Package ‘sasLM’

June 1, 2024

Version 0.10.4
Title ‘SAS’ Linear Model
Description This is a core implementation of ‘SAS’ procedures for linear models - 
GLM, REG, ANOVA, TTEST, FREQ, and UNIVARIATE. Some R packages pro-
vide type II and type III SS. However, the results of nested and complex designs are often differ-
ent from those of ‘SAS.’ Different results does not necessarily mean incorrectness. How-
ever, many wants the same results to SAS. This package aims to achieve that.
Depends R (>= 3.5.0), mvtnorm
Imports methods
Suggests MASS
Author Kyun-Seop Bae [aut]
Maintainer Kyun-Seop Bae <k@acr.kr>
Copyright 2020-, Kyun-Seop Bae
License GPL-3
Repository CRAN
URL https://cran.r-project.org/package=sasLM
NeedsCompilation no
Date/Publication 2024-06-01 03:20:02 UTC

R topics documented:
sasLM-package ......................................................... 4
af ................................................................. 5
aov1 ............................................................. 5
aov2 ............................................................. 7
aov3 ............................................................. 8
aspirinCHD .......................................................... 9
BEdata .......................................................... 10
bk .............................................................. 10
BY ............................................................. 12
R topics documented:

Ctest .................................................. 13
Coll .................................................. 14
CONTR ............................................. 15
Cor.test ............................................. 16
corFisher .......................................... 17
cSS .................................................. 18
CumAlpha .......................................... 19
CV .................................................... 20
Diffogram .......................................... 20
Drift ................................................ 21
e1 ..................................................... 22
e2 ..................................................... 23
e3 ..................................................... 24
EMS .................................................. 24
est ..................................................... 25
ESTM ................................................ 26
estmb ............................................... 27
ExitP ............................................... 28
g2inv ............................................... 29
G2SWEEP ........................................... 30
geoCV ............................................... 31
geoMean ......................................... 32
GLM ............................................... 33
is.cor ............................................. 34
Kurtosis .......................................... 35
KurtosisSE ....................................... 36
LCL .................................................. 36
lfit .................................................. 37
lr .................................................... 38
lr0 .................................................. 39
LSM ................................................ 40
Max ............................................... 41
Mean ............................................... 42
Median .......................................... 42
Min ................................................ 43
ModelMatrix ..................................... 44
mtest .............................................. 45
N .................................................... 46
OR ................................................... 46
ORcmh ............................................. 47
ORinv .............................................. 48
ORMn .............................................. 49
ORMn1 .......................................... 51
pB .................................................. 52
Pcor.test ......................................... 53
pD .................................................. 54
PDIFF .............................................. 54
PocockBound ..................................... 56
topics documented:

- pResD .................................................. 57
- QuartileRange ........................................ 57
- Range .................................................. 58
- RanTest ................................................ 59
- RD ...................................................... 60
- RDinv .................................................. 61
- RDmn .................................................... 62
- RDmn1 .................................................. 63
- REG ...................................................... 64
- regD ..................................................... 66
- RR ...................................................... 67
- RRinv ................................................... 68
- RRmn .................................................... 69
- RRmn1 .................................................. 70
- satt ..................................................... 71
- ScoreCI ................................................ 72
- SD ..................................................... 73
- SEM ..................................................... 73
- seqBound .............................................. 74
- seqCI ................................................... 75
- Skewness ............................................... 76
- SkewnessSE ........................................... 77
- SLICE ................................................... 77
- SS ....................................................... 78
- T3MS .................................................... 79
- T3test ................................................... 80
- tmtest ................................................... 81
- trimmedMean .......................................... 82
- tsum ..................................................... 82
- tsum0 ................................................... 83
- tsum1 ................................................... 84
- tsum2 ................................................... 85
- tsum3 ................................................... 86
- TTEST ................................................... 87
- UCL ..................................................... 88
- UNIV .................................................... 89
- vtest .................................................... 90
- WhiteTest ............................................. 91
- ztest .................................................... 92
Description

This is a core implementation of 'SAS' procedures for linear models - GLM, REG, and ANOVA. Some packages provide type II and type III SS. However, the results of nested and complex designs are often different from those of 'SAS'. A different result does not necessarily mean incorrectness. However, many want the same result with 'SAS'. This package aims to achieve that. Reference: Littell RC, Stroup WW, Freund RJ (2002, ISBN:0-471-22174-0).

Details

This will serve those who want SAS PROC GLM, REG, and ANOVA in R.

Author(s)

Kyun-Seop Bae k@acr.kr

Examples

```R
## SAS PROC GLM Script for Typical Bioequivalence Data
# PROC GLM DATA=BEdata;
# CLASS SEQ SUBJ PRD TRT;
# MODEL LNCMAX = SEQ SUBJ(SEQ) PRD TRT;
# RANDOM SUBJ(SEQ)/TEST;
# LSMEANS TRT / DIFF=CONTROL("R") CL ALPHA=0.1;
# ODS OUTPUT LSMeanDiffCL=LSMD;

# DATA LSMD; SET LSMD;
# PE = EXP(DIFFERENCE);
# LL = EXP(LowerCL);
# UL = EXP(UpperCL);
# PROC PRINT DATA=LSMD; RUN;
##
## SAS PROC GLM equivalent
BEdata = af(BEdata, c("SEQ", "SUBJ", "PRD", "TRT")) # Columns as factor
formula1 = log(CMAX) ~ SEQ/SUBJ + PRD + TRT # Model
GLM(formula1, BEdata) # ANOVA tables of Type I, II, III SS
RanTest(formula1, BEdata, Random="SUBJ") # Hypothesis test with SUBJ as random
ci0 = Clest(formula1, BEdata, "TRT", c(-1, 1), 0.90) # 90$ CI
exp(ci0[, c("Estimate", "Lower CL", "Upper CL")]) # 90% CI of GMR

## 'nlme' or SAS PROC MIXED is preferred for an unbalanced case
## SAS PROC MIXED equivalent
# require(nlme)
# Result = lme(log(CMAX) ~ SEQ + PRD + TRT, random=-1|SUBJ, data=BEdata)
# summary(Result)
# VarCorr(Result)
```
af

# ci = intervals(Result, 0.90) ; ci
# exp(ci$fixed["TRTT",])
##

---

### af

**Convert some columns of a data.frame to factors**

**Description**

Conveniently convert some columns of data.frame into factors.

**Usage**

```r
af(DataFrame, Cols)
```

**Arguments**

- `DataFrame`: a data.frame
- `Cols`: column names or indices to be converted

**Details**

It performs conversion of some columns in a data.frame into factors conveniently.

**Value**

Returns a data.frame with converted columns.

**Author(s)**

Kyun-Seop Bae k@acr.kr

---

### aov1

**ANOVA with Type I SS**

**Description**

ANOVA with Type I SS.

**Usage**

```r
aov1(Formula, Data, BETA=FALSE, Resid=FALSE)
```
Arguments

Formula: a conventional formula for a linear model.
Data: a data.frame to be analyzed.
BETA: if TRUE, coefficients (parameters) of REG will be returned. This is equivalent to SOLUTION option of SAS PROC GLM.
Resid: if TRUE, fitted values (y hat) and residuals will be returned.

Details

It performs the core function of SAS PROC GLM, and returns Type I SS. This accepts continuous independent variables also.

Value

The result table is comparable to that of SAS PROC ANOVA.

Df: degree of freedom
Sum Sq: sum of square for the set of contrasts
Mean Sq: mean square
F value: F value for the F distribution
Pr(>F): probability of larger than F value

Next returns are optional.

Parameter: Parameter table with standard error, t value, p value. TRUE is 1, and FALSE is 0 in the Estimable column. This is returned only with BETA=TRUE option.
Fitted: Fitted value or y hat. This is returned only with Resid=TRUE option.
Residual: Weigthed residuals. This is returned only with Resid=TRUE option.

Author(s)

Kyun-Seop Bae k@acr.kr

Examples

aov1(uptake ~ Plant + Type + Treatment + conc, CO2)
aov1(uptake ~ Plant + Type + Treatment + conc, CO2, BETA=TRUE)
aov1(uptake ~ Plant + Type + Treatment + conc, CO2, Resid=TRUE)
aov1(uptake ~ Plant + Type + Treatment + conc, CO2, BETA=TRUE, Resid=TRUE)
**Description**

ANOVA with Type II SS.

**Usage**

```r
aov2(Formula, Data, BETA=FALSE, Resid=FALSE)
```

**Arguments**

- **Formula**: a conventional formula for a linear model.
- **Data**: a `data.frame` to be analyzed.
- **BETA**: if TRUE, coefficients (parameters) of `REG` will be returned. This is equivalent to the SOLUTION option of SAS PROC GLM.
- **Resid**: if TRUE, fitted values (y hat) and residuals will be returned.

**Details**

It performs the core function of SAS PROC GLM, and returns Type II SS. This accepts continuous independent variables also.

**Value**

The result table is comparable to that of SAS PROC ANOVA.

- **Df**: degree of freedom
- **Sum Sq**: sum of square for the set of contrasts
- **Mean Sq**: mean square
- **F value**: F value for the F distribution
- **Pr(>F)**: probability of larger than F value

Next returns are optional.

- **Parameter**: Parameter table with standard error, t value, p value. TRUE is 1, and FALSE is 0 in the Estimable column. This is returned only with BETA=TRUE option.
- **Fitted**: Fitted value or y hat. This is returned only with Resid=TRUE option.
- **Residual**: Weighted residuals. This is returned only with Resid=TRUE option.

**Author(s)**

Kyun-Seop Bae k@acr.kr
Examples

\begin{verbatim}
\texttt{aov2(uptake ~ Plant + Type + Treatment + conc, CO2)}
\texttt{aov2(uptake ~ Plant + Type + Treatment + conc, CO2, BETA=TRUE)}
\texttt{aov2(uptake ~ Plant + Type + Treatment + conc, CO2, Resid=TRUE)}
\texttt{aov2(uptake ~ Plant + Type + Treatment + conc, CO2, BETA=TRUE, Resid=TRUE)}
\texttt{aov2(uptake ~ Type, CO2)}
\texttt{aov2(uptake ~ Type - 1, CO2)}
\end{verbatim}

\section*{Description}
\textit{ANOVA with Type III SS.}

\section*{Usage}
\texttt{aov3(Formula, Data, BETA=FALSE, Resid=FALSE)}

\section*{Arguments}
\begin{description}
\item[Formula] a conventional formula for a linear model.
\item[Data] a data.frame to be analyzed
\item[BETA] if TRUE, coefficients (parameters) of REG will be returned. This is equivalent to SOLUTION option of SAS PROC GLM
\item[Resid] if TRUE, fitted values (y hat) and residuals will be returned
\end{description}

\section*{Details}
It performs the core function of SAS PROC GLM, and returns Type III SS. This accepts continuous independent variables also.

\section*{Value}
The result table is comparable to that of SAS PROC ANOVA.
\begin{description}
\item[Df] degree of freedom
\item[Sum Sq] sum of square for the set of contrasts
\item[Mean Sq] mean square
\item[F value] F value for the F distribution
\item[Pr(>F)] probability of larger than F value
\end{description}
Next returns are optional.
\begin{description}
\item[Parameter] Parameter table with standard error, t value, p value. TRUE is 1, and FALSE is 0 in the Estimable column. This is returned only with BETA=TRUE option.
\item[Fitted] Fitted value or y hat. This is returned only with Resid=TRUE option.
\item[Residual] Weighthed residuals. This is returned only with Resid=TRUE option.
\end{description}
**aspirinCHD**

**Author(s)**

Kyun-Seop Bae k@acr.kr

**Examples**

```r
aov3(uptake ~ Plant + Type + Treatment + conc, CO2)
aov3(uptake ~ Plant + Type + Treatment + conc, CO2, BETA=TRUE)
aov3(uptake ~ Plant + Type + Treatment + conc, CO2, Resid=TRUE)
aov3(uptake ~ Plant + Type + Treatment + conc, CO2, BETA=TRUE, Resid=TRUE)
```

**Description**

The data is from 'Canner PL. An overview of six clinical trials of aspirin in coronary heart disease. Stat Med. 1987'

**Usage**

```r
aspirinCHD
```

**Format**

A data frame with 6 rows.

- `y1` death event count of aspirin group
- `n1` total subject of aspirin group
- `y2` death event count of placebo group
- `n2` total subject of placebo group

**Details**

This data is for educational purpose.

**References**

**BEdata**

*An Example Data of Bioequivalence Study*

**Description**

Contains Cmax data from a real bioequivalence study.

**Usage**

BEdata

**Format**

A data frame with 91 observations on the following 6 variables.

- **ADM** Admission or Hospitalization Group Code: 1, 2, or 3
- **SEQ** Group or Sequence character code: 'RT' or 'TR'
- **PRD** Period numeric value: 1 or 2
- **TRT** Treatment or Drug code: 'R' or 'T'
- **SUBJ** Subject ID
- **CMAX** Cmax values

**Details**

This contains a real data of 2x2 bioequivalence study, which has three different hospitalization groups. See Bae KS, Kang SH. Bioequivalence data analysis for the case of separate hospitalization. Transl Clin Pharmacol. 2017;25(2):93-100. doi.org/10.12793/tcp.2017.25.2.93

---

**bk**

*Beautify the output of knitr::kable*

**Description**

Trailing zeros after integer is somewhat annoying. This removes those in the vector of strings.

**Usage**

bk(ktab, rpltag=c("n", "N"), dig=10)

**Arguments**

- **ktab**: an output of knitr::kable
- **rpltag**: tag string of replacement rows. This is usually "n" which means the sample count.
- **dig**: maximum digits of decimals in the kable output
Details

This is convenient if used with tsum0, tsum1, tsum2, tsum3. This requires \texttt{knitr::kable}.

Value

A new processed vector of strings. The class is still \texttt{knitr_kable}.

Author(s)

Kyun-Seop Bae k@acr.kr

See Also

\texttt{tsum0, tsum1, tsum2, tsum3}

Examples

```r
# OUTPUT example
# t0 = tsum0(CO2, "uptake", c("mean", "median", "sd", "length", "min", "max"))
# bk(kable(t0)) \# requires knitr package
#
# |   x |
# |:----|
# | mean | 27.21310|
# | median | 28.30000|
# | sd | 10.81441|
# | n | 84 |
# | min | 7.70000|
# | max | 45.50000|
#
# t1 = tsum(uptake ~ Treatment, CO2,
# e=c("mean", "median", "sd", "min", "max", "length"),
# ou=c("chilled", "nonchilled"),
# repl=list(c("median", "length"), c("med", "N")))
#
# bk(kable(t1, digits=3)) \# requires knitr package
#
# |   x |
# |:----|
# | chilled| nonchilled| Combined|
# | mean | 23.783| 30.643| 27.213|
# | median | 19.700| 31.300| 28.300|
# | sd | 10.884| 9.705| 10.814|
# | min | 7.700| 10.600| 7.700|
# | max | 42.400| 45.500| 45.500|
# | N | 42 | 42 | 84 |
```
Description

GLM, REG, aov1 etc. functions can be run by levels of a variable.

Usage

BY(FUN, Formula, Data, By, ...)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FUN</td>
<td>Function name to be called such as GLM, REG</td>
</tr>
<tr>
<td>Formula</td>
<td>a conventional formula for a linear model.</td>
</tr>
<tr>
<td>Data</td>
<td>a data.frame to be analyzed</td>
</tr>
<tr>
<td>By</td>
<td>a variable name in the Data</td>
</tr>
<tr>
<td>...</td>
<td>arguments to be passed to FUN function</td>
</tr>
</tbody>
</table>

Details

This mimics SAS procedures’ BY clause.

Value

a list of FUN function outputs. The names are after each level.

Author(s)

Kyun-Seop Bae k@acr.kr

Examples

BY(GLM, uptake ~ Treatment + as.factor(conc), CO2, By="Type")
BY(REG, uptake ~ conc, CO2, By="Type")
Description

Get point estimate and its confidence interval with given contrast and alpha value using t distribution.

Usage

CIest(Formula, Data, Term, Contrast, conf.level=0.95)

Arguments

- Formula: a conventional formula for a linear model
- Data: a data.frame to be analyzed
- Term: a factor name to be estimated
- Contrast: a level vector. Level is alphabetically ordered by default.
- conf.level: confidence level of confidence interval

Details

Get point estimate and its confidence interval with given contrast and alpha value using t distribution.

Value

- Estimate: point estimate of the input linear contrast
- Lower CL: lower confidence limit
- Upper CL: upper confidence limit
- Std. Error: standard error of the point estimate
- t value: value for t distribution
- Df: degree of freedom
- Pr(>|t|): probability of larger than absolute t value from t distribution with residual’s degree of freedom

Author(s)

Kyun-Seop Bae k@acr.kr

Examples

CIest(log(CMAX) ~ SEQ/SUBJ + PRD + TRT, BEdata, "TRT", c(-1, 1), 0.90) # 90% CI
Collinearity Diagnostics

Description

Collearity diagnostics with tolerance, VIF, eigenvalue, condition index, variance proportions

Usage

Coll(Formula, Data)

Arguments

Formula  formula of the model
Data     input data as a matrix or data.frame

Details

Sometimes collinearity diagnostics after multiple linear regression are necessary.

Value

Tol  tolerance of independent variables
VIF  variance inflation factor of independent variables
Eigenvalue  eigenvalue of Z'Z (crossproduct) of standardized independent variables
Cond. Index  condition index
Proportions of variances  under the names of coefficients

Author(s)

Kyun-Seop Bae k@acr.kr

Examples

Coll(mpg ~ disp + hp + drat + wt + qsec, mtcars)
CONTR

F Test with a Set of Contrasts

Description
Do F test with a given set of contrasts.

Usage
CONTR(L, Formula, Data, mu=0)

Arguments
L contrast matrix. Each row is a contrast.
Formula a conventional formula for a linear model
Data a data.frame to be analyzed
mu a vector of mu for the hypothesis L. The length should be equal to the row count of L.

Details
It performs F test with a given set of contrasts (a matrix). It is similar to the CONTRAST clause of SAS PROC GLM. This can test the hypothesis that the linear combination (function)'s mean vector is mu.

Value
Returns sum of square and its F value and p-value.
Df degree of freedom
Sum Sq sum of square for the set of contrasts
Mean Sq mean square
F value F value for the F distribution
Pr(>F) probability of larger than F value

Author(s)
Kyun-Seop Bae k@acr.kr

See Also
cSS

Examples
CONTR(t(c(0, -1, 1)), uptake ~ Type, CO2) # sum of square
GLM(uptake ~ Type, CO2) # compare with the above
Cor.test  

Description
Testing correlation between numeric columns of data with Pearson method.

Usage
Cor.test(Data, conf.level=0.95)

Arguments
Data  a matrix or a data.frame
conf.level  confidence level

Details
It uses all numeric columns of input data. It uses "pairwise.complete.obs" rows.

Value
Row names show which columns are used for the test
Estimate  point estimate of correlation
Lower CL  upper confidence limit
Upper CL  lower confidence limit
t value  t value of the t distribution
Df  degree of freedom
Pr(>|t|)  probability with the t distribution

Author(s)
Kyun-Seop Bae k@acr.kr

Examples
Cor.test(mtcars)
corFisher

Correlation test by Fisher’s Z transformation

Description

Testing correlation between two numeric vectors by Fisher’s Z transformation

Usage

corFisher(x, y, conf.level=0.95, rho=0)

Arguments

x the first input numeric vector
y the second input numeric vector
conf.level confidence level
rho population correlation rho under null hypothesis

Details

This accepts only two numeric vectors.

Value

N sample size, length of input vectors
r sample correlation
Fisher.z Fisher’s z
bias bias to correct
rho.hat point estimate of population rho
conf.level confidence level for the confidence interval
lower lower limit of confidence interval
upper upper limit of confidence interval
rho0 population correlation rho under null hypothesis
p.value p value under the null hypothesis

Author(s)

Kyun-Seop Bae k@acr.kr

References

Fisher RA. Statistical Methods for Research Workers. 14e. 1973

Examples

corFisher(mtcars$disp, mtcars$hp, rho=0.6)
cSS  

_Sum of Square with a Given Contrast Set_

**Description**

Calculates sum of squares of a contrast from a `lfit` result.

**Usage**

```r
cSS(K, rx, mu=0, eps=1e-8)
```

**Arguments**

- **K**  
  contrast matrix. Each row is a contrast.
- **rx**  
  a result of `lfit` function
- **mu**  
  a vector of `mu` for the hypothesis `K`. The length should be equal to the row count of `K`.
- **eps**  
  Less than this value is considered as zero.

**Details**

It calculates sum of squares with given a contrast matrix and a `lfit` result. It corresponds to SAS PROC GLM CONTRAST. This can test the hypothesis that the linear combination (function)’s mean vector is `mu`.

**Value**

Returns sum of square and its F value and p-value.

- **Df**  
  degree of freedom
- **Sum Sq**  
  sum of square for the set of contrasts
- **Mean Sq**  
  mean square
- **F value**  
  F value for the F distribution
- **Pr(>F)**  
  probability of larger than F value

**Author(s)**

Kyun-Seop Bae k@acr.kr

**See Also**

`CONTR`

**Examples**

```r
rx = REG(uptake ~ Type, CO2, summarize=FALSE)
cSS(t(c(0, -1, 1)), rx)  # sum of square
GLM(uptake ~ Type, CO2)  # compare with the above
```
CumAlpha

**Cumulative Alpha for the Fixed Z-value**

**Description**

Cumulative alpha values with repeated hypothesis with a fixed upper bound z-value.

**Usage**

```r
CumAlpha(x, K=2, side=2)
```

**Arguments**

- **x**
  - fixed upper z-value bound for the repeated hypothesis test
- **K**
  - total number of tests
- **side**
  - 1=one-side test, 2=two-side test

**Details**

It calculates cumulative alpha-values for the even-interval repeated hypothesis test with a fixed upper bound z-value. It assumes linear (proportional) increase of information amount and Brownian motion of z-value, i.e. the correlation is sqrt(t_i/t_j).

**Value**

The result is a matrix.

- **ti**
  - time of test, Even-interval is assumed.
- **cum.alpha**
  - cumulative alpha valued

**Author(s)**

Kyun-Seop Bae k@acr.kr

**References**


**Examples**

```r
CumAlpha(x=qnorm(1 - 0.05/2), K=10) # two-side Z-test with alpha=0.05 for ten times
```
**CV**

*Coefficient of Variation in percentage*

---

**Description**

Coefficient of variation in percentage.

**Usage**

`CV(y)`

**Arguments**

- `y`: a numeric vector

**Details**

It removes `NA`.

**Value**

Coefficient of variation in percentage.

**Author(s)**

Kyun-Seop Bae k@acr.kr

**Examples**

`CV(mtcars$mpg)`

---

**Diffogram**

*Plot Pairwise Differences*

---

**Description**

Plot pairwise differences by a common.

**Usage**

`Diffogram(Formula, Data, Term, conf.level=0.95, adj="lsd", ...)`
Drift

Arguments

- **Formula**: a conventional formula for a linear model
- **Data**: a `data.frame` to be analyzed
- **Term**: a factor name to be estimated
- **conf.level**: confidence level of confidence interval
- **adj**: "lsd", "tukey", "scheffe", "bon", or "duncan" to adjust p-value and confidence limit
- **...**: arguments to be passed to `plot`

Details

This usually shows the shortest interval. It corresponds to SAS PROC GLM PDIFF. For adjust method "dunnett", see PDIFF function.

Value

no return value, but a plot on the current device

Author(s)

Kyun-Seop Bae k@acr.kr

See Also

LSM, PDIFF

Examples

```r
Diffogram(uptake ~ Type*Treatment + as.factor(conc), CO2, "as.factor(conc)"
```

Drift

Drift defined by Lan and DeMets for Group Sequential Design

Description

Calculate the drift value with given upper bounds (z-valuse), times of test, and power.

Usage

```r
Drift(bi, ti=NULL, Power=0.9)
```

Arguments

- **bi**: upper bound z-values
- **ti**: times of test. These should be in the range of [0, 1]. If omitted, even-interval is assumed.
- **Power**: target power at the final test
Details

It calculates the drift value with given upper bound z-values, times of test, and power. If the times of test is not given, even-interval is assumed. `mvtnorm::pmvt` (with noncentrality) is better than `pmvnorm` in calculating power and sample size. But, Lan-DeMets used multi-variate normal rather than multi-variate noncentral t distribution. This function followed Lan-DeMets for the consistency with previous results.

Value

Drift value for the given condition

Author(s)

Kyun-Seop Bae k@acr.kr

References


Examples

```r
Drift(seqBound(ti=(1:5)/5)[, "up.bound"])
```

---

### e1

**Get a Contrast Matrix for Type I SS**

Description

Makes a contrast matrix for type I SS using forward Doolittle method.

Usage

```r
e1(XpX, eps=1e-8)
```

Arguments

- **XpX**: crossproduct of a design or model matrix. This should have appropriate column names.
- **eps**: Less than this value is considered as zero.

Details

It makes a contrast matrix for type I SS. If `zapsmall` is used, the result becomes more inaccurate.

Value

A contrast matrix for type I SS.
**Author(s)**
Kyun-Seop Bae k@acr.kr

**Examples**

```r
x = ModelMatrix(uptake ~ Plant + Type + Treatment + conc, CO2)
round(e1(crossprod(x$X)), 12)
```

---

**e2**

*Get a Contrast Matrix for Type II SS*

**Description**
Makes a contrast matrix for type II SS.

**Usage**

```r
e2(x, eps=1e-8)
```

**Arguments**

- `x`: an output of ModelMatrix
- `eps`: Less than this value is considered as zero.

**Details**
It makes a contrast matrix for type II SS. If zapsmall is used, the result becomes more inaccurate.

**Value**
A contrast matrix for type II SS.

**Author(s)**
Kyun-Seop Bae k@acr.kr

**Examples**

```r
round(e2(ModelMatrix(uptake ~ Plant + Type + Treatment + conc, CO2)), 12)
round(e2(ModelMatrix(uptake ~ Type, CO2)), 12)
round(e2(ModelMatrix(uptake ~ Type - 1, CO2)), 12)
```
**e3**

*Get a Contrast Matrix for Type III SS*

**Description**

Makes a contrast matrix for type III SS.

**Usage**

\[ e3(x, \text{eps}=1e^{-8}) \]

**Arguments**

- **x**: an output of ModelMatrix
- **eps**: Less than this value is considered as zero.

**Details**

It makes a contrast matrix for type III SS. If zapsmall is used, the result becomes more inaccurate.

**Value**

A contrast matrix for type III SS.

**Author(s)**

Kyun-Seop Bae k@acr.kr

**Examples**

\[ \text{round(e3(ModelMatrix(uptake ~ Plant + Type + Treatment + conc, CO2)), 12)} \]

---

**EMS**

*Expected Mean Square Formula*

**Description**

Calculates a formula table for expected mean square of the given contrast. The default is for Type III SS.

**Usage**

\[ \text{EMS(Formula, Data, Type=3, eps=1e^{-8})} \]
est

Arguments

- **Formula**: a conventional formula for a linear model
- **Data**: a `data.frame` to be analyzed
- **Type**: type of sum of squares. The default is 3. Type 4 is not supported yet.
- **eps**: Less than this value is considered as zero.

Details

This is necessary for further hypothesis tests of nesting factors.

Value

A coefficient matrix for Type III expected mean square

Author(s)

Kyun-Seop Bae <k@acr.kr>

Examples

```r
f1 = log(CMAX) ~ SEQ/SUBJ + PRD + TRT
EMS(f1, BEdata)
EMS(f1, BEdata, Type=1)
EMS(f1, BEdata, Type=2)
```

Description

Estimates Linear Functions with a given GLM result.

Usage

```r
est(L, X, rx, conf.level=0.95, adj="lsd", paired=FALSE)
```

Arguments

- **L**: a matrix of linear contrast rows to be tested
- **X**: a model (design) matrix from `ModelMatrix`
- **rx**: a result of `lfit` function
- **conf.level**: confidence level of confidence limit
- **adj**: adjustment method for grouping. This supports "tukey", "bon", "scheffe", "duncan", and "dunnett". This only affects grouping, not the confidence interval.
- **paired**: If this is TRUE, L matrix is for the pairwise comparison such as `PDIFF` function.
Details

It tests rows of linear function. Linear function means linear combination of estimated coefficients. It corresponds to SAS PROC GLM ESTIMATE. Same sample size per group is assumed for the Tukey adjustment.

Value

<table>
<thead>
<tr>
<th>Description</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Estimate</td>
<td>point estimate of the input linear contrast</td>
</tr>
<tr>
<td>Lower CL</td>
<td>lower confidence limit by &quot;lsd&quot; method</td>
</tr>
<tr>
<td>Upper CL</td>
<td>upper confidence limit by &quot;lsd&quot; method</td>
</tr>
<tr>
<td>Std. Error</td>
<td>standard error of the point estimate</td>
</tr>
<tr>
<td>t value</td>
<td>value for t distribution for other than &quot;scheffe&quot; method</td>
</tr>
<tr>
<td>F value</td>
<td>value for F distribution for &quot;scheffe&quot; method only</td>
</tr>
<tr>
<td>Df</td>
<td>degree of freedom of residuals</td>
</tr>
<tr>
<td>Pr(&gt;</td>
<td>t</td>
</tr>
<tr>
<td>Pr(&gt;F)</td>
<td>probability of larger than F value from F distribution with residual’s degree of freedom, for &quot;scheffe&quot; method only</td>
</tr>
</tbody>
</table>

Author(s)

Kyun-Seop Bae k@acr.kr

See Also

ESTM, PDIFF

Examples

```r
x = ModelMatrix(uptake ~ Type, CO2)
rx = REG(uptake ~ Type, CO2, summarize=FALSE)
est(t(c(0, -1, 1)), x$X, rx) # Quebec - Mississippi
t.test(uptake ~ Type, CO2) # compare with the above
```

Description

Estimates Linear Function with a formula and a dataset.

Usage

```r
ESTM(L, Formula, Data, conf.level=0.95)
```
Arguments

- `L`: a matrix of linear functions rows to be tested
- `Formula`: a conventional formula for a linear model
- `Data`: a `data.frame` to be analyzed
- `conf.level`: confidence level of confidence limit

Details

It tests rows of linear functions. Linear function means linear combination of estimated coefficients. It is similar to SAS PROC GLM ESTIMATE. This is a convenient version of `est` function.

Value

- `Estimate`: point estimate of the input linear contrast
- `Lower CL`: lower confidence limit
- `Upper CL`: upper confidence limit
- `Std. Error`: standard error of the point estimate
- `t value`: value for t distribution
- `Df`: degree of freedom
- `Pr(>|t|)`: probability of larger than absolute t value from t distribution with residual’s degree of freedom

Author(s)

Kyun-Seop Bae k@acr.kr

See Also

- `est`

Examples

```r
ESTM(t(c(0, -1, 1)), uptake ~ Type, CO2) # Quevec - Mississippi
```

Description

Check the estimability of row vectors of coefficients.

Usage

```r
estmb(L, X, g2, eps=1e-8)
```
**Arguments**

- **L**: row vectors of coefficients
- **X**: a model (design) matrix from `ModelMatrix`
- **g2**: \( g2 \) generalized inverse of \( \text{crossprod}(X) \)
- **eps**: absolute value less than this is considered to be zero.

**Details**


**Value**

a vector of logical values indicating which row is estimable (as TRUE)

**Author(s)**

Kyun-Seop Bae k@acr.kr

**See Also**

- `G2SWEEP`

---

**ExitP**  
Exit Probability with cumulative Z-test in Group Sequential Design

**Description**

Exit probabilities with given drift, upper bounds, and times of test.

**Usage**

\[
\text{ExitP}(\Theta, \ bi, \ ti=NULL)
\]

**Arguments**

- **Theta**: drift value defined by Lan-DeMets. See the reference.
- **bi**: upper bound z-values
- **ti**: times of test. These should be in the range of [0, 1]. If omitted, even-interval is assumed.
**Details**

It calculates exit probabilities and cumulative exit probabilities with given drift, upper z-bounds and times of test. If the times of test is not given, even-interval is assumed. `mvtnorm::pmvt` (with noncentrality) is better than `pmvnorm` in calculating power and sample size. But, Lan-DeMets used multi-variate normal rather than multi-variate noncentral t distribution. This function followed Lan-DeMets for the consistency with previous results.

**Value**

The result is a matrix.

- `ti` time of test
- `bi` upper z-bound
- `cum.alpha` cumulative alpha-value

**Author(s)**

Kyun-Seop Bae k@acr.kr

**References**


**Examples**

```r
b0 = seqBound(ti=(1:5)/5)[, "up.bound"]
ExitP(Theta = Drift(b0), bi = b0)
```

---

**g2inv**

*Generalized type 2 inverse matrix, g2 inverse*

**Description**

Generalized inverse is usually not unique. Some programs use this algorithm to get a unique generalized inverse matrix. This uses SWEEP operator and works for non-square matrix also.

**Usage**

```r
g2inv(A, eps=1e-08)
```

**Arguments**

- `A` a matrix to be inverted
- `eps` Less than this value is considered as zero.
Details

Value
g2 inverse

Author(s)
Kyun-Seop Bae k@acr.kr

References

See Also
G2SWEEP

Examples
```r
A = matrix(c(1, 2, 4, 3, 3, -1, 2, -2, 5, -4, 0, -7), byrow=TRUE, ncol=4) ; A
g2inv(A)
```

---

G2SWEEP

*Generalized inverse matrix of type 2 for linear regression*

Description
Generalized inverse is usually not unique. Some programs use this algorithm to get a unique generalized inverse matrix.

Usage
```r
G2SWEEP(A, Augmented=FALSE, eps=1e-08)
```

Arguments
- **A**: a matrix to be inverted. If A is not a square matrix, G2SWEEP calls g2inv function.
- **Augmented**: If this is TRUE and A is a model(design) matrix X, the last column should be X’y, the last row y’X, and the last cell y’y. See the reference and example for the detail. If the input matrix A is not a square matrix, Augmented option cannot be TRUE.
- **eps**: Less than this value is considered as zero.
Details


Value

when Augmented=FALSE
  ordinary g2 inverse
when Augmented=TRUE
  g2 inverse and beta hats in the last column and the last row, and sum of square error (SSE) in the last cell
attribute "rank"
  the rank of input matrix

Author(s)

Kyun-Seop Bae k@acr.kr

See Also

lfit, ModelMatrix

Examples

```r
f1 = uptake ~ Type + Treatment # formula
x = ModelMatrix(f1, CO2) # Model matrix and relevant information
y = model.frame(f1, CO2)[, 1] # observation vector
nc = ncol(x$X) # number of columns of model matrix
XpY = crossprod(x$X, y)
aXPX = rbind(cbind(crossprod(x$X), XpY), cbind(t(XpY), crossprod(y)))
ag2 = G2SWEEP(aXPX, Augmented=TRUE)
b = ag2[1:nc, (nc + 1)] ; b # Beta hat
iXPX = ag2[1:nc, 1:nc] ; iXPX # g2 inverse of X'X
SSE = ag2[(nc + 1), (nc + 1)] ; SSE # Sum of Square Error
DFr = nrow(x$X) - attr(ag2, "rank") ; DFr # Degree of freedom for the residual
```

# Compare the below with the above
REG(f1, CO2)
aov1(f1, CO2)

---

**geoCV**

*Geometric Coefficient of Variation in percentage*

Description

Geometric coefficient of variation in percentage.
Usage

geoCV(y)

Arguments

y  a numeric vector

Details

It removes NA. This is \( \sqrt{\exp(\text{var}(\log(x))) - 1} \times 100 \).

Value

Geometric coefficient of variation in percentage.

Author(s)

Kyun-Seop Bae k@acr.kr

See Also

gemoan

Examples

geoCV(mtcars$mpg)


gemoan  \textit{Geometric Mean without NA}

Description

mean without NA values.

Usage

geoMean(y)

Arguments

y  a vector of numerics

Details

It removes NA in the input vector.

Value

gometric mean value
GLM

Author(s)
Kyun-Seop Bae k@acr.kr

See Also
gcov

Examples
geoMean(mtcars$mpg)

---

GLM General Linear Model similar to SAS PROC GLM

Description
GLM is the main function of this package.

Usage
GLM(Formula, Data, BETA=FALSE, EMEAN=FALSE, Resid=FALSE, conf.level=0.95, Weights=1)

Arguments
- **Formula**: a conventional formula for a linear model.
- **Data**: a data.frame to be analyzed
- **BETA**: if TRUE, coefficients (parameters) of REG will be returned. This is equivalent to SOLUTION option of SAS PROC GLM
- **EMEAN**: if TRUE, least square means (or expected means) will be returned. This is equivalent to LSMEANS clause of SAS PROC GLM
- **Resid**: if TRUE, fitted values (y hat) and residuals will be returned
- **conf.level**: confidence level for the confidence limit of the least square mean
- **Weights**: weights for the weighted least square

Details
It performs the core function of SAS PROC GLM. Least square means for the interaction term of three variables is not supported yet.
Value

The result is comparable to that of SAS PROC GLM.

ANOVA ANOVA table for the model
Fitness Some measures of goodness of fit such as R-square and CV
Type I Type I sum of square table
Type II Type II sum of square table
Type III Type III sum of square table
Parameter Parameter table with standard error, t value, p value. TRUE is 1, and FALSE is 0 in the Estimable column. This is returned only with BETA=TRUE option.
Expected Mean Least square (or expected) mean table with confidence limit. This is returned only with EMEAN=TRUE option.
Fitted Fitted value or y hat. This is returned only with Resid=TRUE option.
Residual Weighted residuals. This is returned only with Resid=TRUE option.

Author(s)

Kyun-Seop Bae k@acr.kr

Examples

GLM(uptake ~ Type*Treatment + conc, CO2[-1,]) # Making data unbalanced
GLM(uptake ~ Type*Treatment + conc, CO2[-1,], BETA=TRUE)
GLM(uptake ~ Type*Treatment + conc, CO2[-1,], EMEAN=TRUE)
GLM(uptake ~ Type*Treatment + conc, CO2[-1,], Resid=TRUE)
GLM(uptake ~ Type*Treatment + conc, CO2[-1,], BETA=TRUE, EMEAN=TRUE)
GLM(uptake ~ Type*Treatment + conc, CO2[-1,], BETA=TRUE, EMEAN=TRUE, Resid=TRUE)

is.cor

Is it a correlation matrix?

Description

Testing if the input matrix is a correlation matrix or not

Usage

is.cor(m, eps=1e-16)

Arguments

m a presumed correlation matrix
eps epsilon value. An absolute value less than this is considered as zero.
Details

A diagonal component should not be necessarily 1. But it should be close to 1.

Value

TRUE or FALSE

Author(s)

Kyun-Seop Bae k@acr.kr

Description

Kurtosis with a conventional formula.

Usage

Kurtosis(y)

Arguments

y a vector of numerics

Details

It removes NA in the input vector.

Value

Estimate of kurtosis

Author(s)

Kyun-Seop Bae k@acr.kr

See Also

KurtosisSE
**KurtosisSE**  
*Standard Error of Kurtosis*

**Description**  
Standard error of the estimated kurtosis with a conventional formula.

**Usage**  
KurtosisSE(y)

**Arguments**  
 y  
a vector of numerics

**Details**  
It removes NA in the input vector.

**Value**  
Standard error of the estimated kurtosis

**Author(s)**  
Kyun-Seop Bae k@acr.kr

**See Also**  
Kurtosis

---

**LCL**  
*Lower Confidence Limit*

**Description**  
The estimate of the lower bound of confidence limit using t-distribution

**Usage**  
LCL(y, conf.level=0.95)

**Arguments**  
 y  
a vector of numerics  
conf.level  
confidence level
Details

It removes NA in the input vector.

Value

The estimate of the lower bound of confidence limit using t-distribution

Author(s)

Kyun-Seop Bae k@acr.kr

See Also

UCL

---

**lfit**  
*Linear Fit*

Description

Fits a least square linear model.

Usage

```
lfit(x, y, eps=1e-8)
```

Arguments

- `x`: a result of ModelMatrix
- `y`: a column vector of response, dependent variable
- `eps`: Less than this value is considered as zero.

Details

Minimum version of least square fit of a linear model

Value

- `coefficients`: beta coefficients
- `g2`: g2 inverse
- `rank`: rank of the model matrix
- `DFr`: degree of freedom for the residual
- `SSE`: sum of squares error
- `SST`: sum of squares total
- `DFr2`: degree of freedom of the residual for beta coefficient
Author(s)

Kyun-Seop Bae k@acr.kr

See Also

ModelMatrix

Examples

```r
f1 = uptake ~ Type*Treatment + conc
x = ModelMatrix(f1, CO2)
y = model.frame(f1, CO2)[,1]
lfit(x, y)
```

---

lr  \textit{Linear Regression with g2 inverse}

Description

Coefficients calculated with g2 inverse. Output is similar to `summary(lm())`.

Usage

```r
lr(Formula, Data, eps=1e-8)
```

Arguments

- **Formula**: a conventional formula for a linear model
- **Data**: a data.frame to be analyzed
- **eps**: Less than this value is considered as zero.

Details

It uses G2SWEEP to get g2 inverse. The result is similar to `summary(lm())` without options.

Value

The result is comparable to that of SAS PROC REG.

- **Estimate**: point estimate of parameters, coefficients
- **Std. Error**: standard error of the point estimate
- **t value**: value for t distribution
- **Pr(>|t|)**: probability of larger than absolute t value from t distribution with residual’s degree of freedom
Author(s)
Kyun-Seop Bae k@acr.kr

Examples

\begin{verbatim}
\texttt{lr(uptake ~ Plant + Type + Treatment + conc, CO2)}
\texttt{lr(uptake ~ Plant + Type + Treatment + conc - 1, CO2)}
\texttt{lr(uptake ~ Type, CO2)}
\texttt{lr(uptake ~ Type - 1, CO2)}
\end{verbatim}

\begin{verbatim}
lr0
\end{verbatim}

\begin{center}
\textbf{Simple Linear Regressions with Each Independent Variable}
\end{center}

Description

Usually, the first step to multiple linear regression is simple linear regressions with a single independent variable.

Usage

\begin{verbatim}
lr0(Formula, Data)
\end{verbatim}

Arguments

\begin{verbatim}
Formula\hspace{0.5em}a conventional formula for a linear model. Intercept will always be added.
Data\hspace{0.5em}a \texttt{data.frame} to be analyzed
\end{verbatim}

Details

It performs simple linear regression for each independent variable.

Value

Each row means one simple linear regression with that row name as the only independent variable.

\begin{verbatim}
Intercept\hspace{0.5em}estimate of the intercept
SE(Intercept)\hspace{0.5em}standard error of the intercept
Slope\hspace{0.5em}estimate of the slope
SE(Slope)\hspace{0.5em}standard error of the slope
Rsq\hspace{0.5em}R-squared for the simple linear model
Pr(>F)\hspace{0.5em}p-value of slope or the model
\end{verbatim}

Author(s)
Kyun-Seop Bae k@acr.kr
Examples

\[
\text{lr0(up\text{t}ake \sim \text{Plant} + \text{Type} + \text{Treatment} + \text{conc, CO2})}
\]
\[
\text{lr0(mpg \sim ., mtcars)}
\]

---

LSM  
Least Square Means

Description

Estimates least square means using g2 inverse.

Usage

\[
\text{LSM(Formula, Data, Term, conf.level=0.95, adj="lsd", hideNonEst=TRUE,}
\]
\[
PLOT=FALSE, descend=FALSE, ...)
\]

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Formula</td>
<td>a conventional formula of model</td>
</tr>
<tr>
<td>Data</td>
<td>data.frame</td>
</tr>
<tr>
<td>Term</td>
<td>term name to be returned. If there is only one independent variable, this can be omitted.</td>
</tr>
<tr>
<td>conf.level</td>
<td>confidence level for the confidence limit</td>
</tr>
<tr>
<td>adj</td>
<td>adjustment method for grouping. &quot;lsd&quot;(default), &quot;tukey&quot;, &quot;bon&quot;, &quot;duncan&quot;, &quot;scheffe&quot; are available. This does not affects SE, Lower CL, Upper CL of the output table.</td>
</tr>
<tr>
<td>hideNonEst</td>
<td>logical. hide non-estimables</td>
</tr>
<tr>
<td>PLOT</td>
<td>logical. whether to plot LSMs and their confidence intervals</td>
</tr>
<tr>
<td>descend</td>
<td>logical. This specifies the plotting order be ascending or descending.</td>
</tr>
<tr>
<td>...</td>
<td>arguments to be passed to plot</td>
</tr>
</tbody>
</table>

Details

It corresponds to SAS PROC GLM LSMEANS. The result of the second example below may be different from emmeans. This is because SAS or this function calculates mean of the transformed continuous variable. However, emmeans calculates the average before the transformation. Interaction of three variables is not supported yet. For adjust method "dunnett", see PDIFF function.

Value

Returns a table of expectations, t values and p-values.

<table>
<thead>
<tr>
<th>Group</th>
<th>group character. This appears with one-way ANOVA or Term or adj argument is provided.</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSmean</td>
<td>point estimate of least square mean</td>
</tr>
</tbody>
</table>
Max

LowerCL  lower confidence limit with the given confidence level by "lsd" method
UpperCL  upper confidence limit with the given confidence level by "lsd" method
SE  standard error of the point estimate
Df  degree of freedom of point estimate

Author(s)
Kyun-Seop Bae k@acr.kr

See Also
PDIF, Diffogram

Examples

LSM(uptake ~ Type, CO2[-1,])
LSM(uptake ~ Type - 1, CO2[-1,])
LSM(uptake ~ Type*Treatment + conc, CO2[-1,])
LSM(uptake ~ Type*Treatment + conc - 1, CO2[-1,])
LSM(log(uptake) ~ Type*Treatment + log(conc), CO2[-1,])
LSM(log(uptake) ~ Type*Treatment + log(conc) - 1, CO2[-1,])
LSM(log(uptake) ~ Type*Treatment + as.factor(conc), CO2[-1,])
LSM(log(uptake) ~ Type*Treatment + as.factor(conc) - 1, CO2[-1,])
LSM(log(CMAX) ~ SEQ/SUBJ + PRD + TRT, BEdata)
LSM(log(CMAX) ~ SEQ/SUBJ + PRD + TRT - 1, BEdata)

Max

Max without NA

Description
maximum without NA values.

Usage
Max(y)

Arguments
y  a vector of numerics

Details
It removes NA in the input vector.

Value
maximum value
Mean

**Mean without NA**

**Description**
mean without NA values.

**Usage**
Mean(y)

**Arguments**
y a vector of numerics

**Details**
It removes NA in the input vector.

**Value**
mean value

**Author(s)**
Kyun-Seop Bae k@acr.kr

Median

**Median without NA**

**Description**
median without NA values.

**Usage**
Median(y)

**Arguments**
y a vector of numerics
**Min**

**Details**

It removes NA in the input vector.

**Value**

median value

**Author(s)**

Kyun-Seop Bae k@acr.kr

---

<table>
<thead>
<tr>
<th>Min</th>
<th>Min without NA</th>
</tr>
</thead>
</table>

**Description**

minimum without NA values.

**Usage**

Min(y)

**Arguments**

y a vector of numerics

**Details**

It removes NA in the input vector.

**Value**

minimum value

**Author(s)**

Kyun-Seop Bae k@acr.kr
ModelMatrix

Description
This model matrix is similar to model.matrix. But it does not omit unnecessary columns.

Usage
ModelMatrix(Formula, Data, KeepOrder=FALSE, XpX=FALSE)

Arguments
Formula a conventional formula for a linear model
Data a data.frame to be analyzed
KeepOrder If KeepOrder is TRUE, terms in Formula will be kept. This is for Type I SS.
XpX If XpX is TRUE, the cross-product of the design matrix (XpX, X'X) will be returned instead of the design matrix (X).

Details
It makes the model(design) matrix for GLM.

Value
Model matrix and attributes similar to the output of model.matrix.

X design matrix, i.e. model matrix
XpX cross-product of the design matrix, X'X
terms detailed information about terms such as formula and labels
termsIndices term indices
assign assignment of columns for each term in order, different way of expressing term indices

Author(s)
Kyun-Seop Bae k@acr.kr
mtest

Independent two groups t-test similar to PROC TTEST with summarized input

Description

This is comparable to SAS PROC TTEST except using summarized input (sufficient statistics).

Usage

mtest(m1, s1, n1, m0, s0, n0, conf.level=0.95)

Arguments

- m1: mean of the first (test, active, experimental) group
- s1: sample standard deviation of the first group
- n1: sample size of the first group
- m0: mean of the second (reference, control, placebo) group
- s0: sample standard deviation of the second group
- n0: sample size of the second group
- conf.level: confidence level

Details

This uses summarized input. This also produces confidence intervals of means and variances by group.

Value

The output format is comparable to SAS PROC TTEST.

Author(s)

Kyun-Seop Bae k@acr.kr

See Also

TTEST, tmtest, ztest

Examples

mtest(5.4, 10.5, 3529, 5.1, 8.9, 5190) # NEJM 388;15 p1386
N

Description
Number of observations excluding NA values

Usage
\( \text{N}(y) \)

Arguments
\( y \) a vector of numerics

Details
It removes NA in the input vector.

Value
Count of the observation

Author(s)
Kyun-Seop Bae k@acr.kr

OR

Description
Odds Ratio between two groups

Usage
\( \text{OR}(y_1, n_1, y_2, n_2, \text{conf.level}=0.95) \)

Arguments
\( y_1 \) positive event count of test (the first) group
\( n_1 \) total count of the test (the first) group
\( y_2 \) positive event count of control (the second) group
\( n_2 \) total count of control (the second) group
\( \text{conf.level} \) confidence level
Details

It calculates odds ratio of two groups. No continuity correction here. If you need percent scale, multiply the output by 100.

Value

The result is a data.frame.

- `odd1`: proportion from the first group
- `odd2`: proportion from the second group
- `OR`: odds ratio, `odd1/odd2`
- `SElog`: standard error of `log(OR)`
- `lower`: lower confidence limit of `OR`
- `upper`: upper confidence limit of `OR`

Author(s)

Kyun-Seop Bae k@acr.kr

See Also

RD, RR, RDmn1, RRmn1, ORmn1, RDmn, RRmn, ORmn

Examples

```r
OR(104, 11037, 189, 11034)  # no continuity correction
```

---

ORcmh

**Odds Ratio of two groups with strata by CMH method**

Description

Odds ratio and its score confidence interval of two groups with stratification by Cochran-Mantel-Haenszel method

Usage

```r
ORcmh(d0, conf.level=0.95)
```

Arguments

- `d0`: A data.frame or matrix, of which each row means a strata. This should have four columns named y1, n1, y2, and n2; y1 and y2 for events of each group, n1 and n2 for sample size of each strata. The second group is usually the control group.
- `conf.level`: confidence level
Details

It calculates odds ratio and its score confidence interval of two groups. This can be used for meta-analysis also.

Value

The following output will be returned for each stratum and common value. There is no standard error.

- odd1: odd from the first group, $y_1/(n_1 - y_1)$
- odd2: odd from the second group, $y_2/(n_2 - y_2)$
- OR: odds ratio, $\text{odd1}/\text{odd2}$. The point estimate of common OR is calculated with MH weight.
- lower: lower confidence limit of OR
- upper: upper confidence limit of OR

Author(s)

Kyun-Seop Bae k@acr.kr

See Also

RRmn, ORmn, RDmn1, RRmn1, ORmn1, RDmn, RRmn, ORmn, RDinv, RRinv, ORinv

Examples

```r
d1 = matrix(c(25, 339, 28, 335, 23, 370, 40, 364), nrow=2, byrow=TRUE)
colnames(d1) = c("y1", "n1", "y2", "n2")
ORcmh(d1)
```

Description

Odds ratio and its score confidence interval of two groups with stratification by inverse variance method

Usage

```r
ORinv(d0, conf.level=0.95)
```

Arguments

- `d0`: A data.frame or matrix, of which each row means a stratum. This should have four columns named y1, n1, y2, and n2; y1 and y2 for events of each group, n1 and n2 for sample size of each strata. The second group is usually the control group.
- `conf.level`: confidence level
Details

It calculates odds ratio and its confidence interval of two groups by inverse variance method. This supports stratification. This can be used for meta-analysis also.

Value

The following output will be returned for each stratum and common value. There is no standard error.

odd1  odd from the first group, y1/(n1 - y1)
odd2  odd from the second group, y2/(n2 - y2)
OR    odds ratio, odd1/odd2. The point estimate of common OR is calculated with MH weight.
lower lower confidence limit of OR
upper upper confidence limit of OR

Author(s)

Kyun-Seop Bae k@acr.kr

See Also

RDmn1, RRmn1, ORmn1, RDmn, RRmn, ORmn, RDinv, RRinv, ORcmh

Examples

d1 = matrix(c(25, 339, 28, 335, 23, 370, 40, 364), nrow=2, byrow=TRUE)
colnames(d1) = c("y1", "n1", "y2", "n2")
ORinv(d1)

ORmn

Odds Ratio and Score CI of two groups with strata by MN method

Description

Odds ratio and its score confidence interval of two groups with stratification by the Miettinen and Nurminen method

Usage

ORmn(d0, conf.level=0.95, eps=1e-8)
Arguments

d0  A data.frame or matrix, of which each row means a strata. This should have four columns named y1, n1, y2, and n2; y1 and y2 for events of each group, n1 and n2 for sample size of each strata. The second group is usually the control group.

conf.level  confidence level

eps  absolute value less than eps is regarded as negligible

Details

It calculates odds ratio and its score confidence interval of the two groups. The confidence interval is asymmetric, and there is no standard error in the output. This supports stratification. This implementation uses uniroot function, which usually gives at least 5 significant digits. Whereas PropCIs::orscoreci function uses incremental or decremental search by the factor of 1.001 which gives only about 3 significant digits. This can be used for meta-analysis also.

Value

The following output will be returned for each stratum and common value. There is no standard error.

odd1  odd from the first group, y1/(n1 - y1)
odd2  odd from the second group, y2/(n2 - y2)
OR  odds ratio, odd1/odd2. The point estimate of common OR is calculated with MN weight.
lower  lower confidence limit of OR
upper  upper confidence limit of OR

Author(s)

Kyun-Seop Bae k@acr.kr

References


See Also

RDmn1, RRmn1, ORmn1, RDmn, RRmn, RDinv, RRinv, ORinv, ORcmh

Examples

d1 = matrix(c(25, 339, 28, 335, 23, 370, 40, 364), nrow=2, byrow=TRUE)
colnames(d1) = c("y1", "n1", "y2", "n2")
ORmn(d1)
**Description**

Odds ratio and its score confidence interval of two groups without stratification

**Usage**

```r
ORmn1(y1, n1, y2, n2, conf.level=0.95, eps=1e-8)
```

**Arguments**

- `y1`: positive event count of test (the first) group
- `n1`: total count of the test (the first) group
- `y2`: positive event count of control (the second) group
- `n2`: total count of control (the second) group
- `conf.level`: confidence level
- `eps`: absolute value less than eps is regarded as negligible

**Details**

It calculates odds ratio and its score confidence interval of the two groups. The confidence interval is asymmetric, and there is no standard error in the output. This does not support stratification. This implementation uses uniroot function, which usually gives at least 5 significant digits. Whereas PropCIs::orscoreci function uses incremental or decremental search by the factor of 1.001 which gives only less than 3 significant digits.

**Value**

There is no standard error.

- `odd1`: odd from the first group, `y1/(n1 - y1)`
- `odd2`: odd from the second group, `y2/(n2 - y2)`
- `OR`: odds ratio, `odd1/odd2`
- `lower`: lower confidence limit of OR
- `upper`: upper confidence limit of OR

**Author(s)**

Kyun-Seop Bae k@acr.kr

**References**

See Also

RDmn1, RRmn1, RDmn, RRmn, ORmn

Examples

ORmn1(104, 11037, 189, 11034)

pB

Plot Confidence and Prediction Bands for Simple Linear Regression

Description

It plots bands of the confidence interval and prediction interval for simple linear regression.

Usage

pB(Formula, Data, Resol=300, conf.level=0.95, lx, ly, ...)

Arguments

  Formula       a formula
  Data          a data.frame
  Resol         resolution for the output
  conf.level    confidence level
  lx            x position of legend
  ly            y position of legend
  ...           arguments to be passed to plot

Details

It plots. Discard return values. If lx or ly is missing, the legend position is calculated automatically.

Value

Ignore return values.

Author(s)

Kyun-Seop Bae k@acr.kr

Examples

pB(hp ~ disp, mtcars)
pB(mpg ~ disp, mtcars)
**Pcor.test**  
*Partial Correlation test of multiple columns*

**Description**

Testing partial correlation between many columns of data with Pearson method.

**Usage**

```r
Pcor.test(Data, x, y)
```

**Arguments**

- `Data` a numeric matrix or data.frame
- `x` names of columns to be tested
- `y` names of control columns

**Details**

It performs multiple partial correlation test. It uses "complete.obs" rows of x and y columns.

**Value**

Row names show which columns are used for the test

- `Estimate` point estimate of correlation
- `Df` degree of freedom
- `t value` t value of the t distribution
- `Pr(>|t|)` probability with the t distribution

**Author(s)**

Kyun-Seop Bae k@acr.kr

**Examples**

```r
Pcor.test(mtcars, c("mpg", "hp", "qsec"), c("drat", "wt"))
```
**pD**  
*Diagnostic Plot for Regression*

**Description**  
Four standard diagnostic plots for regression.

**Usage**  
```r
pD(rx, Title=NULL)
```

**Arguments**  
- `rx`: a result of `lm`, which can give `fitted`, `residuals`, and `rstandard`.
- `Title`: title to be printed on the plot

**Details**  
Most frequently used diagnostic plots are 'observed vs. fitted', 'standardized residual vs. fitted', 'distribution plot of standard residuals', and 'Q-Q plot of standardized residuals'.

**Value**  
Four diagnostic plots in a page.

**Author(s)**  
Kyun-Seop Bae k@acr.kr

**Examples**  
```r
pD(lm(uptake ~ Plant + Type + Treatment + conc, CO2), "Diagnostic Plot")
```

---

**PDIFF**  
*Pairwise Difference*

**Description**  
Estimates pairwise differences by a common method.

**Usage**  
```r
PDIFF(Formula, Data, Term, conf.level=0.95, adj="lsd", ref, PLOT=FALSE, reverse=FALSE, ...)
```
Arguments

- **Formula**: a conventional formula for a linear model
- **Data**: a data.frame to be analyzed
- **Term**: a factor name to be estimated
- **conf.level**: confidence level of confidence interval
- **adj**: "lsd", "tukey", "scheffe", "bon", "duncan", or "dunnett" to adjust p-value and confidence limit
- **ref**: reference or control level for Dunnett test
- **PLOT**: whether to plot or not the diffogram
- **reverse**: reverse A - B to B - A
- **...**: arguments to be passed to plot

Details

It corresponds to PDIFF option of SAS PROC GLM.

Value

Returns a table of expectations, t values and p-values. Output columns may vary according to the adjustment option.

- **Estimate**: point estimate of the input linear contrast
- **Lower CL**: lower confidence limit
- **Upper CL**: upper confidence limit
- **Std. Error**: standard error of the point estimate
- **t value**: value for t distribution
- **Df**: degree of freedom
- **Pr(>|t|)**: probability of larger than absolute t value from t distribution with residual’s degree of freedom

Author(s)

Kyun-Seop Bae k@acr.kr

See Also

LSM, Diffogram

Examples

```r
PDIFF(uptake ~ Type*Treatment + as.factor(conc), CO2, "as.factor(conc)")
PDIFF(uptake ~ Type*Treatment + as.factor(conc), CO2, "as.factor(conc)", adj="tukey")
```
**PocockBound**

*Pocock (fixed) Bound for the cumulative Z-test with a final target alpha-value*

---

**Description**

Cumulative alpha values with cumulative hypothesis test with a fixed upper bound z-value in group sequential design.

**Usage**

```r
PocockBound(K=2, alpha=0.05, side=2)
```

**Arguments**

- `K` total number of tests
- `alpha` alpha value at the final test
- `side` 1=one-side test, 2=two-side test

**Details**

Pocock suggested a fixed upper bound z-value for the cumulative hypothesis test in group sequential designs.

**Value**

a fixed upper bound z-value for the K times repeated hypothesis test with a final alpha-value. Attributes are:

- `ti` time of test, Even-interval is assumed.
- `cum.alpha` cumulative alpha valued

**Author(s)**

Kyun-Seop Bae k@acr.kr

**References**


**Examples**

```r
PocockBound(K=2) # Z-value of upper bound for the two-stage design
```
**pResD**

*Residual Diagnostic Plot for Regression*

---

**Description**

Nine residual diagnostics plots.

**Usage**

```
pResD(rx, Title=NULL)
```

**Arguments**

- `rx`: a result of `lm`, which can give fitted, residuals, and rstandard.
- `Title`: title to be printed on the plot

**Details**

SAS-style residual diagnostic plots.

**Value**

Nine residual diagnostic plots in a page.

**Author(s)**

Kyun-Seop Bae k@acr.kr

**Examples**

```
pResD(lm(uptake ~ Plant + Type + Treatment + conc, CO2), "Residual Diagnostic Plot")
```

---

**QuartileRange**

*Inter-Quartile Range*

---

**Description**

Interquartile range (Q3 - Q1) with a conventional formula.

**Usage**

```
QuartileRange(y, Type=2)
```

**Arguments**

- `y`: a vector of numerics
- `Type`: a type specifier to be passed to IQR function
Details
It removes NA in the input vector. Type 2 is SAS default, while Type 6 is SPSS default.

Value
The value of an interquartile range

Author(s)
Kyun-Seop Bae k@acr.kr
RanTest

Test with Random Effects

Description
Hypothesis test of with specified type SS using random effects as error terms. This corresponds to SAS PROC GLM’s RANDOM /TEST clause.

Usage
RanTest(Formula, Data, Random="", Type=3, eps=1e-8)

Arguments
- Formula: a conventional formula for a linear model
- Data: a data.frame to be analyzed
- Random: a vector of random effects. All should be specified as primary terms, not as interaction terms. All interaction terms with random factor are regarded as random effects.
- Type: Sum of square type to be used as contrast
- eps: Less than this value is considered as zero.

Details
Type can be from 1 to 3. All interaction terms with random factor are regarded as random effects. Here the error term should not be MSE.

Value
Returns ANOVA and E(MS) tables with specified type SS.

Author(s)
Kyun-Seop Bae k@acr.kr

Examples
RanTest(log(CMAX) ~ SEQ/SUBJ + PRD + TRT, BEdata, Random="SUBJ")
fBE = log(CMAX) ~ ADM/SEQ/SUBJ + PRD + TRT
RanTest(fBE, BEdata, Random=c("ADM", "SUBJ"))
RanTest(fBE, BEdata, Random=c("ADM", "SUBJ"), Type=2)
RanTest(fBE, BEdata, Random=c("ADM", "SUBJ"), Type=1)
**RD**  

*Risk Difference between two groups*

**Description**

Risk (proportion) difference between two groups

**Usage**

```r
RD(y1, n1, y2, n2, conf.level=0.95)
```

**Arguments**

- `y1`: positive event count of test (the first) group
- `n1`: total count of the test (the first) group
- `y2`: positive event count of control (the second) group
- `n2`: total count of control (the second) group
- `conf.level`: confidence level

**Details**

It calculates risk difference between the two groups. No continuity correction here. If you need percent scale, multiply the output by 100.

**Value**

The result is a data.frame.

- `p1`: proportion from the first group
- `p2`: proportion from the second group
- `RD`: risk difference, `p1 - p2`
- `SE`: standard error of `RD`
- `lower`: lower confidence limit of `RD`
- `upper`: upper confidence limit of `RD`

**Author(s)**

Kyun-Seop Bae k@acr.kr

**See Also**

`RR`, `OR`, `RDmn1`, `RRmn1`, `ORmn1`, `RDmn`, `RRmn`, `ORmn`

**Examples**

```r
RD(104, 11037, 189, 11034) # no continuity correction
```
**RDinv**

*Risk Difference between two groups with strata by inverse variance method*

---

**Description**

Risk difference and its score confidence interval between two groups with stratification by inverse variance method

**Usage**

```
RDinv(d0, conf.level=0.95)
```

**Arguments**

- `d0`: A data.frame or matrix, of which each row means a stratum. This should have four columns named y1, n1, y2, and n2; y1 and y2 for events of each group, n1 and n2 for the sample size of each stratum. The second group is usually the control group.
- `conf.level`: confidence level

**Details**

It calculates risk difference and its confidence interval between two groups by inverse variance method. If you need percent scale, multiply the output by 100. This supports stratification. This can be used for meta-analysis also.

**Value**

The following output will be returned for each stratum and common value. There is no standard error.

- `p1`: proportion from the first group, y1/n1
- `p2`: proportion from the second group, y2/n2
- `RD`: risk difference, p1 - p2. The point estimate of common RD is calculated with MH weight.
- `lower`: lower confidence limit of RD
- `upper`: upper confidence limit of RD

**Author(s)**

Kyun-Seop Bae k@acr.kr

**See Also**

`RDmn1`, `RRmn1`, `ORmn1`, `RDmn`, `RRmn`, `ORmn`, `RRinv`, `ORinv`, `ORcmh`
Examples

d1 = matrix(c(25, 339, 28, 335, 23, 370, 40, 364), nrow=2, byrow=TRUE)
colnames(d1) = c("y1", "n1", "y2", "n2")
RDinv(d1)

Description

Risk difference and its score confidence interval between two groups with stratification by the Mi-ettinen and Nurminen method

Usage

RDmn(d0, conf.level=0.95, eps=1e-8)

Arguments

d0 A data.frame or matrix, of which each row means a stratum. This should have four columns named y1, n1, y2, and n2; y1 and y2 for events of each group, n1 and n2 for sample size of each stratum. The second group is usually the control group. Maximum allowable value for n1 and n2 is 1e8.

conf.level confidence level

eps absolute value less than eps is regarded as negligible

Details

It calculates risk difference and its score confidence interval between the two groups. The confidence interval is asymmetric, and there is no standard error in the output. If you need percent scale, multiply the output by 100. This supports stratification. This implementation uses uniroot function which usually gives at least 5 significant digits. This can be used for meta-analysis also.

Value

The following output will be returned for each stratum and common value. There is no standard error.

p1 proportion from the first group, y1/n1

p2 proportion from the second group, y2/n2

RD risk difference, p1 - p2. The point estimate of common RD is calculated with MN weight.

lower lower confidence limit of RD

upper upper confidence limit of RD
RDmn1

Author(s)
Kyun-Seop Bae k@acr.kr

References

See Also
RDmn1, RRmn1, ORmn1, RRmn, ORmn, RDinv, RRinv, ORinv, ORcmh

Examples

d1 = matrix(c(25, 339, 28, 335, 23, 370, 40, 364), nrow=2, byrow=TRUE)
colnames(d1) = c("y1", "n1", "y2", "n2")
RDmn(d1)

| RDmn1 | Risk Difference and Score CI between two groups without strata by the MN method |

Description
Risk difference and its score confidence interval between two groups without stratification

Usage
RDmn1(y1, n1, y2, n2, conf.level=0.95, eps=1e-8)

Arguments
- y1: positive event count of test (the first) group
- n1: total count of the test (the first) group. Maximum allowable value is 1e8.
- y2: positive event count of control (the second) group
- n2: total count of control (the second) group. Maximum allowable value is 1e8.
- conf.level: confidence level
- eps: absolute value less than eps is regarded as negligible

Details
It calculates risk difference and its score confidence interval between the two groups. The confidence interval is asymmetric, and there is no standard error in the output. If you need percent scale, multiply the output by 100. This does not support stratification. This implementation uses uniroot function which usually gives at least 5 significant digits.
Value

There is no standard error.

- \( p_1 \): proportion from the first group, \( y_1/n_1 \)
- \( p_2 \): proportion from the second group, \( y_2/n_2 \)
- \( RD \): risk difference, \( p_1 - p_2 \)
- \( \text{lower} \): lower confidence limit of RD
- \( \text{upper} \): upper confidence limit of RD

Author(s)

Kyun-Seop Bae k@acr.kr

References


See Also

\( \text{RRmn1, ORmn1, RDmn, RRmn, ORmn} \)

Examples

\( \text{RDmn1(104, 11037, 189, 11034)} \)

---

### REG

**Regression of Linear Least Square, similar to SAS PROC REG**

**Description**

REG is similar to SAS PROC REG.

**Usage**

\[
\text{REG}(\text{Formula, Data, conf.level=0.95, HC=FALSE, Resid=FALSE, Weights=1, summarize=TRUE})
\]

**Arguments**

- \( \text{Formula} \): a conventional formula for a linear model
- \( \text{Data} \): a data.frame to be analyzed
- \( \text{conf.level} \): confidence level for the confidence limit
- \( \text{HC} \): heteroscedasticity related output is required such as HC0, HC3, White’s first and second moment specification test
- \( \text{Resid} \): if TRUE, fitted values (y hat) and residuals will be returned
- \( \text{Weights} \): weights for each observation or residual square. This is usually the inverse of each variance.
- \( \text{summarize} \): If this is FALSE, REG returns just lfit result.
Details

It performs the core function of SAS PROC REG.

Value

The result is comparable to that of SAS PROC REG.
The first part is ANOVA table.
The second part is measures about fitness.
The third part is the estimates of coefficients.

<table>
<thead>
<tr>
<th>Estimate</th>
<th>point estimate of parameters, coefficients</th>
</tr>
</thead>
<tbody>
<tr>
<td>Estimable</td>
<td>estimability: 1=TRUE, 0=FALSE. This appears only when at least one inestimability occurs.</td>
</tr>
<tr>
<td>Std. Error</td>
<td>standard error of the point estimate</td>
</tr>
<tr>
<td>Lower CL</td>
<td>lower confidence limit with conf.level</td>
</tr>
<tr>
<td>Upper CL</td>
<td>lower confidence limit with conf.level</td>
</tr>
<tr>
<td>Df</td>
<td>degree of freedom</td>
</tr>
<tr>
<td>t value</td>
<td>value for t distribution</td>
</tr>
<tr>
<td>Pr(&gt;</td>
<td>t</td>
</tr>
</tbody>
</table>

The above result is repeated using HC0 and HC3, with following White’s first and second moment specification test, if HC option is specified. The t values and their p values with HC1 and HC2 are between those of HC0 and H3.

| Fitted | Fitted value or y hat. This is returned only with Resid=TRUE option. |
| Residual | Weighted residuals. This is returned only with Resid=TRUE option. |

If summarize=FALSE, REG returns;

| coefficients | beta coefficients |
| g2 | g2 inverse |
| rank | rank of the model matrix |
| DFr | degree of freedom for the residual |
| SSE | sum of square error |

Author(s)

Kyun-Seop Bae k@acr.kr

See Also

1r
Examples

REG(uptake ~ Plant + Type + Treatment + conc, CO2)
REG(uptake ~ conc, CO2, HC=TRUE)
REG(uptake ~ conc, CO2, Resid=TRUE)
REG(uptake ~ conc, CO2, HC=TRUE, Resid=TRUE)
REG(uptake ~ conc, CO2, summarize=FALSE)

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Formula</td>
<td>a conventional formula for a linear model</td>
</tr>
<tr>
<td>Data</td>
<td>a data.frame to be analyzed</td>
</tr>
</tbody>
</table>

Details

It performs the conventional regression analysis. This does not use g2 inverse, therefore it cannot handle a singular matrix. If the model (design) matrix is not full rank, use REG or fewer parameters.

Value

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coefficients</td>
<td>conventional coefficients summary with Wald statistics</td>
</tr>
<tr>
<td>Diagnostics</td>
<td>Diagnostics table for detecting outlier or influential/leverage points. This includes fitted (Predicted), residual (Residual), standard error of residual (se_resid), studentized residual (RStudent), hat(Leverage), Cook's D, studentized deleted residual (sdResid), DFFITS, and COVRATIO.</td>
</tr>
<tr>
<td>DFBETAS</td>
<td>Column names are the names of coefficients. Each row shows how much each coefficient is affected by deleting the corresponding row of observation.</td>
</tr>
</tbody>
</table>

Author(s)

Kyun-Seop Bae k@acr.kr

Examples

regD(uptake ~ conc, CO2)
Description

Relative Risk between the two groups

Usage

RR(y1, n1, y2, n2, conf.level=0.95)

Arguments

- y1: positive event count of test (the first) group
- n1: total count of the test (the first) group
- y2: positive event count of control (the second) group
- n2: total count of control (the second) group
- conf.level: confidence level

Details

It calculates relative risk of the two groups. No continuity correction here. If you need percent scale, multiply the output by 100.

Value

The result is a data.frame.

- p1: proportion from the first group
- p2: proportion from the second group
- RR: relative risk, p1/p2
- SElog: standard error of log(RR)
- lower: lower confidence limit of RR
- upper: upper confidence limit of RR

Author(s)

Kyun-Seop Bae k@acr.kr

See Also

RD, OR, RDmn1, RRmn1, ORmn1, RDmn, RRmn, ORmn

Examples

RR(104, 11037, 189, 11034) # no continuity correction
Relative Risk of two groups with strata by inverse variance method

Description

Relative risk and its score confidence interval of two groups with stratification by inverse variance method

Usage

RRinv(d0, conf.level=0.95)

Arguments

d0 A data.frame or matrix, of which each row means a stratum. This should have four columns named y1, n1, y2, and n2; y1 and y2 for events of each group, n1 and n2 for sample size of each stratum. The second group is usually the control group.

conf.level confidence level

Details

It calculates relative risk and its confidence interval of two groups by inverse variance method. This supports stratification. This can be used for meta-analysis also.

Value

The following output will be returned for each stratum and common value. There is no standard error.

p1 proportion from the first group, y1/n1

p2 proportion from the second group, y2/n2

RR relative risk, p1/p2. The point estimate of common RR is calculated with MH weight.

lower lower confidence limit of RR

upper upper confidence limit of RR

Author(s)

Kyun-Seop Bae k@acr.kr

See Also

RDmn1, RRmn1, ORmn1, RDmn, RRmn, ORmn, RDinv, ORinv, ORcmh
Examples

d1 = matrix(c(25, 339, 28, 335, 23, 370, 40, 364), nrow=2, byrow=TRUE)
colnames(d1) = c("y1", "n1", "y2", "n2")
RRinv(d1)

RRmn

Relative Risk and Score CI of two groups with strata by the MN method

Description

Relative risk and its score confidence interval of two groups with stratification by the Miettinen and Nurminen method

Usage

RRmn(d0, conf.level=0.95, eps=1e-8)

Arguments

d0 A data.frame or matrix, of which each row means a strata. This should have four columns named y1, n1, y2, and n2: y1 and y2 for events of each group, n1 and n2 for sample size of each stratum. The second group is usually the control group.
conf.level confidence level
eps absolute value less than eps is regarded as negligible

Details

It calculates relative risk and its score confidence interval of the two groups. The confidence interval is asymmetric, and there is no standard error in the output. This supports stratification. This implementation uses uniroot function, which usually gives at least 5 significant digits. Whereas PropCIs::riskscoreci function uses cubic equation approximation which gives only about 2 significant digits. This can be used for meta-analysis also.

Value

The following output will be returned for each strata and common value. There is no standard error.
p1 proportion from the first group, y1/n1
p2 proportion from the second group, y2/n2
RR relative risk, p1/p2. Point estimate of common RR is calculated with MN weight.
lower lower confidence limit of RR
upper upper confidence limit of RR

Author(s)

Kyun-Seop Bae k@acr.kr
References


See Also

RRmn1, RDmn1, ORmn1, RDmn, ORmn, RDinv, RRinv, ORinv, ORcmh

Examples

d1 = matrix(c(25, 339, 28, 335, 23, 370, 40, 364), nrow=2, byrow=TRUE)
colnames(d1) = c("y1", "n1", "y2", "n2")
RRmn(d1)

RRmn1 Relative Risk and Score CI of two groups without strata by by MN method

Description

Relative risk and its score confidence interval of the two groups without stratification

Usage

RRmn1(y1, n1, y2, n2, conf.level=0.95, eps=1e-8)

Arguments

y1 positive event count of test (the first) group
n1 total count of the test (the first) group
y2 positive event count of control (the second) group
n2 total count of control (the second) group
conf.level confidence level
eps absolute value less than eps is regarded as negligible

Details

It calculates the relative risk and its score confidence interval of the two groups. The confidence interval is asymmetric, and there is no standard error in the output. This does not support stratification. This implementation uses uniroot function, which usually gives at least 5 significant digits. Whereas PropCIs::riskscoreci function uses cubic equation approximation which gives only about 2 significant digits.
Value

There is no standard error.

- p1: proportion from the first group, y1/n1
- p2: proportion from the second group, y2/n2
- RR: relative risk, p1/p2
- lower: lower confidence limit of RR
- upper: upper confidence limit of RR

Author(s)

Kyun-Seop Bae k@acr.kr

References


See Also

RDmn1, ORmn1, RDmn, RRmn, ORmn

Examples

RRmn1(104, 11037, 189, 11034)

satt

Description

Calculates pooled variance and degree of freedom using Satterthwaite equation.

Usage

satt(vars, dfs, ws=c(1, 1))

Arguments

- vars: a vector of variances
- dfs: a vector of degree of freedoms
- ws: a vector of weights

Details

The input can be more than two variances.
ScoreCI

Value

<table>
<thead>
<tr>
<th>Variance</th>
<th>approximated variance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Df</td>
<td>degree of freedom</td>
</tr>
</tbody>
</table>

Author(s)

Kyun-Seop Bae k@acr.kr

---

ScoreCI  Score Confidence Interval for a Proportion or a Binomial Distribution

Description

Score confidence of a proportion in one group

Usage

ScoreCI(y, n, conf.level=0.95)

Arguments

<table>
<thead>
<tr>
<th>y</th>
<th>positive event count of a group</th>
</tr>
</thead>
<tbody>
<tr>
<td>n</td>
<td>total count of a group</td>
</tr>
<tr>
<td>conf.level</td>
<td>confidence level</td>
</tr>
</tbody>
</table>

Details

It calculates score confidence interval of a proportion in one group. The confidence interval is asymmetric and there is no standard error in the output. If you need percent scale, multiply the output by 100.

Value

The result is a data.frame. There is no standard error.

<table>
<thead>
<tr>
<th>PE</th>
<th>point estimation for the proportion</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lower</td>
<td>lower confidence limit of Prop</td>
</tr>
<tr>
<td>Upper</td>
<td>upper confidence limit of Prop</td>
</tr>
</tbody>
</table>

Author(s)

Kyun-Seop Bae k@acr.kr

See Also

binom.test, prop.test
**SD**

*Standard Deviation*

**Examples**

ScoreCI(104, 11037)

---

**SD**

Standard Deviation

**Description**

Standard deviation of a sample.

**Usage**

SD(y)

**Arguments**

- **y**: a vector of numerics

**Details**

It removes NA in the input vector. The length of the vector should be larger than 1.

**Value**

Sample standard deviation

**Author(s)**

Kyun-Seop Bae k@acr.kr

---

**SEM**

*Standard Error of the Sample Mean*

**Description**

The estimate of the standard error of the sample mean

**Usage**

SEM(y)

**Arguments**

- **y**: a vector of numerics
Details

It removes NA in the input vector.

Value

The estimate of the standard error of the sample mean

Author(s)

Kyun-Seop Bae k@acr.kr

seqBound

Sequential bounds for cumulative Z-test in Group Sequential Design

Description

Sequential upper bounds for cumulative Z-test on accumaltive data. Z values are correlated. This is usually used for group sequential design.

Usage

seqBound(ti, alpha = 0.05, side = 2, t2 = NULL, asf = 1)

Arguments

ti	times for test. These should be [0, 1].
alpha
goal alpha value for the last test at time 0.
side	1=one-side test, 2=two-side test
t2	fractions of information amount. These should be [0, 1]. If not available, ti will be used instead.
asf	alpha spending function. 1=O’Brien-Flemming, 2=Pocock, 3=alpha*ti, 4=alpha*ti^1.5, 5=alpha*ti^2

Details

It calculates upper z-bounds and cumulative alpha-values for the repeated test in group sequential design. The correlation is assumed to be sqrt(t_i/t_j).

Value

The result is a matrix.
ti
time of test
bi
upper z-bound
cum. alpha
cumulative alpha-value
**seqCI**

**Author(s)**
Kyun-Seop Bae k@acr.kr

**References**

**Examples**
```
seqBound(ti=(1:5)/5)
seqBound(ti=(1:5)/5, asf=2)
```

---

**seqCI**

*Confidence interval with the last Z-value for the group sequential design*

**Description**
Confidence interval with given upper bounds, time of tests, the last Z-value, and confidence level.

**Usage**
```
seqCI(bi, ti, Zval, conf.level=0.95)
```

**Arguments**
- **bi**: upper bound z-values
- **ti**: times for test. These should be [0, 1].
- **Zval**: the last z-value from the observed data. This is not necessarily the planned final Z-value.
- **conf.level**: confidence level

**Details**
It calculates confidence interval with given upper bounds, time of tests, the last Z-value, and confidence level. It assumes two-side test. `mvtnorm::pmvt` (with noncentrality) is better than `pmvnorm` in calculating power, sample size, and confidence interval. But, Lan-DeMets used multi-variate normal rather than multi-variate noncentral t distribution. This function followed Lan-DeMets for the consistency with previous results. For the theoretical background, see the reference.

**Value**
Confidence interval of Z-value for the given confidence level.

**Author(s)**
Kyun-Seop Bae k@acr.kr
References


Examples

```r
seqCI(bi = c(2.53, 2.61, 2.57, 2.47, 2.43, 2.38),
     ti = c(.2292, .3333, .4375, .5833, .7083, .8333), Zval=2.82)
```

Skewness

Skewness with a conventional formula.

Usage

```r
Skewness(y)
```

Arguments

- `y` a vector of numerics

Details

It removes `NA` in the input vector.

Value

Estimate of skewness

Author(s)

Kyun-Seop Bae k@acr.kr

See Also

`SkewnessSE`
SkewnessSE

Description
Standard error of the skewness with a conventional formula.

Usage
SkewnessSE(y)

Arguments
y a vector of numerics

Details
It removes NA in the input vector.

Value
Standard error of the estimated skewness

Author(s)
Kyun-Seop Bae k@acr.kr

See Also
Skewness

SLICE

Description
Do F test with a given slice term.

Usage
SLICE(Formula, Data, Term, By)
Arguments

- **Formula**: a conventional formula for a linear model
- **Data**: a data.frame to be analyzed
- **Term**: a factor name (not interaction) to calculate the sum of square and do F test with least square means
- **By**: a factor name to be used for slice

Details

It performs F test with a given slice term. It is similar to the SLICE option SAS PROC GLM.

Value

Returns sum of square and its F value and p-value. Row names are the levels of the slice term.

- **Df**: degree of freedom
- **Sum Sq**: sum of square for the set of contrasts
- **Mean Sq**: mean square
- **F value**: F value for the F distribution
- **Pr(>F)**: probability of larger than F value

Author(s)

Kyun-Seop Bae k@acr.kr

Examples

SLICE(uptake ~ Type*Treatment, CO2, "Type", "Treatment")
SLICE(uptake ~ Type*Treatment, CO2, "Treatment", "Type")

<table>
<thead>
<tr>
<th>SS</th>
<th>Sum of Square</th>
</tr>
</thead>
</table>

Description

Sum of squares with ANOVA.

Usage

\[ SS(x, rx, L, eps=1e-8) \]

Arguments

- **x**: a result of ModelMatrix containing design information
- **rx**: a result of lfit
- **L**: linear hypothesis, a full matrix matching the information in x
- **eps**: Less than this value is considered as zero.
Details

It calculates sum of squares and completes the ANOVA table.

Value

ANOVA table a classical ANOVA table without the residual(Error) part.

Author(s)

Kyun-Seop Bae k@acr.kr

See Also

ModelMatrix, lfit

Description

Calculates a formula table for expected mean square of Type III SS.

Usage

T3MS(Formula, Data, L0, eps=1e-8)

Arguments

Formula a conventional formula for a linear model
Data a data.frame to be analyzed
L0 a matrix of row linear contrasts, if missed, e3 is used
eps Less than this value is considered as zero.

Details

This is necessary for further hypothesis tests of nesting factors.

Value

A coefficient matrix for Type III expected mean square

Author(s)

Kyun-Seop Bae k@acr.kr

Examples

T3MS(log(CMAX) ~ SEQ/SUBJ + PRD + TRT, BEdata)
T3test

Test Type III SS using error term other than MSE

Description

Hypothesis test of Type III SS using an error term other than MSE. This corresponds to SAS PROC GLM’s RANDOM /TEST clause.

Usage

T3test(Formula, Data, H='", E='", eps=1e-8)

Arguments

- **Formula**: a conventional formula for a linear model
- **Data**: a `data.frame` to be analyzed
- **H**: Hypothesis term
- **E**: Error term
- **eps**: Less than this value is considered as zero.

Details

It tests a factor of type III SS using some other term as an error term. Here the error term should not be MSE.

Value

Returns one or more ANOVA table(s) of type III SS.

Author(s)

Kyun-Seop Bae k@acr.kr

Examples

T3test(log(CMAX) ~ SEQ/SUBJ + PRD + TRT, BEdata, E=c("SEQ:SUBJ"))
T3test(log(CMAX) ~ SEQ/SUBJ + PRD + TRT, BEdata, H="SEQ", E=c("SEQ:SUBJ"))
### tmtest

**Independent two means test similar to t.test with summarized input**

**Description**

This produces essentially the same to t.test except using summarized input (sufficient statistics).

**Usage**

```r
tmtest(m1, s1, n1, m0, s0, n0, conf.level=0.95, nullHypo=0, var.equal=F)
```

**Arguments**

- `m1`: mean of the first (test, active, experimental) group
- `s1`: sample standard deviation of the first group
- `n1`: sample size of the first group
- `m0`: mean of the second (reference, control, placebo) group
- `s0`: sample standard deviation of the second group
- `n0`: sample size of the second group
- `conf.level`: confidence level
- `nullHypo`: value for the difference of means under null hypothesis
- `var.equal`: assumption on the variance equality

**Details**

The default is Welch t-test with Satterthwaite approximation.

**Value**

The output format is very similar to t.test

**Author(s)**

Kyun-Seop Bae k@acr.kr

**See Also**

`mtest`, `TTEST`, `ztest`

**Examples**

```r
tmtest(5.4, 10.5, 3529, 5.1, 8.9, 5190)  # NEJM 388;15 p1386
tmtest(5.4, 10.5, 3529, 5.1, 8.9, 5190, var.equal=TRUE)
```
### trimmedMean

**Trimmed Mean**

**Description**
Trimmed mean wrapping mean function.

**Usage**
```
trimmedMean(y, Trim=0.05)
```

**Arguments**
- `y`: a vector of numerics
- `Trim`: trimming proportion. Default is 0.05

**Details**
It removes NA in the input vector.

**Value**
The value of trimmed mean

**Author(s)**
Kyun-Seop Bae k@acr.kr

### tsum

**Table Summary**

**Description**
Summarize a continuous dependent variable with or without independent variables.

**Usage**
```
tsum(Formula=NULL, Data=NULL, ColNames=NULL, MaxLevel=30, ...)
```

**Arguments**
- `Formula`: a conventional formula
- `Data`: a data.frame or a matrix
- `ColNames`: If there is no Formula, this will be used.
- `MaxLevel`: More than this will not be handled.
- `...`: arguments to be passed to tsum0, tsum1, tsum2, or tsum3
Details
A convenient summarization function for a continuous variable. This is a wrapper function to
tsum0, tsum1, tsum2, or tsum3.

Value
A data.frame of descriptive summarization values.

Author(s)
Kyun-Seop Bae k@acr.kr

See Also
tsum0, tsum1, tsum2, tsum3

Examples

tsum(lh)
t(tsum(CO2))
t(tsum(uptake ~ Treatment, CO2))
tsum(uptake ~ Type + Treatment, CO2)
print(tsum(uptake ~ conc + Type + Treatment, CO2), digits=3)

Description
Summarize a continuous dependent(y) variable without any independent(x) variable.

Usage
tsum0(d, y, e=c("Mean", "SD", "N"), repl=list(c("length"), c("n")))

Arguments
d a data.frame or matrix with colnames
y y variable name, a continuous variable
e a vector of summarize function names
repl list of strings to replace after summarize. The length of list should be 2, and both should have the same length.

Details
A convenient summarization function for a continuous variable.
Value
A vector of summarized values

Author(s)
Kyun-Seop Bae k@acr.kr

See Also
tsum, tsum1, tsum2, tsum3

Examples
```r
tsum0(CO2, "uptake")
tsum0(CO2, "uptake", repl=list(c("mean", "length"), c("Mean", "n")))
```

---

### tsum1

<table>
<thead>
<tr>
<th>Table Summary 1 independent(x) variable</th>
</tr>
</thead>
</table>

Description
Summarize a continuous dependent(y) variable with one independent(x) variable.

Usage
```r
tsum1(d, y, u, e=c("Mean", "SD", "N"), ou="", repl=list(c("length"), ("n")))
```

Arguments
- **d**: a data.frame or matrix with colnames
- **y**: y variable name. a continuous variable
- **u**: x variable name, upper side variable
- **e**: a vector of summarize function names
- **ou**: order of levels of upper side x variable
- **repl**: list of strings to replace after summarize. The length of list should be 2, and both should have the same length.

Details
A convenient summarization function for a continuous variable with one x variable.

Value
A data.frame of summarized values. Row names are from e names. Column names are from the levels of x variable.
Author(s)

Kyun-Seop Bae k@acr.kr

See Also

tsum, tsum0, tsum2, tsum3

Examples

```r
# Example 1
tsum1(CO2, "uptake", "Treatment")
# Example 2
tsum1(CO2, "uptake", "Treatment",
  e=c("mean", "median", "sd", "min", "max", "length"),
  ou=c("chilled", "nonchilled"),
  repl=list(c("median", "length"), c("med", "n")))
```

**tsum2**  
Table Summary 2 independent(x) variables

Description

Summarize a continuous dependent(y) variable with two independent(x) variables.

Usage

```r
tsum2(d, y, l, u, e=c("Mean", "SD", "N"), h=NULL, ol="", ou="", rm.dup=TRUE,
  repl=list(c("length"), c("n")))
```

Arguments

- **d**: a data.frame or matrix with colnames
- **y**: y variable name. a continuous variable
- **l**: x variable name to be shown on the left side
- **u**: x variable name to be shown on the upper side
- **e**: a vector of summarize function names
- **h**: a vector of summarize function names for the horizontal subgroup. If NULL, it becomes the same as e argument.
- **ol**: order of levels of left side x variable
- **ou**: order of levels of upper side x variable
- **rm.dup**: if TRUE, duplicated names of levels are specified on the first occurrence only.
- **repl**: list of strings to replace after summarize. The length of list should be 2, and both should have the same length.

Details

A convenient summarization function for a continuous variable with two x variables; one on the left side, the other on the upper side.
Value
A data.frame of summarized values. Column names are from the levels of u. Row names are basically from the levels of l.

Author(s)
Kyun-Seop Bae k@acr.kr

See Also
tsum, tsum0, tsum1, tsum3

Examples
tsum2(CO2, "uptake", "Type", "Treatment")
tsum2(CO2, "uptake", "Type", "conc")
tsum2(CO2, "uptake", "Type", "Treatment", e=c("mean", "median", "sd", "min", "max", "length"), ou=c("chilled", "nonchilled"), repl=list(c("median", "length"), c("med", "n")))

tsum3

Description
Summarize a continuous dependent(y) variable with three independent(x) variables.

Usage
tsum3(d, y, l, u, e=c("Mean", "SD", "N"), h=NULL, o11="", o12="", ou="", rm.dup=TRUE, repl=list(c("length"), c("n")))

Arguments
d a data.frame or matrix with colnames
y y variable name. a continuous variable
l a vector of two x variable names to be shown on the left side. The length should be 2.
u x variable name to be shown on the upper side
e a vector of summarize function names
h a list of two vectors of summarize function names for the first and second horizontal subgroups. If NULL, it becomes the same as e argument.
o11 order of levels of 1st left side x variable
o12 order of levels of 2nd left side x variable
TTEST

order of levels of upper side x variable

if TRUE, duplicated names of levels are specified on the first occurrence only.

list of strings to replace after summarize. The length of list should be 2, and both should have the same length.

Details

A convenient summarization function for a continuous variable with three x variables; two on the left side, the other on the upper side.

Value

A data.frame of summarized values. Column names are from the levels of u. Row names are basically from the levels of l.

Author(s)

Kyun-Seop Bae k@acr.kr

See Also

tsum, tsum0, tsum1, tsum2

Examples

tsum3(CO2, "uptake", c("Type", "Treatment"), "conc")
tsum3(CO2, "uptake", c("Type", "Treatment"), "conc",
  e=c("mean", "median", "sd", "min", "max", "length"),
  h=list(c("mean", "sd", "length"), c("mean", "length")),
  ol2=c("chilled", "nonchilled"),
  repl=list(c("median", "length"), c("med", "n")))

TTEST

Independent two groups t-test comparable to PROC TTEST

Description

This is comparable to SAS PROC TTEST.

Usage

TTEST(x, y, conf.level=0.95)

Arguments

x a vector of data from the first (test, active, experimental) group
y a vector of data from the second (reference, control, placebo) group
conf.level confidence level
Details

Caution on choosing the row to use in the output.

Value

The output format is comparable to SAS PROC TTEST.

Author(s)

Kyun-Seop Bae k@acr.kr

See Also

mtest, tmtest, ztest

Examples

TTEST(mtcars[mtcars$am==1, "mpg"], mtcars[mtcars$am==0, "mpg"])

<table>
<thead>
<tr>
<th>UCL</th>
<th>Upper Confidence Limit</th>
</tr>
</thead>
</table>

Description

The estimate of the upper bound of the confidence limit using t-distribution

Usage

UCL(y, conf.level=0.95)

Arguments

<table>
<thead>
<tr>
<th>y</th>
<th>a vector of numerics</th>
</tr>
</thead>
<tbody>
<tr>
<td>conf.level</td>
<td>confidence level</td>
</tr>
</tbody>
</table>

Details

It removes NA in the input vector.

Value

The estimate of the upper bound of the confidence limit using t-distribution

Author(s)

Kyun-Seop Bae k@acr.kr
Univariate Descriptive Statistics

Description

Returns descriptive statistics of a numeric vector.

Usage

UNIV(y, conf.level = 0.95)

Arguments

y a numeric vector
conf.level confidence level for confidence limit

Details

A convenient and comprehensive function for descriptive statistics. NA is removed during the calculation. This is similar to SAS PROC UNIVARIATE.

Value

nAll count of all elements in the input vector
nNA count of NA element
nFinite count of finite numbers
Mean mean excluding NA
SD standard deviation excluding NA
CV coefficient of variation in percent
SEM standard error of the sample mean, the sample mean divided by nFinite
LowerCL lower confidence limit of mean
UpperCL upper confidence limit of mean
TrimmedMean trimmed mean with trimming 1 - confidence level
Min minimum value
Q1 first quartile value
Median median value
Q3 third quartile value
Max maximum value
Range range of finite numbers. maximum - minimum
IQR inter-quartile range type 2, which is SAS default
MAD median absolute deviation
VarLL  lower confidence limit of variance
VarUL  upper confidence limit of variance
Skewness  skewness
SkewnessSE  standard error of skewness
Kurtosis  kurtosis
KurtosisSE  kurtosis
GeometricMean  geometric mean, calculated only when all given values are positive.
GeometricCV  geometric coefficient of variation in percent, calculated only when all given values are positive.

Author(s)
Kyun-Seop Bae k@acr.kr

Examples
UNIV(lh)

vttest  \textit{F-Test for the ratio of two groups’ variances}

Description
F-test for the ratio of two groups’ variances. This is similar to var.test except using the summarized input.

Usage
\texttt{vttest(v1, n1, v0, n0, ratio=1, conf.level=0.95)}

Arguments
\begin{itemize}
  \item \texttt{v1}  sample variance of the first (test, active, experimental) group
  \item \texttt{n1}  sample size of the first group
  \item \texttt{v0}  sample variance of the second (reference, control, placebo) group
  \item \texttt{n0}  sample size of the second group
  \item \texttt{ratio}  value for the ratio of variances under null hypothesis
  \item \texttt{conf.level}  confidence level
\end{itemize}

Details
For the confidence interval of one group, use UNIV function.
WhiteTest

Value
The output format is very similar to var.test.

Author(s)
Kyun-Seop Bae k@acr.kr

Examples
vtest(10.5^2, 5190, 8.9^2, 3529) # NEJM 388;15 p1386
vtest(2.3^2, 13, 1.5^2, 11, conf.level=0.9) # Red book p240

WhiteTest White’s Model Specification Test

Description
This is shown in SAS PROC REG as the Test of First and Second Moment Specification.

Usage
WhiteTest(rx)

Arguments
rx a result of lm

Details
This is also called as White’s general test for heteroskedasticity.

Value
Returns a direct test result by more coomplex theorem 2, not by simpler corollary 1.

Author(s)
Kyun-Seop Bae k@acr.kr

References

Examples
WhiteTest(lm(mpg ~ disp, mtcars))
ztest

Test for the difference of two groups’ means

Description

This is similar to two groups t-test, but using standard normal (Z) distribution.

Usage

ztest(m1, s1, n1, m0, s0, n0, conf.level=0.95, nullHypo=0)

Arguments

m1  mean of the first (test, active, experimental) group
s1  known standard deviation of the first group
n1  sample size of the first group
m0  mean of the second (reference, control, placebo) group
s0  known standard deviation of the second group
n0  sample size of the second group
conf.level  confidence level
nullHypo  value for the difference of means under null hypothesis

Details

Use this only for known standard deviations (or variances) or very large sample sizes per group.

Value

The output format is very similar to t.test

Author(s)

Kyun-Seop Bae k@acr.kr

See Also

mtest, tmtest, TTEST

Examples

ztest(5.4, 10.5, 3529, 5.1, 8.9, 5190) # NEJM 388;15 p1386
Index

* 2x2 Table
  ORcmh, 47
  ORmn, 49
  ORmn1, 51
  RD, 60
  RDmn, 62
  RDmn1, 63
  RR, 67
  RRmn, 69
  RRmn1, 70

* Descriptive Statistics
  UNIV, 89

* Group Sequential Design
  CumAlpha, 19
  Drift, 21
  ExitP, 28
  PocockBound, 56
  seqBound, 74
  seqCI, 75

* Meta-analysis
  ORinv, 48
  RDinv, 61
  RRinv, 68

* Plot
  pB, 52
  pD, 54
  pResD, 57

* Table Summary
  tsum, 82
  tsum0, 83
  tsum1, 84
  tsum2, 85
  tsum3, 86

* datasets
  aspirinCHD, 9
  BEdata, 10

* t-test
  mtest, 45
  tmtest, 81

TTEST, 87
ztest, 92
af, 5
ao1, 5
ao2, 7
ao3, 8
aspirinCHD, 9
BEdata, 10
binom.test, 72
bk, 10
BY, 12
C1est, 13
Coll, 14
CONTR, 15, 18
Cor.test, 16
corFisher, 17
cSS, 15, 18
CumAlpha, 19
CV, 20
Diffogram, 20, 41, 55
Drift, 21
e1, 22
e2, 23
e3, 24
EMS, 24
est, 25, 27
ESTM, 26, 26
estmb, 27
ESTIMATE (ESTM), 26
ExitP, 28
g2inv, 29
G2SWEEP, 28, 30, 30
goCV, 31, 33
geoMean, 32, 32
GLM, 33
is.cor, 34
Kurtosis, 35, 36
KurtosisSE, 35, 36
LCL, 36
lfit, 31, 37, 79
lr, 38, 65
lr0, 39
LSM, 21, 40, 55
Max, 41
Mean, 42
Median, 42
Min, 43
ModelMatrix, 31, 38, 44, 79
mtest, 45, 81, 88, 92
N, 46
OR, 46, 60, 67
ORcmh, 47, 49, 50, 61, 63, 68, 70
ORinv, 48, 48, 50, 61, 63, 68, 70
ORmn, 47–49, 49, 52, 60, 61, 63, 64, 67, 68, 70, 71
ORmn1, 47–50, 51, 60, 61, 63, 64, 67, 68, 70, 71
pB, 52
Pcor.test, 53
pD, 54
PDIFF, 21, 26, 41, 54
PocockBound, 56
pResD, 57
prop.test, 72
QuartileRange, 57
Range, 58
RanTest, 59
RD, 47, 60, 67
RDinv, 48–50, 61, 63, 68, 70
RDmn, 47–50, 52, 60, 61, 62, 64, 67, 68, 70, 71
RDmn1, 47–50, 52, 60, 61, 63, 67, 68, 70, 71
REG, 64
regD, 66
RR, 47, 60, 67
RRinv, 48–50, 61, 63, 68, 70
RRmn, 47–50, 52, 60, 61, 63, 64, 67, 68, 69, 71
RRmn1, 47–50, 52, 60, 61, 63, 64, 67, 68, 70, 70
sasLM (sasLM-package), 4
sasLM-package, 4
satt, 71
ScoreCI, 72
SD, 73
SEM, 73
seqBound, 74
seqCI, 75
Skewness, 76, 77
SkewnessSE, 76, 77
SLICE, 77
SS, 78
T3MS, 79
T3test, 80
tmtest, 45, 81, 88, 92
trimmedMean, 82
tsum, 82, 84–87
tsum0, 11, 83, 85–87
tsum1, 11, 83, 85–87
TTEST, 45, 81, 87, 92
UCL, 37, 88
UNIV, 89
vtest, 90
WhiteTest, 91
ztest, 45, 81, 88, 92