas to estimate good cut-offs.
t.max	Test upper range limit to be set as numeric. It will be used to simulate the parametric curve. If left NULL TGROC will assume that the sample maximum value is the upper limit of the test.
t.min	Test lower range limit to be set as numeric. It will be used to simulate the parametric curve. If left NULL TGROC will assume that the sample minimum value is the lower limit of the test.
precision	The test precision is the unit of variation of the test and should be set as numeric. It will be used to simulate the parametric curve. It will express how many estimations the network will do between each test unit. It is interesting the precision to be something between 1/2 to 1/10 of the test unit. The higher the precision, smoother the parametric curve will look. However, if too much precision is set the function may give an error as a result. This also may depends on the amount of observations in the dataset
n.neurons	Numeric vector containing the number of neurons of each layer. See newff .
learning.rate.global	Learning rate at which every neuron is trained. See newff .
momentum.global	Momentum for every neuron. See newff .
error.criterium	Criteria used to measure to proximity of the neural network prediction to its target. See newff .
Stao	Stao parameter for the TAO error criteria. See newff .
hidden.layer	Activation function of the hidden layer neurons. See newff .
output.layer	Activation function of the hidden layer neurons. See newff .
method	Preferred training method. See newff .
report	Logical value indicating whether the training function should keep quiet. See train .
show.step	Number of epochs to train non-stop until the training function is allow to report. See train .
n.shows	Number of times to report (if report is TRUE).See train .
Plot	Possible values are: "None", "Both", "Parametric" and "Non-parametric". TGROC may plot parametric, non-parametric, both or no plot at all depending of this option. Default is to plot both curves.
Plot.inc.range	Plot inconclusive range. If TRUE, the lines representing the limits of the inconclusive range will be displayed. Default is TRUE. Parametric inconclusive range will be displayed if Plot = "Parametric" or Plot = "Both" and non-parametric inconclusive range otherwise. If Plot is FALSE then Plot.inc.range is not considered.

Plot.Cl	Plot confidence limits. If TRUE, confidence bands for sensitivity and specificity curves will be displayed. If Plot = "Parametric" or Plot = "Both" than parametric bands are displayed and non-parametric otherwise. Default is FALSE. If Plot is FALSE than Plot.Cl is not considered.
Plot.cutoff	A line representing the estimated best cut-off (threshold) will be displayed. If Plot is FALSE then Plot.cutoff is not considered. If Plot = "Parametric" or Plot = "Both" than the parametric values are represented and non-parametric otherwise. Default is "None". Possible values are: "Se=Sp" - the cut-off which Sensitivity is equal to Specificity; "Max.Efficiency" - the cut-off which maximize the efficiency; "Min.MCT" - the cut-off which minimizes the misclassification cost term.
cex	See par . A numerical value giving the amount by which plotting text and symbols should be magnified relative to the default.
cex.sub	See par . Controls the font size in the subtitle. If Plot is FALSE than cex.sub is not considered.
Print	If FALSE, statistics estimated by TGROC will not be displayed in the output window. Default is TRUE.
x	For the plot function, x is an object created TGROC function.
...	Other plot parameters form plot.default

Details

There are two main advantages of TG-ROC over ROC analysis: (1) for the uninitiated is much easier to understand how sensitivity and specificity changes with different cut-offs; (2) and because of the graphical display is much easier to understand and estimate reasonable inconclusive test ranges. Occasionally the MCT or Efficiency cut-offs may be set outside the inconclusive range. This may happens with extreme values of Cost and population prevalence. If this is the case, perhaps the inconclusive range may not be of interest or not applicable. Also, if the test is too good or inconclusive range tolerance is set too low, then there may be no inconclusive range at all, because sensitivity and specificity may not be below this tolerance at the same time. If this is the case, setting a higher inconclusive tolerance may work. Tests results matching the cut-off values will be considered a positive test. TGROC assumes that subjects with higher values of the test are with the target condition and those with lower values are without the target condition. Tests that behave like glucose (middle values are supposed to be normal and extreme values are supposed to be abnormal) and immunofluorescence (lower values - higher dilutions - are suppose to be abnormal) will not be correctly analyzed. In the latter, multiplying the test results by -1 or other transformation before analysis could make it work. The validity measures such as Sensitivity, Specificity and Likelihood ratios and its confidence limits are estimated as in [diagnosis](#) function. MCT and Efficiency are estimated as in [ROC](#) function. Non-parametric confidence bands are estimated by binomial confidence interval and parametric with normal confidence interval. The parametric curve and validity measures are estimated with a neural network strategy using the AMORE package. Usually neural networks, uses a subset of the data to estimate weights, a subset to calibrate/validate the weights and a third subset to simulate the function. TGROC uses only the estimate and simulate steps, therefore there is no stopping rule for the neural network parametric estimation. The only way to check the fit of the neural network is to visually compare with the non-parametric curve. If the curve looks weird or not good enough, than progressive slight changes in the momentum, learning rate, number of layers and / or other parameters should work fine.

Value

Sample size	Amount of subjects analyzed.
Sample prevalence	Prevalence of target condition in the sample.
Population prevalence.	Informed prevalence in the population.
Test summary	A summary of central and dispersion tendencies of test results.
Non-parametric inconclusive limits.	Estimate of the inconclusive limits of the tests and its corresponding validity measures.
Non-parametric best cut-offs.	The cut-offs estimated by different methods and its corresponding validity measures.
Parametric inconclusive limits.	Estimate of the inconclusive limits of the tests and its corresponding validity measures with the parametric simulation.
Parametric best cut-off	The cut-offs estimated by different methods and its corresponding validity measures with the parametric simulation.

Note

Bug reports, malfunctioning, or suggestions for further improvements or contributions can be sent, preferentially, through the DiagnosisMed email list, or R-Forge website <https://r-forge.r-project.org/projects/diagnosismed/>.

Author(s)

Pedro Brasil; - <diagnosismed-list@lists.r-forge.r-project.org>

References

- Greiner, M. (1996) Two-graph receiver operating characteristic (TG-ROC): update version supports optimization of cut-off values that minimize overall misclassification costs. *J.Immunol.Methods* 191:93-94.
- M. Greiner (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.
- M. Greiner, D. Sohr, P. Gobel (1995) A modified ROC analysis for the selection of cut-off values and the definition of intermediate results of serodiagnostic tests. *Journal of immunological methods*. 185(1):123-132.

See Also

[interact.ROC,ROC,diagnosis,performance,binom.conf.int,contROC](#)

Examples

```
# Loading a dataset.
data(tutorial)
# Attaching dataset
attach(tutorial)
# Running the analysis
TGROC(gold=Gold, test=Test_B)
rm(tutorial)
```

tutorial

Example data set for continuous tests results

Description

This data set gives results of three serological tests - continuous scale - and its corresponding reference test classification. Also gives age as four ordinal groups.

Usage

```
tutorial
```

Format

A data.frame containing 5 variables and 170 observations.

Source

Not informed

References

Not informed

Index

- *Topic **datasets**
 - rocdata, [13](#)
 - tutorial, [19](#)
- *Topic **dynamic**
 - interact.ROC, [5](#)
- *Topic **htest**
 - diagnosis, [2](#)
 - ROC, [9](#)
 - TGROC, [14](#)
- *Topic **iplot**
 - interact.ROC, [5](#)
 - LRgraph, [7](#)
 - ROC, [9](#)
 - TGROC, [14](#)
- *Topic **univar**
 - diagnosis, [2](#)
 - ROC, [9](#)
 - TGROC, [14](#)
- as.factor, [10](#), [15](#)
- binom.conf.int, [4](#), [12](#), [18](#)
- contROC, [7](#), [12](#), [18](#)
- diagnosis, [2](#), [7](#), [8](#), [11](#), [12](#), [17](#), [18](#)
- epi.nomogram, [4](#)
- epi.tests, [4](#)
- interact.ROC, [5](#), [12](#), [18](#)
- legend, [7](#)
- LRgraph, [4](#), [7](#)
- newff, [16](#)
- nomogram, [4](#)
- numeric, [6](#)
- par, [7](#), [10](#), [11](#), [17](#)
- performance, [7](#), [12](#), [18](#)
- plot.default, [2](#), [7](#), [11](#), [17](#)
- plot.diag (diagnosis), [2](#)
- plot.ROC (ROC), [9](#)
- plot.TGROC (TGROC), [14](#)
- points, [7](#), [11](#)
- print, [2](#)
- print.diag (diagnosis), [2](#)
- print.ROC (ROC), [9](#)
- print.TGROC (TGROC), [14](#)
- ROC, [4](#), [7](#), [9](#), [17](#), [18](#)
- roc.demo, [7](#)
- rocdata, [13](#)
- sensSpec, [4](#)
- summary, [2](#)
- summary.diag (diagnosis), [2](#)
- TGROC, [7](#), [12](#), [14](#)
- train, [16](#)
- tutorial, [19](#)