Package ‘BSagri’

February 19, 2015

Title  Statistical methods for safety assessment in agricultural field trials
Version 0.1-8
Date 2013-11-21
Author Frank Schaarschmidt
Description Collection of functions, data sets and code examples for evaluations of field trials with the objective of equivalence assessment.
Maintainer Frank Schaarschmidt <schaarschmidt@biostat.uni-hannover.de>
Depends gamlss, multcomp, MCPAN
Imports mvtnorm, boot, mratios
License GPL-2
NeedsCompilation no
Repository CRAN
Date/Publication 2013-11-21 11:50:03

R topics documented:

alignment ................................................................. 2
BOOT/Simpson .......................................................... 3
Brachycera ............................................................... 4
c2compnames .............................................................. 6
Cica1 ................................................................. 7
Cica2 ................................................................. 8
CIGLM ............................................................... 10
CInp ................................................................. 11
CountRep ............................................................ 13
Decomp ............................................................ 13
Diptera ............................................................ 15
ExNBCov ............................................................ 16
ExPCov ............................................................ 17
fakeln ............................................................ 18
Feeding ............................................................ 18
Allignment according to one factor

Description
Substracts the mean or median from observations belonging to the same level of a factor.

Usage
allignment(response, block, type = c("mean", "median"), ...)

Arguments
response a numeric vector
block a factor of the same length as response
type type of location measure to calculate and substract; only the choices "mean" and "median" are supported
... further arguments to be passed to mean or median

Details
Splits response according to the levels of block, calculates and substracts the mean or median and returns the resulting vector in appropriate order.

Value
A numeric vector.
**DESCRIPTION**

NOTE: This is a Test-version and is not sufficiently checked for correctness so far. Simultaneous confidence intervals for differences and ratios of Simpsons indices of diversity are calculated from data sets with repeated samples of communities and designs with more than two treatments groups. The intervals are calculated based on a stratified, non-parametric ordinary bootstrap sample of Simpsonindices, and applying the Algorithm of Besag et al.(1995) on the joint empirical distribution of differences (BOOTSimpsonD) or ratios (BOOTSimpsonR) of the original distribution.

**Usage**

BOOTSimpsonD(X, f, type = "Dunnett", cmat = NULL, conf.level = 0.95, alternative=c("two.sided", "less", "greater"), madj=TRUE, ...)

BOOTSimpsonR(X, f, type = "Dunnett", cmat = NULL, conf.level = 0.95, alternative=c("two.sided", "less", "greater"), madj=TRUE, ...)

**Arguments**

- **X**
  a data.frame of dimension n times p, containing integer entries as species counts of p species from n independent samplings

- **f**
  a factor, usually with more than two levels. Must be of length n, when X is an n times p matrix

- **type**
  a single character string, defining a contrast type. Supported options are 'Dunnett', 'Tukey', 'Sequen'; for more options, see ?contrMatratio in mratios when using BOOTSimpsonR or contrMat in multcomp when using BOOTSimpsonD

- **cmat**
  user defined contrasts: when using BOOTSimpsonD, this should be a numeric contrast matrix, with k columns when the factor f has k levels and should define differences of these levels; when using BOOTSimpsonR, this should be a list with two numeric contrast matrices in the entries numC and denC for numerators and denominator of the ratios, respectively; each having k columns when the factor f has k levels

- **conf.level**
  a single numeric value between 0.5 and 1

- **alternative**
  a single character string, one of 'two.sided', 'less' and 'greater'

- **madj**
  a single logical value, indicating whether simultaneous (if TRUE, the default) or marginal confidence intervals (if FALSE) shall be computed

- **...**
  Further arguments to be passed to the function boot from package boot; most importantly, the number of Bootstrap samples can be chosen via the parameter R (default is R=999); see ?boot for further options.
Details

NOTE: This is a test version!

Value

If madj=TRUE, an object of class "SCSnp", see SCSnp for details. If madj=FALSE, an object of class "CInp", see CInp for details.

Author(s)

Frank Schaarschmidt

See Also

SCSnp, these function internally make use of CCDiff.boot, CCDiff.default, CCRatio.boot, CCRatio.default, boot and estSimpsonf.

Examples

```r
X <- t(rmultinom(n=40, size=100,
    prob=c(0.3,0.2,0.2,0.1,0.05,0.05))

colnames(X) <- paste("Sp", 1:7, sep="")

DAT <- as.data.frame(X)

f <- as.factor(rep(c("A","B","C","D"), each=10))

SCIdunnettd <- BOOSTSimpsonD(X=DAT, f=f, type = "Dunnett",
    conf.level = 0.95, alternative = "two.sided")

SCIdunnettd
```

---

**Brachycera**

*Eklektor counts of Brachycera*

---

Description

In a field trial, 4 treatments were arranged in a randomized complete block design with 8 blocks and 32 plots. Soil eklektor traps were placed in each plot, on six dates from 2005-07-12 to 2005-09-25, the number of individuals of Brachycera (Flies, Order Diptera) hatching from soil were counted. The individuals were classified to the family level. Interest was in assessing potential effects of the novel treatment (Novum) on the abundance of Brachycera, compared to a near standard (Standard) and two additional standard treatments, A and B.
Usage

data(Brachycera)

Format

A data frame with 192 observations on the following 15 variables.

*Date* a POSIXt variable, the time of counting the individuals in the eklektor trap
*Treatment* a factor with 4 levels A B Standard Novum, where Novum is the novel treatment of interest in safety assessment, and Standard is the nearest standard treatment which commonly accepted. A and B are two additional standard treatments.
*Block* a numeric vector, specifying the eight blocks 1-8
*Plot* a factor with levels A1 A2 to Standard8, indicator of the individuals plots
*Agromy* a numeric vector, counts of individuals
*Anthom* a numeric vector, counts of individuals
*Callip* a numeric vector, counts of individuals
*Chloro* a numeric vector, counts of individuals
*Ephyd* a numeric vector, counts of individuals
*Droso* a numeric vector, counts of individuals
*Hybo* a numeric vector, counts of individuals
*Musci* a numeric vector, counts of individuals
*Phori* a numeric vector, counts of individuals
*Sphaer* a numeric vector, counts of individuals
*Total* a numeric vector, counts of individuals

Source

...

Examples

data(Brachycera)

par(mar=c(11,5,3,1))

boxplot(Total ~ Treatment*Date, data=Brachycera, las=2, col=c("white","white","blue","green"))

legend(x=15, y=80, legend=levels(Brachycera$Treatment), fill=c("white","white","blue","green"))
Define row names of a contrast matrix, depending on its column names

Description

Define row names of a contrast matrix, depending on its column names, as can be necessary for contrasts matrices. Currently, two options to do that are available.

Usage

c2compnames(cmat, ntype = "aggr")

Arguments

cmat: a contrast matrix

ntype: a single character string, defining how to build names from the column names of cmat, currently only "aggr" (aggregates names of groups with negative and positive coefficients), and "sequ"

Value

The input matrix cmat, with its row names replaced.

See Also

contrMat in multcomp to define contrast matrices of different types

Examples

# names for interaction contrasts:

n1<-c(10,10,10,10)
names(n1)<-c("A","B","C","D")
n2<-c(3,3,3)
names(n2)<-c(1,2,3)

library(multcomp)
CMT1<-contrMat(n1, type="Tukey")
CMT2<-contrMat(n2, type="Tukey")
IAC<-IAcontrastsCMAT(CMAT1=CMT1, CMAT2=CMT2)
c2compnames(IAC, ntype="aggr")
Cica1  

Catches of Planthoppers and Leafhoppers

Description

Data of a field trial concerning the impact of a genetically modified variety on the abundance of Planthoppers and Leafhoppers. The trial was designed as a randomized complete block design with 8 blocks (Row). In each block, three treatments were randomized: a conventional variety treated with insecticides (Insecticide), a genetically modified variety (GM), and the near-isogenic line (Iso) to the genetically modified line.

Usage

data(Cica1)

Format

A data frame with 24 observations on the following 6 variables.

Field a factor with levels 1 2, separating the two major sites of the trial. On field 1, the blocks 1-5 were situated, on field 2, blocks 6-8 were situated.

Row a factor with 8 levels, specifying the blocks: R1 R2 R3 R4 R5 R6 R7 R8

Year a numeric vector, for year 1 of the trial

Treatment a factor with 3 levels, specifying the genetically modified variety GM, the conventional variety treated with insecticides Insecticide, and the variety that was near-isogenic to the GM variety Iso

```r
# names for Williams-type contrasts:
C1 <- c(10,10,10,10)
names(C1) <- c("C0","D1","D5","D10")
CMW <- contrMat(C1, type="Williams")
c2compnames(CMW, ntype="aggr")
c2compnames(CMW, ntype="sequ")
```
Counts of Auchenorrhyncha by visual assessment
Counts of the major species Zyginidia scutellaris, caught by sweep nets

Source

Examples

data(Cica1)

layout(matrix(1:2,ncol=1))

ylim<-range(Cica1[,c("Au_Bonitur","Zs_sweep_netting")])

boxplot(Au_Bonitur ~ Treatment, data=Cica1,
main="Aucherrhyncha, visual assessment", ylim=ylim)

boxplot(Zs_sweep_netting ~ Treatment, data=Cica1,
main="Z.scutellaris, sweep netting", ylim=ylim)

Catches of Planthoppers and Leafhoppers

Description

Data of a field trial concerning the impact of a genetically modified variety on the abundance of Planthoppers and Leafhoppers. The trial was designed as a randomized complete block design with 8 blocks (Row). In each block, three treatments were randomized: a conventional variety treated with insecticides (Insecticide), a genetically modified variety (GM), and the near-isogenic line (Iso) the to genetically modified line. These data originate from the second year of the trial in Cica1.

Usage

data(Cica2)

Format

A data frame with 24 observations on the following 8 variables.

Field a factor with 2 levels, 1 2, separating the two major sites of the trial. On field 1, the blocks 1-5 were situated, on field 2, blocks 6-8 were situated.

Row a factor with 8 levels, specifying the blocks: R1 R2 R3 R4 R5 R6 R7 R8

Year a numeric vector, with value 2 for the second year
Treatment a factor with 3 levels, specifying the genetically modified variety GM, the conventional variety treated with insecticides Insecticide, and the variety that was near-isogenic to the GM variety Iso

Au_Bonitur counts of Auchenorhyncha by visual assessment
Zs_sweep_netting counts of the major species Zyginidia scutellaris, caught by sweep nets
Zs_yellow_traps counts of Zyginidia scutellaris, caught by yellow traps
Zs_stick_traps counts of Zyginidia scutellaris, caught by sticky traps

Source

...

References

See Cica1 for data of the same trial a year earlier

Examples

data(Cica2)

# A comparison of the treatments:
layout(matrix(1:4,ncol=4))
ylim<range(Cica2[,c("Au_Bonitur","Zs_sweep_netting","Zs_yellow_traps","Zs_stick_traps")])
boxplot(Au_Bonitur ~ Treatment, data=Cica2,
main= "Aucherrhyncha, visual assessment", ylim=ylim, horizontal=TRUE, las=1)

boxplot(Zs_sweep_netting ~ Treatment, data=Cica2,
main="Z.scutellaris, sweep netting", ylim=ylim, horizontal=TRUE, las=1)

boxplot(Zs_yellow_traps ~ Treatment, data=Cica2,
main="Z.scutellaris, yellow traps", ylim=ylim, horizontal=TRUE, las=1)

boxplot(Zs_stick_traps ~ Treatment, data=Cica2,
main="Z.scutellaris, sticky traps", ylim=ylim, horizontal=TRUE, las=1)

# A comparison of sampling methods:
pairs(Cica2[,c("Au_Bonitur","Zs_sweep_netting","Zs_yellow_traps","Zs_stick_traps")])
Description

Computes confidence intervals from the output of a glm, by calling to glht(multcomp).

Usage

CIGLM(x, conf.level = 0.95, method = c("Raw", "Adj", "Bonf"))

Arguments

- **x**: a object of class "glm", "gamlss", or "glm.nb" as can be obtained by calling to function glm, function gamlss in package gamlss, or function glm.nb in package MASS
- **conf.level**: confidence level, a single numeric value between 0.5 and 1
- **method**: a single character string, with "Raw" for unadjusted intervals, "Adj" for multiplicity adjusted intervals taking the correlations into account, or "Bonf" for a Bonferroni adjustment

Details

This is just a wrapper to confint.glht of package multcomp. Note that except for the simple general linear model with assumption of Gaussian response, the resulting intervals are exact intervals. In other cases, the methods are only asymptotically correct, hence might give misleading results for small sample sizes!

Value

An object of class "confint.glht"

See Also

confint.glht in package multcomp for the function that is used internally, UnlogCI for a simple function to bring confidence intervals back to the original scales when there is a log or logit link, with appropriate naming.

Examples

data(Diptera)
library(multcomp)

modelfit <- glm(Ges ~ Treatment, data=Diptera, family=quasipoisson)
comps <- glht(modelfit, mcp(Treatment="Tukey"))
CIs<-CIGLM(comps, method="Raw")
CInp

CIs
CIsAdj<-CIGLM(comps, method="Adj")
CIsAdj

CIsBonf<-CIGLM(comps, method="Bonf")
CIsBonf

library(gamlss)

modelfit2 <- gamlss(Ges ~ Treatment, data=Diptera, family=NBI)
comps2 <- glht(modelfit2, mcp(Treatment="Tukey"))
CIs2<-CIGLM(comps2, method="Raw")
CIs2

CIsAdj2<-CIGLM(comps2, method="Adj")
CIsAdj2

CIsBonf2<-CIGLM(comps2, method="Bonf")
CIsBonf2

CInp  

Construct local confidence intervals from joint empirical distribution.

Description

Construct local confidence intervals for each parameter from the empirical joint distribution of a parameter vector of length P.

Usage

## Default S3 method:
CInp(x, conf.level = 0.95,
alternative = "two.sided", ...)

## S3 method for class 'CCRatio'
CInp(x, ...)

## S3 method for class 'CCDiff'
CInp(x, ...)

## S3 method for class 'bugs'
CInp(x, conf.level = 0.95,
alternative = "two.sided", whichp = NULL, ...)
Arguments

- **x**: an N-times-P matrix, or an object of class CCRatio, CCDiff, bugs, as can be obtained by calling the functions CCRatio, CCDiff, or openbugs in package R2WinBUGS.
- **conf.level**: a single numeric value between 0.5 and 1, specifying the local confidence level for each of the P parameters.
- **alternative**: a single character string, one of "two.sided", "less", "greater", for two-sided, upper and lower limits.
- **whichp**: a single character string, naming an element of the sims.list if x is a bugs object, ignored otherwise.
- **...**: currently not used.

Details

Construct simple confidence intervals based on order statistics applied to the marginal empirical distributions in x.

Value

An object of class "CInp", a list with elements:

- **conf.int**: a P-times-2 matrix containing the lower and upper confidence limits.
- **estimate**: a numeric vector of length P, containing the medians of the P marginal empirical distributions.
- **x**: the input object.
- **k**: the number of values outside each confidence interval, i.e. conf.level*N.
- **N**: the number of values used to construct each confidence interval.
- **conf.level**: a single numeric value, the nominal confidence level, as input.
- **alternative**: a single character string, as input.

See Also

The function internally used is quantile with its default settings. See SCSnp for simultaneous sets.

Examples

```r
# Assume a 100 times 4 matrix of 4 mutually independent normal variables:
X<-cbind(rnorm(100), rnorm(100), rnorm(100), rnorm(100))

lcits<-CInp(x=X, conf.level=0.95, alternative="two.sided")

lcits

ci1<-lcits$conf.int[1,]
```
Simulated count data incl. repeated measurements

Description
Simulated data set of repeated count data within subjects.

Usage
data(CountRep)

Format
A data frame with 160 observations on the following 4 variables.

Abundance a numeric vector with counts simulated from overdispersed and autocorrelated Poisson distributions
ID a factor with levels N1 N2 ..., n40, specifying the subject
Time a factor with levels T1 T2 T3 T4, specifying the time
Treatment a factor with levels N S1 S2 S3

Examples
data(CountRep)

Decomp
A simulated data set

Description
A simulated data set, resembling an experiment on the decomposition of plant material from four different varieties in soil.

Usage
data(Decomp)
**Format**

A data frame with 1152 observations on the following 5 variables.

- **Block** a numeric vector, giving the names of the Blocks
- **PID** a numeric vector, giving the number of the experimental unit
- **Variety** a factor with levels N S1 S2 S3, specifying the variety assigned to the experimental unit, randomized within each Block
- **Time** a factor with levels t1 t2 t3 t4 t5 t6, specifying time points at which measurements were taken
- **Perc** a numeric vector, giving the percentage of material

**Details**

The experiment is a randomized complete block design, with repeated measurements within each experimental unit and additional subsamplings within each time point. Plant material from four different varieties was deposited in bags in soil of 32 experimental units (coded by the variable PID), where the varieties had been grown in the vegetation period before. A total number of 36 bags was placed in each experimental unit. At six different time points, plant material was excavated and the content of the bags was analysed concerning the percentage of decomposition relative to the content at the begin of the experiment. At each time point, six bags were excavated at each experimental unit. Some bags could not be found anymore (data missing). The objective was to assess whether properties of the varieties obstruct the decomposition of plant material in the soil. The variety N is of special interest, while the varieties S1, S2, S3 are standard varieties.

Note, that this data set merely serves as an example to evaluate clustered data. Neither in the mean effects nor in the actual data points it does resemble a true experiment!

**Source**

Simulated.

**References**

Resembling the experimental structure of an experiment performed by Sabine Prescher (JKI Braunschweig).

**Examples**

```r
data(Decomp)
```
Description

Hatching of some families of Diptera was recorded in summer 2005 using eklektors covering 2 square meters of soil surface each. A total of 32 eklektors were arranged in a randomized field trial. Total counts of individuals over the whole season are reported. Aim was to assess the impact of a novel treatment on the abundance of Diptera with larval development in the soil, compared to three standard treatments.

Usage

data(Diptera)

Format

A data frame with 32 observations on the following 7 variables.

- Callip: a numeric vector
- Chloro: a numeric vector
- Ephyd: a numeric vector
- Droso: a numeric vector
- Ges: a numeric vector, total number of species
- Chiro: a numeric vector
- Treatment: a factor, specifying the four different treatments, with levels S1, S2 for two standard treatments, SNovum for the standard treatment most similar to the novel treatment, and Novum, for the novel treatment

Source

personal communications S. Prescher, JKI Braunschweig, Germany

Examples

data(Diptera)

layout(matrix(1:6, nrow=3))

boxplot(Callip~Treatment, data=Diptera, horizontal=TRUE, las=1, main="Abundanz Callip", col=c("white","white","blue","red"))

boxplot(Chloro~Treatment, data=Diptera, horizontal=TRUE, las=1, main="Abundanz Chloro", col=c("white","white","blue","red"))

boxplot(Ephyd~Treatment, data=Diptera, horizontal=TRUE, las=1,
Simulated example data, drawn from a Negative Binomial Distribution

Description

Simulated example data, response drawn from a Negative Binomial Distribution, Covariables follow a multivariate normal distribution.

Usage

data(ExNBCov)

Format

A data frame with 32 observations on the following 12 variables.

Resp a numeric vector, a response of counts
Group a factor with levels A1 A2 A3 A4, e.g. varieties
X1 a numeric covariable
X2 a numeric covariable
X3 a numeric covariable
X4 a numeric covariable
X5 a numeric covariable
X6 a numeric covariable
X7 a numeric covariable
X8 a numeric covariable
X9 a numeric covariable
X10 a numeric covariable
Examples

data(ExNB Cov)

boxplot(Resp ~ Group, data=ExNB Cov)
pairs(ExNB Cov)

Description

Response follows a Poisson distribution, the covariables follow a multivariate normal on the log-scale.

Usage

data(ExPCov)

Format

A data frame with 32 observations on the following 12 variables.

- resp  a response of counts
- A a factor with levels A1 A2 A3 A4, e.g. varieties
- C1 a numeric covariable
- C2 a numeric covariable
- C3 a numeric covariable
- C4 a numeric covariable
- C5 a numeric covariable
- C6 a numeric covariable
- C7 a numeric covariable
- C8 a numeric covariable
- C9 a numeric covariable
- C10 a numeric covariable

Examples

data(ExPCov)

boxplot(resp ~ A, data=ExPCov)
pairs(ExPCov)
fakeln

A simulated data set of lognormal data

Description

A simulated data set of lognormal data, could be concentrations

Usage

data(fakeln)

Format

A data frame with 32 observations on the following 2 variables.

- **Concmug**: a numeric vector, serving as response variable
- **Treat**: a factor with levels N S S\textsubscript{a} S\textsubscript{b}

Examples

data(fakeln)
boxplot(Concmug ~ Treat, data=fakeln)

Feeding

Pupation and Hatching rate in a feeding experiment with four varieties

Description

Larvae of a non-target organism were fed with plant material derived from a novel variety (Novum), material from three standard varieties (NStandard: the standard variety most similar to Novum, and two additional standard varieties S\textsubscript{1} and S\textsubscript{2}). Objective was to assess the impact of Novum on the pupation and hatching rate of an animal that potentially feeds on plant material compared to accepted standard varieties.

Usage

data(Feeding)
**IAcontrasts**

**Format**

A data frame with 32 observations on the following 5 variables.

- **Rep**: a factor with 32 levels indexing the 32 replications
- **Variety**: a factor with 4 levels: S1 and S2 are two standard varieties, Novum is a novel variety, and NStandard is the standard variety most similar to Novum
- **Total**: the total number of animals in each experimental unit
- **Pupating**: number of individuals pupating in each unit, the others died
- **Hatching**: number of individuals hatching from the pupae

**Examples**

```r
data(Feeding)
# Larval mortality:
Feeding$Lmort <- Feeding$Total - Feeding$Pupating
fit1<-.glm(cbind(Pupating,Lmort)~Variety,data=Feeding, family=quasibinomial)
anova(fit1, test="F")
```

---

**IAcontrasts**

**Interaction contrasts for 2-factorial designs**

**Description**

Builds a family of intercation contrasts for complete two-factorial designs.

**Usage**

```r
IAcontrasts(type, k)
```

**Arguments**

- **type**: a vector of two character strings, specifying the contrast type, currently, "Tukey", "Dunnett", "Sequence" or "Identity" are accepted
- **k**: a vector of two integers, specifying the number of groups in each factor of the two-factorial design

**Details**

Creates contrast matrices using `contrMat` from package `multcomp`, creates the kronecker product of both and creates suitable columnnames.
**IAcontrastsCMAT**

**Value**

A matrix with k[1]*k[2] columns and a number of rows depending on type.

**See Also**

for 2-way interaction contrasts directly based on 2 contrasts matrices, see **IAcontrastsCMAT**; two possibilities to specify appropriate rownames are implemented in function **c2compnames**

**Examples**

```r
IAC<-IAcontrasts(type=c("Tukey", "Tukey"), k=c(3,4))
IAC
IAC2<-c2compnames(IAC, ntype="sequ")
IAC2
```

**Description**

Builds a family of interaction contrasts for complete two-factorial designs.

**Usage**

`IAcontrastsCMAT(CMAT1, CMAT2)`

**Arguments**

- **CMAT1** a (named) contrast matrix
- **CMAT2** a (named) contrast matrix

**Details**

Builds the kronecker product of CMAT1 and CMAT2 and creates suitable columnnames. Note that CMAT1 and CMAT2 are not checked, and hence its up to the user to define them suitably.

**Value**


**See Also**

for interaction contrasts based on contrast definition and the number of levels of the factors in a two-way layout, see **IAcontrasts**; two possibilities to specify appropriate rownames are implemented in function **c2compnames**
Examples

library(multcomp)

n1<-c(10,10,10,10)
names(n1)<-c("A","B","C","D")
n2<-c(3,3,3)
names(n2)<-c(1,2,3)

CMT1<-contrMat(n1, type="Tukey")
CMT2<-contrMat(n2, type="Tukey")
IAC<-IAcontrastsCMAT(CMAT1=CMT1, CMAT2=CMT2)
c2compnames(IAC, ntype="sequ")

###

n1<-c(10,10,10,10)
names(n1)<-c("A","B","C","D")
n2<-c(3,3,3)
names(n2)<-c(1,2,3)

CMD1<-contrMat(n1, type="Dunnett")
CMD2<-contrMat(n2, type="Dunnett")
IAC<-IAcontrastsCMAT(CMAT1=CMD1, CMAT2=CMD2)
c2compnames(IAC, ntype="sequ")

---

Lepi Insect counts of 12 Species

Description

Simulated data, inspired by a real field investigating the potential impact of genetically modified crop on several insect species belonging to the same order. The trial was designed as a randomized complete block design with 8 blocks (Block), and a total of 24 plots. In each block, three treatments (Treatment) were randomized: a conventional variety treated with insecticides (Ins), a genetically modified variety (GM) without insecticide treatment, and the near-isogenic variety (Iso)
the to genetically modified variety, without insecticide treatment. Individuals were counted (after classification to the species level) in two different dates in each year of the trial, where the the second date was of higher importance for assessment of impacts of GM variety on non-target species. In total 12 Species were observed during the trial.

Usage

data(Lepi)

Format

A data frame with 144 observations on the following 17 variables.

Year  a numeric vector, the year 1, 2, 3
Date  a numeric vector, 1 and 2 separating the 2 sampling date in each year
Block a numeric vector, with values 1-8, indicator variable for the 8 blocks
Treatment a factor with three levels identifying the varieties: gm is the genetically modified variety, ins the conventional variety with insecticide treatment and iso the near isogenic line without insecticide treatment
Plot  a factor with 24 levels, identifying the individual plots
Sp1  counts of taxon 1
Sp2  counts of taxon 2
Sp3  counts of taxon 3
Sp4  counts of taxon 4
Sp5  counts of taxon 5
Sp6  counts of taxon 6
Sp7  counts of taxon 7
Sp8  counts of taxon 8
Sp9  counts of taxon 9
Sp10 counts of taxon 10
Sp11 counts of taxon 11
Sp12 counts of taxon 12

Source

Simulated data.

Examples

data(Lepi)
str(Lepi)
summary(Lepi)
Simulated data set for a simple mixed model

Description

Simulated data set for a simple mixed model

Usage

data(MM1)

Format

A data frame with 160 observations on the following 3 variables.

Y a numeric vector, the response, sampled from a normal distribution
F a factor with levels F1 F2 F3 F4, representing fixed effects
R a factor with levels R1 R2 R3 R4 R5, representing random effects, sampled from a normal distribution
Examples

    data(MMPois)
    boxplot(Y+R+F, data=MMPois)

Description

A fixed factor F (four levels) and a random factor (five levels), modifying the mean response (random intercept) Y is a variable following a Poisson distribution.

Usage

    data(MMPois)

Format

A data frame with 160 observations on the following 3 variables.

Y  a numeric vector, the Poisson distributed response.
F  a factor with levels F1 F2 F3 F4
R  a factor with levels R1 R2 R3 R4 R5

Source

simulation

Examples

    data(MMPois)
    boxplot(Y+R+F, data=MMPois, las=2)
Simulated data for a simple mixed model with Poisson response

Description

Simulated data with a fixed factor cult (4 levels), with 8 randomized replications each, a (fixed) factor time (6 levels), which are repeated measurements taken from the same experimental units. The 32 experimental (plotid) units differ in their mean response following a gaussian distribution. The response Y follows a Poisson distribution.

Usage

```r
data(MMPoisRep)
```

Format

A data frame with 192 observations on the following 4 variables.

- `plotid` a factor with 32 levels, representing the 32 experimental units (plots)
- `cult` a factor with 4 levels (`c1 cR cS cT`), representing a fixed factor (e.g. the cultivar)
- `time` a factor with 6 levels (`t1 tR tS tT tU tV`) specifying repeated measurements on the same experimental units (plotid) over time
- `Y` a numeric vector, following a Poisson distribution

Source

simulation

Examples

```r
data(MMPoisRep)

boxplot(Y ~ cult*time, data=MMPoisRep, las=TRUE)
```

Nematocera

Trap counts of Nematocera

Description

In a field trial, 4 treatments (A,B, Standard, Novum) were arranged in a randomized complete block design with 8 blocks and 32 plots. In summer 2005 soil eklektor traps were placed in each plot, on six dates from 2005-07-12 to 2005-09-25, the number of individuals of Nematocera (gnats, midges and others) hatching from soil were counted. The individuals were classified to the family level. Interest was in assessing potential effects of a novel agricultural practice (Novum) on the abundance of Nematocera.
Usage

data(Nematocera)

Format

A data frame with 192 observations on the following 14 variables.

Date  a POSIXt, the time of counting the individuals in the eklektor trap

Treatment  a factor with 4 levels, A, B, Standard and Novum, where Novum is the novel treatment, Standard is the standard treatment most similar to Novum, and A and B are additional standard treatments.

Block  a numeric vector, specifying the blocks 1-8

Plot  a factor with 32 levels A1 to Standard8, indicator variables for the individual eklektors

Bibio  a numeric vector, counts of individuals, belonging to the family

Cecido  a numeric vector, counts of individuals, belonging to the family

Cerato  a numeric vector, counts of individuals, belonging to the family

Chiro  a numeric vector, counts of individuals, belonging to the family

Myceto  a numeric vector, counts of individuals, belonging to the family

Psycho  a numeric vector, counts of individuals, belonging to the family

Scato  a numeric vector, counts of individuals, belonging to the family

Sciari  a numeric vector, counts of individuals, belonging to the family

Tipuli  a numeric vector, counts of individuals, belonging to the family

Total  a numeric vector, total count of individuals belonging to the suborder Nematocera

Source

personal communications, S.Prescher, JKI Braunschweig, Germany

Examples

data(Nematocera)

par(mar=c(11,5,3,1))

boxplot(Total ~ Treatment*Date, data=Nematocera, las=2, col=c("white","white","blue","green"))
legend(x=15, y=100, legend=levels(Nematocera$Treatment), fill=c("white","white","blue","green"))

pairs(Nematocera[,c("Cecido","Cerato","Chiro","Myceto","Psycho","Sciari")])
**plotCI.Methods**

---

**Plot confidence intervals calculated by pairwiseCI**

---

**Description**

Plot confidence intervals calculated by calling pairwiseCI or UnlogCI.

**Usage**

```r
## S3 method for class 'UnlogCI'
plotCI(x, ...)
## S3 method for class 'simplesimint'
plotCI(x, ...)
```

**Arguments**

- `x` an object of class "UnlogCI" "simplesimint" as can be obtained by calling `UnlogCI` or `simplesimint`
- `...` further arguments to be passed to plotCI, see ?plotCI in package MCPAN for details

**Value**

A plot.

---

**SCSnp**

---

**Simultaneous confidence sets from empirical joint distribution.**

---

**Description**

Calculate simultaneous confidence sets according to Besag et al. (1995) from a empirical joint distribution of a parameter vector. Joint empirical distributions might be obtained from WinBUGS or OpenBUGS calls.

**Usage**

```r
## Default S3 method:
SCSnp(x, conf.level = 0.95,
       alternative = "two.sided", ...)

## S3 method for class 'bugs'
SCSnp(x, conf.level = 0.95,
       alternative = "two.sided", whichp = NULL, ...)
```
## S3 method for class 'CCRatio'
SCSnp(x, ...)

## S3 method for class 'CCDiff'
SCSnp(x, ...)

### Arguments
- **x**: a matrix N-times-P matrix or an object of class CCRatio or CCDiff
- **conf.level**: a single numeric value between 0.5 and 1, the simultaneous confidence level
- **alternative**: a single character string, one of "two.sided", "less", "greater", for two-sided, upper and lower limits
- **whichp**: a single character string, naming an element of the sims.list if x is a bugs object, ignored otherwise
- **...**: further arguments, currently not used

### Details
Let P be the number of parameters in the parameter vector and N be the total number of values obtained for the empirical joint distribution of the parameter vector, e.g. as can be obtained e.g., from Gibbs sampling.

### Value
An object of class "SCSnp", a list with elements

- **conf.int**: a P-times-2 matrix containing the lower and upper confidence limits
- **estimate**: a numeric vector of length P, containing the medians of the P marginal empirical distributions
- **x**: the input object
- **k**: the number of values outside the SCS, i.e. conf.level*N
- **N**: the number of values used to construct the confidence set
- **conf.level**: a single numeric value, the nominal confidence level, as input
- **alternative**: a single character string, as input

### Author(s)
Frank Schaarschmidt, adapting an earliere version of Gemechis D. Djira

### References

### See Also
- CInp for a wrapper to quantile to compute simple percentile intervals on each of P marginal distributions
Examples

# Assume a 1000 times 4 matrix of 4 mutually independent normal variables:
X<-cbind(rnorm(1000), rnorm(1000), rnorm(1000), rnorm(1000))

SCSts<-SCSnp(x=X, conf.level=0.9, alternative="two.sided")
SCSts

SCS<-SCSts$conf.int

in1<-X[,1]<-SCS[1,1] & X[,1]<-SCS[1,2]

sum(in1*in2*in3*in4)

simplesimint Simultaneous confidence intervals from raw estimates

Description

Calculates simultaneous confidence intervals for multiple contrasts based on a parameter vector, its variance-covariance matrix and (optionally) the degrees of freedom, using quantiles of the multivar

Usage

simplesimint(coef, vcov, cmat, df = NULL, conf.level = 0.95, alternative = c("two.sided", "less", "greater"))

Arguments

coeff a single numeric vector, specifying the point estimates of the parameters of interest
vcov the variance-covariance matrix corresponding to coef, should be of dimension P-times-P, when coef is of length P
cmat the contrasts matrix specifying the comparisons of interest with respect to coef, should have P columns, when coef is of length P
df

optional, the degree of freedom for the multivariate t-distribution; if specified, quantiles from the multivariate t-distribution are used for confidence interval estimation, if not specified (default), quantiles of the multivariate normal distribution are used.

conf.level

a single numeric value between 0.5 and 1.0; the simultaneous confidence level

alternative

a single character string, "two.sided" for intervals, "less" for upper limits, and "greater" for lower limits

Details

Implements the methods formerly available in package `multcomp`, function `csimint`. Input values are a vector of parameter estimates \( \mu \) of length \( P \), a corresponding estimate for its variance-covariance matrix \( \Sigma \) (P times P), and a contrast matrix \( C \) of dimension \( M \times P \). The contrasts \( L = C\mu \) are computed, the variance-covariance matrix (being a function of \( C \) and \( \Sigma \)) and the corresponding correlation matrix \( R \) are computed. Finally, confidence intervals for \( L \) are computed: if \( df \) is given, quantiles of an M-dimensional t distribution with correlation matrix \( R \) are used, otherwise quantiles of an M-dimensional standard normal distribution with correlation matrix \( R \) are used.

Value

An object of class "simplesimint"

- estimate: the estimates of the contrasts
- lower: the lower confidence limits
- upper: the upper confidence limits
- cmat: the contrast matrix, as input
- alternative: a character string, as input
- conf.level: a numeric value, as input
- quantile: a numeric value, the quantile used for confidence interval estimation
- df: a numeric value or NULL, as input
- stderr: the standard error of the contrasts
- vcovC: the variance covariance matrix of the contrasts

Note

This is a testversion and has not been checked extensively.

Author(s)

Frank Schaarschmidt

See Also

See `?coef` and `?vcov` for extracting of parameter vectors and corresponding variance covariance matrices from various model fits.
Examples

# For the simple case of Gaussian response
# variables with homoscedastic variance,
# see the following example

library(mratios)
data(angina)

boxplot(response ~ dose, data=angina)

# Fit a cell means model,
fit<-lm(response ~ 0+dose, data=angina)

# extract cell means, the corresponding
# variance-covariance matrix and the
# residual degree of freedom,
cofi<-coef(fit)
vcofi<-vcov(fit)
dofi<-fit$df.residual

# define an appropriate contrast matrix,
# here, comparisons to control
n<-unlist(lapply(split(angina$response, f=angina$dose), length))
names(n)<-names(cofi)
cmat<-contrMat(n=n, type="Dunnett")
cmat

#
test<-simplesimint(coef=cofi, vcov=vcofi, df=dofi, cmat=cmat, alternative="greater")
test

summary(test)

plotCI(test)

### Note, that the same result can be achieved much more conveniently
### using confint.glht in package multcomp
Description

Produce a detailed print out for objects of class "simplesimint".

Usage

```r
## S3 method for class 'simplesimint'
summary(object, ...)
```

Arguments

- `object`: an object of class "simplesimint" as can be obtained by calling `simplesimint`
- `...`: further arguments to be passed to `print`

Value

Print

Examples

```r
library(multcomp)
data(waste)

anova(lm(waste ~ temp*envir, data=waste))

# a cell-means model instead
# of a two-way-layout

fit <- lm(waste ~ 0 + temp:envir, data=waste)

COEF <- coef(fit)
VCOV <- vcov(fit)

DF <- fit$df.residual

IAC <- IAcontrasts(type=c("Identity", "Dunnett"), k=c(5,3))
IACn <- c2comppnames(IAC)

SCI <- simplesimint(coef=COEF, vcov=VCOV, df=DF, cmat=IACn)

# the print function just gives this:

SCI

# the summary function produces some more output:

summary(SCI)


```

# long names currently give nasty outputs!

colnames(IAC)<-names(COEF)

IACn <- c2compnames(IAC)

SCI<- simplesimint(coef=COEF, vcov=VCOV, df=DF, cmat=IACn)

SCI

# the summary function produces some more output:

summary(SCI)

---

UnlogCI  

**Transform confidence intervals from glm fits.**

---

**Description**

Transform confidence intervals derived from glm fits back to original scale and give appropriate names.

**Usage**

```r
## S3 method for class 'glht'
UnlogCI(x)
```

**Arguments**

- `x` an object of class "confint.glht"

**Details**

Applies exponential function on the estimates and confidence limits and creates useful names for the comparisons and parameters.

**Value**

An object of class "UnlogCI".

**See Also**

`plotCI.UnlogCI` for plotting the result
Examples

```r
# # # CI for odds ratios
# # # for models on the logit-link

data(Feeding)

# Larval mortality:
Feeding$Lmortality <- Feeding$Total - Feeding$Pupating

fit1 <- glm(cbind(Pupating, Lmortality) ~ Variety, data = Feeding, family = quasibinomial)
anova(fit1, test = "F")

library(multcomp)

comp <- glht(fit1, mcp(Variety = "Tukey"))

CIraw <- CIGLM(comp, method = "Raw")

CIraw

UnlogCI(CIraw)

plotCI(UnlogCI(CIraw), lines = c(0.25, 0.5, 2, 4),
       lineslwd = c(1, 2, 2, 1), linescol = c("red", "black", "black", "red"))

# # # # # #

# # # CI for ratios of means
# # # for models on the log-link

data(Diptera)

# Larval mortality:

fit2 <- glm(Ges ~ Treatment, data = Diptera, family = quasipoisson)
anova(fit2, test = "F")

library(multcomp)

comp <- glht(fit2, mcp(Treatment = "Tukey"))

CIadj <- CIGLM(comp, method = "Adj")

CIadj

UnlogCI(CIadj)

plotCI(UnlogCI(CIadj), lines = c(0.5, 1, 2), lineslwd = c(2, 1, 1))
```
vcov.gamlss

Extract variance covariance matrix from objects of class gamlss

Description

Only for internal use. Extract the covariance matrix corresponding to the mu parameters of a gamlss-fit.

Usage

```r
## S3 method for class 'gamlss'
vcov(object, ...)
```

Arguments

- `object`: An object of class "gamlss" as can be created by calling `gamlss`
- `...`: Currently not used.

Details

Only for internal use. Needs implementation of warnings.

Value

A matrix of dimension mxm, if m is the length of mu-parameters from a gamlss fit.

Author(s)

Daniel Gerhard

See Also

packages gamlss and multcomp
vcov.geeglm

Extract variance covariance matrix from objects of class geeglm

Description

Only for internal use with function glht: Extract the variance covariance matrix corresponding to
the mu parameters of a gamlss-fit. Merely a wrapper of the method internally used in function
summary.geeglm, package geepack.

Usage

## S3 method for class 'geeglm'
vcov(object, ...)

Arguments

object An object of class "geeglm" as can be created by calling geeglm in package
geepack.

... Currently not used.

Details

Test version. Only for internal use. Needs implementation of warnings.

Value

A matrix of dimension m times m, if m is the length of coefficients from a geeglm fit.

See Also

packages gamlss and multcomp
Index

*Topic datasets
  Brachycera, 4
  Cica1, 7
  Cica2, 8
  CountRep, 13
  Decompo, 13
  Diptera, 15
  ExNB Cov, 16
  ExPCov, 17
  fakeln, 18
  Feeding, 18
  Lepi, 21
  MM1, 23
  MMPois, 24
  MMPoisRep, 25
  Nematocera, 25
*Topic hplot
  plotCI.Methods, 27
*Topic htest
  BOOTSimpson, 3
  CIGLM, 10
  Cinp, 11
  IA contrasts, 19
  IA contrastsCMAT, 20
  SCSn, 27
  simplesimint, 29
  UnlogCI, 33
  vcov.gam lss, 35
  vcov.gee glml, 36
*Topic misc
  allignment, 2
  c2compnames, 6
*Topic print
  summary.simplesimint, 31
allignment, 2

BOOTSimpson, 3
BOOTSimpsonD (BOOTSimpson), 3
BOOTSimpsonR (BOOTSimpson), 3

Brachycera, 4
  c2compnames, 6, 20
  CCDiff, 12
  CCDiff.boot, 4
  CCDiff.default, 4
  CCRatio, 12
  CCRatio.boot, 4
  CCRatio.default, 4
  Cica1, 7, 9
  Cica2, 8
  CIGLM, 10
  Cinp, 4, 11, 28
  CountRep, 13
  Decompo, 13
  Diptera, 15
  ExNB Cov, 16
  ExPCov, 17
  fakeln, 18
  Feeding, 18
  IA contrasts, 19, 20
  IA contrastsCMAT, 20, 20
  Lepi, 21
  MM1, 23
  MMPois, 24
  MMPoisRep, 25
  Nematocera, 25

plotCI.Methods, 27
  plotCI.simplesimint (plotCI.Methods), 27
  plotCI.UnlogCI, 33
  plotCI.UnlogCI (plotCI.Methods), 27
  quantile, 12
SCSnp, 4, J2, 27
simplesimint, 27, 29, 32
summary.simplesimint, 31

UnlogCI, J0, 27, 33

vcov.gamlss, 35
vcov.geeglm, 36