Inference in mixed models in R - beyond the usual asymptotic likelihood ratio test

Søren Højsgaard [1] Ulrich Halekoh [2]

[1] Department of Mathematical Sciences
Aalborg University, Denmark
sorenh@math.aau.dk

[2] Department of Epidemiology, Biostatistics and Biodemography
University of Southern Denmark, Denmark
uhalekoh@health.sdu.dk

November 14, 2016

Contents

- History
  - The degree of freedom police...
  - Motivation: Sugar beets - A split–plot experiment
  - Motivation: A random regression problem
  - Our contribtion
  - Our goal
- The Kenward–Roger approach
  - The Kenward–Roger modification of the *F*–statistic
  - Shortcommings of Kenward-Roger
- Parametric bootstrap
  - Parallel computations
- Small simulation study: A random regression problem
- Final remarks

History

- Years ago, Ulrich Halekoh and SH colleagues at “Danish Institute for Agricultural Sciences”
- That was SAS-country back then
- Many studies called for random effects models - and for `PROC MIXED`
- `PROC MIXED` reports (by default) *p*–values from asymptotic likelihood ratio test.
- Main concern: Effects should be “tested against” the correct variance component in order not to make effects appear more significant than they really are.

History

- Common advice: Use Satterthwaite or Kenward-Roger approximation of denominator degrees of freedom in *F*-test – in an attempt not to get things “too wrong”.
- Then `R` came along; we advocated the use of `R`.
- Random effects models were fitted with the **nlme** package – but there was no Satterthwaite or Kenward-Roger approximation, so our common advice fell apart.

History

The degree of freedom police...

R-help - 2006: `[R] how calculation degrees freedom`

`https://stat.ethz.ch/pipermail/r-help/2006-January/087013.html`

SH: Along similar lines ... probably in recognition of the degree of freedom problem. It could be nice, however, if anova() produced ...

Doug Bates: I don’t think the ”degrees of freedom police” would find that to be a suitable compromise. :-)

In reply to another question:

Doug Bates: I will defer to any of the ”degrees of freedom police” who post to this list to give you an explanation of why there should be different degrees of freedom.

History

Motivation: Sugar beets - A split–plot experiment

- Model how sugar percentage in sugar beets depends on harvest time and sowing time.
- Five sowing times (*s*) and two harvesting times (*h*).
- Experiment was laid out in three blocks (*b*).

```
Experimental plan for sugar beets experiment

Sowing times:
 1: 4/4, 2: 12/4, 3: 21/4, 4: 29/4, 5: 18/5

Harvest times:
 1: 2/10, 2: 21/10

Plot allocation:
      |  Block 1           |  Block 2           |  Block 3           |
      +--------------------|--------------------|--------------------+
Plot  | h1  h1  h1  h1  h1 | h2  h2  h2  h2  h2 | h1  h1  h1  h1  h1 | Harvest time
1-15  | s3  s4  s5  s2  s1 | s3  s2  s4  s5  s1 | s5  s2  s3  s4  s1 | Sowing time
      |--------------------|--------------------|--------------------|
Plot  | h2  h2  h2  h2  h2 | h1  h1  h1  h1  h1 | h2  h2  h2  h2  h2 | Harvest time
16-30 | s2  s1  s5  s4  s3 | s4  s1  s3  s2  s5 | s1  s4  s3  s2  s5 | Sowing time
      +--------------------|--------------------|--------------------+
```

History

Motivation: Sugar beets - A split–plot experiment

```
data(beets, package='pbkrtest')
head(beets)
```

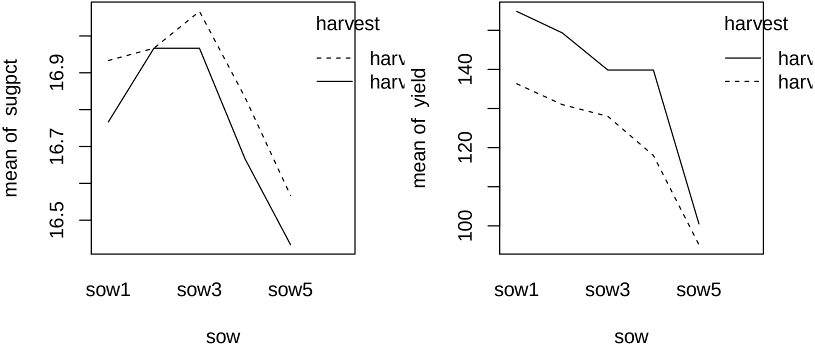
```
##   harvest  block  sow yield sugpct
## 1   harv1 block1 sow3 128.0   17.1
## 2   harv1 block1 sow4 118.0   16.9
## 3   harv1 block1 sow5  95.0   16.6
## 4   harv1 block1 sow2 131.0   17.0
## 5   harv1 block1 sow1 136.5   17.0
## 6   harv2 block2 sow3 136.5   17.0
```

```
library(doBy)
library(lme4)
```

History

Motivation: Sugar beets - A split–plot experiment

```
par(mfrow=c(1,2))
with(beets, interaction.plot(sow, harvest, sugpct))
with(beets, interaction.plot(sow, harvest, yield))
```



History

Motivation: Sugar beets - A split–plot experiment

- For simplicity we assume that there is no interaction between sowing and harvesting times.
- A typical model for such an experiment would be:

$$y_{hbs} = \mu + \alpha_h + \beta_b + \gamma_s + U_{hb} + \epsilon_{hbs}, \tag{1}$$

  where $U_{hb} \sim N(0, \omega^2)$ and $\epsilon_{hbs} \sim N(0, \sigma^2)$.
- Notice that U_{hb} describes the random variation between whole–plots (within blocks).

History

Motivation: Sugar beets - A split–plot experiment

As the design is balanced we may make F–tests for each of the effects as:

```
beets$bh <- with(beets, interaction(block, harvest))
summary(aov(sugpct ~ block + sow + harvest +
                Error(bh), data=beets))
```

```
##
## Error: bh
##           Df Sum Sq Mean Sq F value Pr(>F)
## block      2 0.0327  0.0163    2.58   0.28
## harvest    1 0.0963  0.0963   15.21   0.06
## Residuals  2 0.0127  0.0063
##
## Error: Within
##           Df Sum Sq Mean Sq F value  Pr(>F)
## sow        4   1.01  0.2525     101 5.7e-13
## Residuals 20   0.05  0.0025
```

Notice: the F–statistics are $F_{1,2}$ for harvest time and $F_{4,20}$ for sowing time.

History

Motivation: Sugar beets - A split–plot experiment

Using `lmer()` from **lme4** we can fit the models and test for no effect of sowing and harvest time as follows:

```
beetLarge <- lmer(sugpct ~ block + sow + harvest +
                      (1 | block:harvest), data=beets, REML=FALSE)
beet_no.harv <- update(beetLarge, .~. - harvest)
beet_no.sow  <- update(beetLarge, .~. - sow)
```

History

Motivation: Sugar beets - A split–plot experiment

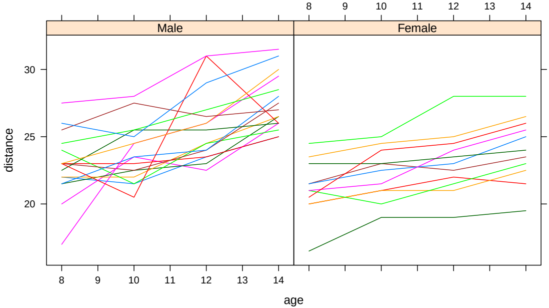
The LRT based *p*–values are anti–conservative: the effect of harvest appears stronger than it is.

```
anova(beetLarge, beet_no.sow)  %>% as.data.frame


##              Df     AIC     BIC logLik deviance Chisq Chi Df Pr(>Chisq)
## beet_no.sow   6  -2.795   5.612  7.398    -14.8    NA     NA         NA
## beetLarge    10 -79.998 -65.986 49.999   -100.0  85.2      4  1.374e-17


anova(beetLarge, beet_no.harv)  %>% as.data.frame


##               Df    AIC    BIC logLik deviance Chisq Chi Df Pr(>Chisq)
## beet_no.harv  9 -69.08 -56.47  43.54   -87.08    NA     NA         NA
## beetLarge    10 -80.00 -65.99  50.00  -100.00 12.91      1  0.0003261
```

History

Motivation: A random regression problem

The change with age of the distance between two cranial distances was observed for 16 boys and 11 girls from age 8 until age 14.



History

Motivation: A random regression problem

Plot suggests:

$$dist_{[i]} = \alpha_{sex[i]} + \beta_{sex[i]} age_{[i]} + A_{Subj[i]} + B_{Subj[i]} age_{[i]} + e_{[i]}$$

with $(A, B) \sim N(0, \boldsymbol{S})$.

ML-test of $\beta_{boy} = \beta_{girl}$:

```
ort1ML<- lmer(distance ~ age + Sex + age:Sex + (1 + age | Subject),
                  REML = FALSE, data=Orthodont)
ort2ML<- update(ort1ML, .~. - age:Sex)
as.data.frame(anova(ort1ML, ort2ML))


##        Df   AIC   BIC logLik deviance Chisq Chi Df Pr(>Chisq)
## ort2ML  7 446.8 465.6 -216.4    432.8    NA     NA         NA
## ort1ML  8 443.8 465.3 -213.9    427.8 5.029      1    0.02492
```

Our goal is to extend the tests provided by `lmer()`.

There are two issues here:

- The choice of test statistic and
- The reference distribution in which the test statistic is evaluated.

Implement Kenward-Roger approximation.

Implement parametric bootstrap.

Implement Satterthwaite approximation (not yet released)

The Kenward–Roger approach

The Kenward–Roger modification of the *F*–statistic

For multivariate normal data

$$Y_{n\times 1} \sim N(\boldsymbol{X}_{n\times p}\boldsymbol{\beta}_{p\times 1}, \boldsymbol{\Sigma})$$

we consider the test of the hypothesis

$$\boldsymbol{L}_{d\times p}\boldsymbol{\beta} = \boldsymbol{\beta}_0$$

where $\boldsymbol{L}$ is a regular matrix of estimable functions of $\boldsymbol{\beta}$.

With $\hat{\boldsymbol{\beta}} \sim N_d(\boldsymbol{\beta}, \boldsymbol{\Phi})$, a Wald statistic for testing $\boldsymbol{L}\boldsymbol{\beta} = \boldsymbol{\beta}_0$ is

$$W = [\boldsymbol{L}(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}_0)]^\top [L\boldsymbol{\Phi}L^\top]^{-1}[\boldsymbol{L}(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}_0)]$$

which is asymptotically $W \sim \chi^2_d$ under the null hypothesis.

The Kenward–Roger approach

The Kenward–Roger modification of the *F*–statistic

A scaled version of W is

$$F = \frac{1}{d} W$$

which is asymptotically $F \sim \frac{1}{d}\chi^2_d$ under the null hypothesis – which we can think of as the limiting distribution of an $F_{d,m}$–distribution as $m \to \infty$

To account for the fact that Φ is estimated from data, we must come up with a better estimate of the denominator degrees of freedom m (better than $m = \infty$).

That was what Kenward and Roger worked on...

The Kenward–Roger approach

The Kenward–Roger modification of the *F*–statistic

The linear hypothesis $\boldsymbol{L}\beta = \beta_0$ can be tested via the Wald-type statistic

$$F = \frac{1}{r}(\hat{\beta} - \beta_0)^\top \boldsymbol{L}^\top (\boldsymbol{L}^\top \boldsymbol{\Phi}(\hat{\boldsymbol{\sigma}})\boldsymbol{L})^{-1} \boldsymbol{L}(\hat{\beta} - \beta_0)$$

- $\boldsymbol{\Phi}(\boldsymbol{\sigma}) = (\boldsymbol{X}^\top \boldsymbol{\Sigma}(\boldsymbol{\sigma})\boldsymbol{X})^{-1} \approx \mathbb{C}\text{ov}(\hat{\beta})$, $\hat{\beta}$ REML estimate of β
- $\hat{\boldsymbol{\sigma}}$: vector of REML estimates of the elements of $\boldsymbol{\Sigma}$

The Kenward–Roger approach

The Kenward–Roger modification of the *F*–statistic

Kenward and Roger (1997) modify the test statistic

- $\mathbf{\Phi}$ is replaced by an improved small sample approximation $\mathbf{\Phi}_A$

Furthermore

- the statistic F is scaled by a factor λ,
- denominator degrees of freedom m are determined

such that the approximate expectation and variance are those of a $F_{d,m}$ distribution.

The Kenward–Roger approach

The Kenward–Roger modification of the *F*–statistic

- Consider only situations where

$$\boldsymbol{\Sigma} = \sum_i \sigma_i \boldsymbol{G}_i, \quad \boldsymbol{G}_i \text{ known matrices}$$

- Variance component and random coefficient models satisfy this restriction.
- $\Phi_A(\hat{\boldsymbol{\sigma}})$ depends now only on the first partial derivatives of $\boldsymbol{\Sigma}^{-1}$:

$$\frac{\partial \boldsymbol{\Sigma}^{-1}}{\partial \sigma_i} = -\boldsymbol{\Sigma}^{-1} \frac{\partial \boldsymbol{\Sigma}}{\partial \sigma_i} \boldsymbol{\Sigma}^{-1}.$$

- $\Phi_A(\hat{\boldsymbol{\sigma}})$ depends also on $\mathbb{V}\mathrm{ar}(\hat{\boldsymbol{\sigma}})$.
- Kenward and Roger propose to estimate $\mathbb{V}\mathrm{ar}(\hat{\boldsymbol{\sigma}})$ via the inverse expected information matrix.

The Kenward–Roger approach

The Kenward–Roger modification of the *F*–statistic

The modification of the F-statistic by Kenward and Roger

- yields the exact F-statistic for balanced mixed classification nested models or balanced split plot models (Alnosaier, 2007).
- Simulation studies (e.g. Spilke, J. et al.(2003)) indicate that the Kenward-Roger approach perform mostly better than alternatives (like Satterthwaite or containment method) for blocked experiments even with missing data.

The Kenward–Roger approach

The Kenward–Roger modification of the *F*–statistic

lme4 (Bates, D., Maechler, M, Bolker, B., Walker, S. 2014) provides efficient estimation of linear mixed models.

lme4 provides most matrices and estimates needed to implement a Kenward-Roger approach.

pbkrtest (Halekoh, U., Højsgaard, S., 2014) provides a “straight forward” transcription of the description in the article of Kenward and Roger, 1997.

The Kenward–Roger approach

The Kenward–Roger modification of the *F*–statistic

The Kenward–Roger approach yields the same results as the anova-test:

```r
beetLarge <- update(beetLarge, REML=TRUE)
beet_no.harv <- update(beet_no.harv, REML=TRUE)
```

Test for harvest effect:

```r
KRmodcomp(beetLarge, beet_no.harv)


## F-test with Kenward-Roger approximation; computing time: 0.06 sec.
## large : sugpct ~ block + sow + harvest + (1 | block:harvest)
## small : sugpct ~ block + sow + (1 | block:harvest)
##       stat  ndf  ddf F.scaling p.value
## Ftest 15.2  1.0  2.0         1    0.06
```

The Kenward–Roger approach

The Kenward–Roger modification of the *F*–statistic

For the cranial distances data the Kenward and Roger modified F-test yields

```
formula(ort1ML)

## distance ~ age + Sex + age:Sex + (1 + age | Subject)

formula(ort2ML)

## distance ~ age + Sex + (1 + age | Subject)

ort1<- update(ort1ML, .~., REML = TRUE)
ort2<- update(ort2ML, .~., REML = TRUE)
```

The Kenward–Roger approach

The Kenward–Roger modification of the *F*–statistic

```
KRmodcomp(ort1, ort2)
```

```
## F-test with Kenward-Roger approximation; computing time: 0.11 sec.
## large : distance ~ age + Sex + (1 + age | Subject) + age:Sex
## small : distance ~ age + Sex + (1 + age | Subject)
##         stat   ndf   ddf F.scaling p.value
## Ftest   5.12  1.00 25.52         1   0.032
```

The p-value form the χ^2-test was 0.0249.

The Kenward–Roger approach

Shortcommings of Kenward-Roger

- The Kenward–Roger approach is no panacea.
- In the computations of the degrees of freedom we need to compute
  $$G_j \Sigma^{-1} G_j$$
  where $\Sigma = \sum_i \sigma_i \boldsymbol{G}_i$. Can be space and time consuming!
- An alternative is a Sattherthwaite–kind approximation which is faster to compute. Will come out in next release of **pbkrtest** (code not tested yet). Way faster...
- What to do with generalized linear mixed models – or even with generalized linear models.
- **pbkrtest** also provides the parametric bootstrap *p*-value. Computationally somewhat demanding, but can be parallelized.

Parametric bootstrap

We have two competing models; a large model $f_1(y; \theta)$ and a null model $f_0(y; \theta_0)$; the null model is a submodel of the large model.

```
lg <- update(beetLarge, REML=FALSE)
sm <- update(beet_no.harv, REML=FALSE)
t.obs <- 2*(logLik(lg)-logLik(sm))
t.obs


## 'log Lik.' 12.91 (df=10)
```

Idea is simple: Draw B parametric bootstrap samples $t^1, \dots, t^B$ under the fitted null model $\hat{\theta}_0$.

That is; simulate B datasets from the fitted null model; fit the large and the null model to each of these datasets; calculate the LR-test statistic for each simulated data:

Parametric bootstrap

```
set.seed(121315)
t.sim <- PBrefdist(lg, sm, nsim=500)
```

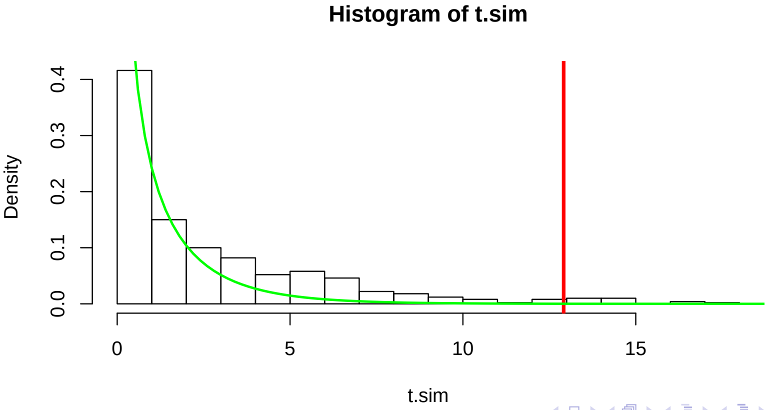
The *p*-value is the fraction of simulated test statistics that are larger or equal to the observed one:

```
head(t.sim)

## [1] 3.1363 0.6829 0.1203 1.1063 6.8241 7.3922

sum( t.sim >= t.obs ) / length( t.sim )

## [1] 0.026
```

Parametric bootstrap

Interesting to overlay limiting χ^2_1 distribution and simulated reference distribution:

```
hist(t.sim, breaks=20, prob=T)
abline(v=t.obs, col="red", lwd=3)
f <- function(x){dchisq(x, df=1)}
curve(f, 0, 20, add=TRUE, col="green", lwd=2)
```



Parametric bootstrap

Do the same for sowing time:

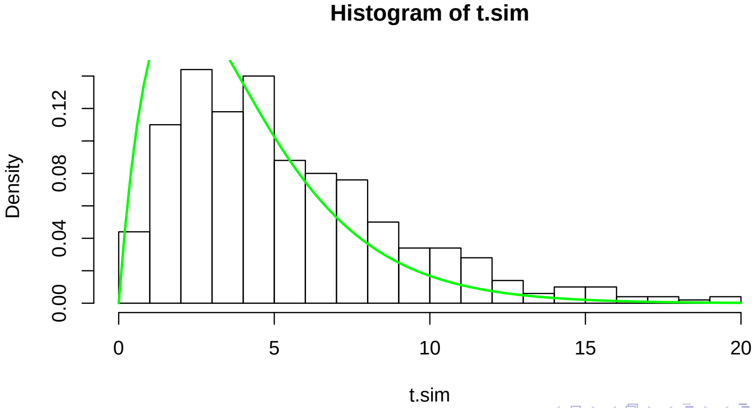
```
lg <- update(beetLarge, REML=FALSE)
sm <- update(beet_no.sow, REML=FALSE)
t.obs <- 2*(logLik(lg)-logLik(sm))
t.obs


## 'log Lik.' 85.2 (df=10)
```

```
set.seed(121315)
t.sim <- PBrefdist(lg, sm, nsim=500)
```

Parametric bootstrap

Interesting to overlay limiting χ^2_1 distribution and simulated reference distribution:

```r
hist(t.sim, breaks=20, prob=T)
abline(v=t.obs, col="red", lwd=3)
f <- function(x){dchisq(x, df=4)}
curve(f, 0, 20, add=TRUE, col="green", lwd=2)
```



Parametric bootstrap

This scheme is implemented as:

R

```
set.seed(121315)
pb <- PBmodcomp(beetLarge, beet_no.harv)
pb
```

```
## Parametric bootstrap test; time: 19.17 sec; samples: 1000 extremes: 40;
## large : sugpct ~ block + sow + harvest + (1 | block:harvest)
## small : sugpct ~ block + sow + (1 | block:harvest)
##         stat df p.value
## LRT     11.8  1 0.00059
## PBtest  11.8    0.04096
```

Parametric bootstrap

In addition we can get p-values

1. directly via the proportion of sampled t_i exceeding t_{obs},
2. approximating the distribution of the scaled statistic $\frac{f}{\bar{t}} \cdot T$ by a χ^2_f distribution (Bartlett type correction) ($\bar{t}$ is the sample average and f the difference in the number of parameters between the null and the alternative model)
3. approximating the bootstrap distribution by a $\Gamma(\alpha, \beta)$ distribution which mean and variance match the moments of the bootstrap sample.
4. approximating the bootstrap distribution by a $F_{d,m}$ distribution which mean is based on matching mean of the bootstrap sample.

Parametric bootstrap

```
summary(pb)
```

```
## Parametric bootstrap test; time: 19.17 sec; samples: 1000 extremes: 40;
## large : sugpct ~ block + sow + harvest + (1 | block:harvest)
## small : sugpct ~ block + sow + (1 | block:harvest)
##           stat    df   ddf p.value
## PBtest   11.82             0.04096
## Gamma    11.82             0.03510
## Bartlett  4.05  1.00       0.04416
## F        11.82  1.00  3.04 0.04042
## LRT      11.82  1.00       0.00059
```

Parametric bootstrap

Parallel computations

Parametric bootstrap is computationally demanding, but multiple cores can be exploited:

```
library(parallel)
nc <- detectCores()
nc


## [1] 4
```

```
clus <- makeCluster(rep("localhost", nc))
```

Parametric bootstrap

Parallel computations

R

```
set.seed(121315)
pb1 <- PBmodcomp(beetLarge, beet_no.harv)
pb1


## Parametric bootstrap test; time: 19.12 sec; samples: 1000 extremes: 40;
## large : sugpct ~ block + sow + harvest + (1 | block:harvest)
## small : sugpct ~ block + sow + (1 | block:harvest)
##         stat df p.value
## LRT     11.8  1 0.00059
## PBtest 11.8    0.04096


pb2 <- PBmodcomp(beetLarge, beet_no.harv, cl=clus)
pb2


## Parametric bootstrap test; time: 10.00 sec; samples: 1000 extremes: 42;
## large : sugpct ~ block + sow + harvest + (1 | block:harvest)
## small : sugpct ~ block + sow + (1 | block:harvest)
##         stat df p.value
## LRT     11.8  1 0.00059
## PBtest 11.8    0.04296
```

Parametric bootstrap

Parallel computations

Results from sugar beets:

Table: p-values (× 100) for removing the harvest or sow effect.

	LRT	KR	ParmBoot	Bartlett	Gamma
harvest	0.03	6	4.1	8.3	4.9
sow	<0.001	<0.001	<0.001	<0.001	<0.001

Results for cranial distance data:

Table: p-values (× 100) testing $\beta_{boy} = \beta_{girl}$.

LRT	KR	ParmBoot	Bartlett	Gamma
2.5	3.3	4.2	4.0	4.2

Parametric bootstrap

Parallel computations

The above approaches are computationally intensive but there are possibilities for speedups:

Instead of simulating a fixed number of values $t^1, \dots, t^M$ for determining the reference distribution used for finding p^{PB} we may instead introduce a stopping rule saying *simulate until we have found, say 20 values t^j larger than t_{obs}.* If J simulations are made then the reported p–value is $20/J$.

Estimating tail–probabilities will require more samples than estimating the mean (and variance) of the reference distribution. Therefore the Bartlett and gamma approaches will require fewer simulations than needed for finding p^{PB}.

The simulation of the reference distribution can be parallelized onto different processors.

Small simulation study: A random regression problem

We consider the simulation from a simple random coefficient model (cf. Kenward and Roger (1997, table 4)):

$$y_{it} = \beta_0 + \beta_1 \cdot t_i + A_i + B_i \cdot t_i + \epsilon_{it}$$

with $cov(A_i, B_i) = \begin{bmatrix} 0.250 & -0.133 \\ -0.133 & 0.250 \end{bmatrix}$ and $var(\epsilon_{it}) = 0.25$.

There are observed $i = 1, \ldots, 24$ subjects divided in groups of 8. For each group observations are at the non overlapping times $t = 0, 1, 2$; $t = 3, 4, 5$ and $t = 6, 7, 8$.

Small simulation study: A random regression problem

Table: Observed test sizes (×100) for $H_0 : \beta_k = 0$ for random coefficient model.

	LR	Wald	ParmBoot	Bartlett	Gamma	KR(R)	KR(SAS
β_0	6.8	4.6	5.2	5.2	5.4	4.0	5.4
β_1	7.3	5.3	6.0	6.0	5.9	5.4	6.3

Final remarks

- The functions `KRmodcomp()` and `PBmodcomp()` described here are available in the `pbkrtest` package.
- The Kenward–Roger approach requires fitting by REML; the parametric bootstrap approaches requires fitting by ML.
- The required fitting scheme is set by the relevant functions, so the user needs not worry about this.
- Parametric bootstrap is parallelized using the `snow` package.

Final remarks

- Halekoh, U., Højsgaard, S. (2014) *A Kenward-Roger Approximation and Parametric Bootstrap Methods for Tests in Linear Mixed Models The R Package pbkrtest*
- Alnosaier, W. (2007) *Kenward-Roger Approximate F Test for Fixed Effects in Mixed Linear Models*, Dissertation, Oregon State University
- Bates, D., Maechler, M. and Bolker, B., Walker, S. (2015) *lme4: Linear mixed-effects models using S4 classes*, R package version 0.999375-39.
- Kenward, M. G. and Roger, J. H. (1997) *Small Sample Inference for Fixed Effects from Restricted Maximum Likelihood*, Biometrics, Vol. 53, pp. 983–997
- Spilke J., Piepho, H.-P. and Hu, X. Hu (2005) *A Simulation Study on Tests of Hypotheses and Confidence Intervals for Fixed Effects in Mixed Models for Blocked Experiments With Missing Data* Journal of Agricultural, Biological, and Environmental Statistics, Vol. 10,p. 374-389