Package ‘pwr’

March 25, 2017

Version 1.2-1
Date 2017-03-25
Title Basic Functions for Power Analysis
Description Power analysis functions along the lines of Cohen (1988).
Imports stats, graphics
Suggests ggplot2, scales, knitr, rmarkdown
License GPL (>= 3)
URL https://github.com/heliosdrm/pwr
VignetteBuilder knitr
RoxygenNote 6.0.1
NeedsCompilation no
Author Stephane Champely [aut],
  Claus Ekstrom [ctb],
  Peter Dalgaard [ctb],
  Jeffrey Gill [ctb],
  Stephan Weibelzahl [ctb],
  Aditya Anandkumar [ctb],
  Clay Ford [ctb],
  Robert Volcic [ctb],
  Helios De Rosario [cre]
Maintainer Helios De Rosario <helios.derosario@gmail.com>
Repository CRAN
Date/Publication 2017-03-25 10:58:41 UTC

R topics documented:

  pwr-package ................................................................. 2
  cohen.ES ................................................................. 3
  ES.h ................................................................. 4
  ES.w1 ................................................................. 5
  ES.w2 ................................................................. 6
pwr-package

Description

Power calculations along the lines of Cohen (1988) using in particular the same notations for effect sizes. Examples from the book are given.

Details

Package: pwr
Type: Package
Version: 1.2-1
Date: 2017-03-25
License: GPL (>= 3)

This package contains functions for basic power calculations using effect sizes and notations from Cohen (1988): pwr.p.test: test for one proportion (ES=h) pwr.2p.test: test for two proportions (ES=h) pwr.2p2n.test: test for two proportions (ES=h, unequal sample sizes) pwr.t.test: one sample t tests for means (ES=d) pwr.t2n.test: two samples (different sizes) t test for means (ES=d) pwr.anova.test: test for one-way balanced anova (ES=f) pwr.r.test: correlation test (ES=r) pwr.chisq.test: chi-squared test (ES=w) pwr.f2.test: test for the general linear model (ES=f2) ES.h: computing effect size h for proportions tests ES.w1: computing effect size w for the goodness of fit chi-squared test ES.w2: computing effect size w for the association chi-squared test cohen.ES: computing effect sizes for all the previous tests corresponding to conventional effect sizes (small, medium, large)

Author(s)

Stephane Champely, based on previous works by Claus Ekstrom and Peter Dalgaard, with contributions of Jeffrey Gill, Stephan Weibelzahl, Clay Ford, Aditya Anandkumar and Robert Volcic.

Maintainer: Helios De Rosario-Martinez <helios.derosario@gmail.com>
cohen.ES

References

See Also
power.t.test, power.prop.test, power.anova.test

Examples
## Exercise 8.1 p. 357 from Cohen (1988)
pwr.anova.test(f=0.28, k=4, n=20, sig.level=0.05)

## Exercise 6.1 p. 198 from Cohen (1988)
pwr.2p.test(h=0.3, n=80, sig.level=0.05, alternative="greater")

## Exercise 7.3 p. 251
pwr.chisq.test(w=0.346, df=(2-1)*(3-1), N=140, sig.level=0.01)

## Exercise 6.5 p. 203 from Cohen (1988)
pwr.p.test(h=0.2, n=60, sig.level=0.05, alternative="two.sided")

<table>
<thead>
<tr>
<th>cohen.ES</th>
<th>Conventional effects size</th>
</tr>
</thead>
</table>

Description
Give the conventional effect size (small, medium, large) for the tests available in this package

Usage
cohen.ES(test = c("p", "t", "r", "anov", "chisq", "f2"),
size = c("small", "medium", "large"))

Arguments
- test: The statistical test of interest
- size: The ES: small, medium, or large?

Value
The corresponding effect size

Author(s)
Stephane CHAMPELY
References

Examples

```r
## medium effect size for the correlation test
cohen.ES(test="r", size="medium")

## sample size for a medium size effect in the two-sided correlation test
## using the conventional power of 0.80
pwr.r.test(r=cohen.ES(test="r",size="medium")$effect.size, 
           power=0.80, sig.level=0.05, alternative="two.sided")
```

**ES.h**

Effect size calculation for proportions

**Description**
Compute effect size h for two proportions

**Usage**

```
ES.h(p1, p2)
```

**Arguments**

- `p1`: First proportion
- `p2`: Second proportion

**Details**
The effect size is $2 \cdot \arcsin(\sqrt{p1}) - 2 \cdot \arcsin(\sqrt{p2})$

**Value**
The corresponding effect size

**Author(s)**
Stephane CHAMPELY

**References**
See Also

pwr.p.test, pwr.2p.test, pwr.2p2n.test, power.prop.test

Examples

```r
## Exercise 6.5 p. 203 from Cohen
h<-ES.w1(0.5,0.4)
h
pwr.p.test(h=h,n=60,sig.level=0.05,alternative="two.sided")
```

ES.w1  
*Effect size calculation in the chi-squared test for goodness of fit*

Description

Compute effect size w for two sets of k probabilities P0 (null hypothesis) and P1 (alternative hypothesis)

Usage

```r
ES.w1(P0, P1)
```

Arguments

- `P0`: First set of k probabilities (null hypothesis)
- `P1`: Second set of k probabilities (alternative hypothesis)

Value

The corresponding effect size w

Author(s)

Stephane CHAMPELY

References


See Also

pwr.chisq.test
Examples

```r
## Exercise 7.1 p. 249 from Cohen
p0<-rep(1/4,4)
p1<-c(0.375,rep((1-0.375)/3,3))
es.w2(P0,P1)
pwr.chisq.test(w=ES.w2(P0,P1),N=100,df=(4-1))
```

Description

Compute effect size w for a two-way probability table corresponding to the alternative hypothesis in the chi-squared test of association in two-way contingency tables.

Usage

```r
ES.w2(P)
```

Arguments

- **P**
  - A two-way probability table (alternative hypothesis)

Value

The corresponding effect size w

Author(s)

Stephane CHAMPELY

References


See Also

pwr.chisq.test

Examples

```r
prob<-matrix(c(0.225,0.125,0.125,0.125,0.125,0.16,0.16,0.04,0.04),nrow=2,byrow=TRUE)
prob
ES.w2(prob)
pwr.chisq.test(w=ES.w2(prob),df=(2-1)*(4-1),N=200)
```
plot.power.htest

Plot diagram of sample size vs. test power

Description
Plot a diagram to illustrate the relationship of sample size and test power for a given set of parameters.

Usage

```r
## S3 method for class 'power.htest'
plot(x, ...)
```

Arguments

- `x` object of class power.htest usually created by one of the power calculation functions, e.g., `pwr.t.test()`
- `...` Arguments to be passed to `ggplot` including xlab and ylab

Details
Power calculations for the following tests are supported: t-test (`pwr.t.test()`, `pwr.t2n.test()`), chi squared test (`pwr.chisq.test()`), one-way ANOVA (`pwr.anova.test()`), standard normal distribution (`pwr.norm.test()`), pearson correlation (`pwr.r.test()`), proportions (`pwr.p.test()`, `pwr.2p.test()`, `pwr.2p2n.test()`)

Value
These functions are invoked for their side effect of drawing on the active graphics device.

Note
By default it attempts to use the plotting tools of `ggplot2` and `scales`. If they are not installed, it will use the basic R plotting tools.

Author(s)
Stephan Weibelzahl <weibelzahl@pfh.de>

See Also

- `pwr.t.test`, `pwr.p.test`, `pwr.2p.test`, `pwr.2p2n.test`, `pwr.r.test`, `pwr.chisq.test`, `pwr.anova.test`, `pwr.t2n.test`
pwr.2p.test

Examples

```r
## Two-sample t-test
p.t.two <- pwr.t.test(d=0.3, power=0.8, type="two.sample", alternative="two.sided")
plot(p.t.two)
plot(p.t.two, xlab="sample size per group")
```

pwr.2p.test

Power calculation for two proportions (same sample sizes)

Description

Compute power of test, or determine parameters to obtain target power (similar to power.prop.test).

Usage

```r
pwr.2p.test(h = NULL, n = NULL, sig.level = 0.05, power = NULL, alternative = c("two.sided","less","greater"))
```

Arguments

- `h`: Effect size
- `n`: Number of observations (per sample)
- `sig.level`: Significance level (Type I error probability)
- `power`: Power of test (1 minus Type II error probability)
- `alternative`: A character string specifying the alternative hypothesis, must be one of "two.sided" (default), "greater" or "less"

Details

Exactly one of the parameters 'h', 'n', 'power' and 'sig.level' must be passed as NULL, and that parameter is determined from the others. Notice that the last one has non-NULL default so NULL must be explicitly passed if you want to compute it.

Value

Object of class "power.htest", a list of the arguments (including the computed one) augmented with 'method' and 'note' elements.

Note

'uniform' is used to solve power equation for unknowns, so you may see errors from it, notably about inability to bracket the root when invalid arguments are given.
pwr.2p2n.test

Author(s)
Stephane Champely <champely@univ-lyon1.fr> but this is a mere copy of Peter Dalgaard work (power.t.test)

References

See Also
ES.h, pwr.2p2n.test, power.prop.test

Examples
## Exercise 6.1 p. 198 from Cohen (1988)
pwr.2p.test(h=0.3,n=80,sig.level=0.05,alternative="greater")

pwr.2p2n.test                  Power calculation for two proportions (different sample sizes)

Description
Compute power of test, or determine parameters to obtain target power.

Usage
pwr.2p2n.test(h = NULL, n1 = NULL, n2 = NULL, sig.level = 0.05, power = NULL,
              alternative = c("two.sided", "less","greater"))

Arguments
- h: Effect size
- n1: Number of observations in the first sample
- n2: Number of observations in the second sample
- sig.level: Significance level (Type I error probability)
- power: Power of test (1 minus Type II error probability)
- alternative: a character string specifying the alternative hypothesis, must be one of "two.sided" (default), "greater" or "less"

Details
Exactly one of the parameters 'h', 'n1', 'n2', 'power' and 'sig.level' must be passed as NULL, and that parameter is determined from the others. Notice that the last one has non-NULL default so NULL must be explicitly passed if you want to compute it.
Value

Object of class "power.htest", a list of the arguments (including the computed one) augmented with 'method' and 'note' elements.

Note

'unroot' is used to solve power equation for unknowns, so you may see errors from it, notably about inability to bracket the root when invalid arguments are given.

Author(s)

Stephane Champely <champely@univ-lyon1.fr> but this is a mere copy of Peter Dalgaard work (power.t.test)

References


See Also

ES.h, pwr.2p.test, power.prop.test

Examples

## Exercise 6.3 P. 200 from Cohen (1988)
pwr.2p2n.test(h=0.30,n1=80,n2=245,sig.level=0.05,alternative="greater")

## Exercise 6.7 p. 207 from Cohen (1988)
pwr.2p2n.test(h=0.20,n1=1600,power=0.9,sig.level=0.01,alternative="two.sided")

pwr.anova.test

Power calculations for balanced one-way analysis of variance tests

Description

Compute power of test or determine parameters to obtain target power (same as power.anova.test).

Usage

pwr.anova.test(k = NULL, n = NULL, f = NULL, sig.level = 0.05, power = NULL)

Arguments

k
  Number of groups
n
  Number of observations (per group)
f
  Effect size

  Significance level (Type I error probability)

  Power of test (1 minus Type II error probability)
Compute power of test or determine parameters to obtain target power (same as power.anova.test).

Usage

pwr.chisq.test(w = NULL, N = NULL, df = NULL, sig.level = 0.05, power = NULL)
Arguments

- **w**: Effect size
- **N**: Total number of observations
- **df**: degree of freedom (depends on the chosen test)
- **sig.level**: Significance level (Type I error probability)
- **power**: Power of test (1 minus Type II error probability)

Details

Exactly one of the parameters 'w','N','power' and 'sig.level' must be passed as NULL, and that parameter is determined from the others. Notice that the last one has non-NULL default so NULL must be explicitly passed if you want to compute it.

Value

Object of class "power.htest", a list of the arguments (including the computed one) augmented with 'method' and 'note' elements.

Note

'unroot' is used to solve power equation for unknowns, so you may see errors from it, notably about inability to bracket the root when invalid arguments are given.

Author(s)

Stephane Champely <champely@univ-lyon1.fr> but this is a mere copy of Peter Dalgaard work (power.t.test)

References


See Also

ES.w1,ES.w2

Examples

```r
## Exercise 7.1 P. 249 from Cohen (1988)
pwr.chisq.test(w=0.289,df=(4-1),N=100,sig.level=0.05)

## Exercise 7.3 p. 251
pwr.chisq.test(w=0.346,df=(2-1)*(3-1),N=140,sig.level=0.01)

## Exercise 7.8 p. 270
pwr.chisq.test(w=0.1,df=(5-1)*(6-1),power=0.80,sig.level=0.05)
```
Power calculations for the general linear model

Description

Compute power of test or determine parameters to obtain target power (same as power.anova.test).

Usage

```r
pwr.f2.test(u = NULL, v = NULL, f2 = NULL, sig.level = 0.05, power = NULL)
```

Arguments

- `u` degrees of freedom for numerator
- `v` degrees of freedom for denominator
- `f2` effect size
- `sig.level` Significance level (Type I error probability)
- `power` Power of test (1 minus Type II error probability)

Details

Exactly one of the parameters `'u'`, `'v'`, `'f2'`, `'power'` and `'sig.level'` must be passed as NULL, and that parameter is determined from the others. Notice that the last one has non-NULL default so NULL must be explicitly passed if you want to compute it.

Value

Object of class `"power.htest"`, a list of the arguments (including the computed one) augmented with `method` and `note` elements.

Note

`uniroot` is used to solve power equation for unknowns, so you may see errors from it, notably about inability to bracket the root when invalid arguments are given.

Author(s)

Stephane Champely <champely@univ-lyon1.fr> but this is a mere copy of Peter Dalgaard work (power.t.test)

References

Examples

```r
## Exercise 9.1 P. 424 from Cohen (1988)
pwr.f2.test(u=5,v=89,f2=0.1/(1-0.1),sig.level=0.05)
```

**pwr.norm.test**

*Power calculations for the mean of a normal distribution (known variance)*

**Description**

Compute power of test or determine parameters to obtain target power (same as power.anova.test).

**Usage**

```r
pwr.norm.test(d = NULL, n = NULL, sig.level = 0.05, power = NULL, alternative = c("two.sided", "less", "greater"))
```

**Arguments**

- `d`  
  Effect size $d=\mu-\mu_0$
- `n`  
  Number of observations
- `sig.level`  
  Significance level (Type I error probability)
- `power`  
  Power of test (1 minus Type II error probability)
- `alternative`  
  a character string specifying the alternative hypothesis, must be one of "two.sided", "less", or "greater"

**Details**

Exactly one of the parameters 'd','n','power' and 'sig.level' must be passed as NULL, and that parameter is determined from the others. Notice that the last one has non-NULL default so NULL must be explicitly passed if you want to compute it.

**Value**

Object of class "power.htest", a list of the arguments (including the computed one) augmented with 'method' and 'note' elements.

**Note**

'unroot' is used to solve power equation for unknowns, so you may see errors from it, notably about inability to bracket the root when invalid arguments are given.

**Author(s)**

Stephane Champely <champely@univ-lyon1.fr> but this is a mere copy of Peter Dalgaard work (power.t.test)
### Examples

```r
## Power at mu=105 for H0:mu=100 vs. H1:mu>100 (sigma=15) 20 obs. (alpha=0.05)
sigma<-15
c<-100
mu<-105
d<-c(mu-c)/sigma
pwr.norm.test(d=d,n=20,sig.level=0.05,alternative="greater")

## Sample size of the test for power=0.80
pwr.norm.test(d=d,power=0.8,sig.level=0.05,alternative="greater")

## Power function of the same test
mu<-seq(95,125,1=100)
d<-c(mu-c)/sigma
plot(d,pwr.norm.test(d=d,n=20,sig.level=0.05,alternative="greater")$power,
    type="l",ylim=c(0,1))
abline(h=0.05)
abline(h=0.80)

## Power function for the two-sided alternative
plot(d,pwr.norm.test(d=d,n=20,sig.level=0.05,alternative="two.sided")$power,
    type="l",ylim=c(0,1))
abline(h=0.05)
abline(h=0.80)
```

### Description

`pwr.p.test` is a function for power calculations for proportion tests (one sample).

**Description**

Compute power of test or determine parameters to obtain target power (same as `power.anova.test`).

**Usage**

```r
pwr.p.test(h = NULL, n = NULL, sig.level = 0.05, power = NULL,
           alternative = c("two.sided", "less", "greater"))
```

**Arguments**

- `h` : Effect size
- `n` : Number of observations
- `sig.level` : Significance level (Type I error probability)
power.test

power
alternative

Power of test (1 minus Type II error probability)
a character string specifying the alternative hypothesis, must be one of "two.sided"
(default), "greater" or "less"

Details

These calculations use arcsine transformation of the proportion (see Cohen (1988))

Exactly one of the parameters 'h', 'n', 'power' and 'sig.level' must be passed as NULL, and that parameter is determined from the others. Notice that the last one has non-NULL default so NULL must be explicitly passed if you want to compute it.

Value

Object of class ""power.htest"", a list of the arguments (including the computed one) augmented with 'method' and 'note' elements.

Note

'uniroot' is used to solve power equation for unknowns, so you may see errors from it, notably about inability to bracket the root when invalid arguments are given.

Author(s)

Stephane Champely <champely@univ-lyon1.fr> but this is a mere copy of Peter Dalgaard work (power.t.test)

References


See Also

ES.h

Examples

## Exercise 6.5 p. 203 from Cohen

```r
h<ES.h(0.5,0.4)

h

pwr.p.test(h=h,n=60,sig.level=0.05,alternative="two.sided")
```

## Exercise 6.8 p. 208

```r
pwr.p.test(h=0.2,power=0.95,sig.level=0.05,alternative="two.sided")
```
pwr.r.test

**Description**

Compute power of test or determine parameters to obtain target power (same as power.anova.test).

**Usage**

```r
pwr.r.test(n = NULL, r = NULL, sig.level = 0.05, power = NULL,
          alternative = c("two.sided", "less","greater"))
```

**Arguments**

- `n`: Number of observations
- `r`: Linear correlation coefficient
- `sig.level`: Significance level (Type I error probability)
- `power`: Power of test (1 minus Type II error probability)
- `alternative`: a character string specifying the alternative hypothesis, must be one of "two.sided" (default), "greater" or "less"

**Details**

These calculations use the $Z'$ transformation of correlation coefficient: $Z'=\arctanh(r)+r/(2*(n-1))$ (see Cohen (1988) p.546).

Exactly one of the parameters 'r', 'n', 'power' and 'sig.level' must be passed as NULL, and that parameter is determined from the others. Notice that the last one has non-NULL default so NULL must be explicitly passed if you want to compute it.

**Value**

Object of class "power.htest", a list of the arguments (including the computed one) augmented with 'method' and 'note' elements.

**Note**

'unroot' is used to solve power equation for unknowns, so you may see errors from it, notably about inability to bracket the root when invalid arguments are given.

**Author(s)**

Stephane Champely <champely@univ-lyon1.fr> but this is a mere copy of Peter Dalgaard work (power.t.test)
References


Examples

```r
## Exercise 3.1 p. 96 from Cohen (1988)
pwr.r.test(r=0.3,n=50,sig.level=0.05,alternative="two.sided")
pwr.r.test(r=0.3,n=50,sig.level=0.05,alternative="greater")

## Exercise 3.4 p. 208
pwr.r.test(r=0.3,power=0.80,sig.level=0.05,alternative="two.sided")
pwr.r.test(r=0.5,power=0.80,sig.level=0.05,alternative="two.sided")
pwr.r.test(r=0.1,power=0.80,sig.level=0.05,alternative="two.sided")
```

### pwr.t.test

**Power calculations for t-tests of means (one sample, two samples and paired samples)**

#### Description

Compute power of tests or determine parameters to obtain target power (similar to `power.t.test`).

#### Usage

```r
pwr.t.test(n = NULL, d = NULL, sig.level = 0.05, power = NULL,
            type = c("two.sample", "one.sample", "paired"),
            alternative = c("two.sided", "less", "greater"))
```

#### Arguments

- **n** Number of observations (per sample)
- **d** Effect size
- **sig.level** Significance level (Type I error probability)
- **power** Power of test (1 minus Type II error probability)
- **type** Type of t test: one- two- or paired-samples
- **alternative** a character string specifying the alternative hypothesis, must be one of "two.sided" (default), "greater" or "less"

#### Details

Exactly one of the parameters 'd','n','power' and 'sig.level' must be passed as NULL, and that parameter is determined from the others. Notice that the last one has non-NULL default so NULL must be explicitly passed if you want to compute it.
Value

Object of class ""power.htest"", a list of the arguments (including the computed one) augmented with 'method' and 'note' elements.

Note

'unioot' is used to solve power equation for unknowns, so you may see errors from it, notably about inability to bracket the root when invalid arguments are given.

Author(s)

Stephane Champely <champely@univ-lyon1.fr> but this is a mere copy of Peter Dalgaard work (power.t.test)

References


See Also

power.prop.test

Examples

```r
## One sample (power)
## Exercise 2.5 p. 47 from Cohen (1988)
pwr.t.test(d=0.2, n=60, sig.level=0.10, type="one.sample", alternative="two.sided")

## Paired samples (power)
## Exercise p. 50 from Cohen (1988)
d<-8/(16*sqrt(2^2*(1-0.06)))
pwr.t.test(d, n=40, sig.level=0.05, type="paired", alternative="two.sided")

## Two independent samples (power)
## Exercise 2.1 p. 40 from Cohen (1988)
d<-2/2.8
pwr.t.test(d, n=30, sig.level=0.05, type="two.sample", alternative="two.sided")

## Two independent samples (sample size)
## Exercise 2.10 p. 59
pwr.t.test(d=0.3, power=0.75, sig.level=0.05, type="two.sample", alternative="greater")
```
pwr.t2n.test

Power calculations for two samples (different sizes) t-tests of means

Description

Compute power of tests or determine parameters to obtain target power (similar to as power.t.test).

Usage

pwr.t2n.test(n1 = NULL, n2= NULL, d = NULL, sig.level = 0.05, power = NULL, alternative = c("two.sided", "less","greater"))

Arguments

n1 Number of observations in the first sample
n2 Number of observations in the second sample
d Effect size
sig.level Significance level (Type I error probability)
power Power of test (1 minus Type II error probability)
alternative a character string specifying the alternative hypothesis, must be one of "two.sided" (default), "greater" or "less"

Details

Exactly one of the parameters 'd','n1','n2',power' and 'sig.level' must be passed as NULL, and that parameter is determined from the others. Notice that the last one has non-NULL default so NULL must be explicitly passed if you want to compute it.

Value

Object of class "power.htest", a list of the arguments (including the computed one) augmented with 'method' and 'note' elements.

Note

'unroot' is used to solve power equation for unknowns, so you may see errors from it, notably about inability to bracket the root when invalid arguments are given.

Author(s)

Stephane Champely <champely@univ-lyon1.fr> but this is a mere copy of Peter Dalgaard work (power.t.test)
References


Examples

```r
## Exercise 2.3 p. 437 from Cohen (1988)
pwr.t2n.test(d=0.6,n1=90,n2=60,alternative="greater")
```
Index

*Topic htest
  cohen.ES, 3
  ES.h, 4
  ES.w1, 5
  ES.w2, 6
  plot.power.htest, 7
  pwr-package, 2
  pwr.2p.test, 8
  pwr.2p2n.test, 9
  pwr.anova.test, 10
  pwr.chisq.test, 11
  pwr.f2.test, 13
  pwr.norm.test, 14
  pwr.p.test, 15
  pwr.r.test, 17
  pwr.t.test, 18
  pwr.t2n.test, 20

*Topic package
  pwr-package, 2

cohen.ES, 3

ES.h, 4
ES.w1, 5
ES.w2, 6

plot.power.htest, 7
pwr (pwr-package), 2
pwr-package, 2
pwr.2p.test, 7, 8
pwr.2p2n.test, 7, 9
pwr.anova.test, 7, 10
pwr.chisq.test, 7, 11
pwr.f2.test, 13
pwr.norm.test, 14
pwr.p.test, 7, 15
pwr.r.test, 7, 17
pwr.t.test, 7, 18
pwr.t2n.test, 7, 20