Package ‘effsize’

April 9, 2020

Type Package
Title Efficient Effect Size Computation
Version 0.8.0
Date 2020-04-09
Description
A collection of functions to compute the standardized effect sizes for experiments (Cohen d, Hedges g, Cliff delta, Vargha-Delaney A). The computation algorithms have been optimized to allow efficient computation even with very large data sets.

URL http://github.com/mtorchiano/effsize/

BugReports https://github.com/mtorchiano/effsize/issues
License GPL-2
NeedsCompilation no
Repository CRAN
Suggests testthat
Author Marco Torchiano [aut, cre]
Maintainer Marco Torchiano <marco.torchiano@polito.it>
Date/Publication 2020-04-09 21:50:02 UTC

R topics documented:
edfsize-package .................................................. 2
cfiff.delta .................................................. 3
cfhen.d .................................................. 5
print.effsize .................................................. 8
VD.A .................................................. 9

Index 11
Description

This package contains functions to compute effect sizes both based on means difference (Cohen’s d and Hedges g), dominance matrices (Cliff’s Delta) and stochastic superiority (Vargha-Delaney A).

The computation (especially for Cliff’s Delta) is carried on with highly efficient algorithms.

Details

The main functions are:
- `cliff.delta`
- `cohen.d`
- `VD.A`

Change history

0.3.1 Fixed a bug in `cohen.d` when PAIRED=TRUE, now the PAIRED parameter has no effect, it is left just for compatibility. In a future code clean-up it may be removed.

0.4 Implemented a new algorithm with improved memory and time complexity. In particular new time complexity is $T = O(n_1 \log(n_2))$ vs. the previous $T = O(n_1 n_2)$, and new memory complexity $M = O(n_1 + n_2)$ vs. the previous $M = O(n_1 \cdot n_2)$. In practice now the computation becomes feasible in a "reasonable" time.

0.4.1 Code clean-up and optimization using vectorized binary partitioning.

0.5 Added Vargha and Delaney A and fixed minor bugs with Cohen.d.

0.5.1 Modified the Vargha and Delaney A computation to minimize accuracy errors.

0.5.2 Fixed bug in `cliff.delta`.

0.5.3 Fixed bug in `cohen.d.formula`.

0.5.4 Fixed minor issue detected by check.

0.5.5 Changed the `effsize` field magnitude to a factor value.

0.6.0 Implemented paired computation and CI computation with non-central t-distributions for `cohen.d`.

0.6.1 Added ability to specify factor vector and data vector for ‘cliff.delta’ function (thanks to Joses W. Ho).

0.6.2 `na.rm` in `cohen.d` removes all incomplete pairs when paired.

0.6.3 Fixed bug in `cohen.d` when `na.rm=TRUE`, minor changes in the documentation (thanks to P.Thomas).

0.6.4 Fixed a bug related to paired `cohen.d` with NAs. Minor documentation changes.

0.7.0 Refactored tests using testthat package. Fixed a bug in `cliff.delta` returning inconsistent results when the dominance matrix is returned. Fixed issue concerning CI. Fixed bug in `cohen.d` when using noncentral parameter for negative effect sizes.
cliff.delta

0.7.1 Fixed minor bugs in cliff.delta and cohen.d
0.7.2 Fixed bugs in cohen.d, order of factors is now observed and CI are computed correctly
0.7.3 Fixed bugs in cohen.d, possible endless loop, cleaned code
0.7.4 Fixed bugs in cliff.delta when values are factors
0.7.5 Fixed bugs in cohen.d for paired data
0.7.6 Fixed bugs in cohen.d for CI of paired data
0.7.7 Fixed bugs in cohen.d for non-pooled SD, plus a few pull requests on documentation
0.7.8 Fixed bug in cohen.d wrong correct type check
0.7.9 Fixed tests to be compatible with upcoming R 4.0, that sets stringsAsFactors to FALSE by default
0.8.0 Added non-central CI estimation for single sample cohen.d, fixed a bug related to order of data and added a subject parameter for paired cohen.d

Author(s)
Marco Torchiano http://softeng.polito.it/torchiano/

cliff.delta \hspace{1cm} Cliff’s Delta effect size for ordinal variables

Description
Computes the Cliff’s Delta effect size for ordinal variables with the related confidence interval using efficient algorithms.

Usage

cliff.delta(d, ... )

## S3 method for class 'formula'
cliff.delta(formula, data=list(), conf.level=.95, use.unbiased=TRUE, use.normal=FALSE, return.dm=FALSE, ...)

## Default S3 method:
cliff.delta(d, f, conf.level=.95, use.unbiased=TRUE, use.normal=FALSE, return.dm=FALSE, ...)


Arguments

d  a numeric vector giving either the data values (if f is a factor) or the treatment
group values (if f is a numeric vector)

f  either a factor with two levels or a numeric vector of values (see Details)

conf.level  confidence level of the confidence interval

use.unbiased  a logical indicating whether to compute the delta’s variance using the "unbiased"
estimate formula or the "consistent" estimate

use.normal  logical indicating whether to use the normal or Student-t distribution for the
confidence interval estimation

return.dm  logical indicating whether to return the dominance matrix.  Warning: the ex-
plicit computation of the dominance uses a sub-optimal algorithm both in terms
of memory and time

formula  a formula of the form y ~ f, where y is a numeric variable giving the data values
and f a factor with two levels giving the corresponding group

data  an optional matrix or data frame containing the variables in the formula formula.
By default the variables are taken from environment(formula).

...  further arguments to be passed to or from methods.

Details

Uses the original formula reported in (Cliff 1996).

If the dominance matrix is required i.e. return.dm=TRUE) the full matrix is computed thus using
the naive algorithm. Otherwise, if treatment and control are factors then the optimized linear
complexity algorithm is used, otherwise the RLE algorithm (with complexity n log n) is used.

Value

A list of class effsize containing the following components:

estimate  the Cliff’s delta estimate

conf.int  the confidence interval of the delta

var  the estimated variance of the delta

conf.level  the confidence level used to compute the confidence interval

dm  the dominance matrix used for computation, only if return.dm is TRUE

magnitude  a qualitative assessment of the magnitude of effect size

method  the method used for computing the effect size, always "Cliff’s Delta"

variance.estimation  the method used to compute the delta variance estimation, either "unbiased" or
"consistent"

CI.distribution  the distribution used to compute the confidence interval, either "Normal" or
"Student-t"

The magnitude is assessed using the thresholds provided in (Romano 2006), i.e. ld<0.147 "negligible",
ld<0.33 "small", ld<0.474 "medium", otherwise "large"
cohen.d

Author(s)

Marco Torchiano http://softeng.polito.it/torchiano/

References


See Also
cohen.d, print.effsize

Examples

## Example data from Hogarty and Kromrey (1999)
treatment <- c(10,10,20,20,30,30,30,40,50)
control <- c(10,20,30,40,40,50)
res = cliff.delta(treatment,control,return.dm=TRUE)
print(res)
print(res$dm)

cohen.d  Cohen’s d and Hedges g effect size

Description

Computes the Cohen’s d and Hedges’ g effect size statistics.

Usage

cohen.d(d, ...)

## S3 method for class 'formula'
cohen.d(formula,data=list(),...)

## Default S3 method:
cohen.d(d,f,pooled=TRUE,paired=FALSE,
na.rm=FALSE, mu=0, hedges.correction=FALSE,
conf.level=0.95,noncentral=FALSE,
within=TRUE, subject=NA, ...)
Arguments

d a numeric vector giving either the data values (if \( f \) is a factor) or the treatment group values (if \( f \) is a numeric vector)

\( f \) either a factor with two levels or a numeric vector of values, if \( f \) a numeric vector effect size is computed

formula a formula of the form \( y \sim f \), where \( y \) is a numeric variable giving the values and \( f \) a factor with two levels giving the corresponding groups.

If using a paired computation (\( \text{paired}=\text{TRUE} \)) it is possible to specify the ids of the subjects using the form \( y \sim f \mid \text{Subject(id)} \) which allow the correct pairing of the pre and post values.

A single sample effect size can be specified with the form \( y \sim . \)

data an optional matrix or data frame containing the variables in the formula \( \text{formula} \).

By default the variables are taken from \( \text{environment(\text{formula})} \).

pooled a logical indicating whether compute pooled standard deviation or the whole sample standard deviation. If \( \text{pooled}=\text{TRUE} \) (default) pooled sd is used, if \( \text{pooled}=\text{FALSE} \) the standard deviation of the the control group (the second argument or the one corresponding the the second level of the factor) is used instead.

hedges.correction logical indicating whether apply the Hedges correction

conf.level confidence level of the confidence interval

noncentral logical indicating whether to use non-central t distributions for computing the confidence interval.

paired a logical indicating whether to consider the values as paired, a warning is issued if \( \text{paired}=\text{TRUE} \) with the formula interface and not \( \mid \text{Subject(id)} \) or with data and factor and no subject is provided

within indicates whether to compute the effect size using the within subject variation, taking into consideration the correlation between pre and post samples.

subject an array indicating the id of the subject for a paired computation, when the formula interface is used it can be indicated in the formula by adding \( \mid \text{Subject(id)} \), where id is the column in the data that contains and id of the subjects to be paired.

\( \mu \) numeric indicating the reference mean for single sample effect size.

na.rm logical indicating whether \( \text{NA} \)s should be removed before computation; if \( \text{paired}=\text{TRUE} \) then all incomplete pairs are removed.

... further arguments to be passed to or from methods.

Details

When \( f \) in the default version is a factor or a character, it must have two values and it identifies the two groups to be compared. Otherwise (e.g. \( f \) is numeric), it is considered as a sample to be compare to \( d \).

In the formula version, \( f \) is expected to be a factor, if that is not the case it is coerced to a factor and a warning is issued.
The function computes the value of Cohen’s d statistics (Cohen 1988). If required (hedges.correction==TRUE) the Hedges g statistics is computed instead (Hedges and Holkin, 1985).

When `paired` is set, the effect size is computed using the approach suggested in (Gibbons et al. 1993). In particular a correction to take into consideration the correlation of the two samples is applied (see Borenstein et al., 2009)

It is possible to perform a single sample effect size estimation either using a formula `~x` or passing `f=NA`.

The computation of the CI requires the use of non-central Student-t distributions that are used when `noncentral==TRUE`; otherwise a central distribution is used.

Also a quantification of the effect size magnitude is performed using the thresholds define in Cohen (1992). The magnitude is assessed using the thresholds provided in (Cohen 1992), i.e. \(|d|<0.2\) "negligible", \(|d|<0.5\) "small", \(|d|<0.8\) "medium", otherwise "large"

The variance of the d is computed using the conversion formula reported at page 238 of Cooper et al. (2009):

\[
S_d^2 = \left( \frac{n_1 + n_2}{n_1n_2} + \frac{d^2}{2df} \right) \left( \frac{n_1 + n_2}{df} \right)
\]

**Value**

A list of class `effsize` containing the following components:

- `estimate` the statistic estimate
- `conf.int` the confidence interval of the statistic
- `sd` the within-groups standard deviation
- `conf.level` the confidence level used to compute the confidence interval
- `magnitude` a qualitative assessment of the magnitude of effect size
- `method` the method used for computing the effect size, either "Cohen’s d" or "Hedges’ g"

**Author(s)**

Marco Torchiano [http://softeng.polito.it/torchiano/](http://softeng.polito.it/torchiano/)

**References**


Cooper, Hedges, and Valentin (2009). The Handbook of Research Synthesis and Meta-Analysis


See Also

cliff.delta, VD.A.print.effsize

Examples

treatment = rnorm(100,mean=10)
control = rnorm(100,mean=12)
d = c(treatment,control))
f = rep(c("Treatment","Control"),each=100)
## compute Cohen's d
## treatment and control
cohen.d(treatment,control)
## data and factor
cohen.d(d,f)
## formula interface
cohen.d(d ~ f)
## compute Hedges' g
cohen.d(d,f,hedges.correction=TRUE)

print.effsize

Prints effect size

Description

Prints the results of an effect size computation

Usage

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

Arguments

x the effect size result

... further parameters are currently ignored

Details

Shows the estimate value and, when available, the confidence interval.
Note
This is still work in progress..

Author(s)
Marco Torchiano http://softeng.polito.it/torchiano/

References
See the main function cliff.delta.

See Also
cliff.delta cohen.d

---

### VD.A

**Vargha and Delaney A measure**

**Description**
Computes the Vargha and Delaney A effect size measure.

**Usage**

```r
VD.A(d, ...)
```

**Arguments**

- `d` a numeric vector giving either the data values (if `f` is a factor) or the treatment group values (if `f` is a numeric vector)
- `f` either a factor with two levels or a numeric vector of values
- `formula` a formula of the form `y ~ f`, where `y` is a numeric variable giving the data values and `f` a factor with two levels giving the corresponding group
- `data` an optional matrix or data frame containing the variables in the formula `formula`. By default the variables are taken from `environment(formula)`.
- `...` further arguments to be passed to or from methods.

**Details**
The function computes the Vargha and Delaney A effect size measure (Vargha and Delaney, 2000).
Value

A list of class effsize containing the following components:

- estimate: the A statistics estimate
- magnitude: a qualitative assessment of the magnitude of effect size
- method: the method used, i.e. "Vargha and Delaney A"

Author(s)

Marco Torchiano [http://softeng.polito.it/torchiano/](http://softeng.polito.it/torchiano/)

References


See Also

cliff.delta, cohen.d, print.effsize

Examples

treatment = rnorm(100,mean=10)
control = rnorm(100,mean=12)
d = c(treatment,control)
f = rep(c("Treatment","Control"),each=100)
## compute Vargha and Delaney A
## treatment and control
VD.A(treatment,control)
## data and factor
VD.A(d,f)
## formula interface
VD.A(d ~ f)
Index

+Topic Cliff
  cliff.delta.3
  effsize-package.2
+Topic Cohen
  cohen.d.5
  effsize-package.2
+Topic Hedges
  cohen.d.5
  effsize-package.2
+Topic Vargha and Delaney
  VD.A.9
+Topic effect size
  cliff.delta.3
  cohen.d.5
  effsize-package.2
  VD.A.9
+Topic package
  effsize-package.2

cliff.delta.2,3,8–10
cohen.d.2,5,5,9,10

effsize(effsize-package).2
effsize-package.2

print.effsize.5,8,8,10
VD.A.2,8,9