Translating lme4 models to sommer

Giovanny Covarrubias-Pazaran

2021-05-05

The sommer package was developed to provide R users a powerful and reliable multivariate mixed model solver. The package is focused on problems of the type p > n (more effects to estimate than observations) and its core algorithm is coded in C++ using the Armadillo library. This package allows the user to fit mixed models with the advantage of specifying the variance-covariance structure for the random effects, and specifying heterogeneous variances, and obtaining other parameters such as BLUPs, BLUEs, residuals, fitted values, variances for fixed and random effects, etc.

The purpose of this vignette is to show how to translate the syntax formula from lme4 models to sommer models. Feel free to remove the comment marks from the lme4 code so you can compare the results.

1) Random slopes with same intercept
2) Random slopes and random intercepts (without correlation)
3) Random slopes and random intercepts (with correlation)
4) Random slopes with a different intercept
5) Other models not available in lme4

1) Random slopes

This is the simplest model people use when a random effect is desired and the levels of the random effect are considered to have the same intercept.

```r
# install.packages("lme4")
# library(lme4)
library(sommer)
data(DT_sleepstudy)
DT <- DT_sleepstudy

###########
## lme4
###########
fm1 <- lmer(Reaction ~ Days + (1 | Subject), data=DT)
summary(fm1) # or vc <- VarCorr(fm1); print(vc, comp=c("Variance"))

# Random effects:
# Groups Name Variance Std.Dev.
# Subject (Intercept) 1378.2 37.12
# Residual 960.5 30.99
# Number of obs: 180, groups: Subject, 18

###########
## sommer
###########
fm2 <- mmer(Reaction ~ Days, random=~ Subject, data=DT, tolparinv = 1e-6, verbose = FALSE)
summary(fm2)$varcomp
```
VarComp VarCompSE Zratio Constraint
Subject.Reaction-Reaction 1377.9758 505.0776 2.728246 Positive
units.Reaction-Reaction 960.4705 107.0638 8.971013 Positive

2) Random slopes and random intercepts (without correlation)

This is the a model where you assume that the random effect has different intercepts based on the levels of another variable. In addition the || in lme4 assumes that slopes and intercepts have no correlation.

```
# lme4

# fm1 <- lmer(Reaction ~ Days + (Days || Subject), data=DT)
# summary(fm1) # or vc <- VarCorr(fm1); print(vc,comp=c("Variance"))
# Random effects:
# Groups Name Variance Std.Dev.
# Subject  (Intercept) 627.57  25.051
# Subject.1 Days 35.86   5.988
# Residual 653.58   25.565
# Number of obs: 180, groups: Subject, 18
```

```
# sommer

fm2 <- mmer(Reaction ~ Days,
             random= ~ Subject + vs(Days, Subject),
             data=DT, tolparinv = 1e-6, verbose = FALSE)
summary(fm2)$varcomp
```

VarComp VarCompSE Zratio Constraint
Subject.Reaction-Reaction 627.54087 283.52939 2.213319 Positive
Days:Subject.Reaction-Reaction 35.86008 14.53187 2.467686 Positive
units.Reaction-Reaction 653.58305 76.72711 8.518281 Positive

Notice that Days is a numerical (not factor) variable.

3) Random slopes and random intercepts (with correlation)

This is the a model where you assume that the random effect has different intercepts based on the levels of another variable. In addition a single | in lme4 assumes that slopes and intercepts have a correlation to be estimated.

```
# lme4

# fm1 <- lmer(Reaction ~ Days + (Days | Subject), data=DT)
# summary(fm1) # or vc <- VarCorr(fm1); print(vc,comp=c("Variance"))
# Random effects:
# Groups Name Variance Std.Dev. Corr
# Subject  (Intercept) 612.10 24.741
# Days 35.07 5.922 0.07
# Residual 654.94 25.592
# Number of obs: 180, groups: Subject, 18
```

```
# sommer
```
### no equivalence in sommer to find the correlation between the 2 vc
### this is the most similar which is equivalent to (intercept // slope)

```r
fm2 <- mmer(Reaction ~ Days,
    random = ~ Subject + vs(Days, Subject),
    data = DT, tolparinv = 1e-6, verbose = FALSE)
summary(fm2)$varcomp
```

<table>
<thead>
<tr>
<th>VarComp</th>
<th>VarCompSE</th>
<th>Zratio</th>
<th>Constraint</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subject.Reaction-Reaction</td>
<td>627.54087</td>
<td>283.52939</td>
<td>2.213319</td>
</tr>
<tr>
<td>Days:Subject.Reaction-Reaction</td>
<td>35.86008</td>
<td>14.53187</td>
<td>2.467686</td>
</tr>
<tr>
<td>units.Reaction-Reaction</td>
<td>653.58305</td>
<td>76.72711</td>
<td>8.518281</td>
</tr>
</tbody>
</table>

4) Random slopes with a different intercept

This is the a model where you assume that the random effect has different intercepts based on the levels of another variable but there’s not a main effect. The 0 in the intercept in lme4 assumes that random slopes interact with an intercept but without a main effect.

### lme4

```r
# fm1 <- lmer(Reaction ~ Days + (0 + Days | Subject), data=DT)
# summary(fm1) # or vc <- VarCorr(fm1); print(vc,comp=c("Variance"))
# Random effects:
# Groups     Name Variance Std.Dev.
# Subject    Days 52.71  7.26
# Residual   842.03 29.02
# Number of obs: 180, groups: Subject, 18
```

### sommer

```r
# fm2 <- mmer(Reaction ~ Days,
#    random = ~ vs(Days, Subject),
#    data = DT, tolparinv = 1e-6, verbose = FALSE)
summary(fm2)$varcomp
```

<table>
<thead>
<tr>
<th>VarComp</th>
<th>VarCompSE</th>
<th>Zratio</th>
<th>Constraint</th>
</tr>
</thead>
<tbody>
<tr>
<td>Days:Subject.Reaction-Reaction</td>
<td>52.70946</td>
<td>19.09984</td>
<td>2.759681</td>
</tr>
<tr>
<td>units.Reaction-Reaction</td>
<td>842.02736</td>
<td>93.84640</td>
<td>8.972399</td>
</tr>
</tbody>
</table>

4) Other models available in sommer but not in lme4

One of the strengths of sommer is the availability of other variance covariance structures. In this section we show 4 models available in sommer that are not available in lme4 and might be useful.

```r
library(orthopolynom)
# diagonal model
fm2 <- mmer(Reaction ~ Days,
    random = ~ vs(ds(Daysf), Subject),
    data = DT, tolparinv = 1e-6, verbose = FALSE)
summary(fm2)$varcomp
```

<table>
<thead>
<tr>
<th>VarComp</th>
<th>VarCompSE</th>
<th>Zratio</th>
<th>Constraint</th>
</tr>
</thead>
<tbody>
<tr>
<td>Days:Subject.Reaction-Reaction</td>
<td>52.70946</td>
<td>19.09984</td>
<td>2.759681</td>
</tr>
<tr>
<td>units.Reaction-Reaction</td>
<td>842.02736</td>
<td>93.84640</td>
<td>8.972399</td>
</tr>
</tbody>
</table>
## Subject.Reaction-Reaction

<table>
<thead>
<tr>
<th>Reaction 1</th>
<th>Reaction 2</th>
<th>Zratio</th>
<th>Constraint</th>
</tr>
</thead>
<tbody>
<tr>
<td>139.5473</td>
<td>399.5095</td>
<td>0.3492967</td>
<td>Positive</td>
</tr>
<tr>
<td>196.8544</td>
<td>411.8262</td>
<td>0.4780037</td>
<td>Positive</td>
</tr>
<tr>
<td>556.0773</td>
<td>501.2665</td>
<td>1.1093445</td>
<td>Positive</td>
</tr>
<tr>
<td>855.2104</td>
<td>581.8190</td>
<td>1.4698910</td>
<td>Positive</td>
</tr>
<tr>
<td>1699.4269</td>
<td>820.4561</td>
<td>2.0713197</td>
<td>Positive</td>
</tr>
<tr>
<td>2910.8975</td>
<td>1175.7872</td>
<td>2.4757011</td>
<td>Positive</td>
</tr>
<tr>
<td>1539.6201</td>
<td>779.1437</td>
<td>1.9760413</td>
<td>Positive</td>
</tr>
<tr>
<td>2597.5337</td>
<td>1089.4522</td>
<td>2.3842568</td>
<td>Positive</td>
</tr>
<tr>
<td>3472.7108</td>
<td>1351.5702</td>
<td>2.5693899</td>
<td>Positive</td>
</tr>
<tr>
<td>879.6958</td>
<td>247.4680</td>
<td>3.5547862</td>
<td>Positive</td>
</tr>
</tbody>
</table>

## units.Reaction-Reaction

<table>
<thead>
<tr>
<th>Reaction 1</th>
<th>Reaction 2</th>
<th>Zratio</th>
<th>Constraint</th>
</tr>
</thead>
<tbody>
<tr>
<td>879.6958</td>
<td>247.4680</td>
<td>3.5547862</td>
<td>Positive</td>
</tr>
</tbody>
</table>

```r
fm2 <- mmer(Reaction ~ Days, random = ~ vs(us(Daysf), Subject), data = DT, tolparinv = 1e-6, verbose = FALSE)
summary(fm2)$varcomp
```

## VarComp  VarCompSE  Zratio    Constraint

<table>
<thead>
<tr>
<th>Reaction 1</th>
<th>Reaction 2</th>
<th>Zratio</th>
<th>Constraint</th>
</tr>
</thead>
<tbody>
<tr>
<td>402.6286</td>
<td>572.0867</td>
<td>0.7037894</td>
<td>Positive</td>
</tr>
<tr>
<td>1022.5098</td>
<td>393.6922</td>
<td>2.5972314</td>
<td>Unconstr</td>
</tr>
<tr>
<td>417.6460</td>
<td>521.3722</td>
<td>0.8010515</td>
<td>Positive</td>
</tr>
<tr>
<td>540.3746</td>
<td>287.1704</td>
<td>1.8817210</td>
<td>Unconstr</td>
</tr>
<tr>
<td>828.5156</td>
<td>325.7576</td>
<td>2.5433499</td>
<td>Unconstr</td>
</tr>
<tr>
<td>0.0000</td>
<td>509.8962</td>
<td>0.0000000</td>
<td>Positive</td>
</tr>
<tr>
<td>798.3750</td>
<td>397.0884</td>
<td>2.0105726</td>
<td>Unconstr</td>
</tr>
<tr>
<td>1137.3863</td>
<td>443.9056</td>
<td>2.5622566</td>
<td>Unconstr</td>
</tr>
<tr>
<td>1057.0708</td>
<td>385.9026</td>
<td>2.7392162</td>
<td>Unconstr</td>
</tr>
<tr>
<td>760.2469</td>
<td>436.7463</td>
<td>1.7407060</td>
<td>Positive</td>
</tr>
<tr>
<td>757.8909</td>
<td>411.2464</td>
<td>1.8429119</td>
<td>Unconstr</td>
</tr>
<tr>
<td>1039.6832</td>
<td>447.5192</td>
<td>2.3232148</td>
<td>Unconstr</td>
</tr>
<tr>
<td>911.1369</td>
<td>377.9651</td>
<td>2.4106377</td>
<td>Unconstr</td>
</tr>
<tr>
<td>1590.6778</td>
<td>566.5376</td>
<td>2.8077180</td>
<td>Unconstr</td>
</tr>
<tr>
<td>957.1797</td>
<td>364.0599</td>
<td>2.6291817</td>
<td>Unconstr</td>
</tr>
<tr>
<td>932.5247</td>
<td>516.7169</td>
<td>1.8047110</td>
<td>Unconstr</td>
</tr>
<tr>
<td>1179.5219</td>
<td>547.9498</td>
<td>2.1526095</td>
<td>Unconstr</td>
</tr>
<tr>
<td>859.1635</td>
<td>440.5250</td>
<td>1.9503173</td>
<td>Unconstr</td>
</tr>
<tr>
<td>1672.9989</td>
<td>664.0846</td>
<td>2.5192556</td>
<td>Unconstr</td>
</tr>
<tr>
<td>2003.0176</td>
<td>738.6399</td>
<td>2.7117633</td>
<td>Unconstr</td>
</tr>
<tr>
<td>2067.9299</td>
<td>553.3254</td>
<td>3.7372765</td>
<td>Positive</td>
</tr>
<tr>
<td>666.1077</td>
<td>565.7589</td>
<td>1.1773702</td>
<td>Unconstr</td>
</tr>
<tr>
<td>850.9395</td>
<td>583.6190</td>
<td>1.4580394</td>
<td>Unconstr</td>
</tr>
<tr>
<td>916.2375</td>
<td>504.0273</td>
<td>1.8178333</td>
<td>Unconstr</td>
</tr>
<tr>
<td>1785.8432</td>
<td>750.7274</td>
<td>2.3788171</td>
<td>Unconstr</td>
</tr>
<tr>
<td>2077.5064</td>
<td>822.0777</td>
<td>2.5271412</td>
<td>Unconstr</td>
</tr>
<tr>
<td>2603.2823</td>
<td>1035.1406</td>
<td>2.5149070</td>
<td>Unconstr</td>
</tr>
<tr>
<td>3123.2005</td>
<td>1049.0352</td>
<td>2.9772123</td>
<td>Positive</td>
</tr>
<tr>
<td>932.8190</td>
<td>490.4744</td>
<td>1.9018709</td>
<td>Unconstr</td>
</tr>
<tr>
<td>927.3416</td>
<td>492.7764</td>
<td>1.8818709</td>
<td>Unconstr</td>
</tr>
<tr>
<td>924.7079</td>
<td>426.2387</td>
<td>2.1694602</td>
<td>Unconstr</td>
</tr>
<tr>
<td>1282.8637</td>
<td>583.3415</td>
<td>2.1991642</td>
<td>Unconstr</td>
</tr>
<tr>
<td>1549.9053</td>
<td>643.7083</td>
<td>2.4077757</td>
<td>Unconstr</td>
</tr>
<tr>
<td>1941.5523</td>
<td>811.3286</td>
<td>2.3930529</td>
<td>Unconstr</td>
</tr>
<tr>
<td>2306.0261</td>
<td>951.5128</td>
<td>2.4235367</td>
<td>Unconstr</td>
</tr>
<tr>
<td>1669.8274</td>
<td>612.0081</td>
<td>2.7284398</td>
<td>Positive</td>
</tr>
</tbody>
</table>
## 8:0: Subject.Reaction-Reaction 920.3110  576.8500  1.5954079 Unconstr
## 8:1: Subject.Reaction-Reaction 1044.9313  592.5243  1.7635247 Unconstr
## 8:2: Subject.Reaction-Reaction 831.4993  486.9625  1.7075221 Unconstr
## 8:3: Subject.Reaction-Reaction 1607.0156  717.6871  2.2391591 Unconstr
## 8:4: Subject.Reaction-Reaction 2029.1022  805.6724  2.5185201 Unconstr
## 8:5: Subject.Reaction-Reaction 3058.1945 1093.4722  2.7967739 Unconstr
## 8:6: Subject.Reaction-Reaction 2927.6051 1177.5589  2.4861644 Unconstr
## 8:7: Subject.Reaction-Reaction 2433.2427  957.7103  2.5406876 Unconstr
## 8:8: Subject.Reaction-Reaction 2947.1635  844.8113  3.4885466 Positive

## 9:0: Subject.Reaction-Reaction 1440.6886  690.1726  2.0874323 Unconstr
## 9:1: Subject.Reaction-Reaction 1514.9679  703.4423  2.1536491 Unconstr
## 9:2: Subject.Reaction-Reaction 967.8504  550.1628  1.7592073 Unconstr
## 9:3: Subject.Reaction-Reaction 1742.6866  797.5934  2.1849310 Unconstr
## 9:4: Subject.Reaction-Reaction 2198.3504  892.7701  2.4623924 Unconstr
## 9:5: Subject.Reaction-Reaction 3236.8715 1391.5584  2.7645359 Unconstr
## 9:6: Subject.Reaction-Reaction 2210.6321 1185.1233  1.8653182 Unconstr
## 9:7: Subject.Reaction-Reaction 2399.5130 1027.8125  2.3345824 Unconstr
## 9:8: Subject.Reaction-Reaction 3847.0132 1390.5381  2.7645359 Unconstr
## 9:9: Subject.Reaction-Reaction 3946.2369 1228.6678  3.2118013 Positive
## 9:10: Subject.Reaction-Reaction 883.2477  577.9203  1.5283210 Positive

## random regression (legendre polynomials)
fm2 <- mmer(Reaction ~ Days,
            random = ~ vs(leg(Days,1), Subject),
            data=DT, tolparinv = 1e-6, verbose = FALSE)
summary(fm2)$varcomp

## unstructured random regression (legendre)
fm2 <- mmer(Reaction ~ Days,
            random = ~ us(leg(Days,1)), Subject),
            data=DT, tolparinv = 1e-6, verbose = FALSE)
summary(fm2)$varcomp

Final remarks

Keep in mind that **sommer** uses the direct inversion (DI) algorithm which can be very slow for large datasets. The package is focused on problems of the type p > n (more random effect levels than observations) and models with dense covariance structures. For example, for an experiment with dense covariance structures with low-replication (i.e. 2000 records from 1000 individuals replicated twice with a covariance structure of 1000x1000), **sommer** will be faster than MME-based software. Also for genomic problems with large number of random effect levels, i.e. 300 individuals (n) with 100,000 genetic markers (p). For highly replicated trials with small number of individuals and covariance structures or n > p (i.e. 2000 records from 200 individuals replicated 10 times with covariance structure of 200x200), **asreml** or other MME-based algorithms will be much faster and we recommend you to opt for those software programs.
Literature


Covarrubias-Pazaran G. 2018. Software update: Moving the R package sommer to multivariate mixed models for genome-assisted prediction. doi: https://doi.org/10.1101/354639


