Package ‘bain’

March 9, 2020

Type Package
Date 2020-03-09
Title Bayes Factors for Informative Hypotheses
Version 0.2.4
License GPL (>= 3)
Encoding UTF-8
LazyData true
URL https://informative-hypotheses.sites.uu.nl/software/bain/
NeedsCompilation yes
RoxygenNote 7.0.2
Depends R (>= 3.0.0), stats
Imports lavaan
Suggests MASS, testthat, knitr, rmarkdown
VignetteBuilder knitr
Author Xin Gu [aut],
Herbert Hoijtink [aut],
Joris Mulder [aut],
Caspar J van Lissa [aut, cre],
Van Zundert Camiel [ctb],
Jeff Jones [ctb],
Niels Waller [ctb]
Maintainer Caspar J van Lissa <c.j.vanlissa@uu.nl>
Repository CRAN
Date/Publication 2020-03-09 12:40:02 UTC
**R topics documented:**

- bain ................................................................. 2
- bain_sensitivity ............................................... 4
- seBeta ............................................................. 5
- sesamesim ......................................................... 7
- t_test ............................................................... 8

**Index**

<table>
<thead>
<tr>
<th>bain</th>
<th>Bayes factors for informative hypotheses</th>
</tr>
</thead>
</table>

**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 by Hoijtink, Mulder, van Lissa, and Gu (2018), was published in Psychological Methods. The preprint of that tutorial is available at DOI:10.31234/osf.io/v3shc, or on 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, fraction = 1, ...)
```

**Arguments**

- **x**
  - An R object containing the outcome of a statistical analysis. Currently, the following objects can be processed: `lm()`, `t_test()`, `lavaan` objects created with the `sem()`, `cfa()`, and `growth()` functions, and named vector objects. See the vignette for elaborations.

- **hypothesis**
  - A character string containing the informative hypotheses to evaluate. See the vignette for elaborations.

- **fraction**
  - A number representing the fraction of information in the data used to construct the prior distribution (see the tutorial DOI: 10.1037/met0000201): The default value 1 denotes the minimal fraction, 2 denotes twice the minimal fraction, etc.

- **...**
  - 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 Hoijtink 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.


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=ngrid,Sigma=covlist,group_parameters=1,joint_parameters = 0)

# SEE THE TUTORIAL AND VIGNETTE FOR MANY ADDITIONAL EXAMPLES

## bain_sensitivity

### Sensitivity analysis for bain

**Description**

Conducts a sensitivity analysis for bain.

**Usage**

bain_sensitivity(x, hypothesis, fractions = 1, ...)

**Arguments**

- **x**: An R object containing the outcome of a statistical analysis. Currently, the following objects can be processed: lm(), t_test(), lavaan objects created with the sem(), cfa(), and growth() functions, and named vector objects. See the vignette for elaborations.
- **hypothesis**: A character string containing the informative hypotheses to evaluate. See the vignette for elaborations.
- **fractions**: A numeric vector, representing the fractions of information in the data to be used to construct the prior distribution (see the tutorial DOI: 10.1037/met0000201): The default value 1 denotes the minimal fraction, 2 denotes twice the minimal fraction, etc.
- **...**: Additional arguments passed to bain.

**Details**

The Bayes factor for equality constraints is sensitive to a scaling factor applied to the prior distribution. The argument fraction adjusts this scaling factor. The function bain_sensitivity is a wrapper for bain, which accepts a vector for the fractions argument, and returns a list of bain results objects. A table with a sensitivity analysis for specific statistics can be obtained using the summary() function, which accepts the argument summary(which_stat = ...). The available statistics are elements of the $fit table (Fit_eq, Com_eq, Fit_in, Com_in, Fit, Com, BF, PMPa, and PMPb), and elements of the BFmatrix, which can be accessed by matrix notation, e.g.: summary(bain_sens,which_stat = "BFmatrix[1,2]").
seBeta

Value

A data.frame of class "bain_sensitivity".

Examples

```r
sesamesim$site <- as.factor(sesamesim$site)
res <- lm(sesamesim$postnumb~sesamesim$site-1)
set.seed(4583)
bain_sens <- bain_sensitivity(res, "site1=site2=site3=site4=site5;
  site2>site5>site1>site3=site4;
  site1=site2>site3=site4>site5",
fractions = c(1,2,3))
summary(bain_sens, which_stat = "BF.c")
summary(bain_sens, which_stat = "BFmatrix[1,3]")
```

seBeta

*Standard Errors and CIs for Standardized Regression Coefficients*

Description

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

Usage

```r
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


Examples

```r
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)
```
Description

This is a simulated counterpart of part of the Sesame Street data presented by Stevens (1996, Appendix A) concerning the effect of the first year of the Sesame street series on the knowledge of 240 children in the age range 34 to 69 months. We will use the following variables: sex; site of child’s origin; setting in which Sesame Street is watched; age; whether or not the child is encouraged to watch; Peabody mental age score; score on numbers test before, after and in a follow up measurement; and scores on knowledge of body parts, letters, forms, numbers, relations, and classifications, both before and after watching Sesame Street for a year.

Usage

data(sesamesim)

Format

A data frame with 240 rows and 21 variables.

Details

<table>
<thead>
<tr>
<th>Variable</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>sex</td>
<td>integer</td>
<td>Sex of the child; 1 = boy, 2 = girl</td>
</tr>
<tr>
<td>site</td>
<td>integer</td>
<td>Site of the child’s origin; 1 = disadvantaged inner city, 2 = advantaged suburban , 3 = advantaged rural , 4 = disadvantaged rural , 5 = disadvantaged Spanish speaking</td>
</tr>
<tr>
<td>setting</td>
<td>integer</td>
<td>Setting in which the child watches Sesame Street; 1 = at home, 2 = at school</td>
</tr>
<tr>
<td>age</td>
<td>integer</td>
<td>Age of the child in months</td>
</tr>
<tr>
<td>viewenc</td>
<td>integer</td>
<td>Whether or not the child is encouraged to watch Sesame Street; 0 = no, 1 = yes</td>
</tr>
<tr>
<td>peabody</td>
<td>integer</td>
<td>Peabody mental age score of the child; the higher the score the higher the mental age</td>
</tr>
<tr>
<td>prenumb</td>
<td>integer</td>
<td>score on a numbers test before watching Sesame Street for a year</td>
</tr>
<tr>
<td>postnumb</td>
<td>integer</td>
<td>score on a numbers test after watching Sesame Street for a year</td>
</tr>
<tr>
<td>funumb</td>
<td>integer</td>
<td>follow up numbers test score measured one year after postnumb</td>
</tr>
<tr>
<td>Bb</td>
<td>integer</td>
<td>Knowledge of body parts before</td>
</tr>
<tr>
<td>Bl</td>
<td>integer</td>
<td>Knowledge of letters before</td>
</tr>
<tr>
<td>Bf</td>
<td>integer</td>
<td>Knowledge of forms before</td>
</tr>
<tr>
<td>Bn</td>
<td>integer</td>
<td>Knowledge of numbers before</td>
</tr>
<tr>
<td>Br</td>
<td>integer</td>
<td>Knowledge of relations before</td>
</tr>
<tr>
<td>Be</td>
<td>integer</td>
<td>Knowledge of classifications before</td>
</tr>
<tr>
<td>Ab</td>
<td>integer</td>
<td>Knowledge of body parts after</td>
</tr>
<tr>
<td>Al</td>
<td>integer</td>
<td>Knowledge of letters after</td>
</tr>
<tr>
<td>Af</td>
<td>integer</td>
<td>Knowledge of forms after</td>
</tr>
<tr>
<td>An</td>
<td>integer</td>
<td>Knowledge of numbers after</td>
</tr>
<tr>
<td>Ar</td>
<td>integer</td>
<td>Knowledge of relations after</td>
</tr>
<tr>
<td>Ac</td>
<td>integer</td>
<td>Knowledge of classifications after</td>
</tr>
</tbody>
</table>
References


---

**t_test**

*Student's t-test*

**Description**

This function is a wrapper for the function `t.test`, which returns group-specific sample sizes and variances, in addition to the usual output of `t.test`.

**Usage**

`t_test(x, ...)`

**Arguments**

- `x` An object for which an S3 method of `t.test` exists (vector or formula).
- `...` arguments passed to `t.test`.

**Details**

This wrapper allows users to enjoy the functionality of bain with the familiar interface of the stats-function `t.test`.

For more documentation, see `t.test`.

**Value**

A list with class "t_test" 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.
- `v` The variance or group-specific variances.
- `n` The sample size, or group-specific sample size.
See Also

\texttt{t.test}

Examples

```r
tmp <- t_test(extra ~ group, data = sleep)
tmp$n
tmp$v
tmp2 <- t_test(extra ~ group, data = sleep)
tmp2$n
tmp2$v
tmp <- t_test(extra ~ group, data = sleep, paired = TRUE)
tmp$n
tmp$v
tmp2 <- t_test(extra ~ group, data = sleep, paired = TRUE)
tmp2$n
tmp2$v
t_test(sesamesim$postnumb)
tmp <- t_test(sesamesim$prenumb)
tmp$n
tmp$v
tmp2 <- t_test(sesamesim$prenumb)
tmp2$n
tmp2$v
tmp <- t_test(sesamesim$prenumb, sesamesim$postnumb)
tmp$n
tmp$v
tmp2 <- t_test(sesamesim$prenumb, sesamesim$postnumb)
tmp2$n
tmp2$v
tmp <- t_test(sesamesim$prenumb, sesamesim$postnumb, paired = TRUE)
tmp$n
tmp$v
tmp2 <- t_test(sesamesim$prenumb, sesamesim$postnumb, paired = TRUE)
tmp2$n
tmp2$v
```
Index

*Topic **Statistics**
  seBeta, 5

*Topic **datasets**
  sesamesim, 7

*Topic **htest**
  t_test, 8

bain, 2, 4
bain_sensitivity, 4

seBeta, 5
sesamesim, 7

t.test, 8, 9
t_test, 8