Output from heifers.R

```
> # heifers.R
>
> library(lsmeans)
> library(lme4)
> #library(pbkrtest)
> library(car)
>
> # get the data
> heifers<-read.table(file="heifers.dat",header=T,
+                 colClasses=c("factor","factor","factor","factor","numeric"))
> head(heifers)
  farm week heifer feed consumpt
1    1    1      1    a      2.7
2    1    1      2    c      2.6
3    2    1      3    b      1.9
4    1    2      1    c      2.1
5    1    2      2    b      0.2
6    2    2      3    a      2.3
> is.factor(heifers$farm)
[1] TRUE
> is.factor(heifers$consumpt)
[1] FALSE
>
> # to fit this model with the lmer function in the lme4 package it is better
> # to avoid implicitly nested factors. Instead create a heifer factor (called
> # allheif below) that has 3*2=6 levels (a unique level for each heifer within
> # each farm).
> heifers <- within(heifers , allheif <- factor(heifer:farm))
> head(heifers)
  farm week heifer feed consumpt allheif
1    1    1      1    a      2.7     1:1
2    1    1      2    c      2.6     2:1
3    2    1      3    b      1.9     3:2
4    1    2      1    c      2.1     1:1
5    1    2      2    b      0.2     2:1
6    2    2      3    a      2.3     3:2
>
> # mixed effects model with random effects of week heifer(farm) farm
> # using reml in the function lmer() from package lme4
> m1<-lmer(consumpt~feed+(1|week)+(1|heifer)+(1|farm),data=heifers)
> summary(m1)
Linear mixed model fit by REML ['lmerMod']
Formula: consumpt ~ feed + (1 | week) + (1 | heifer) + (1 | farm)
   Data: heifers

REML criterion at convergence: 32.2943

Random effects:
 Groups   Name        Variance  Std.Dev.
 heifer   (Intercept) 7.361e-03 8.580e-02
 week     (Intercept) 2.565e-12 1.602e-06
 farm     (Intercept) 8.380e-02 2.895e-01
 Residual             2.731e-01 5.225e-01
Number of obs: 18, groups: heifer, 3; week, 3; farm, 2
```

```
Fixed effects:
            Estimate Std. Error t value
(Intercept)   2.0389     0.2440   8.357
feed1         0.6111     0.1742   3.508
feed2        -0.7889     0.1742  -4.529

Correlation of Fixed Effects:
      (Intr) feed1
feed1  0.000
feed2  0.000 -0.500
> anova(m1,type=3)
Analysis of Variance Table
     Df Sum Sq Mean Sq F value
feed  2 6.1644  3.0822  11.288
>
> # get the lsmeans for each level of feed
> lsmeans(m1,specs=~feed)
$`feed lsmeans`
 feed   lsmean        SE       df  lower.CL upper.CL
    a 2.650000 0.2997675 2.289433 1.5045241 3.795476
    b 1.250000 0.2997675 2.289433 0.1045241 2.395476
    c 2.216667 0.2997675 2.289433 1.0711908 3.362143

>
>
> # Notice that the results above, including the type 3 F test for feed,
> # estimates of variance components, and standard errors of the estimated
> # means for each feed type, do not agree with the results from PROC MIXED
> # in heifers.sas. The reason for this is that to fit this model in heifers.sas
> # I used method=type3, whereas the lmer function uses REML estimation.
> # REML and Type3 estimation typically agree for this model, but that agreement
> # breaks down when the type 3 variance component estimates are negative. In
> # that case, REML constrains the corresponding variance component estimates
> # to be non-negative, and that affects other results like the F tests on
> # the fixed effects and standard errors on the estimated means. Type 3
> # estimation of a mixed effect model is not implemented in lmer or any of
> # the other functions designed for linear mixed models that are available in
> # R. However, the model can be fit with lm with heifers, farms and weeks
> # treated as fixed effects. That approach will give the an F test for feeds
> # that agrees with the type 3 result from PROC MIXED that we obtained in
> # heifers.sas. However, other results form that analysis, including standard
> # errors and confidence intervals on estimated means, will not be correct.
>
> # to get the same Type 3 Analysis of Variance as in SAS
> # we need to fit the model treating week heifer farm as fixed effects using lm
>
> # first, change the handling of unordered factors to use the sum-to-zero
constraints
> op<-options(contrasts=c("contr.sum","contr.poly"))
> options()$contrasts
[1] "contr.sum"  "contr.poly"
>
> m1a<-aov(consumpt~feed+week+farm/heifer+farm,data=heifers)
> # get the type III SSs
> Anova(m1a,type=3)
Anova Table (Type III tests)
```

```
Response: consumpt
            Sum Sq Df  F value    Pr(>F)
(Intercept) 74.827  1 233.4298 3.341e-07 ***
feed         6.164  2   9.6153   0.00745 **
week         0.444  2   0.6932   0.52765
farm         1.027  1   3.2045   0.11122
farm:heifer  0.902  4   0.7036   0.61119
Residuals    2.564  8
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
> # Alternatively, if we are not interested in separating farm and
> # heifer within farm variability, we can just account for 6 heifer effects
> # in the model. In that case, the Type 3 F test for feed types produced
> # by the Type 3 anova estimation method (as implemented in PROC MIXED using
> # using method=type3) or in a a fixed-effect model fitting routine (like
> # PROC GLM or the lm function in R) will not change.
>
> m2a<-aov(consumpt~feed+week+allheif,data=heifers)
> # get the type III SSs
> Anova(m2a,type=3)
Anova Table (Type III tests)

Response: consumpt
            Sum Sq Df  F value    Pr(>F)
(Intercept) 74.827  1 233.4298 3.341e-07 ***
feed         6.164  2   9.6153   0.00745 **
week         0.444  2   0.6932   0.52765
allheif      1.929  5   1.2038   0.38750
Residuals    2.564  8
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
> # reset contrasts to their original values
> options(op)
>
>
```