

Changes to the Mixed Effects Models chapters in ELM

Julian Faraway

Contents

Introduction	1
Revisions to Chapter 8	3
8.1 Estimation	3
8.2 Inference	4
8.3 Predicting Random Effects	4
8.4 Blocks as Random Effects	5
8.5 Split plots	6
8.6 Nested Effects	7
8.7 Crossed Effects	8
8.8 Multilevel Models	9
Revisions to Chapter 9	12
9.1 Longitudinal Data	12
9.2 Repeated Measures	13
9.3 Multiple Response Multilevel Models	15
References	15

[1] "Created: Wed Oct 1 15:27:16 2014"

Introduction

The book *Extending the Linear Model with R* (ELM) (Faraway 2006) first appeared in 2005 and was based on R version 2.2.0. R is updated regularly and so it is natural that some incompatibilities with the current version have been introduced. For most of the chapters, these changes have been minor and have been addressed in the errata and/or subsequent reprintings of the text. However, for chapter 8 and 9, the changes have been much more substantial. These chapters are based on the `lme4` package. (Bates 2005) with a more recent description of `lme4` in (Bates et al. 2014). The test-statistic and p -value information has been removed from the `summary` and `anova` output. This makes doing inference difficult.

For standard linear models (such as those considered in *Linear Models with R* (Faraway 2005)), the recommended way to compare an alternative hypothesis of a larger model compared to a null hypothesis of a smaller model nested within this larger model, is to use an F -test. Under the standard assumptions and when the errors are normally distributed, the F -statistic has an exact F -distribution with degrees of freedom that can be readily computed given the sample size and the number of parameters used by each model.

For linear mixed effects models, that is models having some random effects, we might also wish to test fixed effect terms using an F -test. One way of approaching this is to assume that the estimates of the parameters

characterizing the random effects of the model are in fact the true values. This reduces the mixed models to fixed effect models where the error has a particular covariance structure. Such models can be fit using generalized least squares and F -tests can be conducted using standard linear models theory. Several statistics software packages take this approach including the `nlme` package developed earlier and still available within R. Earlier versions of `lme4` also took this approach and hence the output seen in the current version of ELM.

However, there are two serious problems with this test. Firstly, the random effects are not actually known, but estimated. This means that the F -statistic does not always follow an F -distribution exactly. In some cases, it may be a good approximation, but not in general. Secondly, even if one were to assume that the F -distribution was a sufficiently good approximation, there remains the problem of degrees of freedom. The concept of “degrees of freedom”, as used in statistics, is not as well defined as many people believe. Perhaps one might think of it as the effective number of independent observations on which an estimate or test is based. Often, this is just the sample size minus the number of free parameters. However, this notion becomes more difficult when considering the dependent and hierarchical data found in mixed effects models. There is no simple way in which the degrees of freedom can be counted. The degrees of freedom are used here just to select the null distribution for a test-statistic i.e. they are used as a mathematical convenience rather than as a concept of standalone value. As such, the main concern is whether they produce the correct null distribution for the test statistic. In the case of F -statistics for mixed models, there has been substantial research on this — see (Kenward and Roger 1997) and related — but there is no simple and general solution. Even if there were, this would still not avoid the problem of the dubious approximation to an F -distribution.

The t -statistics presented in the `lme4` model summary outputs are based on the square roots of F -statistics and so the same issues with testing still arise. In some cases, one may appeal to asymptotics to allow for simple normal and chi-squared approximations to be used. But it is not simply a matter of sample size — the number of random effects parameters and the model structure all make a difference to the quality of the approximation. There is no simple rule to say when the approximation would be satisfactory.

All this poses a problem for the writers of statistical software for such models. One approach is to simply provide the approximate solution even though it is known to be poor in some cases. Or one can take the approach that no answer (at least for now) is better than a possibly poor answer, which is the approach currently taken in `lme4`. In some simpler models, specialized solutions are possible. For example, in (Scheffe 1959), F -tests for a range of simple balanced (i.e. equal numbers of observations per group) designs are provided. For some simple but unbalanced datasets, progress has been made — see (Crainiceanu and Ruppert 2004). However, such straightforward solutions are not available for anything more complex. Such partial specialized solutions are not satisfactory for a package as general in scope as `lme4` — we need a solution that works reasonable well in all cases.

We do have some viable alternatives. The parametric bootstrap approach based on the likelihood ratio statistic is discussed in ELM. We can add to this methods based on Markov chain Monte Carlo (MCMC). For some simple balanced models, solutions are available using the `aoV()` command. The `pbkrtest` and `RLRsim` packages can also help.

We present here the changes to the text for chapters 8 and 9. The intent here is to list the changes and suggest simple replacements to achieve much the same result. Further changes to `lme4` may occur so more changes to this document are likely to be necessary.

To help keep this document up-to-date, it is written using [R markdown](#) which inserts the output from R commands directly into the text. However, this does mean all the output which is sometimes more than the edited version that appears in the book. In particular, I have modified the printing of some summaries of `lmer` fits to prevent the printing of correlation matrices.

Let’s start by loading the packages we need and verifying the version of R and `lme4`:

```
library(faraway)
library(lme4)
sessionInfo("lme4")
```

R version 3.1.1 (2014-07-10)

Platform: x86_64-apple-darwin13.1.0 (64-bit)

locale:

[1] en_GB.UTF-8/en_GB.UTF-8/en_GB.UTF-8/C/en_GB.UTF-8/en_GB.UTF-8

attached base packages:

character(0)

other attached packages:

[1] lme4_1.1-7

loaded via a namespace (and not attached):

[1] base_3.1.1	datasets_3.1.1	digest_0.6.4	evaluate_0.5.3
[5] faraway_1.0.6	formatR_0.10	graphics_3.1.1	grDevices_3.1.1
[9] grid_3.1.1	htmltools_0.2.4	knitr_1.6	lattice_0.20-29
[13] MASS_7.3-33	Matrix_1.1-4	methods_3.1.1	minqa_1.2.3
[17] nlme_3.1-117	nloptr_1.0.4	Rcpp_0.11.1	rmarkdown_0.2.68
[21] splines_3.1.1	stats_3.1.1	stringr_0.6.2	tools_3.1.1
[25] utils_3.1.1	yaml_2.1.11		

We also set the seed on the random number generator to ensure exact repeatability for the bootstraps and other simulations:

```
set.seed(123)
```

Revisions to Chapter 8

8.1 Estimation

The first call to `lmer` occurs on p157 where the output now becomes:

```
mmod <- lmer(bright ~ 1+(1|operator), pulp)
summary(mmod)
```

Linear mixed model fit by REML ['lmerMod']

Formula: bright ~ 1 + (1 | operator)

Data: pulp

REML criterion at convergence: 18.6

Scaled residuals:

Min	1Q	Median	3Q	Max
-1.467	-0.759	-0.124	0.628	1.601

Random effects:

Groups	Name	Variance	Std.Dev.
operator	(Intercept)	0.0681	0.261
	Residual	0.1062	0.326

Number of obs: 20, groups: operator, 4

Fixed effects:

	Estimate	Std. Error	t value
(Intercept)	60.400	0.149	404

In the fixed effects part of the output, there is no longer a degrees of freedom and a p -value. In this case, we do not miss the test because the t -value is so large and the intercept so obviously different from zero. The following maximum likelihood based version of the output is also changed:

```
smod <- lmer(bright ~ 1+(1|operator), pulp, REML=FALSE)
summary(smod)
```

```
Linear mixed model fit by maximum likelihood ['lmerMod']
Formula: bright ~ 1 + (1 | operator)
Data: pulp
```

AIC	BIC	logLik	deviance	df.resid
22.5	25.5	-8.3	16.5	17

Scaled residuals:

Min	1Q	Median	3Q	Max
-1.5055	-0.7812	-0.0635	0.6585	1.5623

Random effects:

Groups	Name	Variance	Std.Dev.
operator	(Intercept)	0.0458	0.214
	Residual	0.1062	0.326

Number of obs: 20, groups: operator, 4

Fixed effects:

	Estimate	Std. Error	t value
(Intercept)	60.400	0.129	467

In addition to the the changes to the df and p -value, there are also smaller numerical changes to the random effects estimates due to improvements in the fitting algorithm. Also the way to specify that maximum likelihood estimates are required has changed from `method="ML"` to `REML=FALSE`.

8.2 Inference

Nested hypotheses can still be tested using the likelihood ratio statistic. The chi-squared approximation can be quite inaccurate, giving p -values that tend to be too small. The parametric bootstrap requires much more computation, but gives better results.

The current text also proposed the use of F - or t - statistics, but as explained above, these are no longer provided in the current version of `lme4`. It would be possible to fit most of the models in this chapter using the older `nlme` package, which has a somewhat different syntax, and thereby obtain these F -statistic. Alternatively, it is possible, as we shall demonstrate, to reconstruct these tests from the output. However, one should realize that these may give poor results and we do not recommend doing this in general.

8.3 Predicting Random Effects

No change.

8.4 Blocks as Random Effects

The output of the model on p164 becomes:

```
op <- options(contrasts=c("contr.sum", "contr.poly"))
mmod <- lmer(yield ~ treat + (1|blend), penicillin)
print(summary(mmod),correlation=FALSE)
```

```
Linear mixed model fit by REML ['lmerMod']
Formula: yield ~ treat + (1 | blend)
Data: penicillin
```

```
REML criterion at convergence: 106.6
```

```
Scaled residuals:
```

```
   Min      1Q  Median      3Q      Max
-1.415 -0.502 -0.164  0.683  1.284
```

```
Random effects:
```

```
Groups   Name          Variance Std.Dev.
blend    (Intercept)  11.8      3.43
Residual                    18.8      4.34
Number of obs: 20, groups:  blend, 5
```

```
Fixed effects:
```

```
              Estimate Std. Error t value
(Intercept)    86.00      1.82    47.3
treat1          -2.00      1.68    -1.2
treat2          -1.00      1.68    -0.6
treat3           3.00      1.68     1.8
```

```
options(op)
```

Again we note the lack of p -values for the t -statistics. If one still wanted to perform a t -test, we could use the normal approximation on the t -statistics. Since the three treatment statistics here are well below 2 in absolute value, we might conclude that these treatment effects are not significant. However, providing a more precise p -value is problematic and for t -statistics around 2 or so, some better method of testing would be needed.

The ANOVA table at the top of p165 becomes:

```
anova(mmod)
```

```
Analysis of Variance Table
      Df Sum Sq Mean Sq F value
treat  3     70    23.3    1.24
```

We no longer have a p -value so we can no longer perform the test in this way. The LRT-based test that follows remains unchanged.

8.5 Split plots

On p168, the first model fails with an error. The `try` command is useful when you suspect a command may cause an error and you do not want to interrupt the execution of a batch of commands.

```
tt <- try(lmod <- lmer(yield ~ irrigation * variety +
                     (1|field) +(1|field:variety),data=irrigation))
cat(tt)
```

```
Error in checkNlevels(reTrms$flist, n = n, control) :
  number of levels of each grouping factor must be < number of observations
```

The reason, as explained in the text, is that this model is over-parameterised. Except now the model fails to fit at all, which is actually helpful since it alerts us to the problem. The second (correct) model does fit:

```
lmodr <- lmer(yield ~ irrigation * variety + (1|field),data=irrigation)
logLik(lmodr)
```

```
'log Lik.' -22.7 (df=10)
```

Note the change in the degrees of freedom. The subsequent model summary is:

```
print(summary(lmodr),correlation=FALSE)
```

```
Linear mixed model fit by REML ['lmerMod']
Formula: yield ~ irrigation * variety + (1 | field)
Data: irrigation
```

```
REML criterion at convergence: 45.4
```

```
Scaled residuals:
```

```
   Min      1Q  Median      3Q      Max
-0.745 -0.551  0.000  0.551  0.745
```

```
Random effects:
```

```
Groups   Name             Variance Std.Dev.
field    (Intercept)  16.20    4.02
Residual                    2.11    1.45
```

```
Number of obs: 16, groups: field, 8
```

```
Fixed effects:
```

```
              Estimate Std. Error t value
(Intercept)      38.50      3.03  12.73
irrigationi2       1.20      4.28   0.28
irrigationi3       0.70      4.28   0.16
irrigationi4       3.50      4.28   0.82
varietyv2          0.60      1.45   0.41
irrigationi2:varietyv2 -0.40      2.05  -0.19
irrigationi3:varietyv2 -0.20      2.05  -0.10
irrigationi4:varietyv2  1.20      2.05   0.58
```

As before, the p -values are gone. Note that the t -statistics for the fixed effects are all small and give us a good indication that there are no significant fixed effects here. The subsequent ANOVA table is:

```
anova(lmodr)
```

Analysis of Variance Table

	Df	Sum Sq	Mean Sq	F value
irrigation	3	2.46	0.818	0.39
variety	1	2.25	2.250	1.07
irrigation:variety	3	1.55	0.517	0.25

Again, no p -values. For this dataset, the small t -values are sufficient to conclude that there are no significant fixed effects. This can be confirmed by computing the LRT and estimating its p -value via the parametric bootstrap.

8.6 Nested Effects

The first model output on p171 becomes:

```
cmod <- lmer(Fat ~ 1 + (1|Lab) + (1|Lab:Technician) + (1|Lab:Technician:Sample), data=eggs)
summary(cmod)
```

Linear mixed model fit by REML ['lmerMod']

Formula:

```
Fat ~ 1 + (1 | Lab) + (1 | Lab:Technician) + (1 | Lab:Technician:Sample)
```

Data: eggs

REML criterion at convergence: -64.2

Scaled residuals:

Min	1Q	Median	3Q	Max
-2.0410	-0.4658	0.0093	0.5971	1.5428

Random effects:

Groups	Name	Variance	Std.Dev.
Lab:Technician:Sample	(Intercept)	0.00306	0.0554
Lab:Technician	(Intercept)	0.00698	0.0835
Lab	(Intercept)	0.00592	0.0769
Residual		0.00720	0.0848

Number of obs: 48, groups:

Lab:Technician:Sample, 24; Lab:Technician, 12; Lab, 6

Fixed effects:

	Estimate	Std. Error	t value
(Intercept)	0.388	0.043	9.02

Again the same changes as seen before. The output of the random effects at the top of p172 becomes:

```
cmodr <- lmer(Fat ~ 1 + (1|Lab) + (1|Lab:Technician), data=eggs)
VarCorr(cmodr)
```

Groups	Name	Std.Dev.
Lab:Technician	(Intercept)	0.0895
Lab	(Intercept)	0.0769
Residual		0.0961

which is effectively the same information as in the text.

8.7 Crossed Effects

On p173, the ANOVA becomes:

```
mmod <- lmer(wear ~ material + (1|run) + (1|position), abrasion)
anova(mmod)
```

```
Analysis of Variance Table
          Df Sum Sq Mean Sq F value
material  3  4621    1540    25.1
```

which is again without the p -values. The model output is:

```
print(summary(mmod), correlation=FALSE)
```

```
Linear mixed model fit by REML ['lmerMod']
Formula: wear ~ material + (1 | run) + (1 | position)
Data: abrasion
```

```
REML criterion at convergence: 100.3
```

```
Scaled residuals:
```

```
   Min      1Q  Median      3Q      Max
-1.090 -0.302  0.027  0.422  1.210
```

```
Random effects:
```

Groups	Name	Variance	Std.Dev.
run	(Intercept)	66.9	8.18
position	(Intercept)	107.1	10.35
Residual		61.3	7.83

```
Number of obs: 16, groups: run, 4; position, 4
```

```
Fixed effects:
```

	Estimate	Std. Error	t value
(Intercept)	265.75	7.67	34.7
materialB	-45.75	5.53	-8.3
materialC	-24.00	5.53	-4.3
materialD	-35.25	5.53	-6.4

Again, the p -values are gone. However, note that the large size of the t -statistics means that we can be confident that there are significant material effects here. This could be verified with an LRT with parametric bootstrap to estimate the p -value but is hardly necessary given the already convincing level of evidence.

8.8 Multilevel Models

The linear models analysis remains unchanged. The first difference occurs at the top of p177 where the ANOVA table becomes:

```
jspr <- jsp[jspr$year==2,]
mmod <- lmer(math ~ raven*social*gender+(1|school)+(1|school:class),data=jspr)
anova(mmod)
```

Analysis of Variance Table

	Df	Sum Sq	Mean Sq	F value
raven	1	10218	10218	374.40
social	8	616	77	2.82
gender	1	22	22	0.79
raven:social	8	577	72	2.64
raven:gender	1	2	2	0.09
social:gender	8	275	34	1.26
raven:social:gender	8	187	23	0.86

We no longer have the p -values. We can reconstruct these as:

```
round(pf(anova(mmod)[,4],anova(mmod)[,1],917,lower.tail=FALSE),4)
```

```
[1] 0.0000 0.0043 0.3738 0.0072 0.7639 0.2605 0.5524
```

The degrees of freedom for the denominator of 917 can be obtained by summing the degrees of freedom from the ANOVA table and subtracting an extra one for the intercept:

```
nrow(jspr)-sum(anova(mmod)[,1])-1
```

```
[1] 917
```

Now, as pointed out earlier, there is good reason to question the results of such F -tests. In this case, the nominal degrees of freedom is large. Given that the number of random effects is not particularly large, the “true” degrees of freedom will still be large. This suggests that these particular p -values will be fairly accurate.

Another possibility is to compute LRTs. For example, we can test the three-way interaction term by fitting the model with and without this term and computing the test:

```
mmod <- lmer(math ~ (raven*social*gender)^2+
             (1|school)+(1|school:class),data=jspr,REML=FALSE)
mmod2 <- lmer(math ~ (raven+social+gender)^2+
             (1|school)+(1|school:class),data=jspr,REML=FALSE)
anova(mmod,mmod2)
```

Data: jspr

Models:

```
mmod2: math ~ (raven + social + gender)^2 + (1 | school) + (1 | school:class)
```

```
mmod: math ~ (raven * social * gender)^2 + (1 | school) + (1 | school:class)
```

	Df	AIC	BIC	logLik	deviance	Chisq	Chi Df	Pr(>Chisq)
mmod2	31	5958	6108	-2948	5896			
mmod	39	5967	6156	-2944	5889	7.1	8	0.53

We notice that the p -value of 0.53 is quite similar to the 0.55 produced by the F -test. For larger datasets where the residual standard error is estimated fairly precisely, the denominator of the F -statistic has little variability so that the test statistic becomes close to chi-squared distributed, just like the LRT. They will not be numerically identical, but we might expect them to be close.

Implementing the parametric bootstrap to estimate the p -value is possible here:

```
nrep <- 1000
lrstat <- numeric(nrep)
for(i in 1:nrep){
  rmath <- unlist(simulate(mmod2))
  rmmod <- refit(mmod,rmath)
  rmmod2 <- refit(mmod2, rmath)
  lrstat[i] <- 2*(logLik(rmmod)-logLik(rmmod2))
}
2*(logLik(mmod)-logLik(mmod2))
```

```
'log Lik.' 7.095 (df=39)
```

```
mean(lrstat > 7.0954)
```

```
[1] 0.542
```

Unsurprisingly, given the sample size, the results are very similar to that obtained by the chi-squared approximation.

If you need to consider more than just two models and wish to select a model, it is typically better to use a criterion-based variable selection methods. Here we can use the AIC to select the model. The computation of the AIC does require the specification of the number of parameters, which could be problematic if we also consider the random effects parameters. However, if we only consider models where the fixed effect are different, the issue does not arise when comparing such models. We fit a sequence of such models here:

```
mmod <- lmer(math ~ raven*social*gender+(1|school)+(1|school:class),data=jspr)
AIC(mmod)
```

```
[1] 5970
```

```
mmod <- lmer(math ~ raven*social+(1|school)+(1|school:class),data=jspr)
AIC(mmod)
```

```
[1] 5963
```

```
mmod <- lmer(math ~ raven+social+(1|school)+(1|school:class),data=jspr)
AIC(mmod)
```

```
[1] 5950
```

```
mmod <- lmer(math ~ raven+(1|school)+(1|school:class),data=jspr)
AIC(mmod)
```

```
[1] 5966
```

```
mmod <- lmer(math ~ social+(1|school)+(1|school:class),data=jspr)
AIC(mmod)
```

```
[1] 6227
```

We see that the main effects model that uses `raven` and `social` gives the lowest AIC. We can examine this fit using the same centering of the Raven score as used in the book:

```
jspr$craven <- jspr$raven-mean(jspr$raven)
mmod <- lmer(math ~ craven+social+(1|school)+(1|school:class),jspr)
print(summary(mmod),correlation=FALSE)
```

```
Linear mixed model fit by REML ['lmerMod']
Formula: math ~ craven + social + (1 | school) + (1 | school:class)
Data: jspr
```

```
REML criterion at convergence: 5924
```

```
Scaled residuals:
```

```
   Min      1Q  Median      3Q      Max
-3.943 -0.548  0.147  0.631  2.853
```

```
Random effects:
```

Groups	Name	Variance	Std.Dev.
school:class	(Intercept)	1.03	1.02
school	(Intercept)	3.23	1.80
Residual		27.57	5.25

```
Number of obs: 953, groups: school:class, 90; school, 48
```

```
Fixed effects:
```

	Estimate	Std. Error	t value
(Intercept)	32.0108	1.0350	30.93
craven	0.5841	0.0321	18.21
social2	-0.3611	1.0948	-0.33
social3	-0.7768	1.1649	-0.67
social4	-2.1197	1.0396	-2.04
social5	-1.3632	1.1585	-1.18
social6	-2.3703	1.2330	-1.92
social7	-3.0482	1.2703	-2.40
social8	-3.5473	1.7027	-2.08
social9	-0.8864	1.1031	-0.80

Now that there are only main effects without interactions, the interpretation is simpler but essentially similar to that seen in the book. We do not have p -values in the table of coefficients. However, the sample size is large here and so a normal approximation could be used to compute reasonable p -values:

```
tval <- summary(mmod)$coef[,3]
pval <- 2*pnorm(abs(tval),lower=FALSE)
cbind(summary(mmod)$coef,"p value"=round(pval,4))
```

```
Estimate Std. Error t value p value
```

(Intercept)	32.0108	1.03499	30.9285	0.0000
craven	0.5841	0.03208	18.2053	0.0000
social2	-0.3611	1.09477	-0.3298	0.7415
social3	-0.7768	1.16489	-0.6668	0.5049
social4	-2.1197	1.03963	-2.0389	0.0415
social5	-1.3632	1.15850	-1.1767	0.2393
social6	-2.3703	1.23302	-1.9224	0.0546
social7	-3.0482	1.27027	-2.3997	0.0164
social8	-3.5473	1.70273	-2.0833	0.0372
social9	-0.8864	1.10314	-0.8035	0.4217

Of course, there are the usual concerns with multiple comparisons for the nine-level factor, `social`. The reference level is social class I and we can see significant differences between this level and levels IV, VII and VIII.

When testing the compositional effects, we need to make two changes. Firstly, we have decided not to have the interaction between Raven score and social class, consistent with the analysis above. Secondly, we cannot use the F -test to make the comparison. We replace this with an LRT:

```
schraven <- lm(raven ~ school, jspr)$fit
mmod <- lmer(math ~ craven+social+(1|school)+(1|school:class),jspr, REML=FALSE)
mmodc <- lmer(math ~ craven+social+schraven+(1|school)+
              (1|school:class), jspr,REML=FALSE)
anova(mmod,mmodc)
```

Data: jspr

Models:

mmod: math ~ craven + social + (1 | school) + (1 | school:class)

mmodc: math ~ craven + social + schraven + (1 | school) + (1 | school:class)

	Df	AIC	BIC	logLik	deviance	Chisq	Chi	Df	Pr(>Chisq)
mmod	13	5954	6018	-2964	5928				
mmodc	14	5956	6024	-2964	5928	0.18	1		0.67

As before, we do not find any compositional effects.

Revisions to Chapter 9

9.1 Longitudinal Data

On p186, the function `lmList()` in `lme4`, can be used for the computation of linear models on groups within the data. For now, the computation in the book is simpler.

On p189, the output of the model becomes:

```
psid$cyear <- psid$year-78
mmod <- lmer(log(income) ~ cyear*sex +age+educ+(cyear|person),psid)
print(summary(mmod),correlation=FALSE)
```

Linear mixed model fit by REML ['lmerMod']

Formula: log(income) ~ cyear * sex + age + educ + (cyear | person)

Data: psid

REML criterion at convergence: 3820

Scaled residuals:

Min	1Q	Median	3Q	Max
-10.231	-0.213	0.079	0.415	2.825

Random effects:

Groups	Name	Variance	Std.Dev.	Corr
person	(Intercept)	0.2817	0.531	
	cyear	0.0024	0.049	0.19
Residual		0.4673	0.684	

Number of obs: 1661, groups: person, 85

Fixed effects:

	Estimate	Std. Error	t value
(Intercept)	6.6742	0.5433	12.28
cyear	0.0853	0.0090	9.48
sexM	1.1503	0.1213	9.48
age	0.0109	0.0135	0.81
educ	0.1042	0.0214	4.86
cyear:sexM	-0.0263	0.0122	-2.15

The omission of p -values is noted. Given the sample size, the normal approximation for the computation of p -values for the t -statistics would be acceptable.

9.2 Repeated Measures

The model output on p193 becomes:

```
mmod <- lmer(acuity~power+(1|subject)+(1|subject:eye),vision)
print(summary(mmod),correlation=FALSE)
```

Linear mixed model fit by REML ['lmerMod']

Formula: acuity ~ power + (1 | subject) + (1 | subject:eye)

Data: vision

REML criterion at convergence: 328.7

Scaled residuals:

Min	1Q	Median	3Q	Max
-3.424	-0.323	0.011	0.441	2.466

Random effects:

Groups	Name	Variance	Std.Dev.
subject:eye	(Intercept)	10.3	3.21
subject	(Intercept)	21.5	4.64
Residual		16.6	4.07

Number of obs: 56, groups: subject:eye, 14; subject, 7

Fixed effects:

	Estimate	Std. Error	t value
(Intercept)	112.643	2.235	50.4

power6/18	0.786	1.540	0.5
power6/36	-1.000	1.540	-0.6
power6/60	3.286	1.540	2.1

The omission of p -values is noted. The ANOVA table becomes:

```
anova(mmod)
```

```
Analysis of Variance Table
      Df Sum Sq Mean Sq F value
power  3   141    46.9    2.83
```

We would like to know whether the power is statistically significant but no longer have the p -value from F -statistic available. We can use the LRT and parametric bootstrap as follows:

```
nrep <- 1000
mmod <- lmer(acuity~power+(1|subject)+(1|subject:eye),vision,REML=FALSE)
nmod <- lmer(acuity~1+(1|subject)+(1|subject:eye),vision,REML=FALSE)
as.numeric(2*(logLik(mmod)-logLik(nmod)))
```

```
[1] 8.262
```

```
pchisq(8.2625,3,lower=FALSE)
```

```
[1] 0.04089
```

```
lrstat <- numeric(nrep)
for(i in 1:nrep){
  racuity <- unlist(simulate(nmod))
  rnull <- refit(nmod,racuity)
  ralt <- refit(mmod, racuity)
  lrstat[i] <- as.numeric(2*(logLik(ralt)-logLik(rnull)))
}
(pval <- mean(lrstat > 8.2625))
```

```
[1] 0.057
```

Using the chi-squared approximation gives a p -value of 0.041 while the parametric bootstrap gives 0.057. These are close to the p -value of 0.048 from the F -statistic. Thus the result is borderline.

We can repeat the calculations for when the 43rd observation is omitted:

```
mmodr <- lmer(acuity~power+(1|subject)+(1|subject:eye),vision,subset=-43)
anova(mmodr)
```

```
Analysis of Variance Table
      Df Sum Sq Mean Sq F value
power  3   89.2    29.8    3.6
```

Again we lack the p -values we had before. However, we do have sufficient sample size to conclude that a t -statistic of 3 is sufficient to indicate the significance of the highest power relative to the baseline. The same would be true for the calculations based on the Helmert contrasts.

8.3 Multiple Response Multilevel Models

The ANOVA table at the top of page 197 can be reconstructed as follows:

```
jspr <- jsp[jspr$year==2,]
mjspr <- data.frame(rbind(jspr[,1:6],jspr[,1:6]),
                   subject=factor(rep(c("english","math"),c(953,953))),
                   score=c(jspr$english/100,jspr$math/40))
mjspr$craven <- mjspr$raven-mean(mjspr$raven)
mmod <- lmer(score ~ subject*gender+craven*subject+social+ (1|school)+(1|school:class)+(1|school:class:
sigmaerr <- attributes(VarCorr(mmod))$sc
(fstat <- anova(mmod)[,3]/sigmaerr^2)
```

```
[1] 3953.673    7.464  444.628    6.469   28.225   15.987
```

```
nrow(mjspr)-sum(anova(mmod)[,1])-1
```

```
[1] 1892
```

```
(pvals <- pf(fstat,anova(mmod)[,1],1892,lower.tail=FALSE))
```

```
[1] 0.000e+00 6.352e-03 8.023e-89 2.470e-08 1.207e-07 6.624e-05
```

```
cbind(anova(mmod),fstat,pvalue=round(pvals,3))
```

	Df	Sum Sq	Mean Sq	F value	fstat	pvalue
subject	1	53.7368	53.73683	3953.673	3953.673	0.000
gender	1	0.1015	0.10145	7.464	7.464	0.006
craven	1	6.0432	6.04322	444.628	444.628	0.000
social	8	0.7034	0.08792	6.469	6.469	0.000
subject:gender	1	0.3836	0.38363	28.225	28.225	0.000
subject:craven	1	0.2173	0.21728	15.987	15.987	0.000

In this case, there are a large number of degrees of freedom for the error and the approximation will be good here, just as in the analysis of this data in the previous chapter. In any case, the interaction terms are clearly significant. The subsequent model summary will lack p -values but these are not necessary for our interpretation. If we wanted them, a normal approximation would suffice.

References

- Bates, Douglas. 2005. "Fitting Linear Mixed Models in R." *R News* 5 (1): 27–30. <http://CRAN.R-project.org/doc/Rnews/>.
- Bates, Douglas, Martin Maechler, Ben Bolker, and Steve Walker. 2014. "Fitting Linear Mixed-Effects Models Using lme4." *ArXiv* 1406.5823 (June): 1–51. <http://arxiv.org/abs/1406.5823>.
- Crainiceanu, C., and D. Ruppert. 2004. "Likelihood Ratio Tests in Linear Mixed Models with One Variance Component." *Journal of the Royal Statistical Society, Series B* 66: 165–85.
- Faraway, J. 2005. *Linear Models with R*. London: Chapman; Hall.

———. 2006. *Extending the Linear Model with R*. London: Chapman; Hall.

Kenward, MG, and JH Roger. 1997. “Small Sample Inference for Fixed Effects from Restricted Maximum Likelihood.” *Biometrics* 53 (3): 983–97.

Scheffe, H. 1959. *The Analysis of Variance*. New York: Wiley.