Package ‘pbkrtest’

February 20, 2015

Version 0.4-2
Title Parametric bootstrap and Kenward-Roger-based methods for mixed model comparison
Author Ulrich Halekoh <uhalekoh@health.sdu.dk> Søren Højsgaard <sorenh@math.aau.dk>
Maintainer Søren Højsgaard <sorenh@math.aau.dk>
Description Test in linear mixed effects models.
Attention is on linear mixed effects models as implemented in the lme4 package.
The package implements a parametric bootstrap test.
The package implements a Kenward-Roger modification of F-tests.
URL http://people.math.aau.dk/~sorenh/software/pbkrtest/
Depends R (>= 3.0.0), lme4
Imports Matrix, parallel, MASS
Suggests gplots
Encoding latin1
ZipData no
License GPL (>= 2)
NeedsCompilation no
Repository CRAN
Date/Publication 2014-11-13 14:43:54

R topics documented:

beets ................................................................. 2
budworm ............................................................. 3
getKR ................................................................. 4
get_Lb_ddf ......................................................... 5
KenwardRoger ...................................................... 6
model2restrictionMatrix .......................................... 8
PBmodcomp ........................................................ 9
PBrefdist .......................................................... 12
vcovAdj ............................................................ 13
Yield and sugar percentage in sugar beets from a split plot experiment.

Description

Data is obtained from a split plot experiment. There are 3 blocks and in each of these the harvest time defines the "whole plot" and the sowing time defines the "split plot". Each plot was $25m^2$ and the yield is recorded in kg. See 'details' for the experimental layout.

Usage

data(beets)

Format

The format is: chr "beets"

Details

Experimental plan
Sowing times
1  4. april
2  12. april
3  21. april
4  29. april
5  18. may
Harvest times
1  2. october
2  21. october

Plot allocation:

| Plot 1 | 1 1 1 1 1 | 2 2 2 2 2 | 1 1 1 1 1 | Harvest time
|--------|-----------|-----------|-----------|
| 1-15   | 3 4 5 2 1 | 3 2 4 5 1 | 5 2 3 4 1 | Sowing time
|--------|-----------|-----------|-----------|
| Plot 2 | 2 2 2 2 2 | 1 1 1 1 1 | 2 2 2 2 2 | Harvest time
| 16-30  | 2 1 5 4 3 | 4 1 3 2 5 | 1 4 3 2 5 | Sowing time

Examples

```r
data(beets)
## maybe str(beets) ; plot(beets) ...

beets$bh <- with(beets, interaction(block, harvest))
summary(aov(yield+block+sow+harvest+Error(bh), beets))
summary(aov(sugpct-block+sow+harvest+Error(bh), beets))
```
Effect of Insecticide on survival of tobacco budworms

**Description**
number of killed budworms exposed to an insecticide

**Usage**
data(budworm)

**Format**
This data frame contains 12 rows and 4 columns:
- **sex**: sex of the budworm
- **dose**: dose of the insecticide trans-cypermethrin in [µg]
- **ndead**: budworms killed in a trial
- **ntotal**: total number of budworms exposed per trial

**Details**
mortality of the moth tobacco budworm 'Heliothis virescens' for 6 doses of the pyrethroid trans-cypermethrin differentiated with respect to sex

**Source**

**References**

**Examples**
data(budworm)

#function to calculate the empirical logits
empirical.logit<- function(nevent,ntotal) {
y<-log ((nevent+0.5)/(ntotal-nevent+0.5))
y}

#plot the empirical logits against log-dose
log.dose<-log(budworm$dose)
emp.logit<-empirical.logit(budworm$ndeal,budworm$ntotal)
plot(log.dose,emp.logit,type='n',xlab='log-dose',ylab='empirical logit')
title("budworm: empirical logits of probability to die")

male<-budworm$sex=='male'
female<-budworm$sex=='female'
lines(log.dose[male],emp.logit[male],type='b',lty=1,col=1)
lines(log.dose[female],emp.logit[female],type='b',lty=2,col=2)
legend(0.5,2,legend=c('male','female'),lty=c(1,2),col=c(1,2))

## Not run:
* SAS example;
data budworm;
infile 'budworm.txt' firstobs=2;
input sex dose ndead ntotal;
run;
## End(Not run)

---

**getKR**

*Extract (or "get") components from a KRmodcomp object.*

**Description**

Extract (or "get") components from a KRmodcomp object, which is the result of the KRmodcomp function.

**Usage**

```r
```

**Arguments**

- **object**
  A KRmodcomp object, which is the result of the KRmodcomp function
- **name**
  The available slots. If name is missing or NULL then everything is returned.

**Author(s)**

Soren Hojsgaard <sorenh@math.aau.dk>

**References**

See Also

KRmodcomp PBmodcomp vcovAdj

Examples

data(beets, package='pbkrtest')
lg <- lmer(sugpct ~ block + sow + harvest + (1|block:harvest),
          data=beets, REML=FALSE)
sm <- update(lg, . - harvest)
xx<-KRmodcomp(lg, sm)
getKR(xx, "ddf") # get denominator degrees of freedom.

---

get_Lb_ddf

Adjusted denominator degrees of freedom for linear estimate for linear mixed model.

Description

Get adjusted denominator degrees of freedom for testing \( Lb=0 \) in a linear mixed model where \( L \) is a restriction matrix.

Usage

get_Lb_ddf(object, L)
Lb_ddf(L, V0, Vadj)

Arguments

- **object**: A linear mixed model object.
- **L**: A vector with the same length as `fixef(object)` or a matrix with the same number of columns as the length of `fixef(object)`.
- **V0, Vadj**: Unadjusted and adjusted covariance matrix for the fixed effects parameters. Unadjusted covariance matrix is obtained with `vcov()` and adjusted with `vcovAdj()`.

Value

Adjusted degrees of freedom (adjustment made by a Kenward-Roger approximation).

Author(s)

Soren Hojsgaard, <sorenh@math.aau.dk>

References

See Also

KRmodcomp, vcovAdj, model2restrictionMatrix, restrictionMatrix2model

Examples

(fmLarge <- lmer(Reaction ~ Days + (Days|Subject), sleepstudy))
## removing Days
(fmSmall <- lmer(Reaction ~ 1 + (Days|Subject), sleepstudy))
anova(fmLarge, fmSmall)

KRmodcomp(fmLarge, fmSmall) ## 17 denominator df's
get_lb_ddf(fmLarge, c(0,1)) ## 17 denominator df's

# Notice: The restriction matrix L corresponding to the test above
# can be found with
L <- model2restrictionMatrix(fmLarge, fmSmall)
L

KenwardRoger

F test and degrees of freedom based on Kenward-Roger approximation

Description

An approximate F-test based on the Kenward-Roger approach.

Usage

KRmodcomp(largeModel, smallModel, betaH=0, details=0)
## S3 method for class 'mer'
KRmodcomp(largeModel, smallModel, betaH=0, details=0)
## S3 method for class 'lmerMod'
KRmodcomp(largeModel, smallModel, betaH=0, details=0)

Arguments

largeModel An lmer model
smallModel An lmer model or a restriction matrix
betaH A number or a vector of the beta of the hypothesis, e.g. L beta=L betaH. betaH=0 if modelSmall is a model not a restriction matrix.
details If larger than 0 some timing details are printed.
... Additional arguments to print function
Details

An F test is calculated according to the approach of Kenward and Roger (1997). The function works for linear mixed models fitted with the `lmer` function of the `lme4` package. Only models where the covariance structure is a sum of known matrices can be compared.

The `largeModel` may be a model fitted with `lmer` either using `REML=TRUE` or `REML=FALSE`. The `smallModel` can be a model fitted with `lmer`. It must have the same covariance structure as `largeModel`. Furthermore, its linear space of expectation must be a subspace of the space for `largeModel`. The model `smallModel` can also be a restriction matrix L specifying the hypothesis $L\beta = L\beta_H$, where $L$ is a $k \times p$ matrix and $\beta$ is a $p$ column vector the same length as `fixef(largeModel)`.

The $\beta_H$ is a $p$ column vector.

Notice: if you want to test a hypothesis $L\beta = c$ with a $k$ vector $c$, a suitable $\beta_H$ is obtained via $\beta_H = Lc$ where $L_n$ is a g-inverse of $L$.

Notice: It cannot be guaranteed that the results agree with other implementations of the Kenward-Roger approach!

Note

This functionality is not thoroughly tested and should be used with care. Please do report bugs etc.

Author(s)

Ulrich Halekoh <ulrich.halekoh@agrsci.dk>, Sören Højsgaard <sorenh@math.aau.dk>

References


See Also

`getKR lmer vcovAdj PBmodcomp`

Examples

```r
(fmlarge <- lmer(Reaction ~ Days + (Days|Subject), sleepstudy))
## removing Days
(fmsmall <- lmer(Reaction ~ 1 + (Days|Subject), sleepstudy))
anova(fmlarge,fmsmall)
KRmodcomp(fmlarge,fmsmall)

## The same test using a restriction matrix
L<-cbind(0,1)
KRmodcomp(fmlarge, L)
```
## Same example, but with independent intercept and slope effects:

```r
m.large <- lmer(Reaction ~ Days + (1|Subject) + (0+Days|Subject), data = sleepstudy)
m.small <- lmer(Reaction ~ 1 + (1|Subject) + (0+Days|Subject), data = sleepstudy)
anova(m.large, m.small)
KRmodcomp(m.large, m.small)
```

---

### model2restrictionMatrix

*Conversion between a model object and a restriction matrix*

#### Description

Testing a small model under a large model corresponds imposing restrictions on the model matrix of the larger model and these restrictions come in the form of a restriction matrix. These functions converts a model to a restriction matrix and vice versa.

#### Usage

- `model2restrictionMatrix(largeModel, smallModel)`
- `restrictionMatrix2model(largeModel, LL)`

#### Arguments

- `largeModel, smallModel`  
  Model objects of the same "type". Possible types are linear mixed effects models and linear models (including generalized linear models)
- `LL`  
  A restriction matrix.

#### Value

- `model2restrictionMatrix`: A restriction matrix.
- `restrictionMatrix2model`: A model object.

#### Note

That these functions are visible is a recent addition; minor changes may occur.

#### Author(s)

Ulrich Halekoh <uhalekoh@health.sdu.dk>, Søren Højsgaard <sorenh@math.aau.dk>

#### References

PBmodcomp

Model comparison using parametric bootstrap methods.

Description

Model comparison of nested models using parametric bootstrap methods. Implemented for some commonly applied model types.

Usage

PBmodcomp(largeModel, smallModel, nsim = 1000, ref = NULL, seed=NULL, cl = NULL, details = 0)

Arguments

largeModel  A model object. Can be a linear mixed effects model or generalized linear mixed effects model (as fitted with lmer() and glmer() function in the lme4 package) or a linear normal model or a generalized linear model. The largeModel must be larger than smallModel (see below).

smallModel  A model of the same type as largeModel or a restriction matrix.
### Arguments

- **nsim**
  - The number of simulations to form the reference distribution.

- **ref**
  - Vector containing samples from the reference distribution. If NULL, this vector will be generated using `PBrefdist()`.

- **seed**
  - A seed that will be passed to the simulation of new datasets.

- **cl**
  - A vector identifying a cluster; used for calculating the reference distribution using several cores. See examples below.

- **details**
  - The amount of output produced. Mainly relevant for debugging purposes.

### Details

Under the fitted hypothesis (i.e. under the fitted small model) nsim samples of the likelihood ratio test statistic (LRT) are generated.

Then p-values are calculated as follows:

- **LRT**: Assuming that LRT has a chi-square distribution.
- **PBtest**: The fraction of simulated LRT-values that are larger or equal to the observed LRT value.
- **Bartlett**: A Bartlett correction is calculated from the mean of the simulated LRT-values.
- **Gamma**: The reference distribution of LRT is assumed to be a gamma distribution with mean and variance determined as the sample mean and sample variance of the simulated LRT-values.

### Author(s)

Soren Hojsgaard <sorenh@math.aau.dk>

### References


### See Also

- `KRmodcomp`
- `PBrefdist`

### Examples

```r
data(beets, package="pbkrtest")
head(beets)

## Linear mixed effects model:
sug <- lmer(sugpct ~ block + sow + harvest + (1|block:harvest), data=beets, REML=FALSE)
sug.h <- update(sug, .~ . -harvest)
sug.s <- update(sug, .~ . -sow)
anova(sug, sug.h)
PBmodcomp(sug, sug.h, nsim=50)
anova(sug, sug.h)
Pbmodcomp(sug, sug.s, nsim=50)
```


## Linear normal model:
sug <- lm(sugpct ~ block + sow + harvest, data=beets)
sug.h <- update(sug, .~ -harvest)
sug.s <- update(sug, .~ -sow)

anova(sug, sug.h)
PBmodcomp(sug, sug.h, nsim=50)
anova(sug, sug.s)
PBmodcomp(sug, sug.s, nsim=50)

## Generalized linear model
counts <- c(18,17,15,20,10,25,13,12)
outcome <- gl(3,1,9)
treatment <- gl(3,3)
d.AD <- data.frame(treatment, outcome, counts)
head(d.AD)
glm.D93 <- glm(counts ~ outcome + treatment, family = poisson())
glm.D93.o <- update(glm.D93, .~ -outcome)
glm.D93.t <- update(glm.D93, .~ -treatment)

anova(glm.D93, glm.D93.o, test="Chisq")
PBmodcomp(glm.D93, glm.D93.o, nsim=50)
anova(glm.D93, glm.D93.t, test="Chisq")
PBmodcomp(glm.D93, glm.D93.t, nsim=50)

## Generalized linear mixed model (it takes a while to fit these)
## Not run:
(gm1 <- glmer(cbind(incidence, size - incidence) ~ period + (1 | herd),
           data = cbpp, family = binomial))
(gm2 <- update(gm1, .~.-period))
anova(gm1, gm2)
PBmodcomp(gm1, gm2)

## Not run:
(fmlarge <- lmer(Reaction ~ Days + (Days|Subject), sleepstudy))
## removing Days
(fmsmall <- lmer(Reaction ~ 1 + (Days|Subject), sleepstudy))
anova(fmlarge, fmsmall)
PBmodcomp(fmlarge, fmsmall)

## The same test using a restriction matrix
L <- cbind(0,1)
PBmodcomp(fmlarge, L)

## Vanilla
PBmodcomp(beet0, beet_no_harv, nsim=1000)

## Simulate reference distribution separately:
refdist <- PBrefdist(beet0, beet_no_harv, nsim=1000)
PBmodcomp(beet0, beet_no_harv, ref=refdist)
## PBrefdist

Calculate reference distribution using parametric bootstrap

### Description

Calculate reference distribution of likelihood ratio statistic in mixed effects models using parametric bootstrap

### Usage

```r
PBrefdist(largeModel, smallModel, nsim = 1000, seed=NULL, cl = NULL, details = 0)
```

### Arguments

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>largeModel</code></td>
<td>A linear mixed effects model as fitted with the <code>lmer()</code> function in the <code>lme4</code> package. This model must be larger than <code>smallModel</code> (see below).</td>
</tr>
<tr>
<td><code>smallModel</code></td>
<td>A linear mixed effects model as fitted with the <code>lmer()</code> function in the <code>lme4</code> package. This model must be smaller than <code>largeModel</code> (see above).</td>
</tr>
<tr>
<td><code>nsim</code></td>
<td>The number of simulations to form the reference distribution.</td>
</tr>
<tr>
<td><code>seed</code></td>
<td>Seed for the random number generation.</td>
</tr>
<tr>
<td><code>cl</code></td>
<td>A vector identifying a cluster; used for calculating the reference distribution using several cores. See examples below.</td>
</tr>
<tr>
<td><code>details</code></td>
<td>The amount of output produced. Mainly relevant for debugging purposes.</td>
</tr>
</tbody>
</table>
vcovAdj

Value
A numeric vector

Author(s)
Søren Hojsgaard <sorenh@math.aau.dk>

References

See Also
 PBmodcomp, KRmodcomp

Examples

data(beets)
head(beets)
beet0 <- lmer(sugpct ~ block + sow + harvest + 1 | block:harvest), data = beets, REML = FALSE)
beet_no.harv <- update(beet0, . ~ . - harvest)
rr <- PBrefdist(beet0, beet_no.harv, nsim = 20)
rr

## Note clearly many more than 10 simulations must be made in practice.

## Computations can be made in parallel using several processors:
## Not run:
cl <- makeSOCKcluster(rep("localhost", 4))
clusterEvalQ(cl, library(lme4))
clusterSetupPRNG(cl)
rr <- PBrefdist(beet0, beet_no.harv, nsim = 20)
stopCluster(cl)

## End(Not run)
## Above, 4 cpu's are used and 5 simulations are made on each cpu.

---

vcovAdj  

Adjusted covariance matrix for linear mixed models according to Kenward and Roger

Description

Kenward and Roger (1997) describbe an improved small sample approximation to the covariance matrix estimate of the fixed parameters in a linear mixed model.
Usage

vcovAdj(object, details=0)
LMM_Sigma_G(object, details=0)

Arguments

object An lmer model
details If larger than 0 some timing details are printed.

Value

phiA the estimated covariance matrix, this has attributed P, a list of matrices used in KR_adjust and the estimated matrix W of the variances of the covariance parameters of the random effects
SigmaG list: Sigma the covariance matrix of Y; G the G matrices that sum up to Sigma; n.ggamma: the number (called M in the article) of G matrices

Note

This functionality is not thoroughly tested and should be used with care. Please do report bugs etc.

Author(s)

Ulrich Halekoh <uhalekoh@health.sdu.dk>, Soren Hojsgaard <sorenh@math.aau.dk>

References


See Also

getKR KRmodcomp lmer PBmodcomp vcovAdj

Examples

(fmLarge <- lmer(Reaction ~ Days + (Days|Subject), sleepstudy))
## removing Day
(vcovAdj(fmLarge, detail=0))
Index

*Topic datasets
  beets, 2
  budworm, 3
*Topic function
  KenwardRoger, 6
  vcovAdj, 13
*Topic inference
  PBmodcomp, 9
*Topic models
  getKR, 4
  model2restrictionMatrix, 8
  PBmodcomp, 9
  PBrefdist, 12
*Topic utilities
  get_lb_ddf, 5
  model2restrictionMatrix, 8
  PBmodcomp, 9
  PBrefdist, 12

beets, 2
budworm, 3

get_lb_ddf, 5
get_SigmaG (vcovAdj), 13
getKR, 4, 7, 14
getLRT (PBmodcomp), 9

KenwardRoger, 6
KRmodcomp, 5, 6, 9, 10, 13, 14
KRmodcomp (KenwardRoger), 6
KRmodcomp_internal (KenwardRoger), 6

Lb_ddf (get_lb_ddf), 5
lmer, 7, 14
LMM_Sigma_G (vcovAdj), 13

model2restrictionMatrix, 6, 8

PBmodcomp, 5, 7, 9, 13, 14
PBrefdist, 9, 10, 12
plot.XXmodcomp (PBmodcomp), 9

restrictionMatrix2model, 6
restrictionMatrix2model
  (model2restrictionMatrix), 8
vcovAdj, 5–7, 13, 14
vcovAdj0 (vcovAdj), 13
vcovAdj2 (vcovAdj), 13
vcovAdj_internal (vcovAdj), 13