Package: bdpv (via r-universe)

August 29, 2024

Type Package

Title Inference and Design for Predictive Values in Diagnostic Tests

Version 1.3

Date 2019-03-11

Author Frank Schaarschmidt [aut, cre]

Maintainer Frank Schaarschmidt <schaarschmidt@biostat.uni-hannover.de>

Imports graphics, stats

Description Computation of asymptotic confidence intervals for negative and positive predictive values in binary diagnostic tests in case-control studies. Experimental design for hypothesis tests on predictive values.

License GPL (>= 2)

LazyLoad yes

NeedsCompilation no

Repository CRAN

Date/Publication 2019-03-11 15:12:40 UTC

Contents

bdpv-package *Confidence intervals and experimental design for negative and positive predictive values in binary diagnostic tests.*

Description

Computing asymptotic confidence intervals for negative and positive predictive values of binary diagnostic test assuming a case-control design. Experimental design based on asymptotic formulas and Monte Carlo simulation for hypothesis tests on predictive values, including some plot functions to explore various experimental designs.

Details

Package:	bdpv
Type:	Package
Version:	1.3
Date:	2018-04-17
License:	GPL
LazyLoad:	yes

1) Computing confidence intervals: The function `BDtest` computes the asymptotic confidence intervals for negative and positive predictive value given in Mercaldo et al. (2007), assuming binomial sampling for obtaining estimates of sensitivity and specificity (leading to a 2x2 table with numbers of diseased and healthy fixed by design) and known prevalence. Alternatively, the functions `CIpvBI` and `CIpvBII` allow to simulate Bayesian intervals for negative and positive predictive values in case-control designs (Stamey and Holt, 2010), where prior knowledge concerning sensitivity, specificity may be included and external data and/or prior knowledge on prevalence may be included. By default, flat, non-informative priors are used, resulting in intervals with improved frequentist small sample performance (Stamey and Holt, 2010).

2) The function `nPV` uses the asymptotic formulas of Steinberg et al.(2009) to calculate the sample size necessary to reject tests witzh H0: PPV>=PPV0, H0NPV>=PNPV0, with a prespecified power in a case-control setting. Further necessary input arguments are sensitivity, specificity, prevalence, NPV0, PPV0, the range and number of steps of proportion of true positives in the trial. The results of this function can be plotted using `plotnPV`, `plotnPV2` and be somewhat edited by `as.data.frame.nPV`.

3) Because the results of these functions may be misleading in small sample or extreme proportion situations, the simulation functions `simPV` and `simPVmat` allow to check power and coverage probability for given parameter stettings.

The remaining functions are meant for internal use.

Author(s)

Frank Schaarschmidt, on behalf of the Institute of Biostatistics, LUH, Hannover, Germany Maintainer: Frank Schaarschmidt <schaarschmidt@biostat.uni-hannover.de>

References

Steinberg DM, Fine J, Chappell R (2009). Sample size for positive and negative predictive value in diagnostic research using case-control designs. Biostatistics 10,1, 94-105.

Mercaldo ND, Lau KF, Zhou XH (2007). Confidence intervals for predictive values with an emphasis to case-control studies. Statistics in Medicine 26:2170-2183.

Stamey JD and Holt MM (2010). Bayesian interval estimation for predictive values for case-control studies. Communications in Statistics - Simulation and Computation. 39:1, 101-110.

Examples

```
# 1) Example data: Mercaldo et al.(2007), Table VIII:

Tab8<-matrix(c(240, 178, 87, 288), ncol=2)
colnames(Tab8)<-c("Case","Control")
rownames(Tab8)<-c("ApoEe4plus","ApoEe4minus")
Tab8

# Assuming prevalence=0.03
BDtest(xmat=Tab8, pr=0.03, conf.level = 0.95)

# Assuming prevalence=0.5
BDtest(xmat=Tab8, pr=0.5, conf.level = 0.95)

# 2) Experimental design acc. to Steinberg et al.(2009)

TEST<-nPV(se=c(0.76, 0.78, 0.80, 0.82, 0.84),
 sp=c(0.93, 0.94, 0.95, 0.96, 0.97),
 pr=0.0625, NPV0=0.98, PPV0=0.25, NPVpower = 0.8, PPVpower = 0.8,
 rangeP = c(0.10, 0.9), nsteps = 20, alpha = 0.05)

TEST

plotnPV(TEST, log="y", legpar=list(x=0.6))


# 3) Simulation of power and coverage probability

simPVmat(se=0.8, sp=0.95, pr=0.0625, n1=c(177, 181),
 n0=c(554, 87), NPV0=0.98, PPV0=c(0.4, 0.25))
```

`as.data.frame.nPV` *Coerce results of "nPV" to a data.frame.*

Description

Coerce the possibly long sample size tables resulting from calling "nPV" to a data.frame.

Usage

```
## S3 method for class 'nPV'
as.data.frame(x, ...)
```

Arguments

x	an object of class "nPV"
...	further arguments to be passed to as.data.frame

Details

The lengthy lists in the output nPV, item nlist are coerced to a data.frame with columns propP, and the different NPV/PPV sample sizes for each of the parameters settings following.

BDtest *Computing confidence intervals for sensitivity, specificity and predictive values assuming a case-control study.*

Description

This function computes confidence intervals for negative and positive predictive values. Confidence intervals for sensitivity, specificity are computed for completeness. All methods assume that data are obtained by binomial sampling, with the number of true positives and true negatives in the study fixed by design. The methods to compute negative and positive predictive values (NPV, PPV) assume that prevalence is a known quantity, based on external knowledge.

Usage

```
BDtest(xmat, pr, conf.level = 0.95)
```

Arguments

xmat	A 2x2 table with 4 (integer) values, where the first column (xmat[,1]) represents the numbers of positive and negative results in the group of true positives, and the second column (xmat[,2]) contains the numbers of positive and negative results in the group of true negatives, i.e. the first row contains numbers of positive results and the second row the number of negative results.
pr	A single numeric value between 0 and 1, specifying the assumed prevalence.
conf.level	A single numeric value between 0 amd 1, specifying the nominal confidence level.

Details

The exact, conservative Clopper Pearson (1934) method is used to compute intervals for the sensitivty and specificity. The asymptotic standard logit intervals (Mercaldo et al. 2007) are used to compute intervals for the predictive values. In case that the table contains any 0, the adjusted logit intervals (Mercaldo et al. 2007) are returned instead to compute intervals for the predictive values.

Value

A list containing:

INDAT	a data.frame containing the input 2x2 table
SESPDAT	a data.frame with four columns containing estimates, lower limit and two.sided interval for the sensitivity and specificity (1. and 2. row)
PPVNPVDAT	a data.frame with four columns containing estimates, lower limit and two.sided interval for the NPV and PPV (1. and 2. row)

Author(s)

Frank Schaarschmidt

References

Mercaldo ND, Lau KF, Zhou XH (2007). Confidence intervals for predictive values with an emphasis to case-control studies. Statistics in Medicine 26:2170-2183.

See Also

CInpvppv for the internally used methods to compute the intervals for predictive values,

Examples

```
# Reproduce the standard logit interval results in
# Table IX, Mercaldo et al.(2007)

# 1) Example data: Mercaldo et al.(2007), Table VIII:

Tab8<-matrix(c(240, 178, 87, 288), ncol=2)
colnames(Tab8)<-c("Case","Control")
rownames(Tab8)<-c("ApoEe4plus","ApoEe4minus")
Tab8

# Assuming prevalence=0.03
BDtest(xmat=Tab8, pr=0.03, conf.level = 0.95)

# Assuming prevalence=0.5
BDtest(xmat=Tab8, pr=0.5, conf.level = 0.95)
```

CInpvppv *Asymptotic confidence intervals for negative and positive predictive values.*

Description

Computes asymptotic confidence intervals for negative and positive predictive values under the assumption of binomial sampling and known prevalence, according to Mercaldo et al. (2007). The standard logit intervals and and adjusted version are available, where the standard logit intervals are recommended.

Usage

```
CIlnpv(x0, x1, p, conf.level = 0.95,
alternative = c("two.sided", "less", "greater"))

CIlppv(x0, x1, p, conf.level = 0.95,
 alternative = c("two.sided", "less", "greater"))

CIlnpvak(x0, x1, p, conf.level = 0.95,
 alternative = c("two.sided", "less", "greater"))

CIlppvak(x0, x1, p, conf.level = 0.95,
 alternative = c("two.sided", "less", "greater"))

CombCInpv(x0, x1, p, conf.level = 0.95,
 alternative = c("two.sided", "less", "greater"))

CombCIppv(x0, x1, p, conf.level = 0.95,
 alternative = c("two.sided", "less", "greater"))
```

Arguments

x0	A vector of two (integer) values, specifying the observed number of positive (x0[1]) and negative (x0[2]) outcomes in the group of true negatives.
x1	A vector of two (integer) values, specifying the observed number of positive (x1[1]) and negative (x1[2]) outcomes in the group of true positives.
p	The assumed prevalence, a single numeric value between 0 and 1.
conf.level	The confidence level, a single numeric value between 0 and 1, defaults to 0.95
alternative	A character string specifying whether two-sided ("two.sided"), only lower bounds ("greater") or only upper bounds ("less") shall be calculated.

Details

CIlnpv and CIlppv implement the standard logit intervals for NPV and PPV, Section 2.2, Eq.(8)-Eq.(11) in Mercaldo et al. (2007). CIlnpvak and CIlppval implement the logit intervals for NPV and PPV with adjusted estimates according to Table II in Mercaldo et al. (2007). The standard logit intervals have better properties, but are not defined in a number of extreme outcomes. The adjusted logit methods do always produce intervals, but have worse frequentist properties (Mercaldo et al. 2007). The functions CombCInpv, CombCIppv combine both methods by computing the stdnard logit method when possible and computing the adjusted methods in those cases where the standard method is not defined. These functions are meant to facilitate simulation, e.g. in simPV, simPVmat.

Value

A list with elements

`conf.int`	the confidence bounds
`estimate`	the point estimate

Note

These functions are meant for internal use. There is not much checking for the validity of input.

Author(s)

Frank Schaarschmidt

References

Mercaldo ND, Lau KF, Zhou XH (2007). Confidence intervals for predictive values with an emphasis to case-control studies. Statistics in Medicine 26: 2170-2183.

See Also

`BDtest` as a user level function

Examples

```
CIlnpv(x0=c(87,288), x1=c(240,178), p=0.03,
 conf.level = 0.95, alternative = "two.sided")

CIlppv(x0=c(87,288), x1=c(240,178), p=0.03,
 conf.level = 0.95, alternative = "two.sided")

CIlnpvak(x0=c(87,288), x1=c(240,178), p=0.03,
 conf.level = 0.95, alternative = "two.sided")

CIlppvak(x0=c(87,288), x1=c(240,178), p=0.03,
 conf.level = 0.95, alternative = "two.sided")
```

CIpvBayes *Confidence intervals for negative and positive predictive values in a case-control setting by simulation from the posterior distribution.*

Description

Computes confidence intervals for negative and positive predictive values by simulation from the posterior beta-distribution (Stamey and Holt, 2010), assuming a case-control design to estimate sensitivity and specificity, while prevalence estimates of an external study and/or prior knowledge concerning prevalence may be introduced additionally.

Usage

```
CIpvBI(x1, x0, pr, conf.level = 0.95,
 alternative = c("two.sided", "less", "greater"),
 B=5000, shapes1=c(1,1), shapes0=c(1,1), ...)

CIpvBII(x1, x0, xpr, conf.level = 0.95,
 alternative = c("two.sided", "less", "greater"),
 B=5000, shapes1=c(1,1), shapes0=c(1,1), shapespr=c(1,1), ...)
```

Arguments

x1	A vector of two (integer) values, specifying the observed number of positive (x1[1]) and negative (x1[2]) test results in the group of true positives.
x0	A vector of two (integer) values, specifying the observed number of positive (x0[1]) and negative (x0[2]) test results in the group of true negatives.
pr	A single numeric value between 0 and 1, defining an assumed fixed (known) prevalence (for CIpvBI), where prevalence is the proportion of positives in the population.
xpr	An optional vector of two (integer) values, specifying the observed number of positive (xpr[1]) and negative (xpr[2]) outcomes from an external study that allows to estimate the prevalence of positives in the population of interest.
conf.level	The confidence level, a single numeric value between 0 and 1, defaults to 0.95
alternative	A character string specifying whether two-sided ("two.sided"), only lower bounds ("greater") or only upper bounds ("less") shall be calculated.
B	A single integer, the number of samples from the posterior to be drawn.
shapes1	Two positive numbers, the shape parameters (a,b) of the beta prior for the sensitivity, by default a flat beta prior (a=1, b=1) is used.
shapes0	Two positive numbers, the shape parameters (a,b) of the beta prior for (1-specificity), by default a flat beta prior (a=1, b=1) is used. Note, that this definition differs from that in Stamey and Holt(2010), where the prior is defined for the specificity directly.
shapespr	Two positive numbers, the shape parameters (a,b) of the beta prior for the prevalence, by default a flat beta prior (a=1, b=1) is used. For CIpvBII only.
...	Arguments to be passed to quantile(), other arguments are ignored without warning. .

Details

CIpvBI implements the method refered to as Bayes I in Stamey and Holt (2010), CIpvBI implements the method refered to as Bayes II in Stamey and Holt (2010), Equation (2) and following description (p. 103-104).

Value

A list with elements

conf.int	the confidence bounds
estimate	the point estimate
tab	a 2x2 matrix showing how the input data in terms of true positives and true negatives

Author(s)

Frank Schaarschmidt

References

Stamey JD and Holt MM (2010). Bayesian interval estimation for predictive values for case-control studies. Communications in Statistics - Simulation and Computation. 39:1, 101-110.

Examples

```
# example data: Stamey and Holt, Table 8 (page 108)
# Diseased
# Test D=1 D=0
# T=1 240 87
# T=0 178 288
#n1,n0: 418 375


# reproduce the results for the Bayes I method
# in Stamey and Holt (2010), Table 9, page 108

# assuming known prevalence 0.03
# ppv 0.0591, 0.0860
# npv 0.9810, 0.9850
CIpvBI( x1=c(240,178), x0=c(87,288), pr=0.03)

# assuming known prevalence 0.04
# ppv 0.0779, 0.1111
# npv 0.9745, 0.9800
CIpvBI( x1=c(240,178), x0=c(87,288), pr=0.04)

# compare with standard logit intervals
tab <- cbind( x1=c(240,178), x0=c(87,288))
tab
BDtest(tab, pr=0.03)
BDtest(tab, pr=0.04)


# reproduce the results for the Bayes II method
# in Stamey and Holt (2010), Table 9, page 108

CIpvBII( x1=c(240,178), x0=c(87,288),  shapespr=c(16,486))

CIpvBII( x1=c(240,178), x0=c(87,288), shapespr=c(21,481))
```

nNPVPPV *Asymptotic experimental design for inference on negative and positive predictive values in case-control studies.*

Description

For internal use. Functions to compute sample size (to reach a pre-specified power) and optimal allocation of true positives and true negatives in case-control designs for binary diagnostic tests (Mercaldo et al. 2007).

Usage

```
nNPV(propP, se, sp, prev, NPV0,
 conf.level = 0.95, power = 0.8)

nPPV(propP, se, sp, prev, PPV0,
 conf.level = 0.95, power = 0.8)

AOppvnpv(se, sp)
```

Arguments

se	a numeric value, specifying the expected sensitivity
sp	a numeric value, specifying the expected specificity
propP	a vector of numeric values of proportions of truely positives in the trial (n1/(n1+n0))
prev	a numeric value, the prevalence
NPV0	a numeric value, the negative predictive value to be rejected under H0: NPV>=NPV0
PPV0	a numeric value, the positive predictive value to be rejected under H0: PPV>=PPV0
conf.level	a single numeric values, the nominal confidence level (1-alpha)
power	a single numeric value, the power that is to be obtained

Details

The functions implement the methods described in section 3.2 of Steinberg et al.(2009), nPPV gives the solution to Eq.(3.6) and NA if necesarry conditions mentioned before are not fulfilled, nNPV gives the solution to Eq.(3.8) and NA if necesarry conditions mentioned before are not fulfilled, AOppvnpv gives the optimal proportion of true poistives as are solutions to Eq.(3.4) and Eq. (3.6) for PPV and NPV, respectively.

Value

For nNPV and nPPV: a list with first element

n	the (vector of) sample size (s), or NA if necessary conditions are not met

and further elements giving the input arguments

Author(s)

Frank Schaarschmidt

References

Steinberg DM, Fine J, Chappell R (2009). Sample size for positive and negative predictive value in diagnostic research using case-control designs. Biostatistics 10,1, 94-105.

See Also

For a combination of PPV and NPV experimental design see `nPV` and `plotnPV`; to validate small sample results of these asymptotic formulas, see `simPVmat`

Examples

```
nPPV(propP=c(0.2,0.4,0.6,0.8), se=0.9, sp=0.9,
prev=0.1, PPV0=0.4, conf.level=0.95, power=0.8)

nNPV(propP=c(0.2,0.4,0.6,0.8), se=0.9, sp=0.9,
prev=0.1, NPV0=0.95, conf.level=0.95, power=0.8)

AOppvnpv(se=0.9, sp=0.9)
```

`nPV` *Asymptotic sample size calculation for inference on negative and positive predictive values in case-control designs.*

Description

Functions to compute sample size (to reach a pre-specified power) and optimal allocation of true positives and true negatives in case-control designs (Steinberg et al., 2008) for binary diagnostic tests (Mercaldo et al. 2007).

Usage

```
nPV(se, sp, prev, NPV0, PPV0,
 NPVpower = 0.8, PPVpower = 0.8,
 rangeP = c(0.05, 0.95), nsteps = 20,
 alpha = 0.05, setnames = NULL)
```

Arguments

`se`	a (vector of) numeric value(s), specifying the expected sensitivity
`sp`	a (vector of) numeric value(s), specifying the expected specificity
`prev`	a (vector of) numeric value(s), specifying the prevalence
`NPV0`	a (vector of) numeric value(s), specifying the negative predictive value to be rejected under H0: NPV>=NPV0

PPV0	a (vector of) numeric value(s), specifying the positive predictive value to be rejected under H0: PPV>=PPV0
NPVpower	a (vector of) numeric value(s), the power that is to be obtained for the test H0: NPV>=NPV0
PPVpower	a (vector of) numeric value(s), the power that is to be obtained for the test H0: PPV>=PV0
rangeP	a vector of two numeric values, giving the range of the proportion of truely positives to be considered in experimental design
nsteps	a single (integer) value, the number of steps in rangeP to be considered
alpha	a single numeric value, the type I error of the test (1-confidence level)
setnames	an optional vector of names for the parameter sets

Details

The function uses nNPVPPV and implement the methods described in section 3.2 of Steinberg et al.(2009). The results for NPV are the smallest integers fulfilling Eq.(3.6) and NA if necesarry conditions mentioned before are not met, the results for PPV are the smallest integers fulfilling Eq.(3.8) and NA if necesarry conditions mentioned before are not met.

The arguments se, sp, prev, NPV0, PPV0, NPVpower, PPVpower can be given as vectors or single values, where shorter values are recycled to the length of the longest. The proportion of true positives is varied over nstep equidistant values over the range specified in argument rangeP. On each resulting parameter set, the asymptotic sample size formulas of Steinberg et al.(2009) are applied.

The result of those calculations may be plot using plotnPV and plotnPV2.

Warnings are returned by the internal function nNPV and nPPV if the validity of asymptotic formulas under binomial sampling may be doubtable, namely when the asymptotic formulas return a total sample size n for given propP, se, sp, such that min(n*propP*se, n*propP*(1-se))<5 or min(n*(1-propP)*sp, n*(1-propP)*(1-sp))<5. That is, a warning is returned if the proposed design of the case-control study (n1, n0) = (n*propP, n*(1-propP)) leads to expected counts < 5 for any cell of the 2x2 table.

Value

A list with elements

outDAT	a data.frame showing the parameter settings (in rows) and the input parameters se, sp, prev, NPV0, PPV0, NPVpower, PPVpower, trueNPV, truePPV
nlist	a list with an element for each parameter setting in OUTDAT, listing the results of nNPV, and nPPV
NSETS	a single (integer), the number of parameter sets
nsteps	a single (integer), the number of steps in the range of proportions of true positives
rangeP	the input range of the proportion of true positives
propP	the resulting sequence of proportions of true positives considered

Author(s)

Frank Schaarschmidt

References

Steinberg DM, Fine J, Chappell R (2009). Sample size for positive and negative predictive value in diagnostic research using case-control designs. Biostatistics 10,1, 94-105.

See Also

plotnPV for showing the results in one graphic, and plotnPV for showing the results in a set of subgraphics,

Examples

```
#Reproducing illustration in Section 3.4 and 4.2 of
#Steinberg et al. (2009)

FIG1<-nPV(se=0.8, sp=0.95, prev=1/16, NPV0=0.98, PPV0=0.4,
 NPVpower = 0.8, PPVpower = 0.8,
 rangeP = c(0.01, 0.99), nsteps = 100, alpha = 0.05)

FIG1

DFIG1<-as.data.frame(FIG1)

plot(x=DFIG1$propP, y=DFIG1[,2], ylim=c(0,2000), lty=1,  type="l",
 ylab="total sample size", xlab="proportion of true positives")
lines(x=DFIG1$propP, y=DFIG1[,3], lty=2 )
```

---

plotnPV *Plot experimental design for different setting in a single figure.*

Description

The function creates a plot from the results of the function nPV.

Usage

```
plotnPV(x, NPVpar = NULL, PPVpar = NULL, legpar = NULL, ...)
```

Arguments

x	an object of class "nPV" as can be obtained by calling function nPV
NPVpar	a named list which specifies plot parameters for the negative predictive values, possible are lty, lwd, col, pch

PPVpar	a named list which specifies plot parameters for the positive predictive values, possible are lty, lwd, col, pch
legpar	a named list to pass arguments to the legend. See ?legend for the possible arguments.
...	further arguments to be passed to plot

Details

Required sample sizes for different experimental settings and prevalences, needed to achieve a prespecified power can be calculated in dependence of the proportion of true negative and true positive compounds in the validation set, using function nPV. This function draws a plot with the proportion of positive on x and the total sample size on y, combining all parameter settings in one plot. Parameter settings my be distinguished bylty, lwd, col, pch in NPVpar and PPVpar. By default a legend is drawn which can be further modified in legpar.

Value

A plot.

Author(s)

Frank Schaarschmidt

References

Steinberg DM, Fine J, Chappell R (2009). Sample size for positive and negative predictive value in diagnostic research using case-control designs. Biostatistics 10, 1, 94-105.

See Also

plotnPV2 for a plot with separate subplots for each parameter setting

Examples

```
TEST<-nPV(se=c(0.9, 0.92, 0.94, 0.96, 0.98), sp=c(0.98, 0.96, 0.94, 0.92, 0.90),
 pr=0.12, NPV0=0.98, PPV0=0.4, NPVpower = 0.8, PPVpower = 0.8,
 rangeP = c(0.05, 0.95), nsteps = 100, alpha = 0.05)

plotnPV(TEST)

# plot parameters maybe introduced via ...
# the legend maybe modified via legpar:

plotnPV(TEST, log="y", legpar=list(x=0.6))

# own colour definitions
plotnPV(TEST, NPVpar=list(col=1:6, lwd=2, lty=1),
 PPVpar=list(col=1:6, lwd=2, lty=3))
```

`plotnPV2` *Plot experimental design for different settings in a set of sub figure.*

Description

The function creates a plot from the results of the function `nPV`.

Usage

```
plotnPV2(x, NPVlty = 1, PPVlty = 3, ...)
```

Arguments

`x`	an object of class `"nPV"` as can be obtained by calling function `nPV`
`NPVlty`	single integer value, the linetype for NPV sample size, see `par` for the options
`PPVlty`	single integer value, the linetype for PPV sample size, see `par` for the options
`...`	further arguments to be passed to `plot`

Details

Required sample sizes for different experimental settings and prevalences, needed to achieve a prespecified power can be calculated in dependence of the proportion of true negative and true positive compounds in the validation set, using function `nPV`. This function draws a plot with the proportion of true positives on x and the total sample size on y, combining all parameter settings in one plot.

Note that for huge numbers of setting this should not work.

Value

A plot.

Author(s)

Frank Schaarschmidt

References

Steinberg DM, Fine J, Chappell R (2009). Sample size for positive and negaitive predictiove value in diagnostic research using case-control designs. Biostatoistics 10, 1, 94-105.

See Also

`plotnPV`, for sample sizes for several settings in one figure

Examples

```
TEST<-nPV(se=c(0.9, 0.92, 0.94, 0.96, 0.98), sp=c(0.98, 0.96, 0.94, 0.92, 0.90),
 pr=0.12, NPV0=0.98, PPV0=0.4, NPVpower = 0.8, PPVpower = 0.8,
 rangeP = c(0.05, 0.95), nsteps = 20, alpha = 0.05)

plotnPV2(TEST, log="x")
```

---

print.BDtest *Detailed print out for BDtest*

Description

Print details of the results of the function BDtest on the screen

Usage

```
## S3 method for class 'BDtest'
print(x, ...)
```

Arguments

x	an object of class "BDtest"
...	further arguments to be passed to print internally

---

print.nPV *Detailed print out for nPV*

Description

Print details of the results of the experimental design function nPV on the screen

Usage

```
## S3 method for class 'nPV'
print(x, ...)
```

Arguments

x	an object of class "nPV"
...	further arguments to be passed to print

See Also

plotnPV, plotnPV2 to plot the results of nPV

simPV *Simulate performance of confidence intervals for predictive values in a case-control design*

Description

The function draws data under the binomial assumption and computes the asymptotic confidence bounds (lower bounds only!) for the positive and negative predictive values. Output are the power (probability to exclude NPV0/PPV0), the realized coverage probability, 0.1,0.2, and 0.5-quantiles of the realized distribution of confidence bounds.

Usage

```
simPV(se, sp, pr, n1, n0, NPV0, PPV0,
 conf.level = 0.95, NSIM = 500)
```

Arguments

se	a numeric value, specifying sensitivity
sp	a numeric value, specifying specitivity
pr	a numeric value, specifying prevalence
n1	an (integer) value, specifying the number of truely positive compounds in the trial
n0	an (integer) value, specifying the number of truely negative compounds in the trial
NPV0	a numeric value, specifying the hypothesized negative predictive value (NPV assumed under H0)
PPV0	a numeric value, specifying the hypothesized positive predictive value (PPV assumed under H0)
conf.level	a numeric value, the confidence level
NSIM	an (integer) value, the number of simulations to be run

Details

The function draws data under the binomial assumption in a case-control design (Mercaldo et al. 2007), where the binomial doistributions are defined by n1, n0, se, sp. Then, for each drawn data set, the asymptotic lower confidence bounds (with confidence level=1-alpha, i.e. as suitable for a one-sided test at level alpha) for the positive and negative predictive values ar computed. (Note, that the standard logit interval is replaced by the adjusted logit interval of Mercaldo et al. 2007, if the standard logit interval is not defined.) Output are the etsimated power (observed probability that NPV0/PPV0 are excluded by the lower confidence bound), the realized coverage probability (observed probability that the true NPV/PPV are included in their interval), as well as the 0.1, 0.2, and 0.5-quantiles of the realized distribution of confidence bounds.

Value

A (2x7) matrix with results for NPV and PPV in rows 1,2 respectively, and the columns giving estimates of the power to reject H0: NPV>=NPV0 / PPV>=PPV0 (pow), coverage probability (cov), the values which are excluded with 10, 20 and 50 percent probability (q10, q20, q50), as well as the true predictive values and the marginn of H0 used to calculated power.

Author(s)

Frank Schaarschmidt

See Also

simPVmat for the same function, allowing vector input for se, sp, pr, n1, n0, NPV0 and PPV0.

Examples

```
simPV(se=0.8, sp=0.95, pr=1/16, n1=177, n0=554, NPV0=0.98, PPV0=0.4)

simPV(se=0.8, sp=0.95, pr=1/16, n1=181, n0=87, NPV0=0.98, PPV0=0.25)
```

---

simPVmat *Simulate performance of confidence intervals for predictive values in case-control design*

Description

Simulate the power (probability to exclude NPV0/PPV0), the coverage probability, and 0.1, 0.2, and 0.5-quantiles of the distribution of (lower!) asymptotic confidence bounds for predictive values. Different experimental setups may be compared. The function draws data under the binomial assumption and computes the asymptotic confidence bounds (lower bounds only!) for the positive and negative predictive values.

Usage

```
simPVmat(se, sp, pr, n1, n0, NPV0, PPV0,
 conf.level = 0.95, NSIM = 500, setnames = NULL)
```

Arguments

se	a (vector of) numeric value(s), specifying sensitivity
sp	a (vector of) numeric value(s), specifying specitivity
pr	a (vector of) numeric value(s), specifying prevalence
n1	a (vector of integer) value(s), specifying the number of truely positive compounds in the trial
n0	a (vector of integer) value(s), specifying the number of truely negative compounds in the trial

NPV0	a (vector of) numeric value(s), specifying the hypothesized negative predictive value (NPV assumed under H0)
PPV0	a (vector of) numeric value(s), specifying the hypothesized positive predictive value (PPV assumed under H0)
conf.level	a single numeric value, the confidence level
NSIM	a single (integer) value, the number of simulations to be run
setnames	optional character vector to the parameter sets in the output

Details

The vector or single values in se, sp, pr, n1, n0, NPV0, PPV0 are put together (shorter vectors recycled to the length of longest vectors). Then each of the resulting parameter settings is simulated as described in simPV

Value

A list with elements

INDAT	a dataframe with rows showing the sets of parameters build from the input values and columns: se, sp, pr, NPV0, PPV0, n1, n0, n (total sample size)
NPV	a matrix with simulation results for the negative predictive value
PPV	a matrix with simulation results for the positive predictive value
NSIM	number of suimulations
conf.level	nominal confidence level

Author(s)

Frank Schaarschmidt

See Also

This function is meantb to check small sample results obtained by the asymptotci formulas for experimental design from nPV, nNPV, nPPV

Examples

```
simPVmat(se=0.8, sp=0.95, pr=1/16,
 n1=c(177, 181), n0=c(554, 87), NPV0=0.98, PPV0=c(0.4, 0.25))


simPVmat(se=0.8, sp=0.95, pr=c(0.05,0.0625, 0.075, 0.1),
 n1=177, n0=554, NPV0=0.98, PPV0=0.4)
```

Index