Package ‘pCalibrate’

March 19, 2020

Type Package

Title Bayesian Calibrations of p-Values

Version 0.2-1

Date 2020-03-19

Maintainer Manuela Ott <manuela.c.ott@gmail.com>

Description Implements transformations of p-values to the smallest possible Bayes factor within the specified class of alternative hypotheses, as described in Held & Ott (2018, <doi:10.1146/annurev-statistics-031017-100307>). Covers several common testing scenarios such as z-tests, t-tests, likelihood ratio tests and the F-test.

Depends exact2x2, MCMCpack

NeedsCompilation no

Author Manuela Ott [aut, cre] (<https://orcid.org/0000-0002-8050-7279>),
Leonhard Held [aut] (<https://orcid.org/0000-0002-8686-5325>)

Repository CRAN

Date/Publication 2020-03-19 12:10:08 UTC

R topics documented:

`pCalibrate-package` *Bayesian Calibrations of p-Values*

Description

Implements transformations of one- and two-sided p-values to minimum Bayes factors. The minimum Bayes factor is the smallest possible Bayes factor for the point null hypothesis against the alternative within the specified class of alternatives.

The function `pCalibrate()` provides minimum Bayes factors for two-sided p-values which consider the p-value as the data and are directly based on the distribution of the p-value under the null hypothesis and the alternative. For one- and two-sided p-values from z-tests, `zCalibrate()` implements minimum Bayes factors for different classes of alternatives. The function `tCalibrate()` provides the same functionality for one- and two-sided p-values from t-tests. The functions `FCalibrate()` and `LRCalibrate()` transform two-sided p-values from the F-test or likelihood ratio test, respectively, to minimum Bayes factors.

Details

Package: pCalibrate

Type: Package

Title: Bayesian Calibrations of p-Values

Version: 0.2-1

Date: 2020-03-19

Author: Manuela Ott [aut, cre], Leonhard Held [aut]

Maintainer: Manuela Ott <manuela.c.ott@gmail.com>

Depends: exact2x2, MCMCpack

License: GPL (>=2)

Author(s)

Manuela Ott, Leonhard Held

Maintainer: Manuela Ott <manuela.c.ott@gmail.com>

References

Held, L. and Ott, M. (2018). On p-values and Bayes factors. *Annual Review of Statistics and Its Application*, **5**, 393–419.

Examples

```
pCalibrate(p=c(0.05, 0.01, 0.001), type="exploratory")
zCalibrate(p=c(0.05, 0.01, 0.005), type="one.sided",
           alternative="simple")
zCalibrate(p=c(0.05, 0.01, 0.005),  type="two.sided",
```

```
          alternative="normal")
tCalibrate(p=c(0.05, 0.01, 0.005), n=c(10, 20, 50), type="two.sided",
          alternative="normal")
FCalibrate(p=c(0.05, 0.01, 0.005), n=20, d=c(2, 5, 10),
          alternative="chi.squared")
LRCalibrate(p=c(0.05, 0.01, 0.005), df=2, alternative="simple")
```

BF2pp *Transform Bayes factors to posterior probabilities*

Description

Transforms the Bayes factor for a hypothesis H_1 against a hypothesis H_2 to the posterior probability of H_1 given the prior probability of H_1 (assuming that the prior probabilities of H_1 and H_2 add up to 1).

Usage

```
BF2pp(BF, prior.prob=0.5)
```

Arguments

BF	a vector of Bayes factors
prior.prob	a vector of prior probabilities for H_1. Defaults to a vector with entries 0.5.

Value

A numeric vector of posterior probabilities of the hypothesis H_1

Examples

```
BF2pp(BF=c(2, 10, 50), prior.prob=c(0.2, 0.5, 0.9))
```

FCalibrate *Calibration of two-sided p-values from the F-test in the linear model*

Description

Transforms two-sided p-values from the F-test of overall significance in the linear model to sample-size adjusted lower bounds on the Bayes factor for the point null hypothesis against the alternative.

Usage

```
FCalibrate(p, n, d, alternative="chi.squared", intercept=TRUE,
           transform="id")
```

Arguments

`p`	a vector of two-sided p-values
`n`	a scalar or a vector of positive integers. Specifies the sample size(s). May be a vector only if `d` is a scalar.
`d`	a scalar or a vector of positive integers. Specifies the dimension(s) of the vector(s) of regression coefficients, i.e. the number(s) of explanatory variables in the linear model(s). May be a vector only if `n` is a scalar.
`alternative`	either `"simple"` or `"chi.squared"`. Defaults to `"chi.squared"`. Specifies the alternative hypotheses on the non-centrality parameter of the F-distribution to consider. `"simple"` only considers simple point alternative hypotheses. `"chi.squared"` assumes a scaled chi-squared distribution.
`intercept`	logical. If TRUE, the linear model contains an unknown intercept term, otherwise the intercept is fixed. Defaults to TRUE.
`transform`	either `"id"`, `"log"`, `"log2"` or `"log10"`. Defaults to `"id"`. Specifies how to transform the lower bound on the Bayes factor. `"id"` corresponds to no transformation. `"log"` refers to the natural logarithm, `"log2"` to the logarithm to the base 2 and `"log10"` to the logarithm to the base 10.

Details

Note that under the point null hypothesis that all regression coefficients are equal to zero, the F-statistic F (which is the $(1-p)$-quantile of the F-distribution with d and $n-d-1$ degrees of freedom) has a central F-distribution with d and $n-d-1$ degrees of freedom if the linear model contains an unknown intercept term (otherwise F has a central F-distribution with d and $n-d$ degrees of freedom). Under a simple point alternative, F has a non-central F-distribution with d and $n-d-1$ degrees of freedom.

To obtain the lower bound on the Bayes factor for `alternative="simple"`, the likelihood under the alternative is then maximized numerically with respect to the non-centrality parameter. That calibration is described in Held & Ott (2018), Section 3.1 (in the last two paragraphs).

For `alternative="chi-squared"`, the calibration is proposed in Held & Ott (2016), Section 3 and also described in Held & Ott (2018), Section 3.2. The corresponding minimum Bayes factor has already been derived in Johnson (2005). As described there, assigning a scaled chi-sqaured distribution to the non-centrality parameter of the F-distribution corresponds to assigning a (multivariate) normal prior distribution centered around the null value to the vector of regression coefficients.

Value

A matrix containing the lower bounds on the Bayes factors as entries, for all combinations of p-value and sample size `n` or dimension `d` (whichever is multidimensional). The values for the k-th sample size or dimension (k-th entry in the vector `n` or `d`) and the different p-values are given in the k-th row.

Note

Computation may fail for `alternative="simple"` if the p-value `p` is extremely small and min{n, n-d} is also small. Warnings will be given in this case and the returned value is `minBF=NaN`.

References

Held , L. and Ott, M. (2016). How the maximal evidence of P -values against point null hypotheses depends on sample size. *American Statistician*, **70**, 335–341

Held, L. and Ott, M. (2018). On p-values and Bayes factors. *Annual Review of Statistics and Its Application*, **5**, 393–419.

Johnson, V. E. (2005). Bayes factors based on test statistics. *Journal of the Royal Statistical Society, Series B* **67**:689–701.

See Also

tCalibrate

Examples

```
FCalibrate(p=c(0.05, 0.01, 0.005), n=20, d=2, alternative="simple")
# chi-squared alternatives
FCalibrate(p=c(0.05, 0.01, 0.005), n=20, d=2, intercept=FALSE)
FCalibrate(p=c(0.05, 0.01, 0.005), n=20, d=c(2, 5, 10))
FCalibrate(p=c(0.05, 0.01, 0.005), n=c(10, 20, 50), d=2)

# plot for chi-squared alternatives: d=2 and different sample sizes n
# note that the minimum Bayes factor decreases with decreasing sample
# size
p <- exp(seq(log(0.0001), log(0.3), by=0.01))
n <- c(5, 10, 20)
minBF <- FCalibrate(p, n, d=2)
# compare to the bound for large n
minTBF <- LRCalibrate(p, df=2)
par(las=1)
matplot(p, t(minBF), ylim=c(0.0003, 1), type="l",
        xlab="two-sided F-test p-value", ylab="Minimum Bayes factor",
        log="xy", lty=1, lwd=2, axes=FALSE,
        main="Local normal alternatives")
lines(p, minTBF, col="gray", lty=2, lwd=2)
axis(1, at=c(0.0001, 0.0003, 0.001, 0.003, 0.01, 0.03, 0.1, 0.3),
     as.character(c(format(c(0.0001,0.0003), nsmall=4, digits=4,
                           scientific=FALSE),
                    c(0.001, 0.003, 0.01, 0.03, 0.1, 0.3))))
my.values <- c(3000, 1000, 300, 100, 30, 10, 3, 1)
my.at <- 1/my.values
my.ylegend <- c(paste("1/", my.values[-length(my.values)], sep=""),
                "1")
axis(2, at=my.at, my.ylegend)
box()
legend("bottomright",
       legend=rev(c("n=5", "n=10", "n=20", "n large")),
       col=rev(c(1:3, "gray")), lty=c(2, rep(1, times=3)), lwd=2)
```

`formatBF` *Formatting of Bayes factors*

Description

Bayes factors <1 are represented as ratios 1/x, where x is rounded to the specified number of digits. Bayes factors >=1 are rounded to the specified number of digits.

Usage

```
formatBF(BF, digits="default")
```

Arguments

`BF`	a numeric vector of Bayes factors
`digits`	either "default" (see Details) or a positive integer specifiying the number of decimal places to round the Bayes factor (for Bayes factors >= 1) or its inverse (for Bayes factors < 1) to

Details

The default formatting, which is recommended in Held and Ott (2018), is as follows : For very small Bayes factors BF < 1/1000, "< 1/1000" is returned. Bayes factors BF with 1/1000 <= BF <=1/10 are formatted as 1/x, where x is an integer and Bayes factors BF with 1/10 < BF < 1 as 1/x, where x is rounded to one decimal place. Accordingly, Bayes factors 1 <= BF < 10 are rounded to one decimal place, Bayes factors 10 <= BF <= 1000 are rounded to the next integer and for larger Bayes factors, "> 1000" is returned.

If `digits` is specified, the Bayes factor (if it is >= 1) or its inverse (if the Bayes factor is <1) is rounded to the number of decimal places specified and returned as a ratio if the Bayes factor is <1.

Value

A character vector of ratios (for inputs <1) or rounded numeric values (for inputs >=1).

References

Held, L. and Ott, M. (2018). On p-values and Bayes factors. *Annual Review of Statistics and Its Application*, **5**, 393–419.

Examples

```
BF <- c(0.0001, 0.0049, 0.0258, 0.24, 2.798)
# use the default formatting
formatBF(BF)
# specify the number of digits
formatBF(c(0.087, 4.65), digits=1)
```

`LRCalibrate` *Calibration of two-sided p-values obtained from the likelihood ratio test statistic*

Description

Transforms two-sided p-values from likelihood ratio (deviance) tests to lower bounds on the Bayes factor for the point null hypothesis against the alternative.

Usage

```
LRCalibrate(p, df, alternative="gamma", transform="id")
```

Arguments

`p`	a vector of two-sided p-values
`df`	a vector of degrees of freedom of the asymptotic chi-squared distribution(s) of likelihood ratio test statistic(s)
`alternative`	either `"simple"` or `"gamma"`. Defaults to `"gamma"`. Specifies the alternative hypotheses on the non-centrality parameter of the chi-squared distribution to consider. `"simple"` only considers simple point alternative hypotheses. `"gamma"` assumes a specific gamma distribution.
`transform`	either `"id"`, `"log"`, `"log2"` or `"log10"`. Defaults to `"id"`. Specifies how to transform the lower bound on the Bayes factor. `"id"` corresponds to no transformation. `"log"` refers to the natural logarithm, `"log2"` to the logarithm to the base 2 and `"log10"` to the logarithm to the base 10.

Details

Under the assumption that the parameter vector of interest (which has dimension `df`) is equal to the vector of zeros, the distribution of the deviance converges to a chi-squared distribution with `df` degrees of freedom. Under a simple point alternative for the parameter vector of interest and some regularity conditions, the distribution of the deviance converges to a non-central chi-squared distribution with `df` degrees of freedom.

For `alternative = "simple"`, the lower bound on the Bayes factor is obtained by mazimizing the (asymptotic) chi-squared distribution under the alternative with respect to the non-centrality parameter. That calibration is described in Held and Ott (2018), Section 4.2.1.

The calibration for `alternative = "normal"` uses the test-based Bayes factors introduced in Johnson (2008). That approach is also outlined in Held and Ott (2018), Section 4.2.2.

Using `alternative = "gamma"` yields a larger bound than `alternative = "simple"`.
Typical applications of these calibrations include generalized linear models.

Value

A matrix containing the lower bounds on the Bayes factors as entries, for all combinations of p-value and degrees of freedom. The values for the k-th degrees of freedom (k-th entry in the vector `df`) and the different p-values are given in the k-th row.

References

Held, L. and Ott, M. (2018). On p-values and Bayes factors. *Annual Review of Statistics and Its Application*, **5**, 393–419.

Johnson, V. E. (2008). Properties of Bayes factors based on test statistics. *Scandinavian Journal of Statistics*, **35**, 354–368.

Examples

```
LRCalibrate(p=c(0.05, 0.01, 0.005), df=2, alternative="simple")
# gamma alternatives
LRCalibrate(p=c(0.05, 0.01, 0.005), df=c(2, 5, 10))

# plot the minimum Bayes factor as a function of the p-value
# for different degrees of freedom df of the LR test statistic
par(mfrow=c(1,2), las=1)

p <- exp(seq(log(0.005), log(0.3), by=0.01))
df <- c(1, 5, 20)
par(las=1)

# for a simple alternative
minBF.sim <- LRCalibrate(p, df=df, alternative="simple")
matplot(p, t(minBF.sim), type="l", ylab="Minimum Bayes factor", log="xy",
        xlab="Two-sided LR-test p-value", lty=1, lwd=2, axes=FALSE,
        main="Simple alternative")
axis(1, at=c(0.01, 0.03, 0.1, 0.3), c(0.01, 0.03, 0.1, 0.3))
my.values <- c(30, 20, 10, 5, 3, 1)
my.at <- 1/my.values
my.ylegend <- c(paste("1/", my.values[-length(my.values)], sep=""), "1")
axis(2, at=my.at, my.ylegend)
box()
legend("bottomright", legend=c("df=1", "df=5", "df=20"),
        lty=1, lwd=2, , col=1:3)

# for gamma alternatives
minBF.loc <- LRCalibrate(p, df=df, alternative="gamma")
matplot(p, t(minBF.loc), type="l", ylab="  Minimum Bayes factor",
        log="xy", xlab="Two-sided LR-test p-value", lty=1, lwd=2,
        axes=FALSE, main="Local alternatives")
axis(1, at=c(0.01, 0.03, 0.1, 0.3), c(0.01, 0.03, 0.1, 0.3))
axis(2, at=my.at, my.ylegend)
box()
legend("bottomright", legend=c("df=1", "df=5", "df=20"),
        lty=1, lwd=2, col=1:3)
```

`p2z` *Transform p-values to z-values*

Description

Transforms the p-value(s) to the corresponding one- or two-sided z-value(s).

Usage

```
p2z(p, type="two.sided")
```

Arguments

`p`	a vector of p-values
`type`	type of test. Either "two.sided" or "one.sided". Defaults to "two.sided".

Value

A numeric vector of z-values

Examples

```
p2z(p=c(0.005, 0.01, 0.05))
p2z(p=c(0.005, 0.01, 0.05), type="one.sided")
```

`pCalibrate` *Calibrations of two-sided p-values directly based on the p-value*

Description

Transforms a two-sided p-value to a minimum Bayes factor. That minimum Bayes factor is obtained by modelling the distributions of the p-value under the null and the alternative hypothesis, respectively.

Usage

```
pCalibrate(p,  type="exploratory", transform="id")
```

Arguments

`p`	a vector of two-sided p-values
`type`	either `"exploratory"` or `"confirmatory"`. Defaults to `"exploratory"`. Corresponds to different distrbutional assumptions for the p-value p under the alternative.
`transform`	either `"id"`, `"log"`, `"log2"` or `"log10"`. Defaults to `"id"`. Specifies how to transform the minimum Bayes factor(s). `"id"` corresponds to no transformation. `"log"` refers to the natural logarithm, `"log2"` to the logarithm to the base 2 and `"log10"` to the logarithm to the base 10.

Details

If `type="exploratory"` is used, the so-called "- e p log(p)" calibration (Sellke et al., 2001) is applied.

If `type="confirmatory"` is used, the the so-called "- e q log(q)" calibration with q=1-p (Held & Ott, 2018) is applied.

`type="exploratory"` gives a larger minimum Bayes factor than `type="confirmatory"`.

Under the null hypothesis, the distribution of the p-value is assumed to be uniform. Under the alternative, the p-value is assumed to have a beta-distribution with monotonically decreasing density function under both alternatives, with different parameters however. If `type="exploratory"`, the prior sample size does not exceed 2, whereas for `type="confirmatory"`, the prior sample size is at least 2. The latter calibration may be appropriate for small sample size, but for larger sample size it is too conservative (Held & Ott, 2018).

Note that for the "- e p log(p)" calibration, alternative derivations which do not assume a beta-distribution under the alternative have also been given, see Sellke et al. (2001).

Value

A numeric vector of minimum Bayes factors for the specified p-values

Note

The argument `type` replaces the argument `alternative` in version 0.1-1 of this package.
`type="exploratory"` corresponds to `alternative="noninformative"` and
`type="confirmatory"` corresponds to `alternative="informative"`.

References

Held, L. and Ott, M. (2018). On p-values and Bayes factors. *Annual Review of Statistics and Its Application*, **5**, 393–419.

Sellke, T., Bayarri, M. J. and Berger, J. O. (2001). Calibration of p values for testing precise null hypotheses. *The American Statistician*, **55**, 62–71.

Vovk, V. G. (1993). A logic of probability, with application to the foundations of statistics (with discussion and a reply by the author). *Journal of the Royal Statistical Society, Series B*, **55**, 317–351.

Examples

```
pCalibrate(p=c(0.05, 0.01, 0.001))
pCalibrate(p=c(0.05, 0.01, 0.001), type="confirmatory")

# plot the 2 calibrations as a function of the p-value
par(las=1)
p <- exp(seq(log(0.0001), log(0.3), by=0.01))
minBF1 <- pCalibrate(p=p)
minBF2 <- pCalibrate(p=p, type="confirmatory")
plot(p, minBF1, type="l", log="xy",
      xlab="Two sided p-value", ylab="Minimum Bayes factor",
     axes=FALSE, lwd=2, col=1)
lines(p, minBF2, col=2, lwd=2)
```

```
axis(1, at=c(0.0001, 0.0003, 0.001, 0.003, 0.01, 0.03, 0.1, 0.3),
     as.character(c(format(c(0.0001,0.0003), nsmall=4, digits=4,
                 scientific=FALSE),
     c(0.001, 0.003, 0.01, 0.03, 0.1, 0.3))))
my.values <- c(3000, 1000, 300, 100, 30, 10, 3, 1)
my.at <- 1/my.values
my.ylegend <- c(paste("1/", my.values[-length(my.values)], sep=""), "1")
axis(2, at=my.at, my.ylegend)
box()
legend("bottomright", lty=1, lwd=2,
        legend=c("- e p log(p)", "- e q log(q)")
                , col=c(1,2))
```

tCalibrate *Calibration of p-values from t-tests*

Description

Transforms one- and two-sided p-values from t-tests to sample-size adjusted lower bounds on the Bayes factor for the point null hypothesis against the alternative.

Usage

```
tCalibrate(p, n, type="two.sided", alternative="normal", transform="id")
```

Arguments

`p`	a vector of p-values
`n`	a vector of sample sizes
`type`	either `"one.sided"` or `"two.sided"`. Defaults to `"two.sided"`. Specifies if the t-test (and the corresponding p-value) is one-sided or two-sided.
`alternative`	either `"simple"` or `"normal"`. Defaults to `"normal"`. Specifies the alternative hypotheses for the mean to consider. See Details for more information.
`transform`	either `"id"`, `"log"`, `"log2"` or `"log10"`. Defaults to `"id"`. Specifies how to transform the lower bound on the Bayes factor. `"id"` corresponds to no transformation. `"log"` refers to the natural logarithm, `"log2"` to the logarithm to the base 2 and `"log10"` to the logarithm to the base 10.

Details

For one-sided t-tests (`type="one.sided"`), `alternative="simple"` considers all simple point alternatives for the mean and `alternative="normal"` all shifted and possibly mirrored (if `direction=less`) half-normal distributions with mode at the null value.

For two-sided t-tests (`type="two.sided"`), `alternative="simple"` considers all two-point distributions on the mean which are symmetric with respect to the null value and `alternative="normal"` all local normal distributions centered around the null value.

Note that for `alternative="normal"`, the lower bound on the Bayes factor is the same for one-sided and two-sided p-values.

For `type="one.sided",alternative="simple"`, the calibration is available in closed form and is given in Held & Ott (2018), equation (19).

For `type="two.sided",alternative="simple"`, the calibration needs to be computed by numerical optimization and is given in Held & Ott (2018), equations (17) and (18).

For `alternative="normal"`, the corresponding lower bound on the Bayes factor has been derived in Johnson (2005) and the calibration is proposed in Held & Ott (2016), Section 3.

If `alternative="normal"`, `tCalibrate()` is a special case of `FCalibrate()`, i.e.
`tCalibrate(p,n,alternative="normal")` is equivalent to
`FCalibrate(p,n,d=1,alternative="chi.squared")`.

One can show that if one considers the class of all alternatives which are symmetric with respect to the null value, one obtains the same lower bound as for the subclass of all symmetric two-point distributions.

Value

A matrix containing the lower bounds on the Bayes factors as entries, for all combinations of p-value and sample size. The values for the k-th sample size (k-th entry in the vector `n`) and the specified p-values are given in the k-th row.

References

Held, L. and Ott, M. (2018). On p-values and Bayes factors. *Annual Review of Statistics and Its Application*, **5**, 393–419.

Held, L. and Ott, M. (2016). How the maximal evidence of P-values against point null hypotheses depends on sample size. *The American Statistician*, **70**, 335–341.

Johnson, V. E. (2005). Bayes factors based on test statistics. *Journal of the Royal Statistical Society, Series B* **67**:689–701.

See Also

`FCalibrate`

Examples

```
tCalibrate(p=c(0.05, 0.01, 0.005), n=10, type="one.sided",
           alternative="simple")
# two-sided alternatives
tCalibrate(p=c(0.05, 0.01, 0.005), n=20, alternative="simple")
tCalibrate(p=c(0.05, 0.01, 0.005), n=c(10, 20, 50))

# for two-sided alternatives, plot the minimum Bayes factors
# as a function of the p-value
par(mfrow=c(1,2), pty="s", las=1)

# plot for simple alternatives
p <- exp(seq(log(0.0001), log(0.3), by=0.01))
```

```
n <- c(10, 20, 50)
minBF <- tCalibrate(p, n, alternative = "simple")
# compare to the bound for large n
bound1 <- zCalibrate(p, alternative="simple")
matplot(p, t(minBF), ylim=c(0.0003, 1), type="l",
        ylab="Min. Bayes factor", log="xy",
        xlab="two-sided t-test p-value", lty=1, lwd=2,
        axes=FALSE, main="Simple alternative")
lines(p, bound1, col="gray", lty=2, lwd=2)
my.at1 <- c(0.0001, 0.0003, 0.001, 0.003, 0.01, 0.03, 0.1, 0.3)
my.values1 <- as.character(c(format(c(0.0001,0.0003), nsmall=4,
                                    digits=4, scientific=FALSE),
                           c(0.001, 0.003, 0.01, 0.03, 0.1, 0.3)))
axis(1, at=my.at1, labels=my.values1)
my.values2 <- c(3000, 1000, 300, 100, 30, 10, 3, 1)
my.at2 <- 1/my.values2
my.ylegend2 <- c(paste("1/", my.values2[-length(my.values2)], sep=""),
                 "1")
axis(2, at=my.at2, labels=my.ylegend2)
box()
legend("bottomright", legend=rev(c("n=10", "n=20", "n=50", "n large")),
        lty=rev(c(rep(1,3), 2)), lwd=2, col=rev(c(1:3, "gray")))

# plot for local normal alternatives
p <- exp(seq(log(0.0001), log(0.3), by=0.01))
n <- c(10, 20, 50)
minBF <- tCalibrate(p, n)
bound2 <- zCalibrate(p)
matplot(p, t(minBF), ylim=c(0.0003, 1), type="l",
        ylab="Min. Bayes factor", log="xy",
        xlab="two-sided t-test p-value", lty=1, lwd=2,
        axes=FALSE, main="Local normal alternatives")
lines(p, bound2, col="gray", lty=2, lwd=2)
axis(1, at=my.at1, labels=my.values1)
axis(2, at=my.at2, my.ylegend2)
box()
legend("bottomright", legend=rev(c("n=10", "n=20", "n=50", "n large")),
        lty=rev(c(rep(1,3), 2)), lwd=2, col=rev(c(1:3, "gray")))
```

`twoby2Calibrate` *Minimum Bayes factors and p-values from Fisher's exact test for 2x2 contingency tables*

Description

Computes a sample-size adjusted lower bound on the Bayes factor (for the point null hypothesis against the alternative) for the given 2x2 contingency table. Also returns p-values from Fisher's exact test (different versions in the two-sided case) and less conservative alternatives such as a mid p-value (see Details for more information).

Usage

```
twoby2Calibrate(x, type="two.sided", alternative="normal",
                direction=NULL, transform.bf="id")
```

Arguments

`x`	a 2x2 contingency table in matrix form
`type`	either `"one.sided"` or `"two.sided"`. Defaults to `"two.sided"`. Specifies if Fisher's exact test (and the corresponding p-value) is one-sided or two-sided.
`alternative`	either `"simple"` or `"normal"`. Defaults to `"normal"`. Specifies the alternative hypotheses for the (log) odds ratio to consider for two-sided tests. Is ignored if `type="one.sided"` (in this case only simple alternatives are available).
`direction`	either `"greater"`, `"less"` or `NULL`. Defaults to `NULL`. Specifies the direction of the alternative for one-sided tests: `"greater"` corresponds to an odds ratio > 1 and `"less"` to an odds ratio < 1. Is ignored if `type="two.sided"`.
`transform.bf`	either `"id"`, `"log"`, `"log2"` or `"log10"`. Defaults to `"id"`. Specifies how to transform the lower bound on the Bayes factor. `"id"` corresponds to no transformation. `"log"` refers to the natural logarithm, `"log2"` to the logarithm to the base 2 and `"log10"` to the logarithm to the base 10.

Details

If `type="two.sided"`, the point null hypothesis that the odds ratio is 1 is tested against specific two-sided alternatives: `alternative="simple"` considers all two-point distributions symmetric around 0 for the log odds ratio. `alternative="normal"` assumes a local normal prior distribution (a so-called g-prior) centered around 0 for the log odds ratio.

In the one-sided case (`type="one.sided"`), `direction="less"` tests the alternative that the odds ratio is less than 1 and considers simple point alternatives in that direction to compute the lower bound on the Bayes factor. `"greater"` does the same for the alternative that the odds ratio is larger than 1.

The calibration obtained with `type="two.sided",alternative="normal"` is based on the methodology proposed in Li & Clyde (2018) and yields an (approximate) lower bound on the Bayes factor in closed form. All the other lower bounds on the Bayes factor are computed by numerical optimization. For `type="two.sided"`, the two calibrations are described in Ott & Held (2019).

For one-sided alternatives, the following p-value and 2 related quantities are computed:

- `p.fi` is the one-sided p-value from Fisher's exact test.
- `p.mid` is a "mid" p-value. It is obtained by subtracting half of the probability mass of the observed table from `p.fi`.
- `p.lie` is a Bayesian posterior probability. If `direction="greater"`, it is the posterior probability that the odds ratio exceeds 1 given the observed table under the assumption of uniform priors on the success probabilities for the two groups. If `direction="less"`, it is the posterior probability that the odds ratio does not exceed 1 given the observed table under the same priors.

For two-sided alternatives, the following 3 p-values and 2 related quantities are computed:

- `p.pb` is the "probability-based" p-value (the classical choice), defined as the sum of the probabilities of all tables which are at most a likely as the observed table and have the same marginals.
- `p.ce` is the "central" p-value, which is twice the minimum one-sided p-value (from Fisher's exact test), bounded by 1.
- `p.bl` is "Blaker's" p-value, which is the minimum one-sided p-value (from Fisher's exact test) plus the largest tail probability from the other tail of the distribution that does not exceed that minimum.
- `p.mid` is a "mid" p-value. It is the mid-p modification of the central p-value, i.e. it equals twice the minimum one-sided mid p-value.
- `p.lie` is a two-sided version of the posterior probability for the one-sided test. Let p.lie.os be the one-sided posterior probability that the odds ratio does not exceed 1 given the observed table, as returned by the one-sided test with `direction="less"`. Then `p.lie` = 2 min{p.lie.os, 1-p.lie.os}.

For one-sided alternatives, the posterior probability `p.lie` was already studied in Liebermeister (1877) and its frequentist properties are investigated in Seneta & Phipps (2001).

For two-sided alternatives, the 3 p-values from Fisher's exact test are defined in equations (2.24)-(2.26) in Kateri (2014) and computed using the function `exact2x2()` in the package `exact2x2`. The "mid" p-value is described in Rothman & Greenland (1998, pp. 222-223). The Bayesian significance measure `p.lie` is proposed in Ott & Held (2019) as a modification of the corresponding one-sided significance measure.

Value

A list of the following two elements:

`minBF`	the lower bound on the Bayes factor
`p.value`	A vector of 3 one-sided p-values/significance measures for one-sided tests, namely first the p-value `p.fi` from Fisher's exact test, second the corresponding mid p-value `p.mid` and third the Bayesian posterior probability `p.lie` (see Details for more information). A vector of 5 two-sided p-values/significance measures for two-sided tests: The first three p-values `p.pb`, `p.ce` and `p.bl` (see Details for the definitions) correspond to two-sided p-values from Fisher's exact test. The 4th quantity `p.mid` is a mid p-value, namely the mid-p modification of the second p-value `p.ce`. The last element `p.lie` is a Bayesian significance measure (see Details for additional information).

Warning

For 2x2 tables with entries equal to 0, the minimum Bayes factor is either not defined (for `alternative="normal"`) or the underlying numerical optimization is unstable (for `alternative="simple"`). A warning is displayed in such cases and `minBF=NA` is returned, but the different p-values/significance measures are still available.

References

Li, Y. and Clyde, M. A. (2018). Mixtures of g-priors in generalized linear models. *Journal of the American Statistical Association*, **113**:524, 1828–1845. https://doi.org/10.1080/01621459.2018.1469992

Liebermeister, C. (1877). Ueber Wahrscheinlichkeitsrechnung in Anwendung auf therapeutische Statistik. Sammlung klinischer Vortraege. *Innere Medizin*, **110**(31–64), 935– 962.

Kateri, M. (2014). Contingency Table Analysis - Methods and Implementation using R. Statistics for Industry and Technology. Birkhaeuser.

Ott, M. and Held, L. (2019). Bayesian calibration of p-values from Fisher's exact test. *International Statistical Review* **87**(2), 285–305. https://doi.org/10.1111/insr.12307

Rothman, K. J. and Greenland, S. (1998). Modern Epidemiology. 2nd ed. Lippincott-Raven.

Seneta, E. and Phipps, M. C. (2001). On the comparison of two observed frequencies. *Biometrical Journal*, **43**(1), 23–43.

See Also

For computation of p-values: exact2x2 in package exact2x2.

Examples

```
tab <- matrix(c(1,15,5,10), nrow=2, byrow=TRUE)
# different minimum Bayes factors
twoby2Calibrate(x=tab, type="one.sided", direction="greater")$minBF
twoby2Calibrate(x=tab, type="one.sided", direction="less")$minBF
twoby2Calibrate(x=tab, type="two.sided", alternative="simple")$minBF
twoby2Calibrate(x=tab)$minBF
# one-sided p-values
twoby2Calibrate(x=tab, type="one.sided", direction="greater")$p.value
twoby2Calibrate(x=tab, type="one.sided", direction="less")$p.value
# two-sided p-values
twoby2Calibrate(x=tab)$p.value
```

---

z2p *Transform z-values to p-values*

Description

Transforms the z-value(s) to the corresponding one- or two-sided p-value(s).

Usage

```
z2p(z, type="two.sided")
```

Arguments

z	a vector of z-values
type	Underlying type of p-value. Either "two.sided" or "one.sided". Defaults to "two.sided".

Value

A numeric vector of p-values

Examples

```
z2p(z=c(1,2,5))
z2p(z=c(1,2,5), type="one.sided")
```

`zCalibrate` *Calibration of p-values from z-tests*

Description

Transforms one- and two-sided p-values from z-tests to minimum Bayes factors.

Usage

```
zCalibrate(p=NULL, z=p2z(p), type="two.sided",
           alternative="normal", transform="id")
```

Arguments

`p`	a vector of p-values, default is `NULL`
`z`	a vector of z-values, is calculated from `p`, if necessary.
`type`	either `"one.sided"` or `"two.sided"`. Defaults to `"two.sided"`. Specifies if the z-test (and the corresponding p-value) is one-sided or two-sided.
`alternative`	either `"simple"`, `"normal"` or `"local"`. Defaults to `"normal"`. Specifies the alternative hypotheses for the mean to consider. See Details for more information.
`transform`	either `"id"`, `"log"`, `"log2"` or `"log10"`. Defaults to `"id"`. Specifies how to transform the minimum Bayes factor(s). `"id"` corresponds to no transformation. `"log"` refers to the natural logarithm, `"log2"` to the logarithm to the base 2 and `"log10"` to the logarithm to the base 10.

Details

For one-sided z-tests (`type="one.sided"`), `alternative="simple"` considers all simple point alternatives for the mean in the specified direction, `alternative="normal"` all shifted and possibly mirrored (if `direction=less`) half-normal distributions with mode at the null value and `alternative="local"` all monotonic density functions with mode at the null value, which are concentrated on the parameter space of the alternative.

For two-sided z-tests (`type="two.sided"`), `alternative="simple"` considers all two-point distributions on the mean which are symmetric with respect to the null value,
`alternative="normal"` all normal distributions centered around the null value and
`alternative="local"` all unimodal symmetric distributions with respect the null value (i.e. these distributions are non-increasing as a function of the distance to the null value).

Note that for `alternative="normal"` and `alternative="local"`, the minimum Bayes factor is the same for `type="one.sided"` and `type="two.sided"`.

For `type="one.sided",alternative="simple"` and for `alternative="normal"`, the calibrations have closed-form expressions and were proposed in Edwards et al. (1963).
For `type="two.sided",alternative="simple"` and for `alternative="local"`, the calibrations need to be computed by numerical optimization and were derived in Berger & Sellke (1987). Most of these calibrations are also described in Held & Ott (2018).

Note that if one considers the class of all alternatives, the resulting minimum Bayes factor is the same as for `type="one.sided",alternative="simple"`.

One can show that if one considers the class of all alternatives which are symmetric around the null value, one obtains the same minimum Bayes factor as for the subclass of all symmetric two-point distributions (Berger & Sellke, 1987).

Value

A numeric vector of minimum Bayes factors for the specified p-values

Note

For small p-values, using the corresponding z-value as an argument instead of the p-value may lead to more accurate computations.

References

Berger, J. O. and Sellke, T. (1987). Testing a point null hypothesis: The irreconcilability of P values and evidence (with discussion). *Journal of the American Statistical Association*, **82**, 112–139.

Edwards, W., Lindman, H. and Savage, L. J. (1963). Bayesian statistical inference for psychological research. *Psychological Review*, **70**, 193–242.

Held, L. and Ott, M. (2018). On p-values and Bayes factors. *Annual Review of Statistics and Its Application*, **5**, 393–419.

See Also

`tCalibrate` for sample-size adjusted calibrations of p-values from t-tests

Examples

```
# two-sided alternatives
zCalibrate(p=c(0.05, 0.01, 0.005), alternative="simple")
zCalibrate(p=c(0.05, 0.01, 0.005))
zCalibrate(p=c(0.05, 0.01, 0.005), alternative="local")
zCalibrate(z=c(2, 3, 4), alternative="local")
# one-sided alternatives
zCalibrate(p=c(0.05, 0.01, 0.005), type="one.sided",
           alternative="simple")
zCalibrate(p=c(0.05, 0.01, 0.005), type="one.sided")
zCalibrate(p=c(0.05, 0.01, 0.005), type="one.sided",
           alternative="local")
```

```
# plot the different calibrations as a function of the p-value
par(las=1)
p <- exp(seq(log(0.0001), log(0.3), by=0.01))
minBF <- matrix(NA, ncol=4, nrow=length(p))
minBF[ ,1] <- zCalibrate(p)
minBF[ ,2] <- zCalibrate(p, type="two.sided", alternative="local")
minBF[ ,3] <- zCalibrate(p, type="two.sided", alternative="simple")
minBF[ ,4] <- zCalibrate(p, type="one.sided", alternative="simple")
matplot(p, minBF, type="l", ylab="Minimum Bayes factor", log="xy",
        xlab="z-test p-value", lty=1, lwd=2, col=c(1,3,2,4), axes=FALSE)
axis(1, at=c(0.0001, 0.0003, 0.001, 0.003, 0.01, 0.03, 0.1, 0.3),
     as.character(c(format(c(0.0001,0.0003), nsmall=4, digits=4,
                  scientific=FALSE),
                  c(0.001, 0.003, 0.01, 0.03, 0.1, 0.3))))
my.values <- c(3000, 1000, 300, 100, 30, 10, 3, 1)
my.at <- 1/my.values
my.ylegend <- c(paste("1/", my.values[-length(my.values)], sep=""), "1")
axis(2, at=my.at, my.ylegend)
box()
legend("bottomright", lty=1, lwd=2,
        legend=c("normal", "unimodal symmetric",
                "two-sided simple", "one-sided simple"), col=c(1,3,2,4))
```

Index