Package 'PowerTOST'

April 12, 2018

Type Package

Title Power and Sample Size Based on Two One-Sided t-Tests (TOST) for (Bio)Equivalence Studies

Version 1.4-7

Date 2018-04-12

Maintainer Detlew Labes <DetlewLabes@gmx.de>

Description Contains functions to calculate power and sample size for various study designs used for bioequivalence studies.
See function known.designs() for study designs covered.
Moreover the package contains functions for power and sample size based on 'expected' power in case of uncertain (estimated) variability and/or uncertain theta0.

Added are functions for the power and sample size for the ratio of two means with normally distributed data on the original scale (based on Fieller's confidence ('fiducial') interval).

Contains further functions for power and sample size calculations based on non-inferiority t-test. This is not a TOST procedure but eventually useful if the question of 'non-superiority' must be evaluated.
The power and sample size calculations based on non-inferiority test may also performed via 'expected' power in case of uncertain (estimated) variability and/or uncertain theta0.

Contains functions power.scABEL() and sampleN.scABEL() to calculate power and sample size for the BE decision via scaled (widened) BE acceptance limits (EMA recommended) based on simulations.
Contains also functions scABEL.ad() and sampleN.scABEL.ad() to iteratively adjust alpha in order to maintain the overall consumer risk in ABEL studies and adapt the sample size for the loss in power.
Contains further functions power.RSABE() and sampleN.RSABE() to calculate power and sample size for the BE decision via reference scaled ABE criterion according to the FDA procedure based on simulations.
Contains further functions power.NTIDFDA() and sampleN.NTIDFDA() to calculate power and sample size for the BE decision via the FDA procedure for NTID's

based on simulations.
Contains further functions power.HVNTID() and sampleN.HVNTID() to calculate
power and sample size for the BE decision via the FDA procedure for
highly variable NTID's (see FDA Dabigatran / rivaroxaban guidances)

Contains functions for power analysis of a sample size plan for ABE
(pa.ABE()), scaled ABE (pa.scABE()) and scaled ABE for NTID's (pa.NTIDFDA())
analysing power if deviating from assumptions of the plan.

Contains further functions for power calculations / sample size estimation
for dose proportionality studies using the Power model.

Imports mvtnorm, stats, utils, graphics, grDevices, cubature (>= 1.3-6), TeachingDemos

Suggests crossdes

ByteCompile yes

LazyData true

URL http://github.com/Detlew/PowerTOST

BugReports http://github.com/Detlew/PowerTOST/issues

License GPL (>= 2)

NeedsCompilation no

Author Detlew Labes [aut, cre],
Helmut Schuetz [aut],
Benjamin Lang [aut]

Repository CRAN

Date/Publication 2018-04-12 19:43:08 UTC

R topics documented:

`bib.CL` *Design matrices of period balanced incomplete block designs*

Description

This function returns the 'design' matrix of incomplete block designs described by Chow & Liu. The design matrices were recoded 1=R, 2=T1, 3=T2, ...

Usage

```
bib.CL(trt, p)
```

Arguments

trt	Number of treatments (3 to 5).
p	Number of periods (2 to trt-1).

Value

Matrix containing the sequences in rows and periods in columns.
The entry (i, j) of the matrix corresponds to the treatment or dose (index) a subject within i-th sequence gets in the j-th period.

Author(s)

D. Labes

References

Chow SC, Liu JP. *Design and Analysis of Bioavailability and Bioequivalence Studies.*
Boca Raton: CRC Press; 3^{rd} edition 2009. Chapter 2.6.

Examples

```
# 4 treatments/doses, 3 periods
bib.CL(4, 3)
# gives 4 sequences
# to see this in Chow & Liu's coding
tmt <- c("R", "T1", "T2", "T3")
matrix(tmt[bib.CL(4, 3)], ncol=3)
```

CI.BE *1–2*alpha confidence interval given point estimate, CV, and n*

Description

Utility function to calculate the $1 - 2\alpha$ CI given point estimate, CV, and n for the various designs covered in this package.

Usage

```
CI.BE(alpha = 0.05, pe, CV, n, design = "2x2", robust = FALSE)
```

Arguments

alpha	Type I error probability, significance level. Defaults to 0.05.
pe	Point estimate (GMR).
CV	Coefficient of variation of error variability as ratio (not percent).
n	Total number of subjects if a scalar is given. Number of subjects in (sequence) groups if given as vector.
design	Character string describing the study' design. See known.designs() for designs covered in this package.
robust	Defaults to FALSE. Setting to TRUE will use the degrees of freedom according to the 'robust' evaluation (aka Senn's basic estimator). These df are calculated as n-seq. See known.designs()$df2 for designs covered in this package.

Value

Returns the $1 - 2\alpha$ confidence interval.
Returns a vector with named elements lower, upper if arguments pe and CV are scalars, else a matrix with columns lower, upper is returned.

Note

The function assumes an evaluation using log-transformed data.
The function assumes equal variances in case of design="parallel" and the higher order crossover designs.
The implemented formula covers balanced and unbalanced designs.

Whether the function vectorizes properly is not thoroughly tested.

Author(s)

D. Labes

Examples

```
# 90% confidence interval for the 2x2 crossover
# n(total) = 24
CI.BE(pe=0.95, CV=0.3, n=24)
# should give
#     lower     upper
# 0.8213465 1.0988055
# same total number but unequal sequences
CI.BE(pe=0.95, CV=0.3, n=c(13, 11))
#     lower     upper
# 0.8209294 1.0993637
```

`CI.RatioF` *1–2*alpha Fieller CI given point est., CV (,CVb) and n*

Description

Utility function to calculate the $1 - 2\alpha$ Fieller confidence interval given the point estimate, CV (, CVb), and n for the parallel group and 2×2 crossover.

Usage

```
CI.RatioF(alpha = 0.025, pe, CV, CVb, n, design = c("2x2", "parallel"))
```

Arguments

`alpha`	Type I error probability, aka significance level. Defaults here to 0.025 because this function is intended for studies with clinical endpoints.
`pe`	Point estimate of ratio T/R.
`CV`	Coefficient of variation as ratio (not percent). In case of `design="parallel"` this is the CV of the total variability, in case of `design="2x2"` the intra-subject CV.
`CVb`	CV of the between-subject variability. Only necessary for `design="2x2"`.
`n`	Total number of subjects if a scalar is given. Number of subjects in (sequence) groups if given as vector.
`design`	A character string describing the study design. `design="parallel"` or `design="2x2"` allowed for a parallel two-group design or a classical TR|RT crossover design.

Details

The CV(within) and CVb(etween) in case of `design="2x2"` are obtained via an appropriate ANOVA from the error term and from the difference `(MS(subject within sequence)-MS(error))/2`.

Value

Returns the $1 - 2\alpha$ confidence interval.

Note

The function assumes an evaluation using un-transformed data.
The function assumes equal variances in case of `design="parallel"`.
The formula implemented covers balanced and unbalanced designs.

Note that when the mean of the denominator of the ratio is close to zero, confidence intervals might be degenerated and are returned as `NA`. In such a case a warning is issued.

Whether the function vectorizes properly is not thoroughly tested.

This function is intended for studies with clinical endpoints. In such studies the 95% confidence intervals are usually used for equivalence testing. Therefore alpha defaults here to 0.025 (see EMA 2000).

Author(s)

D. Labes

References

Locke CS. *An exact confidence interval from untransformed data for the ratio of two formulation means.*
J Pharmacokin Biopharm. 1984;12(6):649–55.

Hauschke D, Steinijans VW, Pigeot I. *Bioequivalence Studies in Drug Development.*
Chichester: John Wiley; 2007. Chapter 10.

European Medicines Agency, Committee for Proprietary Medicinal Products.
Points to consider on switching between superiority and non-inferiority
London, 27 July 2000. CPMP/EWP/482/99

See Also

`CI.BE`, `power.RatioF`

Examples

```
# 95% Fieller CI for the 2x2 crossover
CI.RatioF(pe=1.05, CV=0.3, CVb=0.6, n=24)
```

`ct5.1+ct5.2+ct5.3+ct5.4.1`

Sample Size Tables for the Classical 2x2 Crossover Design

Description

These data.frames give sample size tables calculated with `sampleN.TOST()` for the 2×2 design.

Details

The data.frames can be accessed by their names.

ct5.1 is Table 5.1 from
Hauschke D, Steinijans V, Pigeot I. *Bioequivalence studies in Drug Development.*
John Wiley & Sons, Chichester (2007)
Multiplicative model, theta1=0.8, theta2=1.25 (1/theta1), exact

ct5.2 is Table 5.2 from the same source
Multiplicative model, theta1=0.75, theta2=1.3333 (1/theta1), exact

ct5.3 is Table 5.3 from the same source
Multiplicative model, theta1=0.9, theta2=1.1111 (1/theta1), exact

ct5.4.1 is Table 5.4.1 from
Chow SC, Liu JP. *Design and Analysis of Bioavailability and Bioequivalence Studies.*
Third edition, CRC Press, Chapman & Hall, Boca Raton (2009)
Additive model, theta1=–0.2, theta2=+0.2 (BE limits 0.80 – 1.20), exact

Note

Scripts for creation of these data.frames can be found in the \test sub-directory of the package.
Comparing the results of these scripts to the corresponding data.frames can be used for validation purposes.

Author(s)

PowerTOST

Examples

```
ct5.1
ct5.2
ct5.3
ct5.4.1
```

`ct9.6.2+ct9.6.6` *Sample Size Tables for the 2x2x3 Replicate Crossover Design*

Description

These data.frames give sample size tables calculated with `sampleN.TOST()` for the 2×2×3 replicate crossover design (2-treatment 2-sequence 3-period design.

Details

The data.frames can be accessed by their names.

ct9.6.2 is Table 9.6.2 from
Chow SC, Liu JP. *Design and Analysis of Bioavailability and Bioequivalence Studies.*
Third edition, CRC Press, Chapman & Hall, Boca Raton (2009)
Additive model, theta1=–0.2, theta2=+0.2 (BE limits 0.80 – 1.20),
approximate power via shifted non-central *t*-distribution.

ct9.6.6 is Table 9.6.6 from the same reference.
Multiplicative model, theta1=0.8, theta2=1.25 (1/theta1), power via shifted non-central *t*-distribution.
Attention! Chow and Liu's CV is se (standard error) of residuals.

Note

Scripts for creation of these data.frames can be found in the \test sub-directory of the package.
Comparing the results of these scripts to the corresponding data.frames can be used for validation purposes.

Author(s)

PowerTOST

Examples

```
ct9.6.2
ct9.6.6
```

`ct9.6.4+ct9.6.8` *Sample Size Tables for the 2x4x4 Replicate Crossover Design*

Description

These data.frames give sample size tables calculated with `sampleN.TOST()` for the 2×4×4 replicate crossover design (2-treatment 4-sequence 4-period design).

Details

The data.frames can be accessed by their names.

ct9.6.4 is Table 9.6.4 from
Chow SC, Liu JP. *Design and Analysis of Bioavailability and Bioequivalence Studies.*
Third edition, CRC Press, Chapman & Hall, Boca Raton (2009)
Additive model, theta1=–0.2, theta2=+0.2 (BE limits 0.80 – 1.20),
approximate power via shifted non-central *t*-distribution.

ct9.6.8 is Table 9.6.8 from the same reference.
Multiplicative model, theta1=0.8, theta2=1.25 (1/theta1), power via shifted non-central *t*-distribution.
Attention! Chow and Liu's CV in case of multiplicative model is se (standard error) of residuals.

Note

Scripts for creation of these data.frames can be found in the \test sub-directory of the package.
Comparing the results of these scripts to the corresponding data.frames can be used for validation purposes.

Author(s)

PowerTOST

Examples

```
ct9.6.4
ct9.6.8
```

`ctSJ.VIII.10+ctSJ.VIII.20+ctCW.III`

Sample Size Tables for the Parallel Group Design

Description

These data.frames give sample size tables calculated with `sampleN.TOST()` for the parallel group design (2 groups).

Details

The data.frames can be accessed by their names.

ctSJ.VIII.10 is Table VIII, column 'level of bioequivalence 10%' from
Julious SA. *Tutorial in Biostatistics. Sample sizes for clinical trials with Normal data.*
Stat Med. 2004;23(12):1921–86. doi: 10.1002/sim.1783

Multiplicative model, theta1=0.9, theta2=1.1111 (1/theta1), target power=90%,
power approximate via non-central *t*-distribution.
Attention! Julious gives sample size per group.

ctSJ.VIII.20 is Table VIII from the same source
column 'level of bioequivalence 20%'
Multiplicative model, theta1=0.8, theta2=1.25 (1/theta1), target power=90%,
power approximate via non-central *t*.

ctCW.III is Table III from
Chow SC, Wang H. *On Sample Size Calculation in Bioequivalence Trials*
J Pharmacokinet Pharmacodyn. 2001. 28(2):155–69.
Additive model, theta1=–0.2, theta2=+0.2 (BE limits 0.80 – 1.20), exact.

Seems the last reference is not very reliable (compare to the Table in the paper).

Note

Scripts for creation of these data.frames can be found in the \test sub-directory of the package.
Comparing the results of these scripts to the corresponding data.frames can be used for validation purposes.

Author(s)

PowerTOST

Examples

```
ctSJ.VIII.10
ctSJ.VIII.20
ctCW.III
```

CV2se+se2CV+CV2mse+mse2CV

Helper functions

Description

Calculates the standard error or the mean squared error from a given CV and vice versa for log-normal data.

Usage

```
CV2se(CV)
se2CV(se)
CV2mse(CV)
mse2CV(mse)
```

Arguments

`CV`	coefficient of variation
`se`	standard error
`mse`	mean squared error

Value

Returns `se = sqrt(log(CV^2+1))`
or `CV = sqrt(exp(se*se)-1)`
or `mse = log(CV^2+1)`
or `CV = sqrt(exp(mse)-1)`

Note

These functions were originally intended for internal use only.
But may be useful for others.

Author(s)

D. Labes

Examples

```
# these functions are one liners:
CV2se <- function(CV) return(sqrt(log(1.0 + CV^2)))
se2CV <- function(se) return(sqrt(exp(se*se)-1))

CV2se(0.3)
# should give: [1] 0.2935604

se2CV(0.2935604)
#[1] 0.3
```

CVCL *Confidence limits of a CV for log-normal data*

Description

The function calculates the 1-alpha confidence limits (either 1-sided or 2-sided) via the chi-squared distribution of the error variance the CV is based on.

Usage

```
CVCL(CV, df, side = c("upper", "lower", "2-sided"), alpha = 0.05)
```

Arguments

CV	Coefficient of variation
df	degrees of freedom of the CV (error variance)
side	Side(s) to calculate the confidence limits for
alpha	Type I error probability, aka significance level

Value

Numeric vector of the confidence limits named as 'lower CL' and 'upper CL'.
In case of the one-sided upper confidence limit the 'lower CL' is = 0.
In case of the one-sided lower confidence limit the 'upper CL' is = Inf.

Author(s)

D. Labes

Examples

```
# upper one-sided 95% CL of a CV=0.3
# from a study with df=22 (f.i. a 2x2 crossover with n=24)
# side="upper" is standard if not explicitly given
CVCL(0.3, df=22)
# should give:
# lower CL  upper CL
#0.0000000 0.4075525
```

`CVfromCI` *CV from a given Confidence interval*

Description

Calculates the CV (coefficient of variation) from a known confidence interval of a BE study. Useful if no CV but the 90% CI was given in literature.

Usage

```
CVfromCI(pe, lower, upper, n, design = "2x2", alpha = 0.05, robust = FALSE)
CI2CV(pe, lower, upper, n, design = "2x2", alpha = 0.05, robust = FALSE)
```

Arguments

`pe`	Point estimate of the BE ratio. The `pe` may be missing. In that case it will be calculated as geometric mean of `lower` and `upper`.
`lower`	Lower confidence limit of the BE ratio.
`upper`	Upper confidence limit of the BE ratio.
`n`	Total number of subjects under study if given as scalar. Number of subjects in (sequence) groups if given as vector.
`design`	Character string describing the study design. See `known.designs()` for designs covered in this package.
`alpha`	Error probability. Set it to `(1-confidence)/2` (*i.e.* to 0.05 for the usual 90% confidence intervals).
`robust`	With `robust=FALSE` (the default) usual degrees of freedom of the designs are used. With `robust=TRUE` the degrees of freedom for the so-called robust evaluation (df2 in known.designs()) will be used. This may be helpful if the CI was evaluated via a mixed model or via intra-subject contrasts (aka Senn's basic estimator).

Details

See Helmut Schütz' lecture for a description of the algebra underlying this function.

Value

Numeric value of the CV as ratio.

Note

The calculations are based on the assumption of evaluation via log-transformed values.
The calculations are further based on a common variance of Test and Reference treatments in replicate crossover studies or parallel group study, respectively.

In case of argument `n` given as n(total) and is not divisible by the number of (sequence) groups the total sample size is partitioned to the (sequence) groups to have small imbalance only. A message is given in such cases.
The estimated CV is conservative (i.e. greater than actually observed) in case of greater unbalancedness.

`CI2CV()` is simply an alias to `CVfromCI()`.

Author(s)

Original by D. Labes with suggestions by H. Schütz.
Reworked and adapted to unbalanced studies by B. Lang.

References

Yuan J, Tong T, Tang M-L. *Sample Size Calculation for Bioequivalence Studies Assessing Drug Effect and Food Effect at the Same Time With a 3-Treatment Williams Design.*
Regul Sci. 2013;47(2):242–7. doi: 10.1177/2168479012474273

Examples

```
# Given a 90% confidence interval (without point estimate)
# from a classical 2x2 crossover with 22 subjects
CVfromCI(lower=0.91, upper=1.15, n=22, design="2x2")
# will give [1] 0.2279405, i.e a CV ~ 23%
#
# unbalanced 2x2 crossover study, but not reported as  such
CI2CV(lower=0.89, upper=1.15, n=24)
# will give a CV ~ 26.3%
# unbalancedness accounted for
CI2CV(lower=0.89, upper=1.15, n=c(16,8))
# should give CV ~ 24.7%
```

CVp2CV *Decompose CV(T) and CV(R) from 'pooled' CV of T/R*

Description

Helper function to calculate CV(T) and CV(R) from a pooled CV(T/R) assuming a ratio of the intra-subject variances.

Usage

```
CVp2CV(CV, ratio = 1.5)
```

Arguments

CV	'pooled' CV of T/R.
ratio	Ratio of the intra-subject variances s^2(T)/s^2(R). May be a vector.

Details

In case of knowing only the CV(T/R) f.i. from an ordinary cross-over you can calculate the components CV(T) and CV(R) assuming a ratio of the intra-subject variances.
The formula the function is based on:
`log(1.0 + CV^2) = (sWT^2 + sWR^2)/2`
Insert sWT^2 = ratio* sWR^2 and solve for sWR^2.

Value

Returns a numeric vector of the CV values for Test and Reference if only one ratio is given.
Returns a matrix with named columns 'CVwT' and C'VwR' if ratio is given as vector.

Author(s)

D. Labes

Examples

```
CVp2CV(0.4, ratio=2)
# gives
# [1] 0.4677952 0.3225018
```

CVpooled *Pooled CV from several studies*

Description

This function pools CVs of several studies.

Usage

```
CVpooled(CVdata, alpha = 0.2, logscale = TRUE, robust = FALSE)
## S3 method for class 'CVp'
print(x, digits = 4, verbose = FALSE, ...)
```

Arguments

CVdata	A data.frame that must contain the columns CV, n and design where CV are the error CVs from the studies, n the number of subjects and design is a character string describing the study design. See known.designs() for designs covered in this package. If the design column is missing the classical 2x2 crossover is assumed for each study. A message is displayed under that circumstances. A data.frame that contains the columns CV and giving the degrees of freedom df directly is also accepted as CVdata.
alpha	Error probability for calculating an upper confidence limit of the pooled CV. Recommended 0.2–0.25 for use in subsequent sample size estimation. See f.i one of H. Schütz' lectures.
logscale	Should the calculations be done for log-transformed data? Defaults to TRUE.
robust	Defaults to FALSE. Set to TRUE will use the degrees of freedom according to the 'robust' evaluation (aka Senn' basic estimator). These dfs are calculated as n-seq. They are also often more appropriate if the CV comes from a 'true' mixed effects model evaluation (FDA model for average bioequivalence). See known.designs()$df2 for the designs covered in this package.
x	An object of class "CVp".
digits	Number of significant digits for the CV and the CL.
verbose	Defaults to FALSE. Prints only the pooled CV and the df. If set to TRUE the upper confidence limit is also printed.
...	More args to print(). None used.

Details

The pooled CV is obtained from the weighted average of the error variances obtained from the CVs of the single studies, weights are the df (degrees of freedom).
If only n is given in the input CVdata, the dfs are calculated via the formulas given in known.designs().
If both n and df are given the df column precedes.

If logscale=TRUE the error variances are obtained via function CV2se(). Otherwise the pooled CV is obtained via pooling the CV^2.

Value

A list of class "CVp" with components

CV	value of the pooled CV
df	pooled degrees of freedom
CVupper	upper confidence interval of the pooled CV
alpha	input value

The class "CVp" has a S3 methods print.CVp.

Warning

Pooling of CVs from parallel group and crossover designs does not make any sense.
Also the function *does not* throw an error if you do so.

Note

The calculations for `logscale=FALSE` are not described in the references. They are implemented by analogy to the case via log-transformed data.
The calculations are based on a common variance of Test and Reference formulations in replicate crossover studies or a parallel group study, respectively.

Author(s)

D. Labes

References

H. Schütz' lectures about sample size challenges at http://bebac.at/lectures.htm.

Patterson S, Jones B. *Bioequivalence and Statistics in Clinical Pharmacology.*
Boca Raton: Chapman & Hall / CRC Press; 2nd ed. 2017. Chapter 5.7 "Determining Trial Size".

See Also

known.designs, CVfromCI

Examples

```
# some data:
# the values for AUC, study 1 and study 2 are Example 3 of H.Schuetz' lecture
CVs <- ("
 PKmetric | CV   |  n |design| source
    AUC   | 0.20 | 24 | 2x2  | study 1
    Cmax  | 0.25 | 24 | 2x2  | study 1
    AUC   | 0.30 | 12 | 2x2  | study 2
    Cmax  | 0.31 | 12 | 2x2  | study 2
    AUC   | 0.25 | 12 | 2x2x4| study 3 (full replicate)
")
txtcon <- textConnection(CVs)
CVdata <- read.table(txtcon, header=TRUE, sep="|", strip.white=TRUE, as.is=TRUE)
close(txtcon)

# evaluation of the AUC CVs
CVsAUC <- subset(CVdata, PKmetric=="AUC")
CVpooled(CVsAUC, alpha=0.2, logscale=TRUE)
# df of the 'robust' evaluation
CVpooled(CVsAUC, alpha=0.2, logscale=TRUE, robust=TRUE)
# print also the upper CL, data example 3
CVsAUC3 <- subset(CVsAUC,design != "2x2x4")
print(CVpooled(CVsAUC3, alpha=0.2, robust=TRUE), digits=3, verbose=TRUE)
# will give the output:
```

```
# Pooled CV = 0.235 with 32 degrees of freedom (robust df's)
# Upper 80% confidence limit of CV = 0.266
#
# Combining CVs from studies evaluated by ANOVA (robust=FALSE) and
# by a mixed effects model (robust=TRUE). dfs have to be provided!
CVs <- ("
  CV    |  n |design| source  | model | df
  0.212 | 24 | 2x2  | study 1 | fixed | 22
  0.157 | 27 | 3x3  | study 2 | fixed | 50
  0.148 | 27 | 3x3  | study 3 | mixed | 24
")
txtcon <- textConnection(CVs)
CVdata <- read.table(txtcon, header=TRUE, sep="|", strip.white=TRUE, as.is=TRUE)
close(txtcon)
print(CVpooled(CVdata, alpha=0.2), digits=3, verbose=TRUE)
# will give the output:
# Pooled CV = 0.169 with 96 degrees of freedom
# Upper 80% confidence limit of CV = 0.181
```

`CVwRfromU` *CVwR from the upper expanded limit (ABEL)*

Description

Calculates the intra-subject CV (coefficient of variation) of the reference from the upper expanded limit of a BE study (replicate design for ABEL). Useful if no CV_{wR} but the expanded limits were given.

Usage

```
CVwRfromU(U, regulator = "EMA")
U2CVwR(U, regulator = "EMA")
```

Arguments

`U`	Upper expanded limit. Must be >1.2500 and <1.4319 (if `regulator="EMA"`) or >1.2500 and <1.5000 (if `regulator="HC"`).
`regulator`	Regulatory body's settings for expanding the BE acceptance limits, given as a string from the choices "EMA" or "HC". Defaults to `regulator="EMA"`.

Details

Only the upper expanded limit is supported since it offers one more significant digit than the lower expanded limit.

Value

Numeric value of the `CVwR` as ratio, where `CVwR = sqrt(exp((log(U)/r_const)^2)-1)`.

Note

`U2CVwR()` is simply an alias to `CVwRfromU()`.

Author(s)

H. Schütz

Examples

```
# Given the upper expanded limit and using the defaults
CVwRfromU(U=1.38)
# should give [1] 0.44355, i.e., a CVwR ~ 44%
# Upper limit from a study according the Health Canada's rules
CVwRfromU(U=1.48, regulator="HC")
# should give [1] 0.55214
```

---

exppower.noninf *Expected power of the non-inferiority test*

Description

Calculates the so-called expected, i.e. unconditional, power for a variety of study designs used in bioequivalence studies.

Usage

```
exppower.noninf(alpha = 0.025, logscale = TRUE, theta0, margin, CV, n,
                design = "2x2", robust = FALSE,
                prior.type = c("CV", "theta0", "both"), prior.parm = list(),
                method = c("exact", "approx"))
```

Arguments

`alpha`	Significance level (one-sided). Defaults here to 0.025.
`logscale`	Should the data be used on log-transformed or on original scale? TRUE or FALSE. Defaults to TRUE.
`theta0`	Assumed 'true' (or 'observed' in case of `prior.type != "CV"`) ratio or difference. Typically set to 0.95 (default if missing) if `logscale=TRUE`. Defaults to -0.05 if `logscale=FALSE`.
`margin`	Non-inferiority margin. In case of `logscale=TRUE` it must be given as ratio, otherwise as difference. Defaults to 0.8 if `logscale=TRUE` or to -0.2 if `logscale=FALSE`.
`CV`	Assumed true or observed coefficient of variation as ratio. Only values > 0 are allowed.

`n` Number of subjects under study.
Is total number if given as scalar, else number of subjects in the (sequence) groups. In the latter case the length of n has to be equal to the number of (sequence) groups.

`design` Character string describing the study design. See `known.designs()` for designs covered in this package.

`robust` Defaults to FALSE. Set to `TRUE` will use the degrees of freedom according to the 'robust' evaluation (aka Senn's basic estimator). These df are calculated as `n-seq`.
See `known.designs()$df2` for designs covered in this package.

`prior.type` Specifies which parameter uncertainty should be accounted for. In case of `prior.type = "CV"` (the default), only the uncertainty with respect to the CV will be considered (i.e. the given treatment effect is assumed to be fix). In case of `prior.type = "theta0"` only uncertainty with respect to the treatment ratio/difference will be accounted for (i.e. the given CV is assumed to be fix). In case of `prior.type = "both"` the power value will be unconditional with respect to both the `CV` and `theta0`.

`prior.parm` A list of parameters expressing the prior information about the variability and/or treatment effect. Possible components are `df`, `SEM`, `m` and `design`.
For `prior.type = "CV"` the degrees of freedom from the prior trial are required. This information can be provided by specifying the single component `df` or the combination consisting of `m` and `design`.
For `prior.type = "theta0"` the standard error of the treatment difference from the prior trial is required. This information can be provided by specifying the single component `SEM` or the combination consisting of `m` and `design`.
For `prior.type = "both"` the degrees of freedom and the standard error of the treatment difference are required. This information can be provided by specifying the combination consisting of `df` and `SEM` or via the combination `m` and `design`.
See 'Details' for a technical description on each component.

`method` Defaults to `method="exact"`. In that case the expected power will be calculated as expected value of the power with respect to the (prior) distribution of the respective parameter(s).
Set to `method="approx"` the expected power according to the approximate formulas given in the book from Julious or in the Julious/Owen paper will be calculated (using non-central *t*); this only affects `prior.type = "CV"`.

Details

This function calculates the so-called expected power taking into account that usually the parameters (CV and/or theta0) are not known but *estimated* from a prior study with some uncertainty. The expected power is an unconditional power and can therefore be seen as probability for success. See references for further details.

The `prior.parm` argument is a list that can supply any of the following components:

`df` Error degrees of freedom from the prior trial (>4, maybe non-integer). `df = Inf` is allowed and for `method = "exact"` the result will then coincide with `power.noninf(...)`.

Note: This corresponds to the df of both the CV and the difference of means.

`SEM` Standard error of the difference of means from the prior trial; must always be on additive scale (i.e. usually log-scale).

`m` Number of subjects from prior trial. Specification is analogous to the main argument `n`.

`design` Study design of prior trial. Specification is analogous to the main argument `design`.

For `prior.parm`, the combination consisting of `df` and `SEM` requires a somewhat advanced knowledge of the prior trial (provided in the raw output from for example the software SAS, or may be obtained via `emmeans::emmeans`). However, it has the advantage that if there were missing data the exact degrees of freedom and standard error of the difference can be used, the former possibly being non-integer valued (e.g. if the Kenward-Roger method was used).

Details on argument `prior.type`:

`CV` The expectation is calculated with respect to the Inverse-gamma distribution.

`theta0` The expectation is calculated with respect to the conditional distribution theta0 | σ^2 = s^2 of the posteriori distribution of (theta0, σ^2) from the prior trial.

`both` The expectation is calculated with respect to the posteriori distribution of (theta0, σ^2) from the prior trial. Numerical calculation of the two-dimensional integral is performed via `cubature::hcubature`.

Notes on the underlying hypotheses

If the supplied margin is < 0 (`logscale=FALSE`) or < 1 (`logscale=TRUE`), then it is assumed higher response values are better. The hypotheses are
`H0: theta0 <= margin vs. H1: theta0 > margin`
where `theta0 = mean(test)-mean(reference)` if `logscale=FALSE`
or
`H0: log(theta0) <= log(margin) vs. H1: log(theta0) > log(margin)`
where `theta0 = mean(test)/mean(reference)` if `logscale=TRUE`.

If the supplied margin is > 0 (`logscale=FALSE`) or > 1 (`logscale=TRUE`), then it is assumed lower response values are better. The hypotheses are
`H0: theta0 >= margin vs. H1: theta0 < margin`
where `theta0 = mean(test)-mean(reference)` if `logscale=FALSE`
or
`H0: log(theta0) >= log(margin) vs. H1: log(theta0) < log(margin)`
where `theta0 = mean(test)/mean(reference)` if `logscale=TRUE`.
This latter case may also be considered as 'non-superiority'.

Value

Value of expected power according to the input.

Author(s)

B. Lang & D. Labes

References

Grieve AP.
Confidence Intervals and Sample Sizes
Biometrics. 1991;47:1597–603. doi: 10.2307/2532411

O'Hagan et al.
Assurance in Clinical Trial Design
Pharm Stat. 2005;4:187–201. doi: 10.1002/pst.175

Julious SA, Owen RJ.
Sample size calculations for clinical studies allowing for uncertainty in variance
Pharm Stat. 2006;5:29–37. doi: 10.1002/pst.197

Julious SA.
Sample sizes for Clinical Trials
Boca Raton: CRC Press / Chapman & Hall; 2010.

Bertsche et al.
The predictive distribution of the residual variability in the linear-fixed effects model for clinical cross-over trials
Biom J. 2016;58(4):797–809. doi: 10.1002/bimj.201500245

Box GEP, Tiao GC.
Bayesian Inference in Statistical Analysis
Boston: Addison-Wesley; 1992.

Held L, Sabanes Bove D.
Applied Statistical Inference. Likelihood and Bayes
Berlin, Heidelberg: Springer; 2014. doi: 10.1007/9783642378874

Senn S.
Cross-over Trials in Clinical Research
Chichester: Wiley; 2002.

Zierhut ML et al.
Ignorance is not bliss: Statistical power is not probability of trial success
Clin Pharmacol Ther. 2015;99:356–9. doi: 10.1002/cpt.257

See Also

`expsampleN.noninf`, `power.noninf`

Examples

```
# Expected power for non-inferiority test for a 2x2 crossover
# with 40 subjects. CV 30% known from a pilot 2x2 study with
# 12 subjects
# using all the defaults for other parameters (theta0 carved in stone)
# should give: [1] 0.6761068
exppower.noninf(CV = 0.3, n = 40, prior.parm = list(df = 12-2))
# or equivalently
exppower.noninf(CV = 0.3, n = 40, prior.parm = list(m = 12, design = "2x2"))

# May be also calculated via exppower.TOST() after setting upper acceptance limit
# to Inf and alpha=0.025
```

```
exppower.TOST(CV = 0.3, n = 40, prior.parm = list(df = 10), theta2 = Inf, alpha=0.025)

# In contrast: Julious approximation
exppower.noninf(CV = 0.3, n = 40, prior.parm = list(df = 10), method = "approx")
# should give: [1] 0.6751358

# Compare this to the usual (conditional) power (CV known, "carved in stone")
power.noninf(CV = 0.3, n = 40)
# should give: [1] 0.7228685
# same as if setting df = Inf in function exppower.noninf()
exppower.noninf(CV = 0.3, n = 40, prior.parm = list(df = Inf))

# Expected power for a 2x2 crossover with 40 subjects
# CV 30% and theta0 = 0.95 known from a pilot 2x2 study with 12 subjects
# using uncertainty with respect to both CV and theta0
exppower.noninf(CV = 0.3, theta0 = 0.95, n = 40,
                prior.parm = list(m = 12, design = "2x2"), prior.type = "both")
# should give a decrease of expected power to 0.5982852
```

`exppower.TOST` *Expected power of the TOST procedure*

Description

Calculates the so-called expected, i.e. unconditional, power for a variety of study designs used in bioequivalence studies.

Usage

```
exppower.TOST(alpha = 0.05, logscale = TRUE, theta0, theta1, theta2,
              CV, n, design = "2x2", robust = FALSE,
              prior.type = c("CV", "theta0", "both"), prior.parm = list(),
              method = c("exact", "approx"))
```

Arguments

`alpha`	Significance level (one-sided). Commonly set to 0.05.
`logscale`	Should the data be used on log-transformed or on original scale? TRUE or FALSE. Defaults to TRUE.
`theta0`	Assumed 'true' (or 'observed' in case of `prior.type != "CV"`) bioequivalence ratio or difference. Typically set to 0.95 (default if missing) if `logscale=TRUE`. Defaults to 0.05 if `logscale=FALSE`.
`theta1`	Lower bioequivalence limit as ratio (if `logscale=TRUE`) or as difference. Can be missing. Defaults then to 0.8 if `logscale=TRUE` or to -0.2 if `logscale=FALSE`.

theta2 — Upper bioequivalence limit as ratio (if logscale=TRUE) or as difference.
If not given theta2 will be calculated as 1/theta1 if logscale=TRUE, else as -theta1.

CV — Assumed true or observed coefficient of variation as ratio. Only values > 0 are allowed. If logscale=FALSE CV is assumed to be the standard deviation.

n — Number of subjects under study.
Is total number if given as scalar, else number of subjects in the (sequence) groups. In the latter case the length of n has to be equal to the number of (sequence) groups.

design — Character string describing the study design. See known.designs() for designs covered in this package.

robust — Defaults to FALSE. Set to TRUE will use the degrees of freedom according to the 'robust' evaluation (aka Senn's basic estimator). These df are calculated as n-seq.
See known.designs()$df2 for designs covered in this package.

prior.type — Specifies which parameter uncertainty should be accounted for. In case of prior.type = "CV" (the default), only the uncertainty with respect to the CV will be considered (i.e. the given treatment effect is assumed to be fix). In case of prior.type = "theta0" only uncertainty with respect to the treatment ratio/difference will be accounted for (i.e. the given CV is assumed to be fix). In case of prior.type = "both" the power value will be unconditional with respect to both the CV and theta0.

prior.parm — A list of parameters expressing the prior information about the variability and/or treatment effect. Possible components are df, SEM, m and design.
For prior.type = "CV" the degrees of freedom from the prior trial are required. This information can be provided by specifying the single component df or the combination consisting of m and design.
For prior.type = "theta0" the standard error of the treatment difference from the prior trial is required. This information can be provided by specifying the single component SEM or the combination consisting of m and design.
For prior.type = "both" the degrees of freedom and the standard error of the treatment difference are required. This information can be provided by specifying the combination consisting of df and SEM or via the combination m and design.
See 'Details' for a technical description on each component.

method — Defaults to method="exact". In that case the expected power will be calculated as expected value of the power with respect to the (prior) distribution of the respective parameter(s).
Set to method="approx" the expected power according to the approximate formulas given in the book from Julious or in the Julious/Owen paper will be calculated (using non-central *t*); this only affects prior.type = "CV".

Details

This function calculates the so-called expected power taking into account that usually the parameters (CV and/or theta0) are not known but estimated from a prior study with some uncertainty. The

expected power is an unconditional power and can therefore be seen as probability for success. See references for further details.

The `prior.parm` argument is a list that can supply any of the following components:

`df` Error degrees of freedom from the prior trial (>4, maybe non-integer). `df = Inf` is allowed and for `method = "exact"` the result will then coincide with `power.TOST(...)`.
Note: This corresponds to the df of both the CV and the difference of means.

`SEM` Standard error of the difference of means from the prior trial; must always be on additive scale (i.e. usually log-scale).

`m` Number of subjects from prior trial. Specification is analogous to the main argument `n`.

`design` Study design of prior trial. Specification is analogous to the main argument `design`.

For `prior.parm`, the combination consisting of `df` and `SEM` requires a somewhat advanced knowledge of the prior trial (provided in the raw output from for example the software SAS, or may be obtained via `emmeans::emmeans`). However, it has the advantage that if there were missing data the exact degrees of freedom and standard error of the difference can be used, the former possibly being non-integer valued (e.g. if the Kenward-Roger method was used).

Details on argument `prior.type`:

`CV` The expectation is calculated with respect to the Inverse-gamma distribution.

`theta0` The expectation is calculated with respect to the conditional distribution theta0 | σ^2 = s^2 of the posteriori distribution of (theta0, σ^2) from the prior trial.

`both` The expectation is calculated with respect to the posteriori distribution of (theta0, σ^2) from the prior trial. Numerical calculation of the two-dimensional integral is performed via `cubature::hcubature`.

Value

Value of expected power according to the input.

Author(s)

B. Lang (thanks to G. Nehmiz for the helpful discussions)
D. Labes

References

Grieve AP.
Confidence Intervals and Sample Sizes
Biometrics. 1991;47:1597–603. doi: 10.2307/2532411

O'Hagan et al.
Assurance in Clinical Trial Design
Pharm Stat. 2005;4:187–201. doi: 10.1002/pst.175

Julious SA, Owen RJ.
Sample size calculations for clinical studies allowing for uncertainty in variance
Pharm Stat. 2006;5:29–37. doi: 10.1002/pst.197

Julious SA.
Sample sizes for Clinical Trials
Boca Raton: CRC Press / Chapman & Hall; 2010.

Bertsche et al.
The predictive distribution of the residual variability in the linear-fixed effects model for clinical cross-over trials
Biom J. 2016;58(4):797–809. doi: 10.1002/bimj.201500245

Box GEP, Tiao GC.
Bayesian Inference in Statistical Analysis
Boston: Addison-Wesley; 1992.

Held L, Sabanes Bove D.
Applied Statistical Inference. Likelihood and Bayes
Berlin, Heidelberg: Springer; 2014. doi: 10.1007/9783642378874

Senn S.
Cross-over Trials in Clinical Research
Chichester: Wiley; 2002.

Zierhut ML et al.
Ignorance is not bliss: Statistical power is not probability of trial success
Clin Pharmacol Ther. 2015;99:356–9. doi: 10.1002/cpt.257

See Also

expsampleN.TOST, power.TOST

Examples

```
# Expected power for a 2x2 crossover with 40 subjects
# CV 30% known from a pilot 2x2 study with 12 subjects
# using all the defaults for other parameters (theta0 carved in stone)
exppower.TOST(CV = 0.3, n = 40, prior.parm = list(df = 12-2))
# should give: [1] 0.7365519
# or equivalently
exppower.TOST(CV = 0.3, n = 40, prior.parm = list(m = 12, design = "2x2"))

# In contrast: Julious approximation
exppower.TOST(CV = 0.3, n = 40, prior.parm = list(df = 10), method = "approx")
# should give: [1] 0.7359771

# Compare this to the usual (conditional) power (CV known, "carved in stone")
power.TOST(CV = 0.3, n = 40)
# should give: [1] 0.8158453
# same as if setting df = Inf in function exppower.TOST()
exppower.TOST(CV = 0.3, n = 40, prior.parm = list(df = Inf))

# Expected power for a 2x2 crossover with 40 subjects
# CV 30% and theta0 = 0.95 known from a pilot 2x2 study with 12 subjects
# using uncertainty with respect to both CV and theta0
exppower.TOST(CV = 0.3, theta0 = 0.95, n = 40,
              prior.parm = list(m = 12, design = "2x2"), prior.type = "both")
# should give [1] 0.5114685
```

`expsampleN.noninf` *Sample size based on expected power for the non-inferiority test*

Description

Calculates the sample size based on the expected power for a variety of designs used in bioequivalence studies. See `known.designs()` for the study designs covered.

Usage

```
expsampleN.noninf(alpha = 0.025, targetpower = 0.8, logscale = TRUE,
                  theta0, margin, CV, design = "2x2", robust = FALSE,
                  prior.type = c("CV", "theta0", "both"), prior.parm = list(),
                  method = c("exact", "approx"), print = TRUE, details)
```

Arguments

`alpha`	Significance level (one-sided). Defaults here to 0.025.
`targetpower`	Power to achieve at least. Must be >0 and <1. Typical values are 0.8 or 0.9.
`logscale`	Should the data used on log-transformed or on original scale? `TRUE` or `FALSE`. Defaults to `TRUE`.
`theta0`	Assumed 'true' (or 'observed' in case of `prior.type != "CV"`) ratio or difference. Typically set to 0.95 (default if missing) if `logscale=TRUE`. Defaults to -0.05 if `logscale=FALSE`.
`margin`	Non-inferiority margin. In case of `logscale=TRUE` it must be given as a ratio, otherwise as a difference. Defaults to 0.8 if `logscale=TRUE` or to -0.2 if `logscale=FALSE`.
`CV`	Assumed true or observed coefficient of variation as ratio (not percent). Only values > 0 are allowed. If `logscale=FALSE` CV is assumed to be the standard deviation. If `prior.type="CV"` may be given as vector: The CVs are then pooled (as a weighted mean with their degrees of freedom as weights).
`design`	Character string describing the study design. See `known.designs()` for designs covered in this package.
`robust`	Defaults to `FALSE`. With that value the usual degrees of freedom will be used. Setting to `TRUE` will use the degrees of freedom according to the 'robust' evaluation (aka Senn's basic estimator). These df are calculated as `n-seq`. See `known.designs()` for designs covered in this package.
`prior.type`	Specifies which parameter uncertainty should be accounted for. In case of `prior.type="CV"` (the default), only the uncertainty with respect to the CV will be considered (*i.e.*, the given treatment effect is assumed to be fix). In case of `prior.type="theta0"` only uncertainty with respect to the treatment ratio/difference will be accounted for (*i.e.*, the given CV is assumed to be fix). In case of `prior.type="both"` the power value will be unconditional with respect to both the `CV` and `theta0`.

prior.parm — A list of parameters expressing the prior information about the variability and/or treatment effect. Possible components are df, SEM, m, design.
For prior.type="CV" the degrees of freedom from the prior trial are required. This information can be provided by specifying the single componentdf or the combination consisting of m and design.
For prior.type = "theta0" the standard error of the treatment difference from the prior trial is required. This information can be provided by specifying the single component SEM or the combination consisting of m and design.
For prior.type = "both" the degrees of freedom and the standard error of the treatment difference are required. This information can be provided by specifying the combination consisting of df and SEM or via the combination m and design.
See section 'Details' for a technical description of each component.

method — Defaults to method="exact". In that case the expected power will be calculated as expected value of the power with respect to the (prior) distribution of the respective parameter(s).
Set to method="approx" the expected power according to the approximate formulas given by Julious or Julious & Owen will be calculated (using the noncentral *t*); this only affects prior.type = "CV".

print — If TRUE (default) the function prints its results.
If FALSE only a data.frame with the results will be returned.

details — If TRUE the design characteristics and the steps during sample size calculations will be shown.
If not specified, the default value is FALSE for prior.type != "both" and TRUE otherwise.

Details

The sample size is calculated based on iterative evaluation of expected power. The starting value of the sample size search is taken from a large sample approximation if prior.type="CV". Else an empirical start value is obtained. Note that in case of prior.type="both" the calculation may still take several seconds.

Note also that the expected power is always bounded above by the so-called probability of technical success (PTS) which may be a value less than 1.Therefore, it may be possible that it is either not possible to calculate the required sample size at all or that the sample size gets very large if the given targetpower is less but close to the PTS.

Notes on the underlying hypotheses

If the supplied margin is < 0 (logscale=FALSE) or < 1 (logscale=TRUE), then it is assumed *higher* response values are better. The hypotheses are
H0: theta0 <= margin
H1: theta0 > margin
where theta0 = mean(test)-mean(reference) if logscale=FALSE
or
H0: log(theta0) <= log(margin)
H1: log(theta0) > log(margin)
where theta0 = mean(test)/mean(reference) if logscale=TRUE.

If the supplied margin is > 0 (`logscale=FALSE`) or > 1 (`logscale=TRUE`), then it is assumed *lower* response values are better. The hypotheses are
H0: theta0 >= margin
H1: theta0 < margin
where `theta0 = mean(test)-mean(reference)` if `logscale=FALSE`
or
H0: log(theta0) >= log(margin)
H1: log(theta0) < log(margin)
where `theta0 = mean(test)/mean(reference)` if `logscale=TRUE`.
This latter case may also be considered as 'non-superiority'.

Value

A data.frame with the input values and the result of the sample size estimation.
The Sample size column contains the *total* sample size in case of all designs implemented.

Author(s)

B. Lang, D. Labes

References

Grieve AP. *Confidence Intervals and Sample Sizes.*
Biometrics. 1991;47:1597–603. doi: 10.2307/2532411

O'Hagan et al. *Assurance in Clinical Trial Design.*
Pharm Stat. 2005;4:187–201. doi: 10.1002/pst.175

Julious SA, Owen RJ. *Sample size calculations for clinical studies allowing for uncertainty in variance.*
Pharm Stat. 2006;5:29–37. doi: 10.1002/pst.197

Julious SA. *Sample sizes for Clinical Trials.*
Boca Raton: CRC Press; 2010.

Bertsche et al. *The predictive distribution of the residual variability in the linear-fixed effects model for clinical cross-over trials.*
Biom J. 2016;58(4):797–809. doi: 10.1002/bimj.201500245

Box GEP, Tiao GC. *Bayesian Inference in Statistical Analysis.*
Boston: Addison-Wesley; 1992.

Held L, Sabanes Bove D. *Applied Statistical Inference. Likelihood and Bayes.*
Berlin, Heidelberg: Springer; 2014. doi: 10.1007/9783642378874

Senn S. *Cross-over Trials in Clinical Research.*
Chichester: Wiley; 2002.

Zierhut ML et al. *Ignorance is not bliss: Statistical power is not probability of trial success.*
Clin Pharmacol Ther. 2015;99:356–9. doi: 10.1002/cpt.257

See Also

`exppower.noninf`, `known.designs`, `sampleN.noninf`

Examples

```
# Classical 2x2 cross-over, target power = 80%,
# assumed true ratio = 95%, margin = 0.8,
# intra-subject CV=30% estimated from prior 2x2 trial
# with m = 12 subjects
expsampleN.noninf(theta0 = 0.95, margin = 0.8, CV = 0.3, design = "2x2",
                  prior.parm = list(m = 12, design = "2x2"))
# gives n = 58 with achieved expected power 0.809148
# Compare this to the usual sample size with CV assumed
# as 'carved in stone'
sampleN.noninf(theta0 = 0.95, margin = 0.8, CV=0.3)

# Perform 'non-superiority' (lower is better) with assumed
# true ratio = 105% and margin 125%
expsampleN.noninf(theta0 = 1.05, margin = 1.25, CV = 0.3, design = "2x2",
                  prior.parm = list(m = 12, design = "2x2"))
# should give n = 56 with achieved expected power 0.806862

# More than one CV with corresponding degrees of freedom
# other settings as above in first example
CVs <- c(0.25, 0.3)
dfs <- c(22, 10)
expsampleN.noninf(theta0 = 0.95, margin = 0.8, CV = CVs,
                  prior.parm = list(df = dfs))
# should give a pooled CV=0.2664927 with 32 df and a sample
# size n=42 with achieved expected power 0.814073 exact
# achieved expected power 0.816163 approximate acc. to Julious

# Uncertainty is accounted for CV and theta0
## Not run:
expsampleN.noninf(CV=0.3, prior.type = "both",
                  prior.parm = list(m = 12, design = "2x2"))
# gives a dramatic increase in sample size (n = 194)
# due to small pilot trial
## End(Not run)
```

---

expsampleN.TOST *Sample size based on expected power*

Description

Calculates the sample size based on the expected power for a variety of study designs used in bioequivalence studies. See known.designs() for the study designs covered.

Usage

```
expsampleN.TOST(alpha = 0.05, targetpower = 0.8, logscale=TRUE, theta0,
                theta1, theta2, CV, design = "2x2", robust = FALSE,
                prior.type = c("CV", "theta0", "both"), prior.parm = list(),
                method = c("exact", "approx"), print = TRUE, details)
```

Arguments

alpha	Significance level (one-sided). Commonly set to 0.05.
targetpower	Power to achieve at least. Must be >0 and <1. Typical values are 0.8 or 0.9.
logscale	Should the data used on log-transformed or on original scale? TRUE or FALSE. Defaults to TRUE.
theta0	Assumed 'true' (or 'observed' in case of `prior.type != "CV"`) bioequivalence ratio or difference. Typically set to 0.95 (default if missing) if `logscale=TRUE`. Defaults to 0.05 if `logscale=FALSE`.
theta1	Lower bioequivalence limit as ratio if `logscale=TRUE` or as difference. Can be missing. Defaults then to 0.8 if `logscale=TRUE` or to -0.2 if `logscale=FALSE`.
theta2	Upper bioequivalence limit as ratio if `logscale=TRUE` or as difference. If not given `theta2` will be calculated as `1/theta1` if `logscale=TRUE`, else as `-theta1`.
CV	Assumed true or observed coefficient of variation as ratio. Only values > 0 are allowed. If `prior.type="CV"` may be given as vector: The CVs are then pooled (as a weighted mean with their degrees of freedoms as weights).
design	Character string describing the study design. See `known.designs()` for designs covered in this package.
robust	Defaults to FALSE. With that value the usual degrees of freedom will be used. Set to `TRUE` will use the degrees of freedom according to the 'robust' evaluation (aka Senn's basic estimator). These df are calculated as `n-seq`. See `known.designs()$df2` for designs covered in this package.
prior.type	Specifies which parameter uncertainty should be accounted for. In case of `prior.type = "CV"` (the default), only the uncertainty with respect to the CV will be considered (i.e. the given treatment effect is assumed to be fix). In case of `prior.type = "theta0"` only uncertainty with respect to the treatment ratio/difference will be accounted for (i.e. the given CV is assumed to be fix). In case of `prior.type = "both"` the power value will be unconditional with respect to both the `CV` and `theta0`.
prior.parm	A list of parameters expressing the prior information about the variability and/or treatment effect. Possible components are `df`, `SEM`, `m`, `design`. For `prior.type = "CV"` the degrees of freedom from the prior trial are required. This information can be provided by specifying the single component `df` or the combination consisting of `m` and `design`. For `prior.type = "theta0"` the standard error of the treatment difference from the prior trial is required. This information can be provided by specifying the single component `SEM` or the combination consisting of `m` and `design`. For `prior.type = "both"` the degrees of freedom and the standard error of the treatment difference are required. This information can be provided by specifying the combination consisting of `df` and `SEM` or via the combination `m` and `design`. See 'Details' for a technical description on each component.

`method`	Defaults to `method="exact"`. In that case the expected power will be calculated as expected value of the power with respect to the (prior) distribution of the respective parameter(s). Set to `method="approx"` the expected power according to the approximate formulas given in the book from Julious or in the Julious/Owen paper will be calculated (using non-central *t*); this only affects `prior.type = "CV"`.
`print`	If TRUE (default) the function prints its results. If FALSE only a data.frame with the results will be returned.
`details`	If TRUE the design characteristics and the steps during sample size calculations will be shown. If not specified, the default value is FALSE for `prior.type != "both"` and TRUE otherwise.

Details

The sample size is calculated based on iterative evaluation of expected power. The starting value of the sample size search is taken from a large sample approximation if `prior.type = "CV"`. Else an empirical start value is obtained. Note that in case of `prior.type = "both"` the calculation may still take several seconds.

Note also that the expected power is always bounded above by the so-called probability of technical success (PTS) which may be a value less than 1. Therefore, it may be possible that it is either not possible to calculate the required sample size at all or that the sample size gets very large if the given targetpower is less but close to the PTS.

Value

A data.frame with the input values and the result of the sample size estimation.
The "Sample size" column contains the **total** sample size in case of all designs implemented.

Author(s)

B. Lang & D. Labes

References

Grieve AP.
Confidence Intervals and Sample Sizes
Biometrics. 1991;47:1597–603. doi: 10.2307/2532411

O'Hagan et al.
Assurance in Clinical Trial Design
Pharm Stat. 2005;4:187–201. doi: 10.1002/pst.175

Julious SA, Owen RJ.
Sample size calculations for clinical studies allowing for uncertainty in variance
Pharm Stat. 2006;5:29–37. doi: 10.1002/pst.197

Julious SA.
Sample sizes for Clinical Trials
Boca Raton: CRC Press / Chapman & Hall; 2010.

Bertsche et al.
The predictive distribution of the residual variability in the linear-fixed effects model for clinical cross-over trials
Biom J. 2016;58(4):797–809. doi: 10.1002/bimj.201500245

Box GEP, Tiao GC.
Bayesian Inference in Statistical Analysis
Boston: Addison-Wesley; 1992.

Held L, Sabanes Bove D.
Applied Statistical Inference. Likelihood and Bayes
Berlin, Heidelberg: Springer; 2014. doi: 10.1007/9783642378874

Senn S.
Cross-over Trials in Clinical Research
Chichester: Wiley; 2002.

Zierhut ML et al.
Ignorance is not bliss: Statistical power is not probability of trial success
Clin Pharmacol Ther. 2015;99:356–9. doi: 10.1002/cpt.257

See Also

exppower.TOST, known.designs, sampleN.TOST

Examples

```
# Classical 2x2 cross-over, target power = 80%,
# BE limits 80 ... 125%, assumed true BE ratio = 95%,
# intra-subject CV=30% estimated from prior 2x2 trial
# with m = 30 subjects
expsampleN.TOST(CV=0.3, prior.parm = list(m = 30, design = "2x2"))
# -> gives n = 42 with achieved expected power 0.806262
# Compare this to the usual sample size with CV assumed known ('carved in stone')
sampleN.TOST(CV=0.3)
# -> gives n = 40 subjects
# Compare this to the case where uncertainty is accounted for CV and theta0
# Not run due to timing policy of CRAN - may run several seconds
## Not run:
expsampleN.TOST(CV=0.3, prior.parm = list(m = 30, design = "2x2"),
                prior.type = "both")
## End(Not run)
# -> gives n = 72 subjects

# More than one CV with corresponding degrees of freedom
# other settings as above in first example
CVs <- c(0.25, 0.3)
dfs <- c(22, 10)
expsampleN.TOST(CV=CVs, prior.parm = list(df = dfs))
# -> gives a pooled CV=0.2664927 with df=32
# and a sample size n=34 with achieved expected power 0.812653 exact
# achieved expected power 0.815019 approximate acc. Julious
```

known.designs *Show the 'known' designs*

Description

Returns the known study designs for which power and sample size can be calculated within this package.

Usage

```
known.designs()
```

Details

This function is for informal purposes and will be used internal
for obtaining characteristics of the designs used in calculation formulas.

Value

Returns a data.frame with

no	= number of the design
design	= character string for identifying the design
df	= degrees of freedom of the design
df2	= 'robust' degrees of freedom of the design
steps	= step width in the iterative sample size estimation
bk	= so-called design constant in terms of total n
bkni	= design constant in terms of number of subjects in (sequence) groups

The design character string has to be used in the functions calls for power and sample size.

Note

The design string for higher order crossover designs is named as:
`treatments x sequences x periods` in case of replicate designs and
`treatments x periods` in case of crossover designs for more then 2 treatments with number of sequences equal number of treatments.

The df for the replicate crossover designs are those without carry-over in the model.
Chen, Chow and Liu used models with carry-over, i.e. one df lower than here.

The design constant bk in case of design 2x2x4 is here bk=1.
Chen, Chow and Liu used bk=1.1 due to carry-over in the model.

n is the **total** number of subjects for all designs implemented.

df2 = degrees of freedom for the so-called 'robust' analysis (aka Senn's basic estimator).
These degrees of freedom are often also more appropriate in case of evaluation via a 'true' mixed model (FDA model for replicate designs).

The design `2x2x2r` is the 2-treatment-2-sequence-2-period design with 2 repeated targets determined in each period (sequences TT|RR or RR|TT) described by Liu. Implemented are the characteristics of this design for the evaluation via assuming no SxF interaction and equal variability for Test and Reference.

Author(s)

D. Labes

References

K.-W. Chen, S.-C. Chow and G. Liu
"A Note on Sample Size Determination for Bioequivalence Studies with Higher-order Crossover Designs"
J. Pharmacokinetics and Biopharmaceutics, Vol. 25, No. 6, p753-765 (1997)

S. Senn
"Cross-over Trials in Clinical Research"
Second Edition, John Wiley & Sons, Chichester 2002

FDA Guidance for Industry.
"Statistical Approaches to Establishing Bioequivalence"
U.S. Department of Health and Human Services,
Food and Drug Administration,
Center for Drug Evaluation and Research (CDER). January 2001

Liu J-P
"Use of the Repeated Crossover design in Assessing Bioequivalence"
Stat. Med. Vol. 14, 1067-1078 (1995)

Examples

```
known.designs()
```

OwensQ *Owen's Q-function*

Description

Calculates Owen's Q function.

Usage

```
OwensQ(nu, t, delta, a=0, b)
```

Arguments

nu	degree of Owen's Q
t	parameter t
delta	parameter delta
a	lower integration limit, only a=0 implemented
b	upper integration limit

Details

Uses the relationship to non-central t-distribution (see Chou YM)

OwensQ = pt(t, df=nu, ncp=delta) - Integal_b_Inf(Q_integrand)

The definite integral is numerically evaluated using `integrate()` from package `stats` after a variables transformation resulting in the integration range from 0 to 1 instead of the semi-infinite original range. This may result in higher precision and better numerical stability.

The arguments to the function must be scalars. No vectors allowed.

Value

Numeric value of Owen's Q-function at given input arguments.

Note

This function is intended for internal use in the power calculations.
But may be useful for others.

Author(s)

D. Labes

References

Owen DB.
A special case of a bivariate non-central t-distribution
Biometrika. 1965;52(3/4):437–46. doi: 10.2307/2333696

Chou YM.
A bivariate noncentral T-distibution with applications
Commun Stat Theory Methods. 1992;21(12):3427–62. doi: 10.1080/03610929208830988

See Also

`OwensQOwen`

Examples

```
# This function is mainly intended for internal use.
OwensQ(10, 2.5, 5, 0, 2)
#should give [1] 9.388137e-06
OwensQ(10, -2.5, -5, 0, 2)
#should give [1] 0.05264363
```

`OwensQOwen` *Owen's Q-function via repeated integration by parts*

Description

This is an implementation of the algorithm given in Owen's original paper (Biometrica 1965) via repeated integration by parts.

Usage

```
OwensQOwen(nu, t, delta, a=0, b)
```

Arguments

`nu`	degree of Owen's Q
`t`	parameter t
`delta`	parameter delta
`a`	lower integration limit. Only a=0 implemented, other values give an error.
`b`	upper integration limit

Value

numeric value of Owen's Q function.

Note

The argument `a=0` could be dropped but is retained for sake of completeness.

Note

This function is mainly for comparative / validation purposes.
The function needs `OwensT()` function.

Author(s)

D. Labes

References

Owen DB.
A special case of a bivariate non-central t-distribution
Biometrika. 1965;52(3/4):437–46. doi: 10.2307/2333696

See Also

OwensQ, OwensT

Examples

```
# comparison of the results of both implementations
# both should give [1] 0.0731726
OwensQ(2, 2.92, 4.2135, 0, 2.0407)
OwensQOwen(2, 2.92, 4.2135, 0, 2.0407)
```

OwensT *Owen's T-function*

Description

Calculates the definite integral from 0 to a of exp(-0.5*h^2*(1+x^2))/(1+x^2)/(2*pi).

Usage

```
OwensT(h, a)
```

Arguments

h	parameter h
a	upper limit of integration

Details

The function is an R port of FORTRAN code given in the references and MATLAB code given on http://people.sc.fsu.edu/~jburkardt/m_src/asa076/asa076.html by John Burkardt under the GNU LGPL license.

The arguments of OwensT() have to be scalars because the implementation doesn't vectorize.

Value

Numerical value of the definite integral.

Note

This function is only needed as auxiliary in OwensQOwen().
But may be useful for others.

Author(s)

MATLAB code by J. Burkardt
R port by D. Labes

References

Goedhart PW, Jansen MJW.
Remark AS R89: A Remark on Algorithm AS 76: An Integral Useful in Calculating Central t and Bivariate Normal Probabilities
J Royal Stat Soc C. 1992;41(2):496–7. doi: 10.2307/2347586

Boys R.
Algorithm AS R80: A Remark on Algorithm AS 76: An Integral Useful in Calculating Noncentral t and Bivariate Normal Probabilities
J Royal Stat Soc C. 1989;38(3):580–2. doi: 10.2307/2347755

Thomas GE.
Remark ASR 65: A Remark on Algorithm AS76: An Integral Useful in Calculating Non-Central t and Bivariate Normal Probabilities
J Royal Stat Soc C. 1986;35(3):310–2. doi: 10.2307/2348031

Chou Y-M.
Remark AS R55: A Remark on Algorithm AS 76: An Integral Useful in Calculating Noncentral T and Bivariate Normal Probabilities
J Royal Stat Soc C. 1985;34(1):100–1. doi: 10.2307/2347894

Thomas GE.
Remark AS R30: A Remark on Algorithm AS 76: An Integral Useful in Calculating Non-Central t and Bivariate Normal Probabilities
J Royal Stat Soc C. 1979;28(1):113. doi: 10.2307/2346833

Young JC, Minder C.
Algorithm AS 76: An Integral Useful in Calculating Non-Central t and Bivariate Normal Probabilities
J Royal Stat Soc C. 1974;23(3):455–7. doi: 10.2307/2347148

Owen DB.
Tables for Computing Bivariate Normal Probabilities
Ann Math Stat. 1956;27(4):1075–90. doi: 10.1214/aoms/1177728074

See Also

`OwensQOwen`, `OwensQ`

Examples

```
OwensT(2.5, 0.75)
# should give [1]  0.002986697
# value from Owen's tables is 0.002987
OwensT(2.5, -0.75)
# should give [1] -0.002986697
```

pa.ABE *Power analysis for average bioequivalence (ABE)*

Description

An analysis tool for exploration/visualization of the impact of expected values (CV, theta0, reduced sample size due to drop-outs) on power of BE decision via ABE if these values deviate from the ones assumed in planning the sample size of the study.

Usage

```
pa.ABE(CV, theta0 = 0.95, targetpower = 0.8, minpower = 0.7, design = "2x2", ...)
## S3 method for class 'pwrA'
print(x, digits = 4, plotit = TRUE, ...)
## S3 method for class 'pwrA'
plot(x, pct = TRUE, ratiolabel = "theta0", cols = c("blue", "red"), ...)
```

Arguments

CV	Coefficient of variation as ratio. In case of cross-over studies this is the within-subject CV.
theta0	'True' or assumed T/R ratio. Often named GMR. Must be given as ratio.
targetpower	Power to achieve at least in sample size estimation. Must be >0 and <1. Typical values are 0.8 or 0.9. Defaults to 0.8. Note that targetpower < 0.5 doesn't make much sense.
minpower	Minimum acceptable power to have if deviating from assumptions for sample size plan. Has to be lower than targetpower. Defaults to 0.7. minpower < 0.5 doesn't make much sense.
design	Character string describing the study design. See known.designs() for designs covered in this package.
...	More arguments to pass to power.TOST(). F.i. alpha, theta1, theta2 or robust if other values then the defaults for these arguments are needed. See man page of power.TOST(). More arguments passed to the S3 methods. Here currently ignored. Additional arguments of the S3 methods:
x	Object of class 'pwrA'.
digits	Digits for rounding power in printing. The '...' argument is currently ignored in print().
plotit	If set to TRUE, the default, the print method calls plot(x) if R is running interactively.

pct	If set to TRUE (the default) scales CV, theta0, and power in percent in plot(). Else they will be given as ratios, the usual standard in PowerTOST.
ratiolabel	Label of the T/R-ratio. Can be set to any string, e.g. to "GMR". Defaults to "theta0", the usual standard in PowerTOST.
cols	Colors for the plots. cols[1] gives the color for plotting points with power>targetpower. From targetpower toward minpower the color changes gradually to cols[2].

Details

Power calculations are done via power.TOST() and calculations of CV and theta0 which gave a power=minpower are derived via R base uniroot(). While one of the parameters (CV, theta0, N) is varied, the respective two others are kept constant. The tool shows the relative impact of single parameters on power.
The tool takes a minimum of 12 subjects as demanded in most BE guidances into account.

It should be kept in mind that this is **not** a substitute for the "Sensitivity Analysis" recommended in ICH-E9. In a real study a combination of all effects occurs simultaneously. It is up to *you* to decide on reasonable combinations and analyze their respective power.

Value

Returns a list with class "pwrA" with the components

plan	A data.frame with the result of the sample size estimation. See output of sampleN.TOST().
paCV	A data.frame with value pairs CV, pwr for impact of deviations from CV.
paGMR	A data.frame with value pairs theta0, pwr for impact of deviations from theta0 (GMR).
paN	A data.frame with value pairs N, pwr for impact of deviations from planned N (dropouts).
method	Method of BE decision. Here fix = "ABE".
minpower	Minimum acceptable power.

The class 'pwrA' has the S3 methods print() and plot(). See pa.scABE for usage.

Note

The code of deviations from planned sample size tries to keep the degree of imbalance as low as possible between (sequence) groups. This results in a lesser decrease of power than more extreme dropout-patterns.

Author(s)

Idea and original code by H. Schütz
with modifications by D. Labes to use PowerTOST infrastructure.

References

See http://forum.bebac.at/mix_entry.php?id=13353.

See Also

power.TOST, known.designs, pa.scABE, pa.NTIDFDA

Examples

```
# using the defaults
# design="2x2", targetpower=0.8, minpower=0.7, theta0/GMR=0.95
# BE margins from defaults of sampleN.TOST() 0.8 ... 1.25
# print & plot implicitly
pa.ABE(CV=0.2)
# print & plot
## Not run:
res <- pa.ABE(CV=0.2)
print(res, plotit=FALSE)              # print only
plot(res, pct=FALSE, ratiolabel="GMR") # changed from defaults
## End(Not run)
```

---

pa.NTIDFDA *Power analysis for scaled ABE for NTIDs according to FDA*

Description

An analysis tool for exploration/visualization of the impact of expected values (CV, theta0, reduced sample size due to drop-outs) on power of BE decision via scABE for narrow therapeutic drugs (NTIDs) if these values deviate from the ones assumed in planning the sample size of the study.
The only implemented design is the full replicate design "2x2x4" according to the FDA Warfarin guidance.

Usage

```
pa.NTIDFDA(CV, theta0 = 0.975, targetpower = 0.8, minpower = 0.7, ...)
```

Arguments

CV	Coefficient of variation of the intra-subject variabilities of Test and Reference as ratio. Here only the case CVwT=CVwR is implemented, i.e. CV has to be a scalar.
theta0	'True' or assumed T/R ratio. Often named GMR. Must be given as ratio. Defaults here to 0.975.
targetpower	Power to achieve at least in sample size estimation. Must be >0 and <1. Typical values are 0.8 or 0.9. Defaults to 0.8. Note that targetpower < 0.5 doesn't make much sense.
minpower	Minimum acceptable power to have if deviating from assumptions for sample size plan. Has to be lower than targetpower. Defaults to 0.7. minpower < 0.5 doesn't make much sense.

...	More arguments to pass to power.NTIDFDA(). F.i. alpha, theta1, theta2 or nsims if other values then the defaults for these arguments are needed. See man page of power.NTIDFDA().

Details

Power calculations are done via power.NTIDFDA() and calculations of CV and theta0 which result in minpower are derived via uniroot().
While one of the parameters (CV, theta0, n) is varied, the respective two others are kept constant. The tool shows the relative impact of single parameters on power.
The tool takes a minimum of 12 subjects into account as demanded in most BE guidances. However, it should be kept in mind that the FDA requires at least 24 subjects to be enrolled in studies intended for reference-scaling.

It should be kept in mind that this is **not** a substitute for the "Sensitivity Analysis" recommended in ICH-E9. In a real study a combination of all effects occurs simultaneously. It is up to *you* to decide on reasonable combinations and analyze their respective power.

Value

Returns a list with class 'pwrA' with the components

plan	A data.frame with the result of the sample size estimation. See output of sampleN.NTIDFDA().
paCV	A data.frame with value pairs CV, pwr for impact of deviations from CV.
paGMR	A data.frame with value pairs theta0, pwr for impact of deviations from theta0 (GMR).
paN	A data.frame with value pairs N, pwr for impact of deviations from planned N (dropouts).
method	Method of BE decision. Here fix = "NTID FDA".
regulator	Here fix = "FDA".
minpower	Minimum acceptable power from the call of the function.

The class 'pwrA' has the S3 methods print() and plot(). See pa.ABE for usage.

Warning

Be extremly carefull if your sample size plan has extremly small CV near or below 0.05 (5%). Adapt in that case your expected true ratio (theta0) to values nearer to 1 to not run into errors and/or long execution times.

Note

The code for impact of deviations from planned sample size tries to keep the degree of imbalance as low as possible between (sequence) groups. This results in a lesser decrease of power than more extreme dropout-patterns.

Author(s)

D. Labes
according to code by H. Schütz for pa.ABE() and pa.scABE()

References

FDA. *Draft Guidance on Warfarin Sodium.*
Recommended Dec 2012
http://www.fda.gov/downloads/Drugs/GuidanceComplianceRegulatoryInformation/Guidances/UCM201283.pdf

Yu LX et al.
Novel bioequivalence approach for narrow therapeutic index drugs
Clin Pharmacol Ther. 2015;97(3):286–91. doi: 10.1002/cpt.28

Jiang W et al.
A Bioequivalence Approach for Generic Narrow Therapeutic Index Drugs: Evaluation of the Reference-Scaled Approach and Variability Comparison Criterion
AAPS J. 2015;17(4):891–901. doi: 10.1208/s1224801597535

Endrényi L, Tóthfalusi L.
Determination of Bioequivalence for Drugs with Narrow Therapeutic Index: Reduction of the Regulatory Burden
J Pharm Pharm Sci. 2013;16(5):676–82. free download

See Also

power.NTIDFDA, print.pwrA, plot.pwrA, pa.ABE, pa.scABE

Examples

```
# using the defaults:
# targetpower=0.8, minpower=0.7, theta0/GMR=0.975
# BE margins from defaults of sampleN.NTIDFDA() 0.9002 ... 1.1108
# 1E5 sims in power.NTIDFDA()
# not run due to timing policy of CRAN for examples
# may run some ten seconds or more
## Not run:
plot(pa.NTIDFDA(CV=0.1))
## End(Not run)
```

---

pa.scABE *Power analysis for scaled average bioequivalence (scABE)*

Description

An analysis tool for exploration/visualization of the impact of expected values (CV, theta0, reduced sample size due to drop-outs) on power of BE decision via scABE (for highly variable drugs) if these values deviate from the ones assumed in planning the sample size of the study.

Usage

```
pa.scABE(CV, theta0 = 0.9, targetpower = 0.8, minpower = 0.7,
         design = c("2x3x3", "2x2x4", "2x2x3"),
         regulator = c("EMA", "HC", "FDA"), ...)
```

Arguments

CV	Coefficient of variation of the intra-subject variability as ratio. Here only the case CVwT=CVwR is implemented, i.e. CV has to be a scalar.
theta0	'True' or assumed T/R ratio. Often named GMR. Must be given as ratio. Defaults to 0.9 here since HVD have a greater scatter in point estimates of T/R.
targetpower	Power to achieve at least in sample size estimation. Must be >0 and <1. Typical values are 0.8 or 0.9. Defaults to 0.8. Note that targetpower < 0.5 doesn't make much sense.
minpower	Minimum acceptable power to have if deviating from assumptions for sample size plan. Has to lower than targetpower. Defaults to 0.7. minpower < 0.5 doesn't make much sense.
design	Character string describing the study design. Defaults to 2x3x3, the partial replicate design (TRR|RTR|RRT).
regulator	Character string describing the scaled ABE method recommended by the regulatory bodies EMA, HC, or FDA. Defaults to EMA, method of scaled (widened) bioequivalence limits.
...	More arguments to pass to power.scABEL() or power.RSABE(). F.i. alpha, theta1, theta2 or nsims if other values then the defaults for these arguments are needed. See man pages of power.scABEL() or power.RSABE().

Details

Power calculations are done via power.scABEL() or power.RSABE() and calculations of CV and theta0 which result in minpower are derived via uniroot().
While one of the parameters (CV, GMR, N) is varied, the respective two others are kept constant. The tool shows the relative impact of single parameters on power.
The tool takes a minimum of 12 subjects as demanded in most BE guidances into account. However, it should be kept in mind that

- the FDA requires at least 24 subjects *enrolled* in studies intended for reference-scaling;
- the EMA requires at least 12 *eligible* subjects in the sequence RTR of the TRT|RTR-design.

You should be aware that this is **not** a substitute for the "Sensitivity Analysis" recommended in ICH-E9. In a real study a combination of all effects occurs simultaneously. It is up to *you* to decide on reasonable combinations and analyze their respective power.

Value

Returns a list with class 'pwrA' with the components

plan	A data.frame with the result of the sample size estimation. See output of sampleN.scABEL() or sampleN.RSABE().
paCV	A data.frame with value pairs CV, pwr for impact of deviations from CV.
paGMR	A data.frame with value pairs theta0, pwr for impact of deviations from theta0 (GMR).
paN	A data.frame with value pairs N, pwr for impact of deviations from planned N (dropouts).
method	Method of BE decision. Here fix = "scABE".
regulator	"EMA", "HC", or "FDA".
minpower	Minimum acceptable power from the call of the function.

The class 'pwrA' has the S3 methods print() and plot(). See pa.ABE for usage.

Note

The code for impact of deviations from planned sample size tries to keep the degree of imbalance as low as possible between (sequence) groups. This results in a lesser decrease of power than more extreme dropout-patterns.

Author(s)

Idea and original code by H. Schütz
with modifications by D. Labes to use PowerTOST infrastructure.

References

See http://forum.bebac.at/mix_entry.php?id=13376.

See Also

power.scABEL, power.RSABE, known.designs, print.pwrA, plot.pwrA, pa.ABE, pa.NTIDFDA

Examples

```
# using the defaults:
# design="2x3x3", targetpower=0.8, minpower=0.7, theta0/GMR=0.90
# widened BE margins from defaults of sampleN.scABEL() 0.7462 ... 1.3402
# 1E5 sims in power.scABEL()
# not run due to timing policy of CRAN, may run some ten seconds
## Not run:
# implicit print & plot
pa.scABE(CV=0.4)
## End(Not run)
```

power.2TOST *Power for two simultaneous TOST procedures*

Description

Calculates the exact power of two simultaneous TOST procedures (where the two parameters of the two TOSTs are correlated with some correlation) for various study designs used in BE studies

Usage

```
power.2TOST(alpha = c(0.05, 0.05), logscale = TRUE, theta0, theta1, theta2,
            CV, n, rho, design = "2x2", robust = FALSE, nsims, setseed = TRUE,
            details = FALSE)
```

Arguments

alpha	Vector; contains one-sided significance level for each of the two TOSTs. For one TOST, by convention mostly set to 0.05.
logscale	Should the data used on log-transformed (TRUE) or on original scale (FALSE)? Defaults to TRUE.
theta1	Vector; contains lower bioequivalence limit for each of the two TOSTs. In case of logscale=TRUE it is given as ratio, otherwise as diff. to 1. Defaults to c(0.8, 0.8) if logscale=TRUE or to c(-0.2, -0.2) if logscale=FALSE.
theta2	Vector; contains upper bioequivalence limit for each of the two TOSTS. If not given theta2 will be calculated as 1/theta1 if logscale=TRUE or as -theta1 if logscale=FALSE.
theta0	Vector; contains 'true' assumed bioequivalence ratio for each of the two TOSTs. In case of logscale=TRUE each element must be given as ratio, otherwise as difference to 1. See examples. Defaults to c(0.95, 0.95) if logscale=TRUE or to c(0.05, 0.05) if logscale=FALSE.
CV	Vector of coefficient of variations (given as as ratio, *e.g.*, 0.2 for 20%). In case of cross-over studies this is the within-subject CV, in case of a parallel-group design the CV of the total variability. In case of logscale=FALSE CV is assumed to be the respective standard deviation.
n	Number of subjects under study. Is total number if given as scalar, else number of subjects in the (sequence) groups. In the latter case the length of n vector has to be equal to the number of (sequence) groups.
rho	Correlation between the two PK metrics (*e.g.*, AUC and Cmax) under consideration. This is defined as correlation between the estimator of the treatment difference of PK metric one and the estimator of the treatment difference of PK metric two.
design	Character string describing the study design. See known.designs() for designs covered in this package.

`robust`	Defaults to FALSE. With that value the usual degrees of freedom will be used. Setting to TRUE will use the degrees of freedom according to the 'robust' evaluation (aka Senn's basic estimator). These df are calculated as `n-seq`. See `known.designs()$df2` for designs covered in this package. Has only effect for higher-order crossover designs.
`nsims`	Number of studies to simulate. Defaults to 1E5.
`setseed`	Logical; if TRUE, the default, a seed of 1234567 is set.
`details`	Logical; if TRUE, run time will be printed. Defaults to FALSE.

Details

Calculations are based on simulations and follow the distributional properties as described in Phillips. This is in contrast to the calculations via the 4-dimensional non-central t-distribution as described in Hua et al which was implemented in versions up to 1.4-6.

The formulas cover balanced and unbalanced studies w.r.t (sequence) groups.

In case of parallel group design and higher order crossover designs (replicate crossover or crossover with more than two treatments) the calculations are based on the assumption of equal variances for Test and Reference products under consideration.

The formulas for the paired means 'design' do not take an additional correlation parameter into account. They are solely based on the paired *t*-test (TOST of differences = zero).

Value

Value of power.

Note

If n is given as scalar (total sample size) and this number is not divisible by the number of (sequence) groups of the design an unbalanced design with small imbalance is assumed. A corresponding message is thrown showing the assumed numbers of subjects in (sequence) groups.
The function does not vectorize properly if design is a vector. Moreover, theta0 and CV must be of length two, thus further vectorizing is not possible.
Other vector input is not tested yet.

Author(s)

Benjamin Lang and Detlew Labes

References

Phillips KF.
Power for Testing Multiple Instances of the Two One-Sided Tests Procedure
Int J Biostat. 2009;5(1):Article 15. doi: 10.2202/15574679.1169

Hua SY, Xu S, D'Agostino RB Sr.
Multiplicity adjustments in testing for bioequivalence
Stat Med. 2015;34(2):215–31. doi: 10.1002/sim.6247

Lang B, Fleischer F.
Letter to the Editor: Comments on 'Multiplicity adjustments in testing for bioequivalence'
Stat Med. 2016;35(14):2479–80. doi: 10.1002/sim.6488

See Also

sampleN.2TOST, known.designs

Examples

```
# Power for the 2x2x2 cross-over design with 24 subjects, intra-subject
# standard deviation of 0.3 (CV = 30.7%) and assumed ratios of 1.05 for both
# parameters, and correlation 0.75 between parameters (using all the other
# default values)
power.2TOST(theta0 = rep(1.05, 2), CV = rep(se2CV(0.3), 2),
            n = 24, rho = 0.75)
# should give: 0.38906

# Setting as before but use rho ~ 1 to replicate result of power.TOST()
p1 <- power.2TOST(theta0 = rep(1.05, 2), CV = rep(se2CV(0.3), 2),
                  n = 24, rho = 0.99999999, nsims=1E6)
p2 <- power.TOST(theta0 = 1.05, CV = se2CV(0.3), n = 24)
all.equal(p1, p2, tolerance = 1e-03)
```

---

power.dp *Power of dose-proportionality studies evaluated via Power model*

Description

Calculates the power of dose-proportionality studies using the Power model for cossover (Latin square) or parallel group designs via a confidence interval equivalence criterion.

Usage

```
power.dp(alpha = 0.05, CV, doses, n, beta0, theta1 = 0.8, theta2 = 1/theta1,
         design = c("crossover", "parallel", "IBD"), dm=NULL, CVb)
```

Arguments

alpha	Type 1 error. Usually taken as 0.05.
CV	Coefficient of variation for intra-subject variability if design="crossover" or CV of total variability in case of design="parallel".
doses	Vector of dose values. At least 2 doses have to be given.

n	Number of subjects. Is total number if given as scalar, else number of subjects in the (sequence) groups. In the latter case the length of n vector has to be the same as length of vector doses. n has to be >2.
beta0	'True' slope of power model. If missing defaults to 1+log(0.95)/log(rd) where rd is the ratio of highest to lowest dose.
theta1	Lower acceptance limit for the ratio of dose normalized means (Rdmn). Transformes into slope acceptance range as described under item beta0.
theta2	Upper acceptance limit for the ratio of dose normalized means (Rdmn).
design	Crossover design (default), parallel group design or incomplete block design (IBD). Crossover design means Latin square design with number of doses as dimension.
dm	'Design matrix' of the incomplete block design (IBD) if design="IBD". This matrix contains the sequences in rows and periods in columns. The entry (i,j) of the design matrix corresponds to the dose (index) a subject with i-th sequence gets in the j-th period. Can be obtained f.i. via functions of package 'crossdes' or via function bib.CL().
CVb	Coefficient of variation of the between-subject variability. Only necessary if design="IBD". Will be set to 2*CV if missing. Set CVb=0 if an all-effects-fixed model shall be used. This model gives higher power than the random subject effects model.

Details

The power calculations are based on TOST for testing equivalence of the slope of the Power model with alternativ hypothesis slope = 1.
Power is calculated via non-central t-approximation only.
The calculations are based on mixed effects model (random intercept aka random subject effect). For design="cossover" or design="parallel" the results coincide with all-effects-fixed model.

Value

Value of power according to the input arguments.

Warning

This function is 'experimental' only since it is not thorougly tested yet. Especially for design="IBD" reliable test cases are missing.

Author(s)

D. Labes

References

Patterson S, Jones B.
Bioequivalence and Statistics in Clinical Pharmacology

Boca Ration: Chapman & Hall/CR: 2006, p.239.
(contains presumably a bug)

Sethuraman VS, Leonov S, Squassante L, Mitchell TR, Hale MD.
Sample size calculation for the Power Model for dose proportionality studies
Pharm Stat. 2007;6(1):35–41. doi: 10.1002/pst.241

Hummel J, McKendrick S, Brindley C, French R.
Exploratory assessment of dose proportionality: review of current approaches and proposal for a practical criterion
Pharm. Stat. 2009;8(1):38–49. doi: 10.1002/pst.326

See Also

sampleN.dp, bib.CL

Examples

```
# using all the defaults, i.e. latin square crossover design, alpha=0.05,
# beta0=1+log(0.95)/log(rd), theta1=0.8, theta2=1.25
power.dp(CV=0.2, doses=c(1,2,8), n=15)
#
# period balanced IBD with 3 doses, 2 periods and 3 sequences,
ibd <- matrix(c(1,2,3,2,3,1), nrow=3, ncol=2)
power.dp(CV=0.2, doses=c(1,2,8), n=12, design="IBD", dm=ibd)
# considerably lower than 3x3 Latin square
```

---

power.HVNTID *(Empirical) Power for BE decision via FDA method for highly variable NTIDs*

Description

This function performs the power calculation of the BE decision via the FDA method for highly variable narrow therapeutic index drugs (NTIDs) as described in the FDA Dabigatran / Rivaroxaban guidances based on simulations. The study design could be the full replicate design 2x2x4 with 4-periods or the 2x2x3 replicate design with 3-periods and sequences TRT|RTR.

Usage

```
power.HVNTID(alpha = 0.05, theta1, theta2, theta0, CV, n, design=c("2x2x4", "2x2x3"),
             nsims = 1e+05, details = FALSE, setseed = TRUE)
```

Arguments

alpha	Type I error probability, significance level. Conventionally mostly set to 0.05.
theta1	Conventional lower ABE limit to be applied in the FDA procedure. Defaults to 0.8 if not given explicitly.

theta2	Conventional upper ABE limit to be applied in the FDA procedure. Defaults to 1.25 if not given explicitly.
theta0	'True' or assumed bioequivalence ratio. Defaults to 0.95 if not given explicitly.
CV	Coefficient(s) of variation as ratio. If `length(CV) = 1` the same CV is assumed for Test and Reference. If `length(CV) = 2` the CV for Test must be given in `CV[1]` and for Reference in `CV[2]`.
n	Number of subjects under study. May be given as vector. In that case it is assumed that n contains the number of subjects per sequence groups. Attention! In case of the 2x2x3 (TRT|RTR) design the order of n's important if given as vector. `n[1]` is for sequence group 'TRT' and `n[2]` is for sequence group 'RTR'. If n is given as single number (total sample size) and this number is not divisible by the number of sequences of the design an unbalanced design is assumed. A corresponding message is thrown showing the numbers of subjects in the sequence groups.
design	Design of the study to be planned. 2x2x4 is the full replicate design with 2 sequences and 4 periods. 2x2x3 is the 3-period replicate design with sequences TRT|RTR. Defaults to `design="2x2x4"`.
nsims	Number of simulations to be performed to obtain the empirical power. Defaults to 100 000 = 1e+5.
details	If set to `TRUE` the computational time is shown as well as the components for the BE decision. p(BE-ABE) is the simulated probability for the conventional ABE test. p(BE-sratio) is the probability that the upper 90% confidence limit of the ratio of sWT/sWR is < 2.5.
setseed	Simulations are dependent on the starting point of the (pseudo) random number generator. To avoid differences in power for different runs a `set.seed(123456)` is issued if `setseed=TRUE`, the default.

Details

For deciding BE the study must pass the conventional ABE test (90% CI within the acceptance range) and additional the test that the ratio of sWT/sWR is < 2.5.

The simulations are done via the distributional properties of the statistical quantities necessary for deciding BE based on this method.
Details can be found in a document "Implementation_scaledABE_sims" located in the doc subdirectory of the package.

Value

Returns the value of the (empirical) power if argument `details=FALSE`.

Returns a named vector if argument `details=TRUE`.
p(BE) is the power, p(BE-ABE) is the power of the ABE test alone and p(BE-sratio) is the power of the criterion 'ratio of sWT/sWR is <= 2.5' alone.

Note

The FDA guidances recommend only the full replicate design 2x2x4. The results for the design "2x2x3" are to be considered as experimental since at present not thorougly tested.

Author(s)

D. Labes

References

FDA *Draft Guidance on Dabigatran Etexilate Mesylate*
Recommended Jun 2012. Revised Sep 2015. download

FDA *Draft Guidance on Rivaroxaban*
Recommended Sep 2015. download

See Also

`sampleN.HVNTID` and `power.NTIDFDA`, `sampleN.NTIDFDA` for NTIDs with low variability

Examples

```
# using the defaults:
# GMR=0.95, theta1=0.8, theta2=1.25, full replicate design 2x2x4, 100 000 sims
# and a CV of 0.3 (=30%) for both Reference and Test, with 24 subjects, balanced
power.HVNTID(CV=0.3, n=24)
# should give a power of 0.86354
```

---

power.noninf *Power of the one-sided non-inferiority t-test*

Description

Function calculates of the power of the one-sided non-inferiority t-test for normal or log-normal distributed data.

Usage

```
power.noninf(alpha = 0.025, logscale = TRUE, margin, theta0, CV, n,
             design = "2x2", robust = FALSE)
```

Arguments

alpha	Type I error probability, significance level. Defaults here to 0.025.
logscale	Should the data used on log-transformed or on original scale? TRUE or FALSE. Defaults to TRUE.
margin	Non-inferiority margin. In case of logscale=TRUE it must be given as ratio, otherwise as diff. to 1. Defaults to 0.8 if logscale=TRUE or to -0.2 if logscale=FALSE.
theta0	'True' or assumed bioequivalence ratio or difference. In case of logscale=TRUE it must be given as ratio, otherwise as difference to 1. See examples. Defaults to 0.95 if logscale=TRUE or to -0.05 if logscale=FALSE.
CV	Coefficient of variation as ratio. In case of cross-over studies this is the within-subject CV, in case of a parallel-group design the CV of the total variability.
n	Number of subjects under study. Is total number if given as scalar, else number of subjects in the (sequence) groups. In the latter case the length of n vector has to be equal to the number of (sequence) groups.
design	Character string describing the study design. See known.designs for designs covered in this package.
robust	Defaults to FALSE. With that value the usual degrees of freedom will be used. Set to TRUE will use the degrees of freedom according to the 'robust' evaluation (aka Senn's basic estimator). These df are calculated as n-seq. See known.designs()$df2 for designs covered in this package. Has only effect for higher-order crossover designs.

Details

The power is calculated exact via non-central t-distribution.

Notes on the underlying hypotheses

If the supplied margin is < 0 (logscale=FALSE) or < 1 (logscale=TRUE), then it is assumed higher response values are better. The hypotheses are

```
H0: theta0 <= margin vs. H1: theta0 > margin
```

where `theta0 = mean(test)-mean(reference)` if `logscale=FALSE`

or

```
H0: log(theta0) <= log(margin) vs. H1: log(theta0) > log(margin)
```

where `theta0 = mean(test)/mean(reference)` if `logscale=TRUE`.

If the supplied margin is > 0 (logscale=FALSE) or > 1 (logscale=TRUE), then it is assumed lower response values are better. The hypotheses are

```
H0: theta0 >= margin vs. H1: theta0 < margin
```

where `theta0 = mean(test)-mean(reference)` if `logscale=FALSE`

or

```
H0: log(theta0) >= log(margin) vs. H1: log(theta0) < log(margin)
```

where `theta0 = mean(test)/mean(reference)` if `logscale=TRUE`.
This latter case may also be considered as 'non-superiority'.

Value

Value of power according to the input arguments.

Warning

The function does not vectorize if design is a vector.
The function vectorize properly if CV or theta0 are vectors.
Other vector input is not tested yet.

Note

This function does not rely on TOST but may be useful in planning BE studies if the question is not equivalence but 'non-superiority'.
Hint: Evaluation of Fluctuation in the EMA MR NfG (1999) between modified release formulation and immediate release product.

Author(s)

D. Labes

References

Julious SA
Sample sizes for clinical trials with Normal data
Stat Med. 2004;23(12):1921–86. doi: 10.1002/sim.1783

See Also

`known.designs`, `sampleN.noninf`

Examples

```
# using all the defaults: margin=0.8, theta0=0.95, alpha=0.025
# log-transformed, design="2x2"
# should give: 0.4916748
power.noninf(CV=0.3, n=24)
```

power.NTIDFDA *(Empirical) Power for BE decision via FDA method for NTIDs*

Description

This function performs the power calculation of the BE decision via the FDA method for narrow therapeutic index drugs (NTID's) by simulations. The study design could be the full replicate design 2x2x4 with 4-periods or the 2x2x3 replicate design with sequences TRT|RTR.

Usage

```
power.NTIDFDA(alpha = 0.05, theta1, theta2, theta0, CV, n, design=c("2x2x4", "2x2x3"),
              nsims = 1e+05, details = FALSE, setseed = TRUE)
```

Arguments

alpha	Type I error probability, significance level. Conventionally mostly set to 0.05.
theta1	Conventional lower ABE limit to be applied in the FDA procedure. Defaults to 0.8 if not given explicitly.
theta2	Conventional upper ABE limit to be applied in the FDA procedure. Defaults to 1.25 if not given explicitly.
theta0	'True' or assumed bioequivalence ratio. Attention! Defaults here to 0.975 if not given explicitly. The value was chosen nearer to 1 because the potency (contents) settings for NTID's are tightened by the FDA.
CV	Coefficient(s) of variation as ratio. If `length(CV) = 1` the same CV is assumed for Test and Reference. If `length(CV) = 2` the CV for Test must be given in CV[1] and for Reference in CV[2].
n	Number of subjects under study. May be given as vector. In that case it is assumed that n contains the number of subjects per sequence groups. Attention! In case of the 2x2x3 (TRT|RTR) design the order of n's important if given as vector. n[1] is for sequence group 'TRT' and n[2] is for sequence group 'RTR'. If n is given as single number (total sample size) and this number is not divisible by the number of sequences of the design an unbalanced design is assumed. A corresponding message is thrown showing the numbers of subjects in the sequence groups.
design	Design of the study to be planned. 2x2x4 is the full replicate design with 2 sequences and 4 periods. 2x2x3 is the 3-period replicate design with sequences TRT|RTR. Defaults to `design="2x2x4"`.

`nsims` Number of simulations to be performed to obtain the empirical power. Defaults to 100 000 = 1e+5.

`details` If set to `TRUE` the computational time is shown as well as the components for the BE decision.
p(BE-ABE) is the simulated probability for the conventional ABE test. p(BE-sABEc) is the probability that the 95% CI of the ABE criterion is <0.
p(BE-sratio) is the probability that the ratio of sWT/sWR is <= 2.5.

`setseed` Simulations are dependent on the starting point of the (pseudo) random number generator. To avoid differences in power for different runs a `set.seed(123456)` is issued if `setseed=TRUE`, the default.

Details

The linearized scaled ABE criterion is calculated according to the SAS code given in the FDA Warfarin guidance. For deciding BE the study must pass that criterion, the conventional ABE test and additional the test that the ratio of sWT/sWR is <= 2.5.

The simulations are done via the distributional properties of the statistical quantities necessary for deciding BE based on these method.
Details can be found in a document "Implementation_scaledABE_sims" located in the doc subdirectory of the package.

Value

Returns the value of the (empirical) power if argument `details=FALSE`.

Returns a named vector if argument `details=TRUE`.
p(BE) is the power, p(BE-sABEc) is the power of the BE test via scaled ABE criterion alone, p(BE-ABE) is the power of the conventional ABE test alone and p(BE-sratio) is the power of the criterion 'ratio of sWT/sWR is <= 2.5' alone.

Note

The FDA method is described for the ABE limits 0.8 ... 1.25 only. Setting theta1, theta2 to other values may not be reasonable and is not tested.

The results for the design "2x2x3" are to be considered as experimental since at present not thoroughly tested.

Author(s)

D. Labes

References

FDA *Draft Guidance on Warfarin Sodium*
Recommended Dec 2012. download

Yu LX et al.
Novel bioequivalence approach for narrow therapeutic index drugs
Clin Pharmacol Ther. 2015;97(3):286–91. doi: 10.1002/cpt.28

Jiang W et al.
A Bioequivalence Approach for Generic Narrow Therapeutic Index Drugs: Evaluation of the Reference-Scaled Approach and Variability Comparison Criterion
AAPS J. 2015;17(4):891–901. doi: 10.1208/s1224801597535

Endrényi L, Tóthfalusi L.
Determination of Bioequivalence for Drugs with Narrow Therapeutic Index: Reduction of the Regulatory Burden
J Pharm Pharm Sci. 2013;16(5):676–82. free download

See Also

`sampleN.NTIDFDA`
and `power.HVNTID`, `sampleN.HVNTID` for NTIDs with high variability

Examples

```
# using the all defaults:
# GMR=0.975, theta1=0.8, theta2=1.25, 100 000 sims
# and a CV of 0.1 (= 10%) with 12 subjects, balanced
power.NTIDFDA(CV=0.1, n=12)
# should give a power of 0.62553
```

`power.RatioF` *Power for equivalence of the ratio of two means with normality on original scale*

Description

Calculates the power of the test of equivalence of the ratio of two means with normality on original scale.
This test is based on Fieller's confidence ('fiducial') interval and Sasabuchi's test (again a TOST procedure).

Usage

```
power.RatioF(alpha = 0.025, theta1 = 0.8, theta2, theta0 = 0.95, CV, CVb, n,
             design = "2x2", setseed=TRUE)
```

Arguments

`alpha`	Type I error probability, aka significance level. Defaults here to 0.025 because this function is intended for studies with clinical endpoints.
`theta1`	Lower bioequivalence limit. Typically 0.8 (default).

theta2	Upper bioequivalence limit. Typically 1.25. Is set to 1/theta1 if missing.
theta0	'True' or assumed bioequivalence ratio. Typically set to 0.95 for planning.
CV	Coefficient of variation as ratio. In case of design="parallel" this is the CV of the total variability, in case of design="2x2" the intra-subject CV (CVw in the reference).
CVb	CV of the between-subject variability. Only necessary for design="2x2".
n	Number of subjects to be planned. n is for both designs implemented the **total** number of subjects.
design	A character string describing the study design. design="parallel" or design="2x2" allowed for a two-parallel group design or a classical TR/RT crossover design.
setseed	If set to TRUE the dependence of the power from the state of the random number generator is avoided. With setseed = FALSE you may see the dependence from the state of the random number generator.

Details

The power is calculated exact using the bivariate non-central t-distribution via function pmvt() from the package mvtnorm.
Due to the calculation method of the used package mvtnorm - randomized Quasi-Monte-Carlo - these probabilities are dependent from the state of the random number generator within the precision of the power. See argument setseed.

Value

Value of power according to the input.

Note

This function is intended for studies with clinical endpoints.
In such studies the 95% confidence intervals are usually used for equivalence testing.
Therefore alpha defaults here to 0.025.
See CPMP/EWP/482/99 "Points to consider on switching between superiority and non-inferiority" EMEA, London (2000).

The formulas given in the references rely on the assumption of equal variances in the two treatment groups for the parallel group design or on assuming equal within-subject and between-subject variabilities for the 2x2 crossover design.

Author(s)

D. Labes

References

Hauschke D, Kieser M, Diletti E, Burke M.
Sample size determination for proving equivalence based on the ratio of two means for normally distributed data
Stat Med. 1999;18(1):93–105. doi: 10.1002/(SICI)1097-0258(19990115)18:1<93::AID-SIM992>3.0.CO;2-8

Hauschke D, Steinijans V, Pigeot I.
Bioequivalence Studies in Drug Development
Chichester: Wiley; 2007. Chapter 10.

See Also

sampleN.RatioF

Examples

```
# power for alpha=0.025, ratio0=0.95, theta1=0.8, theta2=1/theta1=1.25
# within-subject CV=0.2, between-subject CV=0.4
# 2x2 crossover study, n=24
# using all the defaults:
power.RatioF(CV=0.2, CVb=0.4, n=24)
# gives [1] 0.7315357
```

---

power.RSABE *(Empirical) Power for BE decision via linearized scaled ABE criterion*

Description

This function performs the power calculation of the BE decision via linearized scaled ABE criterion by simulations as recommended by the FDA.

Usage

```
power.RSABE(alpha = 0.05, theta1, theta2, theta0, CV, n,
            design = c("2x3x3", "2x2x4", "2x2x3"), regulator,
            nsims = 1e+05, details = FALSE, setseed=TRUE)
```

Arguments

alpha	Type I error probability, significance level. Conventionally mostly set to 0.05.
theta1	Conventional lower ABE limit to be applied in the mixed procedure if CVsWR <= CVswitch. Also lower limit for the point estimate constraint. Defaults to 0.8 if not given explicitly.
theta2	Conventional upper ABE limit to be applied in the mixed procedure if CVsWR <= CVswitch. Also upper limit for the point estimate constraint. Defaults to 1.25 if not given explicitly.

`theta0` 'True' or assumed bioequivalence ratio.
Defaults to 0.90 according to the two Laszlo's if not given explicitly.

`CV` Coefficient(s) of variation as ratio.
If `length(CV) = 1` the same CV is assumed for Test and Reference.
If `length(CV) = 2` the CV for Test must be given in `CV[1]` and for Reference in `CV[2]`.

`n` Number of subjects under study.
May be given as vector. In that case it is assumed that n contains the number of subjects in the sequence groups.

If n is given as single number (total sample size) and this number is not divisible by the number of sequences of the design an unbalanced design is assumed. A corresponding message is thrown showing the numbers of subjects in sequence groups used.
Attention! In case of the 2x2x3 (TRT|RTR) design the order of sample sizes / sequence is important if given as a vector. `n[1]` is for sequence group 'TRT' and `n[2]` is for sequence group 'RTR'.

`design` Design of the study to be planned.
2x3x3 is the partial replicate design (TRR|RTR|RRT).
2x2x4 is the full replicate design with 2 sequences and 4 periods.
2x2x3 is the 3-period design with sequences TRT|RTR.
Defaults to `design="2x3x3"`.

`regulator` Regulatory settings for RSABE.
May be given as character from the choices "EMA" or "FDA" or as an object of class 'regSet' (see `reg_const`).
Defaults to `regulator="FDA"` if missing.
This argument may be given also in lower case if given as character.

Also the linearized scaled ABE criterion is usually calculated with the FDA constant r_const=log(1.25)/0.25 you can override this behavior to use the EMA setting r_const=0.76 to avoid the discontinuity at CV=30% and be more stringent.

`nsims` Number of simulations to be performed to obtain the empirical power. Defaults to 100 000 = 1e+5.
If simulations are aimed for empirical alpha `nsims=1e+06` is recommended.

`details` If set to `TRUE` the computational time is shown as well as the components for the BE decision.
p(BE-sABEc) is the probability that the 95% CI of the ABE criterion is <0.
p(BE-PE) is the probability that the point estimate is within theta1 ... theta2.
p(BE-ABE) is the simulated probability for the conventional ABE test given for comparision purposes.

`setseed` Simulations are dependent on the starting point of the (pseudo) random number generator. To avoid differences in power for different runs a set.seed() is issued if `setseed=TRUE`, the default.

Details

The linearized scaled ABE criterion is calculated according to the SAS code given in the FDA progesterone guidance.

The simulations are done via the distributional properties of the statistical quantities necessary for deciding BE based on scaled ABE criterion.
Details can be found in a document "Implementation_scaledABE_simsVx.yy.pdf" located in the doc subdirectory of the package.

If a CVcap is defined for the regulator, the BE decision is based on the inclusion of the CI in the capped widened acceptance limits in case of CVwR > CVcap. This resembles method Howe-EMA in Munoz et al. and is the standard behavior now if `regulator="EMA"` is choosen.

Value

Returns the value of the (empirical) power if argument `details=FALSE`.

Returns a named vector if argument `details=TRUE`.
p(BE) is the power, p(BE-sABEc) is the power of the scaled ABE criterion alone and p(BE-pe) is the power of the criterion 'point estimat within acceptance range' alone.
p(BE-ABE) is the power of the conventional ABE test given for comparative purposes.

Warning

In case of the design 2x2x3 heteroscedasticity (i.e. CVwT not equal to CVwR) may lead to poor agreement of the power values compared to those calculated via the 'classical' way of subject data simulations if the design is unbalanced in respect to the number of subjects in the sequence groups. Therefore, the function issues a warning for that cases.

Author(s)

D. Labes

References

FDA *Draft Guidance on Progesterone*
Recommended Apr 2010. Revised Feb 2011. download

Tóthfalusi, L, Endrényi, L.
Sample Sizes for Designing Bioequivalence Studies for Highly Variable Drugs
J Pharm Pharmaceut Sci. 2011;15(1):73–84. free download

Tóthfalusi L, Endrényi L, García Arieta A.
Evaluation of Bioequivalence for Highly Variable Drugs with Scaled Average Bioequivalence
Clin Pharmacokin. 2009;48(11):725–43. doi: 10.2165/11318040000000000000

Muñoz J, Alcaide D, Ocaña J.
Consumer's risk in the EMA and FDA regulatory approaches for bioequivalence in highly variable drugs
Stat Med. 2015;35(12):1933–43. doi: 10.1002/sim.6834

See Also

sampleN.RSABE, power.scABEL

Examples

```
# using all the defaults:
# design="2x3x3" -> partial replicate
# ABE limits, PE constraint 0.8-1.25
# true ratio =0.90, 1E+5 simulations
power.RSABE(CV=0.4, n=36)
# should give
# [1] 0.83634
#
# to explore the simulation error due to the state of the
# random number generator
power.RSABE(CV=0.4, n=36, setseed=FALSE)
# will give something like
# [1] 0.83725
#
# explore pure RSABE (without mixed method, without pe constraint)
rs <- reg_const("FDA")
rs$CVswitch <- 0
rs$pe_constr <- FALSE
power.RSABE(CV=0.4, n=36, regulator=rs)
# should give
# [1] 0.84644
```

---

power.RSABE2L.sdsims *(Empirical) Power of BE Decision via Reference Scaled ABE*

Description

These function performs the power calculation of the BE decision via the reference scaled ABE based on **subject data** simulations. Implemented are the methods ABEL, Hyslop and 'exact' as described in the references.
The estimation method of the key statistics needed to perform the RSABE decision is the usual ANOVA.

Usage

```
power.RSABE2L.sdsims(alpha = 0.05, theta1, theta2, theta0, CV, n,
                     design = c("2x3x3", "2x2x4", "2x2x3"), design_dta=NULL,
                     SABE_test="exact", regulator, nsims = 1e+05,
                     details = FALSE, setseed = TRUE, progress)
```

Arguments

alpha	Type I error probability, significance level. Conventionally mostly set to 0.05.
theta1	Conventional lower ABE (Average Bioequivalence) limit to be applied in the mixed procedure if CVsWR <= CVswitch. Also lower limit for the point estimate constraint. Defaults to 0.8 if not given explicitly.
theta2	Conventional upper ABE limit to be applied in the mixed procedure if CVsWR <= CVswitch. Also upper limit for the point estimate constraint. Defaults to 1.25 if not given explicitly.
theta0	'True' or assumed bioequivalence ratio. Defaults to 0.90 according to the two Laszlos if not given explicitly.
CV	Coefficient(s) of variation as ratio. If length(CV) = 1 the same CV is assumed for Test and Reference. If length(CV) = 2 the CV for Test must be given in CV[1] and for Reference in CV[2].
n	Number of subjects under study. May be given as vector. In that case it is assumed that n contains the number of subjects in the sequence groups. If n is given as single number (total sample size) and this number is not divisible by the number of sequences of the design an unbalanced design is assumed. A corresponding message is thrown (showing the numbers of subjects in sequence groups). Attention! In case of the "2x2x3" (TRT|RTR) design the order of sample sizes per sequence is important if given as vector. n[1] is for sequence group 'TRT' and n[2] is for sequence group 'RTR'.
design	Design of the study to be planned. "2x3x3" is the partial replicate design (TRR|RTR|RRT). "2x2x4" is the full replicate design with 2 sequences and 4 periods (TRTR|RTRT). "2x2x3" is the 3-period design with sequences TRT|RTR. Defaults to design="2x3x3".
design_dta	Alternatively to using the arguments design and n the design may be defined via a data.frame with columns subject, sequence, period and tmt. This feature is experimental in the sense that the data.frame is not checked for complying with the assumed structure. If you use the argument design_dta you don't need to specify the arguments design and n. The default design_dta=NULL means that design and n are used for the internal construction of the design data.frame.
SABE_test	This argument specifies the test method to be used for the reference scaled ABE decision. Default is the "exact" ncTOST method of the two Laszlos. Other choices are "ABEL", "hyslop" and "fda". See Details.
regulator	Regulatory settings for the widening of the BE acceptance limits. May be given as character "EMA" or as an object of class 'regSet' (see reg_const). Defaults to regulator="EMA" if missing.

	This argument may be given also in lower case if given as character. If given as object of class 'regSet' the component `est_method` can not be "ISC".
`nsims`	Number of simulations to be performed to obtain the empirical power. Defaults to 100,000 = 1e+05. If simulations are aimed for empirical alpha `nsims=1e+06` is recommended.
`details`	If set to TRUE the computational time is shown as well as the components for the BE decision. p(BE-RSABE) is the probability of a positive outcome of the SABE test. p(BE-PE) is the probability that the point estimate is within theta1 ... theta2. p(BE-ABE) is the simulated probability for the conventional ABE test.
`setseed`	Simulations are dependent on the starting point of the (pseudo) random number generator. To avoid differences in power for different runs a set.seed() is issued if `setseed=TRUE`, the default.
`progress`	Should a progressbar be shown? Defaults to TRUE if missing and `nsims >5e5`.

Details

The methods rely on the analysis of log-transformed data, *i.e.*, assumes a log-normal distribution on the original scale.

The data.frame with columns subject, sequence, period and tmt necessary for evalution of simulated subject data is constructed internally from the arguments design and n or may be given user defined via the argument design_dta. The last option is usefull if missing data have to be considered or if designs have to be evaluated which are not in the list of argument design.

The estimation method for obtaining the statistics necessary to perform the reference scaled ABE decision is the usual ANOVA with effects treatment, period, sequence and subject within sequence for the evaluation of all data and period, sequence and subject within sequence for the evaluation of the Reference formulation data only.

The SABE tests implemented are:

"exact"	- 'exact' based method of the two Laszlos (see references, called there ncTOST)
"ABEL"	- Average bioequivalence with expanding limits
"hyslop"	- BE decision via the linearized RSABE criterion and its upper 95% CI
"fda"	- Hyslop with an additional bias correction term as implemented in the SAS code of the FDA Guidance on Progeste

Value

Returns the value of the (empirical) power if argument `details=FALSE`.

Returns a named vector if argument `details=TRUE`.
p(BE) is the power, p(BE-RSABE) is the power of using the reference scaled ABE alone, and p(BE-pe) is the power of the criterion 'point estimat within acceptance range' alone. p(BE-ABE) is the power of the conventional ABE test given for comparative purposes.

Note

The function is relatively slow. The run-time for 1 Mio. simulations is between ~ 1 up to 6 minutes for n=12 or n=120 and 1 Mio. sim's (see the call under examples) on a machine with an Intel core i7 processor.
Thus be patient and go for a cup of coffee if you use this function with higher sample sizes and aim for estimating the type 1 error!

Author(s)

D. Labes

References

FDA. *Draft Guidance on Progesterone*
Recommended Apr 2010. Revised Feb 2011. download

Tóthfalusi L, Endrényi L.
An Exact Procedure for the Evaluation of Reference-Scaled Average Bioequivalence
AAPS J. 2016;18(2):476–89. doi: 10.1208/s1224801698736.

Tóthfalusi L, Endrényi L.
Algorithms for evaluating reference scaled average bioequivalence: power, bias, and consumer risk
Stat Med. 2017;36(27):4378–4390. doi: 10.1002/sim.7440

See Also

power.RSABE, reg_const

Examples

```
# Not run due to timing policy of CRAN
## Not run:
# pure EMA settings without mixed procedure, cap on widening and PE constraint
# as in the reference from 2017
reg           <- reg_const("EMA")
reg$CVswitch  <- 0
reg$CVcap     <- Inf
reg$pe_constr <- FALSE
reg$name      <- "EMA pure"
power.RSABE2L.sds(CV=0.4, n=12, theta0=exp(0.05), design="2x2x4",
                  regulator=reg, nsims=50000)
# should give:
# [1] 0.46504 (compared to 47.1% in the 2017 reference)
## End(Not run)
```

power.scABEL *(Empirical) Power of BE decision via scaled (widened) BE acceptance limits*

Description

These function performs the power calculation of the BE decision via scaled (widened) BE acceptance limits by simulations.

Usage

```
power.scABEL(alpha = 0.05, theta1, theta2, theta0, CV, n,
             design = c("2x3x3", "2x2x4", "2x2x3"), regulator,
             nsims = 1e+05, details = FALSE, setseed = TRUE)
```

Arguments

alpha	Type I error probability, significance level. Conventionally mostly set to 0.05.
theta1	Conventional lower ABE limit to be applied in the mixed procedure if CVsWR <= CVswitch. Also lower limit for the point estimate constraint. Defaults to 0.8 if not given explicitly.
theta2	Conventional upper ABE limit to be applied in the mixed procedure if CVsWR <= CVswitch. Also upper limit for the point estimate constraint. Defaults to 1.25 if not given explicitly.
theta0	'True' or assumed bioequivalence ratio. Defaults to 0.90 according to the two Laszlo's if not given explicitly.
CV	Coefficient(s) of variation as ratio. If `length(CV) = 1` the same CV is assumed for Test and Reference. If `length(CV) = 2` the CV for Test must be given in CV[1] and for Reference in CV[2].
n	Number of subjects under study. May be given as vector. In that case it is assumed that n contains the number of subjects in the sequence groups. If n is given as single number (total sample size) and this number is not divisible by the number of sequences of the design an unbalanced design is assumed. A corresponding message is thrown showing the numbers of subjects in sequence groups. Attention! In case of the 2x2x3 (TRT|RTR) design the order of n's is important if given as vector. n[1] is for sequence group 'TRT' and n[2] is for sequence group 'RTR'.
design	Design of the study to be planned. 2x3x3 is the partial replicate design (TRR/RTR/RRT). 2x2x4 is the full replicate design with 2 sequences and 4 periods. 2x2x3 is the 3-period design with sequences TRT|RTR. Defaults to `design="2x3x3"`.

regulator	Regulatory settings for the widening of the BE acceptance limits. May be given as character from the choices "EMA", "HC" "FDA" or as an object of class 'regSet' (see reg_const). Defaults to regulator="EMA" if missing. This argument may be given also in lower case if given as character. The former regulator="ANVISA" is no longer allowed. Use "EMA" since ANVISA now recommends the use of EMA regulatory settings.
nsims	Number of simulations to be performed to obtain the empirical power. Defaults to 100 000 = 1e+05. If simulations are aimed for empirical alpha nsims=1e+06 is recommended.
details	If set to TRUE the computational time is shown as well as the components for the BE decision. p(BE-wABEL) is the probability that the CI is within (widened) limits. p(BE-PE) is the probability that the point estimate is within theta1 ... theta2. p(BE-ABE) is the simulated probability for the conventional ABE test.
setseed	Simulations are dependent on the starting point of the (pseudo) random number generator. To avoid differences in power for different runs a set.seed() is issued if setseed=TRUE, the default.

Details

The methods rely on the analysis of log-transformed data, i.e. assume a log-normal distribution on the original scale.

The widened BE acceptance limits will be calculated by the formula
[lBEL,uBEL] =exp(-+ r_const*sWR)
with r_const the regulatory constant and sWR the standard deviation of the within subjects variability of the Reference. r_const=0.76 (~log(1.25)/0.29356) is used in case of regulator="EMA" or regulator="HC" and in case of regulator="FDA" r_const=0.89257...(=log(1.25)/0.25). If the CVwR of the Reference is < CVswitch=0.3 the conventional ABE limits apply (mixed procedure).

In case of regulator="EMA" a cap is placed on the widened limits if CVwr>0.5, i.e. the widened limits are held at value calculated for CVwR=0.5. In case of regulator="HC" the capping is done such that the acceptance limits are 0.6666 ... 1.5 at maximum.

The former inofficial regulatory settings for regulator="ANVISA" are now covered by regulator="EMA".

The simulations are done via the distributional properties of the statistical quantities necessary for deciding BE based on widened ABEL.
For more details see a document "Implementation_scaledABE_simsVx.yy.pdf" in the /doc subdirectory of the package.

Function power.scABEL() implements the simulation via distributional characteristics of the 'key' statistics obtained from the EMA recommended evaluation via ANOVA if regulator="EMA" or if the regulator component est_method is set to "ANOVA" if regulator is an object of class 'regSet'. Otherwise the simulations are based on the distributional characteristis of the 'key' statistics obtained from evaluation via intra-subject contrasts (ISC), as recommended by the FDA.

Value

Returns the value of the (empirical) power if argument `details=FALSE`.

Returns a named vector if argument `details=TRUE`.
p(BE) is the power, p(BE-wABEL) is the power of the widened ABEL criterion alone and p(BE-pe) is the power of the criterion 'point estimat within acceptance range' alone. p(BE-ABE) is the power of the conventional ABE test given for comparative purposes.

Warning

Cross-validation of the simulations as implemented here and via the 'classical' subject data simulation have shown somewhat unsatisfactory results for the 2x3x3 design if the variabilities for Test and Reference are different.
The function `power.scABEL()` therefore gives a warning if calculations with different CVwT and CVwR are requested for the 2x3x3 partial replicate design. For more details see the above mentioned document "Implementation_scaledABE_simsVy.xx.pdf".

Note

In case of `regulator="FDA"` the (empirical) power is only approximate since the BE decision method is not exactly what is expected by the FDA. But the "Two Laszlos" state that the scABEL method should be 'operational equivalent' to the FDA method.
To get the power for the FDA favored method via linearized scaled ABE criterion use function `power.RSABE`.

In case of `regulator="HC"` in function `power.scABEL()`, based on ISC, power is also only approximative since the Health Canada recommends an evaluation via mixed model approach. This could only implemented via subject data simulations which are very time consuming. But ISC may be a good substitute.

Author(s)

D. Labes

References

Tóthfalusi L, Endrényi L.
Sample Sizes for Designing Bioequivalence Studies for Highly Variable Drugs
J Pharm Pharmaceut Sci. 2011;15(1):73–84. free download

See Also

`sampleN.scABEL`, `power.RSABE`, `reg_const`

Examples

```
# using all the defaults:
# design="2x3x3", EMA regulatory settings
# PE constraint 0.8-1.25, cap on widening if CV>0.5
# true ratio =0.90, 1E+6 simulations
```

```
power.scABEL(CV=0.4, n=29)
# should give:
# Unbalanced design. n(i)=10/10/9 assumed.
# [1] 0.66113
#
# with details=TRUE to view the computational time and components
power.scABEL(CV=0.5, n=54, theta0=1.15, details=TRUE)
# should give (times may differ depending on your machine):
# 1e+05sims. Time elapsed (sec): 0.07
#
#      p(BE) p(BE-wABEL)    p(BE-pe)   p(BE-ABE)
#    0.81727     0.82078     0.85385     0.27542
#
# exploring pure ABEL with the EMA regulatory constant
# (without mixed method, without capping, without pe constraint)
rs <- reg_const("EMA")
rs$CVswitch  <- 0
rs$CVcap     <- Inf
rs$pe_constr <- FALSE
power.scABEL(CV=0.5, n=54, theta0=1.15, regulator=rs)
# should give
# [1] 0.8519
```

`power.scABEL.sdsims` *(Empirical) Power of BE decision via scaled (widened) BE acceptance limits*

Description

These function performs the power calculation of the BE decision via scaled (widened) BE acceptance limits based on **subject data** simulations.

Usage

```
power.scABEL.sdsims(alpha = 0.05, theta1, theta2, theta0, CV, n,
                    design = c("2x3x3", "2x2x4", "2x2x3"), design_dta=NULL,
                    regulator, nsims = 1e+05, details = FALSE, setseed = TRUE,
                    progress)
```

Arguments

`alpha`	Type I error probability, significance level. Conventionally mostly set to 0.05.
`theta1`	Conventional lower ABE limit to be applied in the mixed procedure if CVsWR <= CVswitch. Also lower limit for the point estimate constraint. Defaults to 0.8 if not given explicitly.
`theta2`	Conventional upper ABE limit to be applied in the mixed procedure if CVsWR <= CVswitch. Also upper limit for the point estimate constraint. Defaults to 1.25 if not given explicitly.

theta0	'True' or assumed bioequivalence ratio. Defaults to 0.90 according to the two Laszlo's if not given explicitly.
CV	Coefficient(s) of variation as ratio. If length(CV) = 1 the same CV is assumed for Test and Reference. If length(CV) = 2 the CV for Test must be given in CV[1] and for Reference in CV[2].
n	Number of subjects under study. May be given as vector. In that case it is assumed that n contains the number of subjects in the sequence groups. If n is given as single number (total sample size) and this number is not divisible by the number of sequences of the design an unbalanced design is assumed. A corresponding message is thrown showing the numbers of subjects in sequence groups. Attention! In case of the 2x2x3 (TRT|RTR) design the order of n's is important if given as vector. n[1] is for sequence group 'TRT' and n[2] is for sequence group 'RTR'.
design	Design of the study to be planned. 2x3x3 is the partial replicate design (TRR|RTR|RRT). 2x2x4 is the full replicate design with 2 sequences and 4 periods (TRTR|RTRT). 2x2x3 is the 3-period design with sequences TRT|RTR. Defaults to design="2x3x3".
design_dta	Alternatively to using the arguments design and n the design may be defined via a data.frame with columns subject, sequence, period and tmt. This feature is experimental in the sense that the data.frame is not checked for complying with the assumed structure. If you use the argument design_dta you don't need to specify the arguments design and n. The default design_dta = NULL means that design and n are used for the internal construction of the design data.frame.
regulator	Regulatory settings for the widening of the BE acceptance limits. May be given as character == "EMA" or as an object of class 'regSet' (see reg_const). Defaults to regulator="EMA" if missing. This argument may be given also in lower case if given as character. If given as object of class 'regSet' the component est_method can not be "ISC".
nsims	Number of simulations to be performed to obtain the empirical power. Defaults to 100 000 = 1e+05. If simulations are aimed for empirical alpha nsims=1e+06 is recommended.
details	If set to TRUE the computational time is shown as well as the components for the BE decision. p(BE-wABEL) is the probability that the CI is within (widened) limits. p(BE-PE) is the probability that the point estimate is within theta1 ... theta2. p(BE-ABE) is the simulated probability for the conventional ABE test.
setseed	Simulations are dependent on the starting point of the (pseudo) random number generator. To avoid differences in power for different runs a set.seed() is issued if setseed=TRUE, the default.

progress Should a progressbar be shown? Defaults to TRUE if missing and nsims >5E5.

Details

The methods rely on the analysis of log-transformed data, i.e. assume a log-normal distribution on the original scale.

The widened BE acceptance limits will be calculated by the formula
`[lBEL,uBEL] =exp(-+ r_const*sWR)`
with r_const the regulatory constant and sWR the standard deviation of the within subjects variability of the Reference. r_const=0.76 (~log(1.25)/0.29356) is used in case of `regulator="EMA"`. If the CVwR of the Reference is < CVswitch=0.3 the conventional ABE limits apply (mixed procedure). In case of `regulator="EMA"` a cap is placed on the widened limits if CVwr>0.5, i.e. the widened limits are held at value calculated for CVwR=0.5.

The simulations are done by simulating subject data (all effects fixed except the residuals) and evaluating these data via ANOVA of all data to get the point esimate of T vs. R along with its 90% CI and an ANOVA of the data under R(eference) only to get an estimate of s2wR.
The data.frame with columns `subject, sequence, period` and `tmt` necessary for evalution of simulated subject data is constructed internally from the arguments `design` and `n` or may be given user defined via the argument `design_dta`. The last option is usefull if missing data have to be considered or if designs have to be evaluated which are not in the list of argument `design`.
This feature is experimental in the sense that the data.frame is not checked for complying with the assumed structure.

Value

Returns the value of the (empirical) power if argument `details=FALSE`.

Returns a named vector if argument `details=TRUE`.
p(BE) is the power, p(BE-wABEL) is the power of the widened ABEL criterion alone and p(BE-pe) is the power of the criterion 'point estimat within acceptance range' alone. p(BE-ABE) is the power of the conventional ABE test given for comparative purposes.

Note

The function is mainly intended for crosscheck of `power.scABEL()` results.
But may be mandatory for cases where `power.scABEL()` results are inaccurate (low samplesizes and/or heteroscedasticity).
It is relatively slow. The run-time of this function doing 1 Mio sims is between ~ 7-8 sec for n=12 and ~ 3-4 min for n=120 on a machine with an Intel core i7 processor.
Thus be patient and go for a cup of coffee if you use this function with high sample sizes!

Author(s)

D. Labes & B. Lang

References

Tóthfalusi L, Endrényi L.
Sample Sizes for Designing Bioequivalence Studies for Highly Variable Drugs
J Pharm Pharmaceut Sci. 2011;15(1):73–84. free download

See Also

power.scABEL, reg_const

Examples

```
# using all the defaults:
# design="2x3x3", EMA regulatory settings
# PE constraint 0.8-1.25, cap on widening if CV>0.5
# true ratio =0.90, 1E+5 simulations
## Not run: power.scABEL.sdsims(CV=0.4, n=36)
# should give:
# [1] 0.74321
```

---

power.TOST *Power of the classical TOST procedure*

Description

Calculates the exact or approximate power of the two-one-sided t-tests (TOST) procedure for various study designs used in BE studies.

Usage

```
power.TOST(alpha = 0.05, logscale = TRUE, theta1, theta2, theta0, CV, n,
           design = "2x2", method="exact", robust=FALSE)
```

Arguments

alpha	Type I error probability, significance level. By convention mostly set to 0.05.
logscale	Should the data used on log-transformed or on original scale? TRUE or FALSE. Defaults to TRUE.
theta1	Lower bioequivalence limit. In case of logscale=TRUE it is given as ratio, otherwise as diff. to 1. Defaults to 0.8 if logscale=TRUE or to -0.2 if logscale=FALSE.
theta2	Upper bioequivalence limit. If not given theta2 will be calculated as 1/theta1 if logscale=TRUE or as -theta1 if logscale=FALSE.
theta0	'True' or assumed bioequivalence ratio. In case of logscale=TRUE it must be given as ratio, otherwise as difference to 1. See examples. Defaults to 0.95 if logscale=TRUE or to 0.05 if logscale=FALSE

CV	Coefficient of variation as ratio. In case of cross-over studies this is the within-subject CV, in case of a parallel-group design the CV of the total variability.
n	Number of subjects under study. Is total number if given as scalar, else number of subjects in the (sequence) groups. In the latter case the length of n vector has to be equal to the number of (sequence) groups.
design	Character string describing the study design. See known.designs() for designs covered in this package.
method	Method for calculation of the power. Defaults to "exact" in which case the calculation is done based on formulas with Owen's Q. The calculation via Owen's Q can also be choosen with method="owenq". Another exact method via direct integration of the bivariate non-central t-distribution may be chosen with method="mvt". This may have somewhat lower precision compared to Owen's Q and longer run-time. Approximate calculations can be choosen via method="noncentral" or method="nct" for the approximation using the non-central t-distribution. With method="central" or method="shifted" the relative crude approximation via 'shifted' central t-distribution is chosen. The strings for method may be abbreviated.
robust	Defaults to FALSE. With that value the usual degrees of freedom will be used. Set to TRUE will use the degrees of freedom according to the 'robust' evaluation (aka Senn's basic estimator). These df are calculated as n-seq. See known.designs()$df2 for designs covered in this package. Has only effect for higher-order crossover designs.

Details

The exact calculations of power are based on Owen's Q-function or via direct integration of the bivariate non-central t-distribution via function pmvt() of package mvtnorm.
The approximate power is implemented via non-central t-distribution or via 'shifted' central t-distribution.

The formulas cover now balanced and unbalanced studies w.r.t (sequence) groups.

In case of parallel group design and higher order crossover designs (replicate crossover or crossover with more than two treatments) the calculations are based on the assumption of equal variances for Test and Reference products under consideration.

The formulas for the paired means 'design' do not take a correlation parameter into account. They are solely based on the paired t-test (TOST of differences = zero).

Value

Value of power according to the input arguments.

Note

Of course it is highly recommended to use the default `method="exact"` :-)).
There is no reason beside testing and comparative purposes to use an approximation if the exact method is available.

If n is given as scalar (total sample size) and this number is not divisible by the number of (sequence) groups of the design an unbalanced design with small imbalance is assumed. A corresponding message is thrown showing the assumed numbers of subjects in (sequence) groups.
The function does not vectorize properly if design is a vector.
The function vectorizes properly if CV or theta0 are vectors.
Other vector input is not tested yet.

Former function `power2.TOST()` designd to handle unbalanced studies is now removed since `power.TOST()` handles balanced as well as unbalanced designs.

Author(s)

D. Labes
Direct integration of bivariate non-central t-distribution by Benjamin Lang.

References

Phillips KF.
Power of the Two One-Sided Tests Procedure in Bioequivalence
J Pharmacokin Biopharm. 1990;18(2):137–44. doi: 10.1007/BF01063556

Diletti D, Hauschke D, Steinijans VW.
Sample Size Determination for Bioequivalence Assessment by Means of Confidence Intervals
Int J Clin Pharmacol Ther Toxicol. 1991;29(1):1–8.

See Also

`sampleN.TOST`, `known.designs`

Examples

```
# power for the 2x2 cross-over design with 24 subjects and CV 25%
# using all the other default values
power.TOST(CV=0.25, n=24)
# should give: [1] 0.7391155
# nct approximation very good for this configuration
power.TOST(CV=0.25, n=24, method="nct")
# gives also: [1] 0.7391155
# shifted-central-t  approximation
power.TOST(CV=0.25, n=24, method="shifted")
# gives:      [1] 0.7328894

# power for the 2x2 cross-over study with 24 subjects, CV 25%
# and 2 drop-outs in the same sequence group (unbalanced study)
power.TOST(CV=0.25, n=c(10,12))
# should give: [1] 0.6912935
```

```
# not the same compared to the balanced setting
power.TOST(CV=0.25, n=22)
# should give: [1] 0.6953401
```

`power.TOST.sim` *Power of the TOST procedure obtained via simulations*

Description

Power is calculated by simulations of studies (PE via it's normal distribution, MSE via it's associated chi-squared distribution) and application of the two one-sided t-tests. Power is obtained via ratio of studies found BE to # of simulated studies.

Usage

```
power.TOST.sim(alpha = 0.05, logscale = TRUE, theta1, theta2, theta0, CV, n,
               design = "2x2", robust = FALSE, setseed = TRUE, nsims = 1e+05)
```

Arguments

`alpha`	Type I error probability, significance level. By convention mostly set to 0.05.
`logscale`	Should the data used on log-transformed or on original scale? TRUE or FALSE. Defaults to TRUE.
`theta1`	Lower bioequivalence limit. In case of `logscale=TRUE` it is given as ratio, otherwise as diff. to 1. Defaults to 0.8 if `logscale=TRUE` or to -0.2 if `logscale=FALSE`.
`theta2`	Upper bioequivalence limit. If not given theta2 will be calculated as `1/theta1` if `logscale=TRUE` or as `-theta1` if `logscale=FALSE`.
`theta0`	'True' or assumed bioequivalence ratio. In case of `logscale=TRUE` it must be given as ratio, otherwise as difference to 1. See examples. Defaults to 0.95 if `logscale=TRUE` or to 0.05 if `logscale=FALSE`
`CV`	Coefficient of variation as ratio. In case of cross-over studies this is the within-subject CV, in case of a parallel-group design the CV of the total variability.
`n`	Number of subjects under study. Is total number if given as scalar, else number of subjects in the (sequence) groups. In the latter case the length of n vector has to be equal to the number of (sequence) groups.
`design`	Character string describing the study design. See `known.designs()` for designs covered in this package.

robust	Defaults to FALSE. With that value the usual degrees of freedom will be used. Set to TRUE will use the degrees of freedom according to the 'robust' evaluation (aka Senn's basic estimator). These df are calculated as n-seq. See known.designs()$df2 for designs covered in this package. Has only effect for higher-order crossover designs.
setseed	Simulations are dependent on the starting point of the (pseudo) random number generator. To avoid differences in power for different runs a set.seed(1234567) is issued if setseed=TRUE, the default. Set this argument to FALSE to view the variation in power between different runs.
nsims	Number of studies to simulate. Defaults to 1E5 = 100 000.

Value

Value of power according to the input arguments.

Note

This function was intended for internal check of the analytical power calculation methods. Use of the analytical power calculation methods (power.TOST()) for real problems is recommended.
For sufficient precision nsims > 1E5 (default) may be necessary. Be patient if using nsims=1E6. May take some seconds.

Author(s)

D. Labes

See Also

power.TOST,

Examples

```
# using the default design 2x2, BE range 0.8 ... 1.25, logscale, theta0=0.95
power.TOST.sim(alpha=0.05, CV=0.3, n=12)
# should give 0.15054, with nsims=1E6 it will be 0.148533
# exact analytical is
power.TOST(alpha=0.05, CV=0.3, n=12)
# should give 0.1484695

# very unusual alpha setting
power.TOST.sim(alpha=0.9, CV=0.3, n=12)
# should give the same (within certain precision) as
power.TOST(alpha=0.95, CV=0.3, n=12)
# or also within certain precision equal to
power.TOST(alpha=0.95, CV=0.3, n=12, method="mvt")
# SAS Proc Power gives here the incorrect value 0.60525
```

pvalue.TOST *p-value(s) of the TOST procedure*

Description

Calculates the p-value(s) of the TOST procedure via students t-distribution given pe, CV and n.

Usage

```
pvalue.TOST(pe, CV, n, logscale = TRUE, theta1, theta2, design = "2x2",
            robust = FALSE, both = FALSE)
pvalues.TOST(pe, CV, n, logscale = TRUE, theta1, theta2, design = "2x2",
             robust = FALSE, both = TRUE)
```

Arguments

pe	Observed point estimate of the ratio Test vs. Reference (if logscale=TRUE) or of the difference (if logscale=FALSE).
CV	Obseved coefficient of variation as ratio (if logscale=TRUE) or residual error standard deviation (if logscale=FALSE).
n	Total number of subjects if given as scalar. Number of subjects in (sequence) groups if given as vector.
logscale	Should the data be used after log-transformation or on original scale? TRUE or FALSE. Defaults to TRUE.
theta1	Lower bioequivalence limit. In case of logscale=TRUE it has to be given as ratio, otherwise as value < 0. Defaults to 0.8 if logscale=TRUE or to log(0.8)= -0.2231 if logscale=FALSE.
theta2	Upper bioequivalence limit. If not given theta2 will be calculated as 1/theta1 if logscale=TRUE or as -theta1 if logscale=FALSE.
design	Character string describing the study design. See known.designs() for designs covered in this package.
robust	If set to TRUE triggers the use of degrees of freedom according to the 'robust' evaluation (aka Senn's basic estimator). These df are calculated as n-seq. See known.designs()$df2.Has only effect for higher-order crossover designs. Defaults to FALSE. With that value the usual degrees of freedom will be used.
both	Indicates if both p-values (t-tests of pe>=theta1 and pe<=theta2) shall be given back or only the maximum. Defaults to FALSE for the function pvalue.TOST() and to TRUE for the function pvalues.TOST().

Value

Returns the p-value(s).
Returns a vector with named elements "p.left", "p.right" if arguments pe and CV are scalars, else a matrix with columns "p.left", "p.right".
p.left names the p-value of testing HA1: theta>=theta1, p.right the p-value of testing HA2: theta<=theta2 against their respective Nulls.

Note

The formulas implemented cover balanced and unbalanced designs.

In case of argument n given as n(total) and is not divisible by the number of (sequence) groups the total sample size is partitioned to the (sequence) groups to have small imbalance only. A message is given in such cases.

SAS procedure TTEST with the TOST option names p.left = Upper, p.right= Lower according to the tail of the t-distribution to be used for obtaining the p-values.

Author(s)

Benjamin Lang
Man page by D. Labes

References

Schuirmann DJ.
A comparison of the two one-sided tests procedure and the power approach for assessing the equivalence of average bioavailability
J Pharmacokin Biopharm. 1987;15:657–80. doi: 10.1007/BF01068419

Hauschke D, Steinijans V, Pigeot I.
Bioequivalence Studies in Drug Development
Chichester: Wiley; 2007.

See Also

CI.BE

Examples

```
# Defaults: 2x2 crossover, log-transformation
# BE acceptance limits 0.8 ... 1.25, usual df's
# interested in both p-values
pvalues.TOST(pe=0.95, CV=0.3, n=12)
# gives the vector (named elements)
#     p.left    p.right
# 0.09105601 0.02250985
# i.e. 'left' hypothesis H01: theta<=theta1 can't be rejected
# 'right' hypothesis H02: theta>=theta2 can be rejected

# max. p-value only as 'overall' pvalue, preferred by Benjamin
```

```
pvalue.TOST(pe=0.912, CV=0.333, n=24)
# should give 0.08777621, i.e inequivalence can't be rejected
# this is operationally identical to
CI.BE(pe=0.912, CV=0.333, n=24)
# lower limit = 0.7766 outside 0.8 ... 1.25, i.e inequivalence can't be rejected
```

`reg_const` *Constructor of an object with class 'regSet' containing the regulatory settings for ABEL*

Description

This function may be used to define regulatory settings not implemented yet in PowerTOST.

Usage

```
reg_const(regulator, r_const, CVswitch, CVcap, pe_constr)
```

Arguments

`regulator`	Name of the regulatory body as character. Implemented settings are for "EMA", "FDA", "HC". The former (inofficial) settings for "ANVISA" are now covered by the EMA settings. In case of `regulator="USER"` the other arguments must be given. Otherwise they may be missing.
`r_const`	Regulatory constant.
`CVswitch`	CV to switch to the widened acceptance limits.
`CVcap`	CV for capping the widening of the acceptance limits.
`pe_constr`	Logical. Shall pe constraint be applied? Defaults to `TRUE`.

Value

Returns an object of class 'regSet', a list with components

`name`	Name of the settings
`CVswitch`	see arguments
`r_const`	Regulatory constant
`CVcap`	see arguments
`pe_constr`	see arguments
`est_method`	"ANOVA" or "ISC"

Class 'regSet' has a S3 print method.

The component `est_method` is automatically set to "ANOVA", except for `regulator="FDA"` or `regulator="HC"` where "ISC" is used.

Note

The former inofficial regulatory settings for `regulator="ANVISA"` are now covered by `regulator="EMA"`.
See http://forum.bebac.at/mix_entry.php?id=16291.

The settings for CVcap of Health Canada (`regulator="HC"`) were chosen such that the limits of the acceptance range are capped nearly exact to 1/1.5 up to 1.5. Literally it was given rounded to 3 significant digits.
See http://www.hc-sc.gc.ca/dhp-mps/prodpharma/activit/announce-annonce/notice-avis-be-hvdp-nb-pphv-php

Author(s)

D. Labes

Examples

```
# to retrieve the EMA settings
reg_const("EMA")
# to define the old ANVISA settings
reg <- reg_const("USER", r_const=0.76, CVswitch=0.4, CVcap=0.5)
reg$name <- "Old ANVISA"
# Use reg as argument in the power / sample size functions
```

---

sampleN.2TOST *Sample size based on power of two TOSTs*

Description

Calculates the necessary sample size to have at least a given power when two parameters are being tested simultaneously.

Usage

```
sampleN.2TOST(alpha = c(0.05, 0.05), targetpower = 0.8, logscale = TRUE,
              theta0, theta1, theta2, CV, rho, design = "2x2", setseed = TRUE,
              robust = FALSE, print = TRUE, details = FALSE, imax = 100,
              nsims = 1e+05)
```

Arguments

`alpha`	Vector; contains one-sided significance level for each of the two TOSTs. For one TOST, by convention mostly set to 0.05.
`targetpower`	Power to achieve at least. Must be >0 and <1. Typical values are 0.8 or 0.9.
`logscale`	Should the data used on log-transformed (TRUE) or on original scale (FALSE)? Defaults to TRUE.

theta0	Vector; contains 'true' assumed bioequivalence ratio for each of the two TOSTs. In case of logscale=TRUE each element must be given as ratio, otherwise as difference to 1. See examples. Defaults to c(0.95, 0.95) if logscale=TRUE or to c(0.05, 0.05) if logscale=FALSE.
theta1	Vector; contains lower bioequivalence limit for each of the two TOSTs. In case of logscale=TRUE it is given as ratio, otherwise as diff. to 1. Defaults to c(0.8, 0.8) if logscale=TRUE or to c(-0.2, -0.2) if logscale=FALSE.
theta2	Vector; contains upper bioequivalence limit for each of the two TOSTs. If not given theta2 will be calculated as 1/theta1 if logscale=TRUE or as -theta1 if logscale=FALSE.
CV	Vector of coefficient of variations (given as as ratio, e.g. 0.2 for 20%). In case of cross-over studies this is the within-subject CV, in case of a parallel-group design the CV of the total variability. In case of logscale=FALSE CV is assumed to be the respective standard deviation.
rho	Correlation between the two PK metrics (*e.g.*, AUC and Cmax) under consideration. This is defined as correlation between the estimator of the treatment difference of PK metric one and the estimator of the treatment difference of PK metric two.
design	Character string describing the study design. See known.designs() for designs covered in this package.
setseed	Logical; if TRUE, the default, a seed of 1234567 is set.
robust	Defaults to FALSE. With that value the usual degrees of freedom will be used. Set to TRUE will use the degrees of freedom according to the 'robust' evaluation (aka Senn's basic estimator). These df are calculated as n-seq. See known.designs()$df2 for designs covered in this package. Has only effect for higher-order crossover designs.
print	If TRUE (default) the function prints its results. If FALSE only the result list will be returned.
details	If TRUE the design characteristics and the steps during sample size calculations will be shown. Defaults to FALSE.
imax	Maximum number of steps in sample size search. Defaults to 100.
nsims	Number of studies to simulate. Defaults to 1E5.

Details

The sample size is calculated via iterative evaluation of power of the 2 TOSTs.
Start value for the sample size search is taken from a large sample approximation (1 TOST) according to Zhang, modified.
The sample size is bound to 4 as minimum.

Value

A list with the input and results will be returned.
The element name "Sample size" contains the total sample size.

Warning

The function does not vectorize properly.
If you need sample sizes with varying CVs f.i. use for-loops or the apply-family.

Note

If both theta0 are near the acceptance limits then the starting value may not be a good approximation resulting in a lot of iteration steps; imax may need to be increased to obtain the required sample size.

Author(s)

Benjamin Lang & Detlew Labes

References

Phillips KF.
Power for Testing Multiple Instances of the Two One-Sided Tests Procedure
Int J Biostat. 2009;5(1):Article 15. doi: 10.2202/15574679.1169

Hua SY, Xu S, D'Agostino RB Sr.
Multiplicity adjustments in testing for bioequivalence
Stat Med. 2015;34(2):215–31. doi: 10.1002/sim.6247

Lang B, Fleischer F.
Letter to the Editor: Comments on 'Multiplicity adjustments in testing for bioequivalence'
Stat Med. 2016;35(14):2479–80. doi: 10.1002/sim.6488

Zhang P.
A Simple Formula for Sample Size Calculation in Equivalence Studies
J Biopharm Stat. 2003;13(3):529–538. doi: 10.1081/BIP120022772

See Also

power.2TOST, known.designs

Examples

```
# Sample size for 2x2x2 cross-over design, intra-subject CV = 30% and assumed
# ratios of 0.95 for both parameters, and correlation 0.9 between parameters
# (using all the other default values)
# Should give n=44 with power=0.80684
sampleN.2TOST(theta0 = rep(0.95, 2), CV = rep(0.3, 2), rho = 0.9)

# Sample size for a parallel group design,
# evaluation on the original (untransformed) scale
# BE limits 80 ... 120% = -20% ... +20% of reference,
# assumed true BE ratio 0.95% = -5% to reference mean for both parameters,
# total CV=20% for both parameters, and correlation 0.9 between parameters
# should give n=54 with power=0.8149
```

```
sampleN.2TOST(logscale=FALSE, theta0 = rep(-0.05, 2), CV = c(0.2, 0.2),
              rho = 0.9, design = "parallel")
```

`sampleN.dp` *Sample size estimation of dose-proportionality studies evaluated via Power model*

Description

Performes a sample size estimation for dose-proportionality studies using the Power model for cossover (Latin square), parallel group designs or incomplete block designs via a confidence interval equivalence criterion.

Usage

```
sampleN.dp(alpha = 0.05, CV, doses, targetpower = 0.8, beta0, theta1 = 0.8,
           theta2 = 1/theta1, design = c("crossover", "parallel", "IBD"),
           dm=NULL, CVb, print = TRUE, details = FALSE, imax = 100)
```

Arguments

`alpha`	Type 1 error. Usually taken as 0.05.
`CV`	Coefficient of variation. Is intra-subject CV for `design="crossover"` and CV of total variability in case of `design="parallel"`
`doses`	Vector of dose values under study. At least 2 doses have to be given.
`targetpower`	Power to achieve at least. Must be >0 and <1. Typical values are 0.8 or 0.9.
`beta0`	'True' slope of power model. If missing defaults to `1+log(0.95)/log(rd)` where rd is the ratio is the ratio of highest to lowest dose. Has to be within slope acceptance range according to `1+log(theta1)/log(rd)` and `1+log(theta2)/log(rd)`. Otherwise the function issues an error.
`theta1`	Lower acceptance limit for the ratio of dose normalized means (Rdmn). Transformes into slope acceptance range as described under item `beta0`.
`theta2`	Upper acceptance limit for the ratio of dose normalized means (Rdmn).
`design`	Crossover design (default), parallel group design or incomplete block design (IBD). Crossover design means Latin square design with number of doses as dimension.
`dm`	'Design matrix' of the incomplete block design (IBD) if `design="IBD"`. This matrix contains the sequences in rows and periods in columns. The entry (i,j) of the design matrix corresponds to the dose (index) a subject with i-th sequence gets in the j-th period. Can be obtained f.i. via functions of package 'crossdes'. See examples. Function `bib.CL` returns some IBD described in Chow, Liu's book "Design and Analysis of Bioavailability and Bioequivalence Studies".

CVb	Coefficient of variation of the between-subject variability. Only necessary if design="IBD". Will be set to 2*CV if missing. Set CVb=0 if all-effects-fixed model shall be used. This model gives lower sample sizes than the mixed model with random subject effects (random intercept).
print	If TRUE (default) the function prints its results. If set to FALSE only the data.frame with the results will be returned.
details	If details=TRUE the steps during sample size search will be shown. Defaults to FALSE.
imax	Maximum number of steps in sample size search. Defaults to 100. Adaption only in rare cases needed, if any.

Details

The sample size is calculated via iterative evaluation of power.dp().
Start value for the sample size search is taken from a large sample approximation.
The sample size is bound to number of dose or sequence groups as minimum.
Balanced designs are used although this is not absolutely necessary.

Value

A data.frame with the input and results will be returned.
The "Sample size" column contains the total sample size.

Warning

This function is 'experimental' only since it is not thorougly tested yet. Especially for design="IBD" reliable test cases are missing.

Author(s)

D. Labes

References

Patterson S, Jones B.
Bioequivalence and Statistics in Clinical Pharmacology
Boca Ration: Chapman & Hall/CR: 2006, p.239.
(contains presumably a bug)

Sethuraman VS, Leonov S, Squassante L, Mitchell TR, Hale MD.
Sample size calculation for the Power Model for dose proportionality studies
Pharm Stat. 2007;6(1):35–41. doi: 10.1002/pst.241

Hummel J, McKendrick S, Brindley C, French R.
Exploratory assessment of dose proportionality: review of current approaches and proposal for a practical criterion
Pharm. Stat. 2009;8(1):38–49. doi: 10.1002/pst.326

See Also

power.dp, bib.CL

Examples

```
# using all the defaults, i.e. crossover design, alpha=0.05
# theta1=0.8, theta2=1.25 but true slope slightly off 1
sampleN.dp(CV=0.2, doses=c(1, 2, 8), beta0=1.02)
# should give n=18, power=0.854528

## Not run:
# incomplete block design mentioned in Sethuraman et al.
# with 5 doses, 20 sequences, 3 periods
# (I hope at least that the design is that they used)
library(crossdes)
# IBD based on mutually orthogonal Latin squares
ibd <- des.MOLS(trt=5, k=3)
CVb <- mse2CV(0.8) # Sethuraman et al. omega squared
sampleN.dp(CV=0.2, doses=c(5, 25, 50, 100, 200), beta0=1, design="IBD", dm=ibd, CVb=CVb)
# power of that design near 90% with n=30, sequence group unbalanced
power.dp(CV=0.2, doses=c(5, 25, 50, 100, 200), n=30, beta0=1, design="IBD", dm=ibd, CVb=CVb)
## End(Not run)
```

`sampleN.HVNTID` *Sample size estimation for BE decision via FDA method for highly variable (HV) narrow therapeutic index drugs (NTIDs)*

Description

This function performs the Sample size estimation for the BE decision via FDA method for highly variable NTIDs as described in the FDA Dabigatran / Rivaroxaban guidances based on simulations. The study designs may be the full replicate design 2x2x4 with 4 periods and the 3-period replicate design 2x2x3 with sequences RTR|TRT.

Usage

```
sampleN.HVNTID(alpha = 0.05, targetpower = 0.8, theta0, theta1, theta2, CV,
               design=c("2x2x4", "2x2x3"), nsims = 1e+05, nstart, imax=100,
               print = TRUE, details = TRUE, setseed = TRUE)
```

Arguments

`alpha`	Type I error probability. Per convention mostly set to 0.05.
`targetpower`	Power to achieve at least. Must be >0 and <1. Typical values are 0.8 or 0.9.
`theta0`	'True' or assumed bioequivalence ratio. Defaults to 0.95 if not given explicitly.
`theta1`	Conventional lower ABE limit to be applied in the FDA procedure. Defaults to 0.8 if not given explicitly.
`theta2`	Conventional upper ABE limit to be applied in the FDA procedure. Defaults to 1.25 if not given explicitly.

CV	Coefficient(s) of variation as ratio. If length(CV) = 1 the same CV is assumed for Test and Reference. If length(CV) = 2 the CV for Test must be given in CV[1] and for Reference in CV[2].
design	Design of the study to be planned. 2x2x4 is the full replicate design with 2 sequences and 4 periods. 2x2x3 is the 3-period replicate design with sequences TRT|RTR. Defaults to design="2x2x4".
nsims	Number of simulations to be performed to obtain the empirical power. Defaults to 100 000 = 1e+5.
nstart	Set this to a start value for the sample size if a previous run failed. May be missing.
imax	Maximum number of steps in sample size search. Defaults to 100.
print	If TRUE (default) the function prints its results. If FALSE only the resulting dataframe will be returned.
details	If set to TRUE, the default, the steps during sample size search are shown. Moreover the details of the method settings are printed.
setseed	Simulations are dependent on the starting point of the (pseudo) random number generator. To avoid differences in power values for different runs a set.seed(123456) is issued if setseed=TRUE, the default.

Details

For deciding BE the study must pass the conventional ABE test and additionally the test that the ratio of sWT/sWR is <= 2.5.

The simulations are done via the distributional properties of the statistical quantities necessary for deciding BE based on these method.
Details can be found in a document "Implementation_scaledABE_sims" located in the doc subdirectory of the package.

Value

Returns a data.frame with the input and sample size results.
The "Sample size" column contains the total sample size.
The "nlast" column contains the last n value. May be useful for re-starting.

Warning

For some input constellations the sample size search may be very time consuming and will eventually also fail since the start values chosen may not really reasonable for them.
In case of a failed sample size search you may restart with setting the argument nstart.

Note

The design recommended by the FDA is the full replicate design 2x2x4.
The sample size estimation is done only for balanced studies since the break down of the total subject number in case of unbalanced sequence groups is not unique. Moreover the formulas used are only valid for balanced designs.
The FDA method is described for the ABE limits 0.8 ... 1.25 only. Setting theta1, theta2 to other values may not be reasonable and is not tested.
The minimum sample size is n=6, even if the power is higher than the intended targetpower.

Author(s)

D. Labes

References

FDA *Draft Guidance on Dabigatran Etexilate Mesylate*
Recommended Jun 2012. Revised Sep 2015. download

FDA *Draft Guidance on Rivaroxaban*
Recommended Sep 2015. download

See Also

power.HVNTID
and power.NTIDFDA, sampleN.NTIDFDA for NTIDs with low variability

Examples

```
# using all defaults but CV
sampleN.HVNTID(CV=0.3)
# should give
# n=22 with an (empirical) power of 0.829700

# Test formulation with lower variability but same pooled CV
CVs <- CVp2CV(0.3, ratio=0.25)
sampleN.HVNTID(CV=CVs)
# should give (no distinct difference to example above)
# n=22 with an (empirical) power of 0.837520
```

---

sampleN.noninf *Sample size for the non-inferiority t-test*

Description

Function for calculating the sample size needed to have a pre-specified power for the one-sided non-inferiority t-test for normal or log-normal distributed data.

Usage

```
sampleN.noninf(alpha = 0.025, targetpower = 0.8, logscale = TRUE, margin,
               theta0, CV, design = "2x2", robust = FALSE,
               details = FALSE, print = TRUE, imax=100)
```

Arguments

alpha	Type I error probability, significance level. Defaults here to 0.025.
targetpower	Power to achieve at least. Must be >0 and <1. Typical values are 0.8 or 0.9.
logscale	Should the data used on log-transformed or on original scale? TRUE or FALSE. Defaults to TRUE.
margin	Non-inferiority margin. In case of logscale=TRUE it must be given as ratio, otherwise as diff. to 1. Defaults to 0.8 if logscale=TRUE or to -0.2 if logscale=FALSE.
theta0	'True' or assumed bioequivalence ratio or difference. In case of logscale=TRUE it must be given as ratio, otherwise as difference to 1. See examples. Defaults to 0.95 if logscale=TRUE or to 0.05 if logscale=FALSE
CV	Coefficient of variation as ratio. In case of cross-over studies this is the within-subject CV, in case of a parallel-group design the CV of the total variability.
design	Character string describing the study design. See known.designs for designs covered in this package.
robust	Defaults to FALSE. With that value the usual degrees of freedom will be used. Set to TRUE will use the degrees of freedom according to the 'robust' evaluation (aka Senn's basic estimator). These df are calculated as n-seq. See known.designs()$df2 for designs covered in this package. Has only effect for higher-order crossover designs.
details	If TRUE the design characteristics and the steps during sample size calculations will be shown. Defaults to FALSE.
print	If TRUE (default) the function prints its results. If FALSE only the data.frame with the results will be returned.
imax	Maximum number of steps in sample size search. Defaults to 100. Adaption only in rare cases needed.

Details

The sample size is calculated via iterative evaluation of power.noninf().
Start value for the sample size search is taken from a large sample approximation.
The sample size is bound to 4 as minimum.

Notes on the underlying hypotheses

If the supplied margin is < 0 (logscale=FALSE) or < 1 (logscale=TRUE), then it is assumed higher

response values are better. The hypotheses are
`H0: theta0 <= margin vs. H1: theta0 > margin`
where `theta0 = mean(test)-mean(reference)` if `logscale=FALSE`
or
`H0: log(theta0) <= log(margin) vs. H1: log(theta0) > log(margin)`
where `theta0 = mean(test)/mean(reference)` if `logscale=TRUE`.

If the supplied margin is > 0 (`logscale=FALSE`) or > 1 (`logscale=TRUE`), then it is assumed lower response values are better. The hypotheses are
`H0: theta0 >= margin vs. H1: theta0 < margin`
where `theta0 = mean(test)-mean(reference)` if `logscale=FALSE`
or
`H0: log(theta0) >= log(margin) vs. H1: log(theta0) < log(margin)`
where `theta0 = mean(test)/mean(reference)` if `logscale=TRUE`.
This latter case may also be considered as 'non-superiority'.

Value

A data.frame with the input settings and results will be returned.
Explore it with `str(sampleN.noninf(...)`

Warning

The function does not vectorize properly.
If you need sample sizes with varying CVs f.i. use for-loops or the apply-family.

Author(s)

D. Labes

References

Julious SA.
Sample sizes for clinical trials with Normal data
Stat Med. 2004;23(12):1921–86. doi: 10.1002/sim.1783

See Also

`known.designs`, `power.noninf`

Examples

```
# using all the defaults: margin=0.8, theta0=0.95, alpha=0.025
# log-transformed, design="2x2"
sampleN.noninf(CV=0.3)
# should give n=48
#
# 'non-superiority' case, log-transformed data
# with assumed 'true' ratio somewhat above 1
sampleN.noninf(CV=0.3, targetpower=0.9, margin=1.25, theta0=1.05)
# should give n=62
```

`sampleN.NTIDFDA` *Sample size estimation for BE decision via FDA method for narrow therapeutic index drugs (NTIDs)*

Description

This function performs the Sample size estimation for the BE decision via FDA method for NTIDs based on simulations. The study design is the full replicate design 2x2x4 or the 3-period replicate design with sequeences TRT|RTR.

Usage

```
sampleN.NTIDFDA(alpha = 0.05, targetpower = 0.8, theta0, theta1, theta2, CV,
                design=c("2x2x4", "2x2x3"), nsims = 1e+05, nstart, imax=100,
                print = TRUE, details = TRUE, setseed = TRUE)
```

Arguments

`alpha`	Type I error probability. Per convention mostly set to 0.05.
`targetpower`	Power to achieve at least. Must be >0 and <1. Typical values are 0.8 or 0.9.
`theta0`	'True' or assumed bioequivalence ratio. Attention! Defaults here to 0.975 if not given explicitly. The value was chosen nearer to 1 because the potency (contents) settings for NTIDs are tightened by the FDA.
`theta1`	Conventional lower ABE limit to be applied in the FDA procedure. Defaults to 0.8 if not given explicitly.
`theta2`	Conventional upper ABE limit to be applied in the FDA procedure. Defaults to 1.25 if not given explicitly.
`CV`	Coefficient(s) of variation as ratio. If `length(CV) = 1` the same CV is assumed for Test and Reference. If `length(CV) = 2` the CV for Test must be given in CV[1] and for Reference in CV[2].
`design`	Design of the study to be planned. 2x2x4 is the full replicate design with 2 sequences and 4 periods. 2x2x3 is the 3-period replicate design with sequences TRT|RTR. Defaults to `design="2x2x4"`.
`nsims`	Number of simulations to be performed to obtain the empirical power. Defaults to 100 000 = 1e+5.
`nstart`	Set this to a start value for the sample size if a previous run failed. May be missing.
`imax`	Maximum number of steps in sample size search. Defaults to 100.
`print`	If `TRUE` (default) the function prints its results. If `FALSE` only the resulting dataframe will be returned.

details	If set to TRUE, the default, the steps during sample size search are shown. Moreover the details of the method settings are printed.
setseed	Simulations are dependent on the starting point of the (pseudo) random number generator. To avoid differences in power values for different runs a set.seed(123456) is issued if setseed=TRUE, the default.

Details

The linearized scaled ABE criterion is calculated according to the SAS code given in the FDA Warfarine guidance. For deciding BE the study must pass that criterion, the conventional ABE test and additional the test that the ratio of sWT/sWR is <= 2.5.

The simulations are done via the distributional properties of the statistical quantities necessary for deciding BE based on these method.
Details can be found in a document "Implementation_scaledABE_sims" located in the doc subdirectory of the package.

Value

Returns a data.frame with the input settings and sample size results.
The "Sample size" column contains the total sample size.
The "nlast" column contains the last n value. May be useful for re-starting.

Warning

For some input constellations the sample size search may be very time consuming and will eventually also fail since the start values chosen may not really reasonable for them. This applies especially for theta0 values near to the implied scaled (tightened/widened) ABE limits according to exp(+-1.053605*swR).
In case of a failed sample size search you may restart with setting the argument nstart.
In case of theta0 values outside the implied scaled (tightened/widened) ABE limits no sample size estimation is possible and the function throws an error (f.i. CV=0.04, theta0=0.95).

Note

The design recommended by the FDA is the full replicate design 2x2x4.
The sample size estimation is done only for balanced studies since the break down of the total subject number in case of unbalanced sequence groups is not unique. Moreover the formulas used are only valid for balanced designs.
The FDA method is described for the ABE limits 0.8 ... 1.25 only. Setting theta1, theta2 to other values may not be reasonable and is not tested.

The results for the design "2x2x3" are to be considered as experimental since at present not thoroughly tested.
The minimum sample size is n=6, even if the power is higher than the intended targetpower.

Author(s)

D. Labes

References

FDA *Draft Guidance on Warfarin Sodium*
Recommended Dec 2012. download

Yu LX et al.
Novel bioequivalence approach for narrow therapeutic index drugs
Clin Pharmacol Ther. 2015;97(3):286–91. doi: 10.1002/cpt.28

Jiang W et al.
A Bioequivalence Approach for Generic Narrow Therapeutic Index Drugs: Evaluation of the Reference-Scaled Approach and Variability Comparison Criterion
AAPS J. 2015;17(4):891–901. doi: 10.1208/s1224801597535

See Also

power.NTIDFDA and power.HVNTID, sampleN.HVNTID for NTIDs with high variability

Examples

```
sampleN.NTIDFDA(CV=0.04,theta0=0.975)
# should give
# n=54 with an (empirical) power of 0.809590
#
# Test formulation with lower variability
sampleN.NTIDFDA(CV=c(0.04,0.06),theta0=0.975)
# should give
# n=20 with an (empirical) power of 0.0.814610
#
# alternative 3-period design
sampleN.NTIDFDA(CV=0.04,theta0=0.975, design="2x2x3")
# should give
# n=86 with power = 0.80364
```

---

sampleN.RatioF *Sample size for equivalence of the ratio of two means with normality on original scale*

Description

Calculates the necessary sample size to have at least a given power based on Fieller's confidence ('fiducial') interval.

Usage

```
sampleN.RatioF(alpha = 0.025, targetpower = 0.8, theta1 = 0.8, theta2,
               theta0 = 0.95, CV, CVb, design = "2x2",
               print = TRUE, details = FALSE, imax=100, setseed=TRUE)
```

Arguments

alpha	Type I error probability. Defaults here to 0.025 because this function is intended for studies with clinical endpoints.
targetpower	Power to achieve at least. Must be >0 and <1. Typical values are 0.8 or 0.9.
theta1	Lower bioequivalence limit. Typically 0.8 (default).
theta2	Upper bioequivalence limit. Typically 1.25. Is set to 1/theta1 if missing.
theta0	'True' or assumed bioequivalence ratio. Typically set to 0.95.
CV	Coefficient of variation as ratio. In case of design="parallel" this is the CV of the total variability, in case of design="2x2" the intra-subject CV (CVw in the reference).
CVb	CV of the between-subject variability. Only necessary for design="2x2".
design	A character string describing the study design. design="parallel" or design="2x2" allowed for a two-parallel group design or a classical TR/RT crossover design.
print	If TRUE (default) the function prints its results. If FALSE only a data.frame with the results will be returned.
details	If TRUE the steps during sample size calculations will be shown. Defaults to FALSE.
imax	Maximum number of steps in sample size search. Defaults to 100. Adaption only in rare cases needed.
setseed	If set to TRUE the dependence of the power from the state of the random number generator is avoided.

Details

The sample size is based on exact power calculated using the bivariate non-central t-distribution via function pmvt() from the package mvtnorm.
Due to the calculation method used in package mvtnorm these probabilities are dependent from the state of the random number generator within the precision of the power.

The CV(within) and CVb(etween) in case of design="2x2" are obtained via an appropriate ANOVA from the error term and from the difference (MS(subject within sequence)-MS(error))/2.

Value

A data.frame with the input values and results will be returned.
The sample size n returned is the **total** sample size for **both** designs.

Note

This function is intended for studies with clinical endpoints.
In such studies the 95% confidence intervals are usually used for equivalence testing.

Therefore alpha defaults here to 0.025.
See CPMP/EWP/482/99 *Points to consider on switching between superiority and non-inferiority* EMEA, London (2000).

Author(s)

D. Labes

References

Hauschke D, Kieser M, Diletti E, Burke M.
Sample size determination for proving equivalence based on the ratio of two means for normally distributed data
Stat Med. 1999;18(1):93–105. doi: 10.1002/(SICI)1097-0258(19990115)18:1<93::AID-SIM992>3.0.CO;2-8

Hauschke D, Steinijans V, Pigeot I.
Bioequivalence Studies in Drug Development
Chichester: Wiley; 2007. Chapter 10.

See Also

power.RatioF

Examples

```
# sample size for a 2x2 cross-over study
# with CVw=0.2, CVb=0.4
# alpha=0.025 (95% CIs), target power = 80%
# 'true' ratio = 95%, BE acceptance limits 80-125%
# using all the defaults:
sampleN.RatioF(CV=0.2, CVb=0.4)
# gives n=28 with an achieved power of 0.807774
# see Hauschke et.al. (2007) Table 10.3a

# sample size for a 2-group parallel study
# with CV=0.4 (total variability)
# alpha=0.025 (95% CIs), target power = 90%
# 'true' ratio = 90%, BE acceptance limits 75-133.33%
sampleN.RatioF(targetpower=0.9, theta1=0.75, theta0=0.90, CV=0.4, design="parallel")
# gives n=236 with an achieved power of 0.900685
# see Hauschke et.al. (2007) Table 10.2

# a rather strange setting of ratio0! have a look at n.
# it would be better this is not the sample size but your account balance ;-).
sampleN.RatioF(theta0=0.801, CV=0.2, CVb=0.4)
```

`sampleN.RSABE` *Sample size estimation for BE decision via linearized scaled ABE criterion*

Description

This function performs the Sample size estimation for the BE decision via linearized scaled ABE criterion based on simulations.

Usage

```
sampleN.RSABE(alpha = 0.05, targetpower = 0.8, theta0, theta1, theta2, CV,
             design = c("2x3x3", "2x2x4", "2x2x3"), regulator = c("FDA", "EMA"),
              nsims = 1e+05, nstart, imax=100,
              print = TRUE, details = TRUE, setseed=TRUE)
```

Arguments

`alpha`	Type I error probability. Per convention mostly set to 0.05.
`targetpower`	Power to achieve at least. Must be >0 and <1. Typical values are 0.8 or 0.9.
`theta0`	'True' or assumed bioequivalence ratio. Defaults to 0.90 according to the two Laszlo's if not given explicitly.
`theta1`	Conventional lower ABE limit to be applied in the mixed procedure if CVsWR <= CVswitch. Also Lower limit for the point estimate constraint. Defaults to 0.8 if not given explicitly.
`theta2`	Conventional upper ABE limit to be applied in the mixed procedure if CVsWR <= CVswitch. Also upper limit for the point estimate constraint. Defaults to 1.25 if not given explicitly.
`CV`	Coefficient(s) of variation as ratio. If `length(CV) = 1` the same CV is assumed for Test and Reference. If `length(CV) = 2` the CV for Test must be given in CV[1] and for Reference in CV[2].
`design`	Design of the study to be planned. 2x3x3 is the partial replicate design (TRR|RTR|RRT). 2x2x4 is the full replicate design with 2 sequences and 4 periods. 2x2x3 is the 3-period design with sequences (TRT|RTR). Defaults to `design="2x3x3"`
`regulator`	Regulatory body settings for the scaled ABE criterion. Defaults to `design="FDA"`. Also the scaled ABE criterion is usually calculated with the FDA constant r_const=log(1.25)/0.25 you can override this behavior to use the EMA setting r_const=0.76 to avoid the discontinuity at CV=30% and be more stringent.
`nsims`	Number of simulations to be performed to obtain the (empirical) power.

nstart	Set this to a start for the sample size search if a previous run failed. After reworking the start n in version 1.1-05 seldom needed.
imax	Maximum number of steps in sample size search. Defaults to 100.
print	If TRUE (default) the function prints its results. If FALSE only the result data.frame will be returned.
details	If set to TRUE, the default, the steps during sample size search are shown.
setseed	Simulations are dependent on the starting point of the (pseudo) random number generator. To avoid differences in power for different runs a set.seed(123456) is issued if setseed=TRUE, the default.

Details

The linearized scaled ABE criterion is calculated according to the SAS code given in the FDA progesterone guidance.

The simulations are done via the distributional properties of the statistical quantities necessary for deciding BE based on scaled ABE. For more details see a document "Implementation_scaledABE_simsVx.yy.pdf" in the doc subdirectory of the package.

If a CVcap is defined for the regulator, the BE decision is based on the inclusion of the CI in the capped widened acceptance limits in case of CVwR > CVcap. This resembles method Howe-EMA in Munoz et al. and is the standard behavior now if regulator="EMA" is choosen.

Value

Returns a data.frame with the input and sample size results.
The "Sample size" column contains the total sample size.
The "nlast" column contains the last n value. May be useful for restarting.

Warning

The sample size estimation for theta0 >1.2 and <0.85 may be very time consuming and will eventually also fail since the start values chosen are not really reasonable in that ranges. This is especially true in the range about CV = 0.3 and regulatory constant according to FDA.
If you really need sample sizes in that range be prepared to restart the sample size estimation via the argument nstart.
Since the dependence of power from n is very flat in the mentioned region you may also consider to adapt the number of simulations not to tap in the simulation error trap.

Note

The sample size estimation is done only for balanced designs since the break down of the total subject number in case of unbalanced sequence groups is not unique. Moreover the formulas used are only for balanced designs.
The minimum sample size is n=6, even if the power is higher than the intended targetpower.

Author(s)

D. Labes

References

FDA *Draft Guidance on Progesterone*
Recommended Apr 2010. Revised Feb 2011. download

Tóthfalusi, L, Endrényi, L.
Sample Sizes for Designing Bioequivalence Studies for Highly Variable Drugs
J Pharm Pharmaceut Sci. 2011;15(1):73–84. free download

Tóthfalusi L, Endrényi L, García Arieta A.
Evaluation of Bioequivalence for Highly Variable Drugs with Scaled Average Bioequivalence
Clin Pharmacokin. 2009;48(11):725–43. doi: 10.2165/11318040000000000000

Muñoz J, Alcaide D, Ocaña J.
Consumer's risk in the EMA and FDA regulatory approaches for bioequivalence in highly variable drugs
Stat Med. 2015;35(12):1933–43. doi: 10.1002/sim.6834

See Also

power.RSABE, power.scABEL

Examples

```
# using all the defaults:
# design=2x3x3 (partial replicate design), theta0=0.90,
# ABE limits, PE constraint 0.8 - 1.25
# targetpower=80%, alpha=0.05, 1E5 sims
sampleN.RSABE(CV=0.3)
# should result in a sample size n=45, power=0.80344
```

`sampleN.scABEL` *Sample size estimation for BE decision via scaled (widened) BE acceptance limits*

Description

These functions performs the Sample size estimation via power calculations of the BE decision via scaled (widened) BE acceptance limits, based on simulations.

Usage

```
sampleN.scABEL(alpha = 0.05, targetpower = 0.8, theta0, theta1, theta2, CV,
               design = c("2x3x3", "2x2x4", "2x2x3"), regulator, nsims = 1e+05,
               nstart, imax=100, print = TRUE, details = TRUE, setseed = TRUE)
```

Arguments

alpha	Type I error probability. Per convention mostly set to 0.05.
targetpower	Power to achieve at least. Must be >0 and <1. Typical values are 0.8 or 0.9.
theta0	'True' or assumed bioequivalence ratio. Defaults to 0.90 according to the *"Two Laszlos"* if not given explicitly.
theta1	Conventional lower ABE limit to be applied in the mixed procedure if CVsWR <= CVswitch. Also Lower limit for the point estimate constraint. Defaults to 0.8 if not given explicitly.
theta2	Conventional upper ABE limit to be applied in the mixed procedure if CVsWR <= CVswitch. Also upper limit for the point estimate constraint. Defaults to 1.25 if not given explicitly.
CV	Coefficient(s) of variation as ratio. If `length(CV) = 1` the same CV is assumed for Test and Reference. If `length(CV) = 2` the CV for Test must be given in CV[1] and for Reference in CV[2].
design	Design of the study to be planned. 2x3x3 is the partial replicate design (TRR|RTR|RRT). 2x2x3 is the 3-period replicate design (TRT|RTR). 2x2x4 is the full replicate design with 2 sequences and 4 periods. Defaults to `design="2x3x3"`
regulator	Regulatory settings for the widening of the BE acceptance limits. May be given as character from the choices "EMA", "HC", "FDA" or as an object of class 'regSet' (see `reg_const`). Defaults to `regulator="EMA"` if missing. This argument may be given also in lower case if given as character. The former `regulator="ANVISA"` is no longer allowed. Use "EMA" since ANVISA now recommends the use of EMA regulatory settings.
nsims	Number of simulations to be performed to obtain the (empirical) power. The default value 100 000 = 1e+5 is usually sufficient. Consider to rise this value if theta0<=0.85 or >=1.25. But see the warning section.
nstart	Set this to a start for the sample size search if a previous run failed. After reworking the start n in version 1.1-05 seldom needed.
imax	Maximum number of steps in sample size search. Defaults to 100.
print	If TRUE (default) the function prints its results.
details	If set to `TRUE`, the default, the steps during sample size search are shown.
setseed	Simulations are dependent on the starting point of the (pseudo) random number generator. To avoid differences in power for different runs a `set.seed(123456)` is issued if `setseed=TRUE`, the default.

Details

The simulations are done via the distributional properties of the statistical quantities necessary for deciding BE based on widened ABEL. For more details see a description in the doc subdirectory of the package.

Function `sampleN.scABEL()` is based on power calculations via simulations using the distributional characteristics of the 'key' statistics obtained from the EMA recommended evaluation via ANOVA if `regulator="EMA"` or if the regulator component `est_method` is set to "ANOVA" if regulator is an object of class 'regSet'.
Otherwise the simulations are based on the distributional characteristis of the 'key' statistics obtained from evaluation via intra-subject contrasts (ISC), as recommended by the FDA.

Function `sampleN.scABEL2()` is soleyly based on power calculations via simulation using the distributional characteristics of the 'key' statistics obtained from evaluation via ISC. This function is deprecated.

Value

Returns a data.frame with the input settings and sample size results.
The "Sample size" column contains the total sample size.
The "nlast" column contains the last n value. May be useful for restarting.

Warning

The sample size estimation for very extreme theta0 (<0.83 or >1.21) may be very time consuming and will eventually also fail since the start values chosen are not really reasonable in that ranges. This is especially true in the range around CV = 0.3 and regulatory constant according to FDA.
If you really need sample sizes in that range be prepared to restart the sample size estimation via the argument nstart.
Since the dependence of power from n is very flat in the mentioned region you may also consider to adapt the number of simulations not to tap in the simulation error trap.

See also the Warning section of the function `power.scABEL()` concerning the power value agreement to those obtained from simulations via subject data.

`sampleN.scABEL2()` is deprecated and will be removed in future. A corresponding warning is thrown if this function is used.

Note

We are doing the sample size estimation only for balanced designs since the break down of the total subject number in case of unbalanced sequence groups is not unique. Moreover the formulas used are only for balanced designs.

In case of `regulator="FDA"` the sample size is only approximate since the BE decision method is not exactly what is expected by the FDA. But the *"Two Laszlos"* state that the scABEL method should be 'operational' equivalent to the FDA method. Thus the sample size should be comparable. Consider in case of `regulator="FDA"` to use the function `sampleN.RSABE()`.

In case of `regulator="HC"` the underlying power is only approximative since the Health Canada recommends as evaluation method a mixed model approach. But this could only implemented via subject data simulations which were very time consuming.

The minimum sample size is n=6, even if the power is higher than the intended targetpower.

Author(s)

D. Labes

References

Tóthfalusi L, Endrényi L.
Sample Sizes for Designing Bioequivalence Studies for Highly Variable Drugs
J Pharm Pharmaceut Sci. 2011;15(1):73–84. free download

See Also

`power.scABEL`, `power.RSABE`, `sampleN.RSABE`, `reg_const`

Examples

```
# using all the defaults:
# partial replicate design, targetpower=80%,
# true assumed ratio = 0.90, 1E+5 simulated studies
# ABE limits, PE constraint 0.8 - 1.25
# EMA regulatory settings
sampleN.scABEL(CV=0.3)
# should result in a sample size n=54, power=0.8159
#
# now with former (inofficial) ANVISA settings, CVswitch=40%
# (now ANVISA uses the same settings as EMA)
reg <- reg_const("USER", r_const=0.76, CVswitch=0.4, CVcap=0.5)
reg$name <- "Old ANVISA"
sampleN.scABEL(CV=0.3, regulator=reg)
# should result in n=60, power=0.8101

# for the full replicate design, target power = 90%
# true assumed ratio = 0.9, FDA regulatory settings
# sims based on evalaution via ISC
sampleN.scABEL(CV=0.4, targetpower=0.9, theta0=0.9, design="2x2x4", regulator="FDA")
# should result in a sample size n=32, power=0.9125
```

---

sampleN.scABEL.ad *Sample size estimation for ABEL and iteratively adjusted alpha*

Description

This function performs a sample size estimation for the BE decision via Average Bioequivalenc with Expanding Limits (ABEL) based on simulations. Simultaneously alpha is iteratively adjusted in order to maintain the consumer risk at the nominal level.

Usage

```
sampleN.scABEL.ad(alpha = 0.05, targetpower = 0.8, theta0, theta1, theta2,
                  CV, design = c("2x3x3", "2x2x4", "2x2x3"), regulator,
                  nstart = NA, nsims = 1e+06, imax = 100, tol, print = TRUE,
                  details = FALSE, alpha.pre = 0.05, setseed = TRUE,
                  sdsims = FALSE, progress)
```

Arguments

alpha	Type I error (TIE) probability (nominal level of the test). Per convention commonly set to 0.05.
targetpower	Power to achieve at least. Must be >0 and <1. Typical values are 0.80 to 0.90 (i.e., 80% to 90%). Defaults to 0.80 if not given explicitly.
theta0	'True' or assumed bioavailability ratio. Defaults to 0.90 if not given explicitly.
theta1	Conventional lower ABE limit to be applied in the mixed procedure if CVwR==CVswitch. Also lower limit for the point estimate constraint. Defaults to 0.80 if not given explicitly.
theta2	Conventional upper ABE limit to be applied in the mixed procedure if CVwR==CVswitch. Also upper limit for the point estimate constraint. Defaults to 1.25 if not given explicitly.
CV	Intra-subject coefficient(s) of variation as ratio (not percent). • If given as a scalar (length(CV)==1) the *same* CV of Test and Reference is assumed (homoscedasticity: CVwT==CVwR). • If given as a vector (length(CV)==2) – assuming heteroscedasticity – the CV of Test **must** be given in the *first* element and the one of Reference in the *second*.
design	Design of the study to be planned. "2x3x3" is the partial replicate design (TRR|RTR|RRT). "2x2x3" is the 2-sequence 3-period full replicate design (TRT|RTR). "2x2x4" is the 2-sequence 4-period full replicate design (TRTR|RTRT). Defaults to "2x3x3".
regulator	Regulatory settings for the widening of the BE acceptance limits. Choose from "EMA" (default) or "HC". This argument may be given also in lower case.
nstart	Best "guess" sample size. If not given (default), simulations start with the sample size estimated for alpha (or alpha.pre, if given), theta0, and targetpower. Can also be set to start the sample size search if a previous run failed. According to regulatory requirements must be >=12 for the EMA and >=24 for ANVISA.
nsims	Number of simulations to be performed to estimate the (empirical) TIE and in each iteration of adjusting alpha. The default value 1,000,000 = 1E+6 should not be lowered.
imax	Maximum number of steps in sample size search. Defaults to 100.
tol	Desired accuracy (convergence tolerance). Defaults to 1E-6.

print	If TRUE (default), the function sends its results to the console.
details	If TRUE (default), the steps during sample size search are shown. Additionally information about the impact on power by adjusting alpha and change of study costs due to the increased sample size is given.
alpha.pre	Pre-specified alpha (optional). Must be <=alpha. ABEL will be performed at level alpha.pre and the TIE assessed at level alpha. Less powerful than adjusting alpha but an alternative in the critical region of maximum inflation of the TIE. In certain scenarios Bonferroni's 0.025 is not sufficient to preserve the Type I Error. Not recommended if CVwR >= 0.45 due to poor power characteristics.
setseed	Simulations are dependent on the starting point of the (pseudo) random number generator. To avoid differences in power for different runs set.seed(123456) is issued if setseed=TRUE (default).
sdsims	If FALSE (default) power is estimated by the respective statistics. Recommended for the full replicate designs for speed reasons. Set to TRUE if results of power.scABEL are expected to be inaccurate (small sample sizes and/or heteroscedasticity) and subject data (via power.scABEL.sdsims) should be simulated instead. Very time consuming (easily 100times slower)! If design="2x3x3" subject simulations are recommended – even if homoscedasticity is assumed. Subject data simulations are not supported for regulator="HC".
progress	Set to TRUE if a progress bar should be displayed. Ignored if sdsims=FALSE.

Details

The simulations are done via the distributional properties of the statistical quantities necessary for assessing BE based on ABEL. Simulations for the TIE are performed at the upper (expanded) limit U of the acceptance range. Due to the symmetry around 1 results are valid for the lower (expanded) limit L as well.
U at the EMA's and Health Canada's CVswitch and CVcap:

```
scABEL(CV=0.3, regulator="EMA")[["upper"]]; scABEL(CV=0.3, regulator="HC")[["upper"]]
[1] 1.25
[1] 1.25
scABEL(CV=0.5, regulator="EMA")[["upper"]]; scABEL(CV=0.57382, regulator="HC")[["upper"]]
[1] 1.43191
[1] 1.5
```

Simulated studies are evaluated by ANOVA ('Method A') as recommended in the EMA's Q&A-document and by intra-subject contrasts if regulator="HC". Health Canada requires a mixed-effects model which cannot be implemented in R. However, intra-subjects contrasts are a sufficiently close approximation.

If an inflation of the TIE is expected (i.e., >alpha), alpha is iteratively adjusted until at least the target power is reached and the consumer risk is maintained (<=alpha). For details about the algorithm see the respective section of scABEL.ad.

Value

Returns a data.frame with the input and results for adjusted alpha, type I error, sample size, and achieved power.
The "Sample size" column contains the total sample size. If no adjustment is necessary, `NA` will be returned in the "alpha.adj" column and other results are identical to the ones obtained by `sampleN.scABEL`.

Warning

The sample size estimation for extreme `theta0` (<0.83 or >1.21) may be time consuming and will eventually also fail since the start values chosen are not really reasonable in that ranges.
If you really need sample sizes in that range be prepared to restart the sample size estimation with `nstart` above the last one before failure.
Since the dependence of power from `n` is very flat in the mentioned region you may also consider to adapt the number of simulations not to tap in the simulation error trap.

See also the Warning section of the function `power.scABEL` concerning the power value agreement to those obtained from simulations via subject data.

Note

We are doing the sample size estimation only for balanced designs since the break down of the total subject number in case of unbalanced sequences is not unique. Moreover the formulas used are only for balanced designs.

Author(s)

H. Schütz

References

Tóthfalusi L, Endrényi L.
Sample Sizes for Designing Bioequivalence Studies for Highly Variable Drugs
J Pharm Pharmaceut Sci. 2011;15(1):73–84. free download

Wonnemann M, Frömke C, Koch A.
Inflation of the Type I Error: Investigations on Regulatory Recommendations for Bioequivalence of Highly Variable Drugs
Pharm Res. 2015;32(1):135–43. doi: 10.1007/s110950141450z

Muñoz J, Alcaide D, Ocaña J.
Consumer's risk in the EMA and FDA regulatory approaches for bioequivalence in highly variable drugs
Stat Med. 2015;35(12):1933–43. doi: 10.1002/sim.6834

Labes D, Schütz H.
Inflation of Type I Error in the Evaluation of Scaled Average Bioequivalence, and a Method for its Control
Pharm Res. 2016;33(11):2805–14. doi: 10.1007/s1109501620061

See Also

scABEL.ad, sampleN.scABEL, power.scABEL, scABEL

Examples

```
# --- Not run due to timing policy of CRAN for examples
# each may run some ten seconds or more
# using all the defaults:
# TRR|RTR|RRT, target power 80%, assumed ratio 0.90, 1E+6 simulated studies,
# EMA regulatory settings (ABE limits, PE constraint 0.8 - 1.25)
## Not run:
sampleN.scABEL.ad(CV = 0.3)
## End(Not run)
# should result in n 60, power 0.8022.
# Note: Without adjustment by sampleN.scABEL(): n 54, power 0.8159
# Easier to show the details:
## Not run:
sampleN.scABEL.ad(CV = 0.3, details = TRUE)
## End(Not run)
#
# TRTR|RTRT, target power 90%, pre-specified alpha 0.025
## Not run:
sampleN.scABEL.ad(CV = 0.3, targetpower = 0.9, design = "2x2x4", alpha.pre = 0.025)
## End(Not run)
# should result in n 60, power 0.9021; pre-specified alpha justified.
```

---

sampleN.TOST *Sample size based on power of TOST*

Description

Calculates the necessary sample size to have at least a given power.

Usage

```
sampleN.TOST(alpha = 0.05, targetpower = 0.8, logscale = TRUE,
             theta0, theta1, theta2, CV, design = "2x2", method="exact",
             robust=FALSE, print = TRUE, details = FALSE, imax=100)
```

Arguments

alpha	Type I error probability. Per convention mostly set to 0.05.
targetpower	Power to achieve at least. Must be >0 and <1. Typical values are 0.8 or 0.9.
logscale	Should the data used on log-transformed (TRUE) or on original scale (FALSE)? Defaults to TRUE.

`theta0` 'True' or assumed bioequivalence ratio.
In case of `logscale=TRUE` it must be given as ratio, otherwise as difference to 1. See examples.
Defaults to 0.95 if `logscale=TRUE` or to 0.05 if `logscale=FALSE`.

`theta1` Lower bioequivalence limit.
In case of `logscale=TRUE` it is given as ratio, otherwise as diff. to 1.
Defaults to 0.8 if `logscale=TRUE` or to -0.2 if `logscale=FALSE`.

`theta2` Upper bioequivalence limit.
If not given theta2 will be calculated as `1/theta1` if `logscale=TRUE` or as `-theta1` if `logscale=FALSE`.

`CV` Coefficient of variation as ratio.

`design` Character string describing the study design.
See `known.designs()` for designs covered in this package.

`method` Method for calculation of the power.
Defaults to "exact" in which case the calculation is done based on formulas with Owen's Q. The calculation via Owen's Q can also be choosen with `method="owenq"`.
Another exact method via direct use of the bivariate non-central *t*-distribution may be chosen with method="mvt". This may have somewhat lower precision compared to Owen's Q and has much longer run-time.
Approximate calculations can be choosen via method="noncentral" or method="nct" for the approximation using the non-central *t*-distribution. With method="central" or method="shifted" the relative crude approximation via 'shifted' central *t*-distribution is chosen.
The strings for method may be abbreviated.

`robust` Defaults to FALSE. With that value the usual degrees of freedom will be used.
Set to TRUE will use the degrees of freedom according to the 'robust' evaluation (aka Senn's basic estimator). These df are calculated as `n-seq`.
See `known.designs()$df2` for designs covered in this package.
Has only effect for higher-order crossover designs.

`print` If TRUE (default) the function prints its results.
If FALSE only the data.frame with the results will be returned.

`details` If TRUE the design characteristics and the steps during sample size calculations will be shown.
Defaults to FALSE.

`imax` Maximum number of steps in sample size search.
Defaults to 100. Adaption only in rare cases needed.

Details

The sample size is calculated via iterative evaluation of power of the TOST procedure.
Start value for the sample size search is taken from a large sample approximation according to Zhang, modified.
The sample size is bound to 4 as minimum.

Value

A data.frame with the input and results will be returned.
The "Sample size" column contains the total sample size.

Warning

The function does not vectorize properly.
If you need sample sizes with varying CVs f.i. use for-loops or the apply-family.

Note

Of course it is highly recommended to use the default method="exact" :-)).
There is no reason beside testing and comparative purposes to use an approximation if the exact method is available at no extra costs.

Author(s)

D. Labes

References

Phillips KF.
Power of the Two One-Sided Tests Procedure in Bioequivalence
J Pharmacokin Biopharm. 1990;18:137–44. doi: 10.1007/BF01063556

Diletti D, Hauschke D, Steinijans VW.
Sample Size Determination for Bioequivalence Assessment by Means of Confidence Intervals
Int J Clin Pharmacol Ther Toxicol. 1991;29(1):1–8.

Diletti D, Hauschke D, Steinijans VW.
Sample size determination: Extended tables for the multiplicative model and bioequivalence ranges of 0.9 to 1.11 and 0.7 to 1.43
Int J Clin Pharmacol Ther Toxicol. 1992;30(Suppl 1):S59–62.

Zhang P.
A Simple Formula for Sample Size Calculation in Equivalence Studies
J Biopharm Stat. 2003;13(3):529–538. doi: 10.1081/BIP120022772

See Also

power.TOST, known.designs

Examples

```
# Exact calculation for a classical 2x2 cross-over (TR/RT),
# BE limits 80 ... 125%, assumed true BE ratio 0.95, intra-subject CV=30%,
# using all the default values
# should give n=40 power=0.815845
sampleN.TOST(CV=0.3)

# Exact calculation for a parallel group design
```

```
# evaluation on the original (untransformed) scale
# BE limits 80 ... 120% = -20% ... +20% of reference,
# assumed true BE ratio 0.95% = -5% to reference mean,
# total CV=20%
# should give n=48 (total) power=0.815435
sampleN.TOST(logscale=FALSE, theta1=-0.2, theta0=-0.05, CV=0.2, design="parallel")

# A rather strange setting of theta0! Have a look at n.
# It would be better this is not the sample size but the running total
# of my bank account. But the first million is the hardest ;-).
sampleN.TOST(CV=0.2, theta0=0.8005, theta1=0.8)
```

---

scABEL *Scaled (widened) BE Acceptance Limits*

Description

The (widened) scaled BE acceptance limits are calculated according to the regulatory settings of EMA, HC, FDA or via user defined regulatory settings.

Usage

```
scABEL(CV, regulator)
```

Arguments

CV	Coefficient of variation (of the Reference) as ratio.
regulator	Regulatory body settings for the widening of the BE acceptance limits. May be given as character from the choices "EMA", "HC", "FDA" or as an object of class 'regSet' (see reg_const). Defaults to regulator="EMA" if missing. The former regulator="ANVISA" is no longer allowed. Use "EMA" since ANVISA now recommends the use of EMA regulatory settings. The former regulator="USER" is no longer accepted but can be handled now via function reg_const() to define an object with class 'regSet'.

Details

The widened BE acceptance limits will be calculated by the formula
[lBEL,uBEL] =exp(-+ r_const*sWR)
with r_const the regulatory constant and sWR the standard deviation of the within subjects variability of the Reference.
r_const=0.76 (~log(1.25)/0.29356) is used in case of regulator="EMA" or regulator="HC" and in case of regulator="FDA" r_const=0.89257...(=log(1.25)/0.25).
If the CVwR of the Reference is < CVswitch=0.3 the conventional ABE limits apply (mixed procedure).

In case of `regulator="EMA"` a cap is placed on the widened limits if CVwr>0.5, i.e. the widened limits are held at value calculated for CVwR=0.5. In case of `regulator="HC"` the capping is done such that the acceptance limits are 0.6666 ... 1.5 at maximum, i.e. CVcap=0.57382. Literally it was given rounded to 3 significant digits as 57.4%.
See http://www.hc-sc.gc.ca/dhp-mps/prodpharma/activit/announce-annonce/notice-avis-be-hvdp-nb-pphv-php

Value

Returns a vector of lenghth 2 if one CV is given or a matrix if CV is given as vector with named components "lower" and "upper" of the scaled acceptance limits.

Note

The scaled acceptance limits are not used directly in the BE evaluation for highly variable drugs recommended by the FDA. They are included here for comparative purposes. Moreover there are controversies where to locate the 'implied' BE acceptance limits. See reference below.

Author(s)

D. Labes

References

Davit et al.
Implementation of a Reference-Scaled Average Bioequivalence Approach for Highly Variable Generic Drug Products by the US Food and Drug Administration
AAPS J. 2012;14(4):915–24. doi: 10.1208/s122480129406x

See Also

`power.scABEL`, `sampleN.scABEL`, `reg_const`

Examples

```
scABEL(CV=0.3, regulator="EMA")
# should give the usual BE limits:
# lower  upper
#  0.80   1.25
scABEL(CV=0.4, regulator="EMA")
# should give the widened limits:
#   lower     upper
# 0.746177 1.340165
#
# define old ANVISA settings via reg_const()
rc <- reg_const("USER", r_const=0.76, CVswitch=0.4, CVcap=0.5)
rc$name <- "ANVISAold"
scABEL(CV=0.4, regulator=rc)
```

```
# should give the not widened limits:
# lower upper
#  0.80  1.25
```

---

scABEL.ad *Iteratively adjusted alpha for ABEL*

Description

This function iteratively adjusts alpha for the BE decision via Average Bioequivalence with Expanding Limits (ABEL) based on simulations in order to maintain the consumer risk at the nominal level.

Usage

```
scABEL.ad(alpha = 0.05, theta0, theta1, theta2, CV,
          design = c("2x3x3", "2x2x4", "2x2x3"), regulator,
          n, alpha.pre = 0.05, imax = 100, tol, print = TRUE,
          details = FALSE, setseed = TRUE, nsims = 1e+06,
          sdsims = FALSE, progress)
```

Arguments

alpha	Type I Error (TIE) probability (nominal level of the test). Per convention commonly set to 0.05.
theta0	'True' or assumed bioavailability ratio. Defaults to 0.90 if not given explicitly.
theta1	Conventional lower ABE limit to be applied in the mixed procedure if CVwR==CVswitch. Also lower limit for the point estimate constraint. Defaults to 0.80 if not given explicitly.
theta2	Conventional upper ABE limit to be applied in the mixed procedure if CVwR==CVswitch. Also upper limit for the point estimate constraint. Defaults to 1.25 if not given explicitly.
CV	Intra-subject coefficient(s) of variation as ratio (not percent). • If given as a scalar (length(CV)==1) the *same* CV of Test and Reference is assumed (homoscedasticity: CVwT==CVwR). • If given as a vector (length(CV)==2) – assuming heteroscedasticity – the CV of Test **must** be given in the *first* element and the one of Reference in the *second*.
design	Design of the study. "2x3x3" is the partial replicate design (TRR|RTR|RRT). "2x2x3" is the 2-sequence 3-period full replicate design (TRT|RTR). "2x2x4" is the 2-sequence 4-period full replicate design (TRTR|RTRT). Defaults to "2x3x3".

regulator	Regulatory settings for the widening of the BE acceptance limits. Choose from "EMA" (default), "HC", or "FDA". This argument may also be given in lower case.
n	Total sample size of the study or a vector of sample size / sequences. If n leads to an unbalanced design (i.e., is not a multiple of two in the full replicate designs or not a multiple of three in the partial replicate), the code tries to keep subjects / sequence as balanced as possible. In evaluating a particular *unbalanced* study **always** give n as a vector. If design="2x2x3" (TRT|RTR) the *order* of sample sizes is important. n[1] is for sequence TRT and n[2] is for sequence RTR. If n is missing, a sample size is estimated with target power 0.80 and pre-specified alpha if defined. Otherwise, alpha is used.
alpha.pre	Pre-specified alpha (optional). Must be <=alpha. ABEL will be performed at level alpha.pre and the TIE assessed at level alpha. Less powerful than adjusting alpha but an alternative in the critical region of maximum inflation of the TIE. In certain scenarios Bonferroni's 0.025 is not sufficient to preserve the Type I Error (e.g., the third example). Not recommended if CVwR >= 0.45 due to poor power characteristics.
imax	Maximum number of steps in sample size search. Defaults to 100.
tol	Desired accuracy (convergence tolerance). Defaults to 1E-6.
print	If TRUE (default), the function sends its results to the console.
details	If TRUE, the *relative* change of the consumer risk in percent is shown. Additionally information about the impact on power (for specified theta0 and target power 0.80), runtime, and number of simulations (iterations) are given. Defaults to FALSE.
setseed	Simulations are dependent on the starting point of the (pseudo) random number generator. To avoid differences in power for different runs set.seed(123456) is issued if setseed=TRUE (default).
nsims	Number of simulations to be performed to estimate the (empirical) TIE error and in each iteration of adjusting alpha. The default value 1,000,000 = 1E+6 should not be lowered.
sdsims	If FALSE (default) power is estimated by the respective statistics. Recommended for the full replicate designs for speed reasons. Set to TRUE if results of power.scABEL are expected to be inaccurate (small sample sizes and/or heteroscedasticity) and subject data (via power.scABEL.sdsims) should be simulated instead. Very time consuming (easily 100times slower)! If design="2x3x3" subject simulations are recommended – even if homoscedasticity is assumed. Subject data simulations are only supported for regulator="EMA".
progress	Set to TRUE if a progress bar should be displayed. Ignored if sdsims=FALSE.

Details

The simulations are done via the distributional properties of the statistical quantities necessary for assessing BE based on ABEL. Simulations for the TIE are performed at the upper (expanded) limit U of the acceptance range. Due to the symmetry around 1 results are valid for the lower (expanded) limit L as well.
U at the EMA's and Health Canada's CVswitch and CVcap:

```
scABEL(CV=0.3, regulator="EMA")[["upper"]]; scABEL(CV=0.3, regulator="HC")[["upper"]]
[1] 1.25
[1] 1.25
scABEL(CV=0.5, regulator="EMA")[["upper"]]; scABEL(CV=0.57382, regulator="HC")[["upper"]]
[1] 1.43191
[1] 1.5
```

Simulated studies are evaluated by ANOVA ('Method A') as recommended in the EMA's Q&A-document and by intra-subject contrasts if `regulator="HC"`. Health Canada requires a mixed-effects model which cannot be implemented in R. However, intra-subjects contrasts are a sufficiently close approximation.
The Type I Error in ABEL depends only on `CVwR` and – to a minor degree – the sample size. Algorithm:

1. The TIE is assessed based on `alpha` (or `alpha.pre`) and compared to the nominal level of the test `alpha`.
2. If no inflation of the TIE is found, the algorithm stops.
3. Otherwise, alpha is iteratively adjusted (i.e., `alpha.adj <alpha`) until no more relevant inflation of the TIE is detected (i.e., `abs(TIE - alpha) <= tol`).

Value

Sends results to the console if argument `print=TRUE` (default).

Returns a list with the input, adjusted alpha, and Type I Error (for nominal and adjusted alpha) if argument `print=FALSE`.
If no adjustment is necessary, `NA`s will be returned for the respective variables (`alpha.adj`, `TIE.adj`, `rel.change`, `pwr.adj`, `rel.loss`).

Warning

See the Warning section of the function `power.scABEL` concerning the power value agreement to the one obtained by simulations via subject data.

Note

Specifying `theta0` is not necessary.
If `theta0` *is not* given, achievable power for the common target of 0.80 (both for `alpha` and adjusted alpha) will be estimated. If `theta0` *is* specified, its value will be used; again for target power 0.80.
If you are interested in other levels of power, use `sampleN.scABEL.ad`.

The EMA's method is currently recommended in other jurisdictions as well (*e.g.*, the WHO; ASEAN States, Australia, Brazil, Egypt, the Eurasian Economic Union, New Zealand, the Russian Federation).

Author(s)

H. Schütz

References

Wonnemann M, Frömke C, Koch A.
Inflation of the Type I Error: Investigations on Regulatory Recommendations for Bioequivalence of Highly Variable Drugs
Pharm Res. 2015;32(1):135–43. doi: 10.1007/s110950141450z

Muñoz J, Alcaide D, Ocaña J.
Consumer's risk in the EMA and FDA regulatory approaches for bioequivalence in highly variable drugs
Stat Med. 2015;35(12):1933–43. doi: 10.1002/sim.6834

Labes D, Schütz H.
Inflation of Type I Error in the Evaluation of Scaled Average Bioequivalence, and a Method for its Control
Pharm Res. 2016;33(11):2805–14. doi: 10.1007/s1109501620061

Tóthfalusi L, Endrényi L.
Algorithms for Evaluating Reference Scaled Average Bioequivalence: Power, Bias, and Consumer Risk
Stat Med. 2017;36(27):4378–90. doi: 10.1002/sim.7440

Molins E, Cobo E, Ocaña J.
Two-Stage Designs Versus European Scaled Average Designs in Bioequivalence Studies for Highly Variable Drugs: Which to Choose?
Stat Med. 2017;36(30):4777–88. doi: 10.1002/sim.7452

European Medicines Agency, Committee for Medicinal Products for Human Use.
Guideline on the Investigation of Bioequivalence
London, 20 January 2010. CPMP/EWP/QWP/1401/98 Rev. 1/ Corr **

European Medicines Agency, Committee for Medicinal Products for Human Use.
Questions & Answers: positions on specific questions addressed to the Pharmacokinetics Working Party (PKWP)
London, 19 November 2015. EMA/618604/2008 Rev. 13

Health Canada, Therapeutic Products Directorate.
Policy on Bioequivalence Standards for Highly Variable Drug Products
Ottawa, 18 April 2016. 16-104293-140

See Also

sampleN.scABEL.ad, power.scABEL, power.scABEL.sdsims, scABEL

Examples

```
# Using all defaults:
# TRR|RTR|RRT, target power 80% for assumed ratio 0.90 (estimated sample size 54),
# EMA regulatory settings (ABE limits and PE constraint 0.80 - 1.25),
# 1E+6 simulated studies.
# Not run: due to timing policy of CRAN for examples
## Not run:
scABEL.ad(CV = 0.3)
## End(Not run)
# Should result in adjusted alpha 0.03389 (TIE 0.5000, TIE for nominal alpha 0.07189).
```

```
#
# As above but subject data simulations.
## Not run:
scABEL.ad(CV = 0.3, sdsims = TRUE)
## End(Not run)
# Should result in adjusted alpha 0.03336 (TIE 0.5000, TIE for nominal alpha 0.07237).
#
# TRT|RTR, heteroscedasticity, sample size 48 (unbalanced), subject data simulations.
## Not run:
scABEL.ad(CV = c(0.25, 0.3), design = "2x2x3", n = c(23, 25), sdsims = TRUE)
## End(Not run)
# Should result in adjusted alpha 0.02465 (TIE 0.5000, TIE for nominal alpha 0.09050).
#
# TRTR|RTRT, CV 0.35, sample size 33 (unbalanced).
## Not run:
scABEL.ad(CV = 0.35, design = "2x2x4", n = c(16, 17))
## End(Not run)
# Should result in adjusted alpha 0.03632 (TIE 0.5000, TIE for nominal alpha 0.06544).
```

`type1error.2TOST` *Type I error rate for two simultaneous TOST procedures*

Description

Was designed to calculate the type I error rate of two simultaneous TOST procedures (where the two parameters of the two TOSTs are correlated with some correlation) for various study designs used in BE studies.

Is defunct now since it suffers from insufficient precision to obtain the type 1 error (TIE) via simulations.
Due to the intersection-union principle the TIE is always upper bounded to alpha by theory.

Index