Mixed Model Analysis

The purpose of this TP is to carry out some mixed model analyses using **R**. As usual, make sure that you read the **help** for any new functions that you use.

Corn Data (revisited)

0. The corn data is called **ant111b** in the **DAAG** package. In the course we looked at modeling harvest weight **harvwt** of corn; now, we will do the same type of modeling but this time with the **ears** variable as the outcome.

We will also be using the packages **lme4** and **lattice** in the analyses. To load these packages and learn about the data, type

```
library(lme4)
library(lmerTest)
library(lattice)
library(DAAG)
?ant111b
```

Now set the class of model objects so that we can obtain *p*-values for the fixed effects coefficients:

```
merModLmerTest <- setClass("merModLmerTest",contains = c("merMod", "lmerMod"))
```

1. Start to explore the data:

```
str(ant111b)
summary(ant111b)
```

You can make a dotplot of the number of ears by site, sorting by mean **ears** as follows:

```
dotplot(reorder(site, ears) ~ ears, ant111b, xlab = "Number of ears of corn",
      ylab = "Site", pch = 19, aspect = 0.32, type = c("p", "a"))
```

Comment on your plot - do any effects seem to contribute to the variation in **ears**?

2. Now we will use **lmer** to fit a random effects model of the form $y_{ij} = \mu + b_i + \varepsilon_{ij}$, and look at the results. You can use extractor methods (e.g. **fixef**, **ranef**, **fitted**, etc.) to extract components from the resulting **mer** object.

```
(ears.lmer <- lmer(ears ~ 1 + (1 | site), data=ant111b))
```

There are two sources of random variation, one for site and one for parcel within site (residual), each with an estimated variance (and SD).

a. Find the grand mean.
b. Make a table showing the sample mean and fitted value for each site (there is sample code in the slides from the lecture). Note that the fitted values are not just the sample means, but are between the grand mean and individual group sample means.
c. Give the estimated variance for each source of variation, σ^2_{site} and $\sigma^2_{Residual}$. Which source of variation is larger? What proportion of variation is due to differences between sites?
d. Make a caterpillar plot for the random effects. Does the plot support your conclusion about the source of variation? Which site(s) are most 'unusual'?

```
# Caterpillar plot:
dotplot(ranef(ears.lmer, condVar = TRUE), strip = FALSE)[[1]]
```

3. Use the function **update** with option **REML = FALSE** to re-fit the model to obtain ML, rather than REML, estimates. How do the estimates compare?

4. It is also a good idea to check the model assumptions with a few diagnostic plots.

a. There should not be any apparent pattern in the residuals. You can check this by making a plot of residuals versus fitted values:

```
plot(fitted(ears.lmer), residuals(ears.lmer), main="residual plot", pch=19)
abline(h=0, lty=2)
```

b. The residuals should also be normally distributed. You can check this by making a normal quantile-quantile (QQ) plot. If the points fall along a straight line, the distribution is approximately normal.

```
qqnorm(resid(ears.lmer))
qqline(resid(ears.lmer))
```

What do you conclude?

5. Now we will consider differences between parcels under a variety of conditions.

a. Suppose that there was also a parcel V at site WLAN where the data are not recorded, but the corn was grown under the same conditions. Estimate the number of ears of corn in this parcel. Include a standard error with the estimate. we'd use the mean of the site, and the variance of the parcels (i.e. 'Residual')
b. Suppose there was also a parcel VI at the same site as parcel V. What is the standard deviation of the estimated difference in the number of ears of corn in parcels V and VI? the std error of a difference is the sqrt of the sum of variances
c. What is the standard deviation of the difference in numbers of ears of corn between two parcels at a *new* site on the island?
variances of sites and of parcels, and the grand mean

Rat Brain Data

0. The rat brain data is called **rat.brain** in the **WWGbook** package. We want to examine the effect of **treatment** on the response variable **activate**, while also taking into account **region** and **animal**. To learn about the data, type

```
library(WWGbook)
?rat.brain
```

1. Start to explore the data:

```
attach(rat.brain)
str(rat.brain)
summary(rat.brain)
```

In order to use **treatment** and **region** correctly in the model, they will each need to be coded as a **factor** (what type of variables are they now?):

```
region.f <- region
region.f[region == 1] <- 1
region.f[region == 2] <- 2
region.f[region == 3] <- 0
region.f <- factor(region.f)
levels(region.f) <- c("VST", "BST", "LS")

treat <- factor(treatment)
levels(treat) <- c("Basal","Carbachol")

rat.brain <- data.frame(rat.brain, region.f, treat)

str(rat.brain)
summary(rat.brain)
```

First try to get some idea what the data look like through graphical exploration. Here are a few different representations of the data. Try them all out - which do you think is most revealing?

```
dotplot(reorder(animal, activate) ~ activate, rat.brain,
      groups = region.f, ylab = "Animal", xlab = "Activate", pch=19,
      type = c("p", "a"), auto.key=list(columns=3, lines=TRUE))
```

Here we have plotted results for each rat (ordered by increasing mean(activate), but this includes both treatment measurements for each rat. Let's look at each rat/treatment combination separately:

```
rat.brain$rt <- with(rat.brain, treat:factor(animal))
dotplot(reorder(rt, activate) ~ activate, rat.brain, groups = region.f,
      ylab = "Animal", xlab = "Activate", pch=19,
      type = c("p", "a"), auto.key=list(columns=3, lines=TRUE))

# Each rat separately:
xyplot(activate ~ treat | animal, rat.brain, aspect = "xy", layout = c(5,1),
      groups=region.f, pch=19, type=c("p", "l", "g"),
      index.cond = function(x,y) coef(lm(y~x))[1], xlab = "Treatment",
      ylab="Activate", auto.key=list(space="top",lines=TRUE,columns=3))

# Separated by treatment group:
xyplot(activate ~ region.f|treat, rat.brain, groups = animal, pch=19,
      ylim=c(0,800), xlab="Region", ylab="Activate",
      type = c("p","a"), auto.key = list(space="top"))
```

Does the treatment appear to have an effect? Why do you say that? Does the effect (if any) appear to be the same in each region? Do there appear to be rat-specific effects?

2. We will start off fitting a model including all fixed effects (main effects and interactions for the treatment and region variables - make sure to use the factor versions) and a random effect for animal. As above, you can use extractor functions to view some of the model components.

```
(rat.brain.lmer1 <- lmer(activate ~ region.f*treat + (1|animal), REML=TRUE, data = rat.brain))
```

Make sure that you know how to interpret the coefficients (the interpretation will be determined by the coding).

3. From the plot above, we saw that between-animal variation was greater for the carbachol treatment than for the basal treatment. To accommodate this difference in variation, we can add a random animal-specific effect of treatment to the model. The effect of treatment is fixed in our original model, therefore constant across all animals. The additional random effect associated with treatment that we include in the new model allows the implied marginal variance of observations for the carbachol treatment to differ from that for the basal treatment. (We can also think of the new model as having two random intercepts per rat, one for the carbachol treatment and an additional one for the basal treatment.)

```
(rat.brain.lmer2 <- lmer(activate ~ region.f*treat + (treat |animal), REML=TRUE, data =
rat.brain))
```

What happens to the estimated fixed effects coefficients? What about their standard errors?

4. We can compare the models using a likelihood ratio (LR) test, carried out with the **anova** function. The **anova** method for **mer** objects carries out a ML (not REML) LR test, even if the model has been fit by REML. The results are not identical for the two methods, but in this case the conclusions are the same. We can also test for the presence of random effects (*i.e.* = 0 or not) using the **rand** function:

```
anova(rat.brain.lmer1, rat.brain.lmer2)
rand(rat.brain.lmer1)
rand(rat.brain.lmer2)
```

Here, there are 2 parameters that are different in the null and alternative models: the variance of the random treatment effects and the covariance of the two random effects (intercept and treatment). The other parameter, the variance of the random intercepts, is retained in both models. Again because of the boundary condition, the (asymptotic) distribution of -2*LR is not χ^2 with 2 df, but is a 50-50

mixture of a χ^2 with 1 df and a χ^2 with 2 df. (Ask me if you really want to know why!) So the p-value will be conservative in this respect (although generally also anti-conservative due to the small sample). The p-value here is very highly significant in any case, so we would reject the null and retain the additional parameters in the model.

The χ^2 distribution relies on asymptotic (large-sample) theory, so we would not usually carry out this type of test (or even fit a model with this many parameters!) for such a small data set (only 5 rats). Rather, in practice, the random effects would probably be retained without testing, so that the appropriate marginal variance-covariance structure would be obtained for the data set. We have looked at these as an illustration and for a little practice in carrying out and interpreting the test results. A different approach to analysis of these data would be to fit a model on the treatment differences for each rat.

a. Which fixed effects are significant? Should any be deleted from the model?
b. Do the variance components appear to be bigger than 0?

5. As above, we check some diagnostics for the final model.

```
# Residual plot:
fit <- fitted(rat.brain.lmer2)
res <- resid(rat.brain.lmer2)
plotres.fit <- data.frame(rat.brain, fit, res)
xyplot(res ~ fit, data=plotres.fit, groups=treat, pch=19, xlab="Predicted value",
      ylab="Residual", abline=0, auto.key=list(space="top", columns=2))

# QQ normal plot:
qqnorm(resid(rat.brain.lmer2))
qqline(resid(rat.brain.lmer2))
```

What do you conclude?

6. **(Optional)** If you want a little more explanation about the technical difficulties with *p*-values in these models, read:

```
?pvalues
```