

Package ‘bain’

February 8, 2019

Type Package

Date 2019-02-08

Title Bayes Factors for Informative Hypotheses

Version 0.2.1

Description Computes approximated adjusted fractional Bayes factors for equality, inequality, and about equality constrained hypotheses. S3 methods are available for specific types of `lm()` models, namely ANOVA, ANCOVA, and multiple regression, and for the `t_test()`. The statistical underpinnings are described in

Hojtink, Mulder, van Lissa, and Gu, (2018) <doi:10.31234/osf.io/v3shc>,

Gu, Mulder, and Hoijtink, (2018) <doi:10.1111/bmsp.12110>,

Hojtink, Gu, and Mulder, (2018) <doi:10.1111/bmsp.12145>, and

Hojtink, Gu, Mulder, and Rosseel, (2018) <doi:10.1037/met0000187>.

License GPL (>= 3)

Encoding UTF-8

LazyData true

URL <https://informative-hypotheses.sites.uu.nl/software/bain/>

NeedsCompilation yes

RoxygenNote 6.1.1

Depends R (>= 3.0.0), stats

Suggests MASS, testthat, knitr, rmarkdown

VignetteBuilder knitr

Author Xin Gu [aut],
Herbert Hoijtink [aut],
Joris Mulder [aut],
Caspar J van Lissa [aut, cre],
Jeff Jones [ctb],
Niels Waller [ctb],
The R Core Team [cph]

Maintainer Caspar J van Lissa <c.j.vanlissa@uu.nl>

Repository CRAN

Date/Publication 2019-02-08 12:13:51 UTC

R topics documented:

bain	2
seBeta	4
sesamesim	5
t_test	6

Index	9
--------------	----------

bain	<i>Bayes factors for informative hypotheses</i>
------	---

Description

bain is an acronym for "Bayesian informative hypotheses evaluation". It uses the Bayes factor to evaluate hypotheses specified using equality and inequality constraints among (linear combinations of) parameters in a wide range of statistical models. A tutorial is provided by Hoijsink, Mulder, van Lissa, and Gu (2018) retrievable from the Psychological Methods website at <https://www.apa.org/pubs/journals/met/> or the bain website at <https://informative-hypotheses.sites.uu.nl/software/bain/> **Users are advised to read the tutorial AND the vignette that is provided with this package before using bain.**

Usage

```
bain(x, hypothesis, ...)
```

Arguments

x	An R object containing the outcome of a statistical analysis. Currently, the following objects can be processed: <code>lm()</code> , <code>t_test()</code> , and named vector objects. See the vignette for elaborations.
hypothesis	A character string containing the informative hypotheses to evaluate. See the vignette for elaborations.
...	Additional arguments. See the vignette for elaborations.

Value

The main output resulting from analyses with bain are Bayes factors and posterior model probabilities associated with the hypotheses that are evaluated. See the **tutorial** and the **vignette** for further elaborations.

Author(s)

The main authors of the bain package are Xin Gu, Caspar van Lissa, Herbert Hoijsink and Joris Mulder. Contributions were made by Marlyne Bosman and Camiel van Zundert. Contact information can be found on the bain website at <https://informative-hypotheses.sites.uu.nl/software/bain/>

References

See the vignette for additional references.

Hojtink, H., Mulder, J., van Lissa, C., and Gu, X. (2018). A tutorial on testing hypotheses using the Bayes factor. *Psychological Methods*. DOI: 10.1037/met0000201

Examples

```
# USING BAIN WITH A LM OBJECT: Bayesian ANOVA
# make a factor of variable site
sesamesim$site <- as.factor(sesamesim$site)
# execute an analysis of variance using lm() which, due to the -1, returns
# estimates of the means per group
anov <- lm(postnumb~site-1,sesamesim)
# take a look at the estimated means and their names
coef(anov)
# set a seed value
set.seed(100)
# use the names to formulate and test hypotheses with bain
results <- bain(anov, "site1=site2=site3=site4=site5; site2>site5>site1>
site3>site4")
#
# USING BAIN WITH A NAMED VECTOR: Bayesian ANOVA

# make a factor of variable site
sesamesim$site <- as.factor(sesamesim$site)
# execute an analysis of variance using lm() which, due to the -1, returns
# estimates of the means per group
anov <- lm(postnumb~site-1,sesamesim)
# collect the estimates means in a vector
estimate <- coef(anov)
# give names to the estimates in anov
names(estimate) <- c("site1", "site2", "site3","site4","site5")
# create a vector containing the sample sizes of each group
ngroup <- table(sesamesim$site)
# compute the variance of the means and collect them in a list
var <- summary(anov)$sigma**2
cov1 <- matrix(var/ngroup[1], nrow=1, ncol=1)
cov2 <- matrix(var/ngroup[2], nrow=1, ncol=1)
cov3 <- matrix(var/ngroup[3], nrow=1, ncol=1)
cov4 <- matrix(var/ngroup[4], nrow=1, ncol=1)
cov5 <- matrix(var/ngroup[5], nrow=1, ncol=1)
covlist <- list(cov1, cov2, cov3, cov4,cov5)
# set a seed value
set.seed(100)
# test hypotheses with bain. Note that there are multiple groups
# characterized by one mean, therefore group_parameters=1. Note that
# there are no joint parameters, therefore, joint_parameters=0.
results <- bain(estimate,
"site1=site2=site3=site4=site5; site2>site5>site1>site3>site4",
n=ngroup,Sigma=covlist,group_parameters=1,joint_parameters = 0)

# SEE THE TUTORIAL AND VIGNETTE FOR MANY ADDITIONAL EXAMPLES
```

seBeta

*Standard Errors and CIs for Standardized Regression Coefficients***Description**

Computes Normal Theory and ADF Standard Errors and CIs for Standardized Regression Coefficients

Usage

```
seBeta(X = NULL, y = NULL, cov.x = NULL, cov.xy = NULL,
       var.y = NULL, Nobs = NULL, alpha = 0.05, estimator = "ADF")
```

Arguments

X	Matrix of predictor scores.
y	Vector of criterion scores.
cov.x	Covariance or correlation matrix of predictors.
cov.xy	Vector of covariances or correlations between predictors and criterion.
var.y	Criterion variance.
Nobs	Number of observations.
alpha	Desired Type I error rate; default = .05.
estimator	'ADF' or 'Normal' confidence intervals - requires raw X and raw y; default = 'ADF'.

Value

cov.Beta	Normal theory or ADF covariance matrix of standardized regression coefficients.
se.Beta	standard errors for standardized regression coefficients.
alpha	desired Type-I error rate.
CI.Beta	Normal theory or ADF (1-alpha) intervals for standardized regression coefficients.
estimator	estimator = "ADF" or "Normal".

Author(s)

Jeff Jones and Niels Waller

References

Jones, J. A, and Waller, N. G. (2015). The Normal-Theory and Asymptotic Distribution-Free (ADF) covariance matrix of standardized regression coefficients: Theoretical extensions and finite sample behavior. *Psychometrika*, 80, 365-378.

Examples

```
set.seed(123)

R <- matrix(.5, 3, 3)
diag(R) <- 1
X <- sesamesim[, c("peabody", "prenumb", "postnumb")]
y <- sesamesim$age
results <- seBeta(X, y, Nobs = nrow(sesamesim), alpha = .05, estimator = 'ADF')
print(results, digits = 3)

library(MASS)

set.seed(123)

R <- matrix(.5, 3, 3)
diag(R) <- 1
X <- mvrnorm(n = 200, mu = rep(0, 3), Sigma = R, empirical = TRUE)
Beta <- c(.2, .3, .4)
y <- X %*% Beta + .64 * scale(rnorm(200))
results <- seBeta(X, y, Nobs = 200, alpha = .05, estimator = 'ADF')
print(results, digits = 3)
```

sesamesim

Simulated Sesame Street Data

Description

A simulated data set inspired by the Sesame Street data set from: Stevens, J. P. (1996). *Applied Multivariate Statistics for the Social Sciences*. Mahwah NJ: Lawrence Erlbaum.

Usage

```
data(sesamesim)
```

Format

A data frame with 240 rows and 9 variables.

Details

The number knowledge of children is measured before watching Sesame Street, after one year, and a follow up measurement after two years.

sex Sex of the child; 1 = boy, 2 = girl

site Site of the child's origin; 1 = disadvantaged inner city, 2 = advantaged suburban, 3 = advantaged rural, 4 = disadvantaged rural, 5 = disadvantaged Spanish speaking

setting Setting in which the child watches Sesame Street; 1 = at home, 2 = at school

age Age of the child in months

viewenc Whether or not the child is encouraged to watch Sesame Street; 0 = no, 1 = yes

peabody Peabody mental age score of the child; the higher the score the higher the mental age

prenumb score on a numbers test before watching Sesame Street for a year

postnumb score on a numbers test after watching Sesame Street for a year

funumb follow up numbers test score measured one year after postnumb

t_test	<i>Student's t-Test</i>
--------	-------------------------

Description

Performs one and two sample t-tests on vectors of data.

Usage

```
t_test(x, ...)

## Default S3 method:
t_test(x, y = NULL, alternative = c("two.sided",
  "less", "greater"), mu = 0, paired = FALSE, var.equal = FALSE,
  conf.level = 0.95, ...)

## S3 method for class 'formula'
t_test(formula, data, subset, na.action, ...)
```

Arguments

x	a (non-empty) numeric vector of data values.
...	further arguments to be passed to or from methods.
y	an optional (non-empty) numeric vector of data values.
alternative	a character string specifying the alternative hypothesis, must be one of "two.sided" (default), "greater" or "less". You can specify just the initial letter.
mu	a number indicating the true value of the mean (or difference in means if you are performing a two sample test).

paired	a logical indicating whether you want a paired t-test.
var.equal	a logical variable indicating whether to treat the two variances as being equal. If TRUE then the pooled variance is used to estimate the variance otherwise the Welch (or Satterthwaite) approximation to the degrees of freedom is used.
conf.level	confidence level of the interval.
formula	a formula of the form $lhs \sim rhs$ where lhs is a numeric variable giving the data values and rhs a factor with two levels giving the corresponding groups.
data	an optional matrix or data frame (or similar: see <code>model.frame</code>) containing the variables in the formula formula. By default the variables are taken from <code>environment(formula)</code> .
subset	an optional vector specifying a subset of observations to be used.
na.action	a function which indicates what should happen when the data contain NAs. Defaults to <code>getOption("na.action")</code> .

Details

The formula interface is only applicable for the 2-sample tests.

`alternative = "greater"` is the alternative that x has a larger mean than y .

If `paired` is TRUE then both x and y must be specified and they must be the same length. Missing values are silently removed (in pairs if `paired` is TRUE). If `var.equal` is TRUE then the pooled estimate of the variance is used. By default, if `var.equal` is FALSE then the variance is estimated separately for both groups and the Welch modification to the degrees of freedom is used.

If the input data are effectively constant (compared to the larger of the two means) an error is generated.

Value

A list with class "htest" containing the following components:

statistic	the value of the t-statistic.
parameter	the degrees of freedom for the t-statistic.
p.value	the p-value for the test.
conf.int	a confidence interval for the mean appropriate to the specified alternative hypothesis.
estimate	the estimated mean or difference in means depending on whether it was a one-sample test or a two-sample test.
null.value	the specified hypothesized value of the mean or mean difference depending on whether it was a one-sample test or a two-sample test.
alternative	a character string describing the alternative hypothesis.
method	a character string indicating what type of t-test was performed.
data.name	a character string giving the name(s) of the data.

Bain t_test

In order to allow users to enjoy the functionality of bain with the familiar stats-function t.test, we have had to make minor changes to the function t.test.default. All rights to, and credit for, the function t.test.default belong to the R Core Team, as indicated in the original license below. We make no claims to copyright and incur no liability with regard to the changes implemented in t_test.

This the original copyright notice by the R core team: File src/library/stats/R/t_test.R Part of the R package, <https://www.R-project.org>

Copyright (C) 1995-2015 The R Core Team

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

A copy of the GNU General Public License is available at <https://www.R-project.org/Licenses/>

See Also

[t.test](#)

Examples

```
require(graphics)

t_test(1:10, y = c(7:20))      # P = .00001855
t_test(1:10, y = c(7:20, 200)) # P = .1245    -- NOT significant anymore

## Classical example: Student's sleep data
plot(extra ~ group, data = sleep)
## Traditional interface
with(sleep, t_test(extra[group == 1], extra[group == 2]))
## Formula interface
t_test(extra ~ group, data = sleep)
```


Index

*Topic **Statistics**

seBeta, 4

*Topic **datasets**

sesamesim, 5

*Topic **htest**

t_test, 6

bain, 2

model.frame, 7

seBeta, 4

sesamesim, 5

t.test, 8

t_test, 6