Introduction to statistics: Linear mixed models

Shravan Vasishth

Universität Potsdam
vasishth@uni-potsdam.de
http://www.ling.uni-potsdam.de/~vasishth

February 22, 2020
Summary

1. We know how to do simple t-tests.
2. We know how to fit simple linear models.
3. We saw that the paired t-test is identical to the varying intercepts linear mixed model.

Now we are ready to look at linear mixed models in detail.
Returning to our SR/OR relative clause data from English (Grodner and Gibson, Expt 1). First we load the data as usual (not shown).

```r
gg1crit <- read.table("data/grodnergibson05data.txt", header = TRUE)

gg1crit$so <- ifelse(gg1crit$condition == "objgap", 1, -1)

dat <- gg1crit
dat$logrt <- log(dat$rawRT)

bysubj <- aggregate(logrt ~ subject + condition, mean, data = dat)
```
Linear models

The simple linear model (incorrect for these data):

```r
summary(m0<-lm(logrt~so,dat))$coefficients
```

|            | Estimate | Std. Error | t value | Pr(>|t|)       |
|------------|----------|------------|---------|----------------|
| (Intercept)| 5.88306 | 0.019052   | 308.78  | 0.000000000    |
| so         | 0.06202 | 0.019052   | 3.2551  | 0.0011907      |
Linear models

We can visualize the different responses of subjects (four subjects shown):
Linear models

Given these differences between subjects, you could fit a separate linear model for each subject, collect together the intercepts and slopes for each subject, and then check if the intercepts and slopes are significantly different from zero.

*We will fit the model using log reading times because we want to make sure we satisfy model assumptions (e.g., normality of residuals).*
Linear models

There is a function in the package lme4 that computes separate linear models for each subject: \texttt{lmList}.

\begin{verbatim}
library(lme4)

## Loading required package: Matrix

lmList.fm1 <- lmList(logrt ~ so | subject, dat)
\end{verbatim}
Linear models

Intercept and slope estimates for three subjects:

\texttt{lmlist.fm1$`1`$coefficients}

\begin{verbatim}
## (Intercept) so
## 5.769617 0.043515
\end{verbatim}

\texttt{lmlist.fm1$`28`$coefficients}

\begin{verbatim}
## (Intercept) so
## 6.10021 0.44814
\end{verbatim}

\texttt{lmlist.fm1$`37`$coefficients}

\begin{verbatim}
## (Intercept) so
## 6.61699 0.35537
\end{verbatim}
Linear models

One can plot the individual lines for each subject, as well as the linear model m0’s line (this shows how each subject deviates in intercept and slope from the model m0’s intercept and slopes).
Linear models
Linear models

To find out if there is an effect of RC type, you can simply check whether the slopes of the individual subjects’ fitted lines taken together are significantly different from zero.
Linear models

```r
t.test(coef(lmlist.fm1)[2])
```

##
## One Sample t-test
##
## data:  coef(lmlist.fm1)[2]
## t = 2.81, df = 41, p-value = 0.0076
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
## 0.017449 0.106585
## sample estimates:
## mean of x
## 0.062017
Linear models

The above test is exactly the same as the paired t-test and the varying intercepts linear mixed model on aggregated data:

\[
t.\text{test}(\text{logrt} \sim \text{condition}, \text{bysubj}, \text{paired} = \text{TRUE})\text{\$statistic}
\]

```
##      t
## 2.8102
```

\# also compare with linear mixed model:
\[
\text{summary(lmer(\text{logrt} \sim \text{condition} + (1 | \text{subject}), \text{bysubj}))\text{\$coefficients}[2,]}
\]

```
## Estimate Std. Error t value
## -0.124033  0.044137 -2.810207
```
Linear models

- The above lmList model we fit is called repeated measures regression. We now look at how to model unaggregated data using the linear mixed model.

- This model is now only of historical interest, and useful only for understanding the linear mixed model, which is the modern standard approach.
Linear mixed models

- The **linear mixed model** does something related to the above by-subject fits, but with some crucial twists, as we see below.

- In the model shown in the next slide, the statement (1|subject) adjusts the grand mean estimates of the intercept by a term (a number) for each subject.
Notice that we did not aggregate the data here.

```r
m0.lmer <- lmer(logrt ~ so + (1 | subject), dat)
```

Abbreviated output:

**Random effects:**
- Groups: `subject`  
  - Name: (Intercept)  
    - Variance: 0.09983  
    - Std. Dev.: 0.3160  
  - Residual:  
    - Variance: 0.14618  
    - Std. Dev.: 0.3823  

Number of obs: 672, groups: subject, 42

**Fixed effects:**
- `(Intercept)`  
  - Estimate: 5.88306  
  - Std. Error: 0.05094  
  - t value: 115.497
- `so`  
  - Estimate: 0.06202  
  - Std. Error: 0.01475  
  - t value: 4.205
Linear mixed models

One thing to notice is that the coefficients (intercept and slope) of the fixed effects of the above model are identical to those in the linear model m0 above. The varying intercepts for each subject can be viewed by typing:

```r
ranef(m0.lmer)$subject[,1][1:10]
```

```
## [1] -0.1039283  0.0771948 -0.2306209  0.2341978  0.0088279 -0.0953633
## [7] -0.2055713 -0.1553708  0.0759436 -0.3643671
```

```r
ranef(m0.lmer)$subject[,7][1:10]
```

```
## [1] -0.2055713 -0.1553708  0.0759436 -0.3643671
## [7] -0.2055713 -0.1553708  0.0759436 -0.3643671
```
Visualizing random effects

Here is another way to summarize the adjustments to the grand mean intercept by subject. The error bars represent 95% confidence intervals.

```r
library(lattice)
print(dotplot(ranef(m0.lmer, condVar=TRUE)))
```
Visualizing random effects

## $subject
Linear mixed models

The model m0.lmer above prints out the following type of linear model. \( i \) indexes subject, and \( j \) indexes items. Once we know the subject id and the item id, we know which subject saw which condition:

\[
y_{ij} = \beta_0 + u_{0i} + \beta_1 \times so_{ij} + \epsilon_{ij} \quad (1)
\]

The \textbf{only} new thing here is the by-subject adjustment to the intercept.
Linear mixed models

- Note that these by-subject adjustments to the intercept $u_{0i}$ are assumed by lmer to come from a normal distribution centered around 0:
  \[ u_{0i} \sim \text{Normal}(0, \sigma_{u0}) \]

- The ordinary linear model m0 has one intercept $\beta_0$ for all subjects, whereas the linear mixed model with varying intercepts m0.lmer has a different intercept \((\beta_0 + u_{0i})\) for each subject $i$.

- We can visualize the adjustments for each subject to the intercepts as shown below.
Linear mixed models
Formal statement of varying intercepts linear mixed model

i indexes subjects, j items.

\[ y_{ij} = \beta_0 + u_{0i} + (\beta_1) \times s_{oi} + \epsilon_{ij} \] 

(2)

Variance components:

- \( u_0 \sim \text{Normal}(0, \sigma_{u0}) \)
- \( \epsilon \sim \text{Normal}(0, \sigma) \)
Linear mixed models

Note that, unlike the figure associated with the lmlist.fm1 model above, which also involves fitting separate models for each subject, the model m0.lmer assumes **different intercepts** for each subject **but the same slope**. We can have lmer fit different intercepts AND slopes for each subject.
Linear mixed models

Varying intercepts and slopes by subject

We assume now that each subject’s slope is also adjusted:

\[ y_{ij} = \beta_0 + u_{0i} + (\beta_1 + u_{1i}) \times so_{ij} + \epsilon_{ij} \]  

That is, we additionally assume that \( u_{1i} \sim Normal(0, \sigma_{u1}) \).

\[
m1.lmer<-lmer(logrt~so+(1+so||subject),dat)
\]

Random effects:

<table>
<thead>
<tr>
<th>Groups</th>
<th>Name</th>
<th>Variance</th>
<th>Std.Dev.</th>
</tr>
</thead>
<tbody>
<tr>
<td>subject</td>
<td>(Intercept)</td>
<td>0.1006</td>
<td>0.317</td>
</tr>
<tr>
<td>subject.1</td>
<td>so</td>
<td>0.0121</td>
<td>0.110</td>
</tr>
<tr>
<td>Residual</td>
<td></td>
<td>0.1336</td>
<td>0.365</td>
</tr>
</tbody>
</table>

Number of obs: 672, groups: subject, 42
Linear mixed models

These fits for each subject are visualized below (the red line shows the model with a single intercept and slope, i.e., our old model m0):

varying intercepts and slopes for each subject
Linear mixed models

Comparing \textit{lmList} model with varying intercepts model

Compare this model with the \textit{lmList.fm1} model we fitted earlier:

\begin{itemize}
  \item \textbf{ordinary linear model}
  \item \textbf{varying intercepts and slopes}
\end{itemize}
Visualizing random effects

```r
print(dotplot(ranef(m1.lmer, condVar=TRUE)))
```
Lecture 6

Linear mixed models

Model type 2: Varying intercepts and slopes model (no correlation)

Visualizing random effects

```
## $subject
```

![Graph showing random effects visualization](image-url)
Formal statement of varying intercepts and varying slopes linear mixed model

\(y_{ij} = \beta_0 + u_{0i} + (\beta_1 + u_{1i}) \times s_{ij} + \epsilon_{ij}\)  \hspace{1cm} (4)

Variance components:

- \(u_0 \sim Normal(0, \sigma_{u0})\)
- \(u_1 \sim Normal(0, \sigma_{u1})\)
- \(\epsilon \sim Normal(0, \sigma)\)
Shrinkage in linear mixed models

▶ The estimate of the effect by participant is smaller than when we fit a separate linear model to the subject’s data.
▶ This is called shrinkage in linear mixed models: the individual level estimates are shunk towards the mean slope.
▶ The less data we have from a given subject, the more the shrinkage.
Shrinkage in linear mixed models

Subject 28’s estimates

Condition

Subject 36’s estimates

Condition

Subject 37’s estimates

Condition
Shrinkage in linear mixed models

The effect of missing data on estimation in LMMs

Let’s randomly delete some data from one subject:

```r
set.seed(4321)
## choose some data randomly to remove:
rand<-rbinom(1,n=16,prob=0.5)
```
Shrinkage in linear mixed models

The effect of missing data on estimation in LMMs

```r
dat[which(dat$subject==37),]$rawRT

## [1]  770  536  686  578  457  487 2419  884 3365  233
## [15] 1081  971

dat$deletedRT <- dat$rawRT

dat[which(dat$subject==37),]$deletedRT <- ifelse(rand, NA, dat[which(dat$subject==37),]$rawRT)
```
Shrinkage in linear mixed models

The effect of missing data on estimation in LMMs

Now subject 37’s estimates are going to be pretty wild:

```r
subset(dat, subject==37)$deletedRT
```

```
## [1] 770  NA  686  578  NA   NA   NA   NA   NA 3365  233
## [15] NA  971
```
Shrinkage in linear mixed models

The effect of missing data on estimation in LMMs

```r
## original no pooling estimate:
lmList.fm1_old <- lmList(log(rawRT) ~ so | subject, dat)
coefs_old <- coef(lmList.fm1_old)
intercepts_old <- coefs_old[1]
colnames(intercepts_old) <- "intercept"
slopes_old <- coefs_old[2]

## subject 37's original estimates:
intercepts_old$intercept[37]

## [1] 6.617

slopes_old$so[37]

## [1] 0.35537
```
Shrinkage in linear mixed models

The effect of missing data on estimation in LMMs

```r
## on deleted data:
lmList.fm1_deleted <- lmList(log(deletedRT) ~ so | subject, dat)
coefs <- coef(lmList.fm1_deleted)
intercepts <- coefs[1]
colnames(intercepts) <- "intercept"
slopes <- coefs[2]
## subject 37's new estimates on deleted data:
intercepts$intercept[37]
```

```
## [1] 6.6879
```

```
slopes$so[37]
```

```
## [1] 0.38843
```
Shrinkage in linear mixed models

The effect of missing data on estimation in LMMs

Subject 37's estimates
Shrinkage in linear mixed models

The effect of missing data on estimation in LMMs

- What we see here is that the estimates from the hierarchical model are barely affected by the missingness, but the estimates from the no-pooling model are heavily affected.

- This means that linear mixed models will give you more robust estimates (think Type M error!) compared to no pooling models.

- This is one reason why linear mixed models are such a big deal.
Crossed subjects and items in LMMs

Subjects and items are fully crossed:

```
head(xtabs(~subject+item,dat))
```

```
## item
## subject 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
## 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
## 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
## 3 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
## 4 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
## 5 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
## 6 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
```
Linear mixed models

Linear mixed model with crossed subject and items random effects.

```r
m2.lmer <- lmer(logrt ~ so + (1 + so || subject) + (1 + so || item), dat)
```
Linear mixed models

Random effects:

<table>
<thead>
<tr>
<th>Groups</th>
<th>Name</th>
<th>Variance</th>
<th>Std. Dev.</th>
</tr>
</thead>
<tbody>
<tr>
<td>subject</td>
<td>(Intercept)</td>
<td>0.10090</td>
<td>0.3177</td>
</tr>
<tr>
<td>subject.1</td>
<td>so</td>
<td>0.01224</td>
<td>0.1106</td>
</tr>
<tr>
<td>item</td>
<td>(Intercept)</td>
<td>0.00127</td>
<td>0.0356</td>
</tr>
<tr>
<td>item.1</td>
<td>so</td>
<td>0.00162</td>
<td>0.0402</td>
</tr>
<tr>
<td>Residual</td>
<td></td>
<td>0.13063</td>
<td>0.3614</td>
</tr>
</tbody>
</table>

Number of obs: 672, groups: subject, 42; item, 16

Fixed effects:

<table>
<thead>
<tr>
<th>Estimate</th>
<th>Std. Error</th>
<th>t value</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Intercept)</td>
<td>5.8831</td>
<td>0.0517</td>
<td>113.72</td>
</tr>
<tr>
<td>so</td>
<td>0.0620</td>
<td>0.0242</td>
<td>2.56</td>
</tr>
</tbody>
</table>
Visualizing random effects
Visualizing random effects
Linear mixed models

Linear mixed model with crossed subject and items random effects.

m3.lmer<-lmer(logrt~so+(1+so|subject)+(1+so|item), dat)

## boundary (singular) fit: see ?isSingular
Linear mixed models

Linear mixed model with crossed subject and items random effects.

Random effects:

Groups | Name      | Variance | Std.Dev. | Corr
--------|-----------|----------|----------|------
subject | (Intercept) | 0.10103  | 0.3178   |      
so      | 0.01228  | 0.1108   | 0.58     |
item    | (Intercept) | 0.00172  | 0.0415   |      
so      | 0.00196  | 0.0443   | 1.00 <= degenerate |
Residual|          | 0.12984  | 0.3603   |      |

Number of obs: 672, groups: subject, 42; item, 16

Fixed effects:

<table>
<thead>
<tr>
<th>Estimate</th>
<th>Std. Error</th>
<th>t value</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Intercept)</td>
<td>5.8831</td>
<td>0.0520</td>
</tr>
<tr>
<td>so</td>
<td>0.0620</td>
<td>0.0247</td>
</tr>
</tbody>
</table>
Formal statement of varying intercepts and varying slopes linear mixed model with correlation

\[ y_{ij} = \alpha + u_0 + w_0 + (\beta + u_1 + w_1) \cdot s_{ij} + \varepsilon_{ij} \quad (5) \]

where \( \varepsilon_{ij} \sim \text{Normal}(0, \sigma) \) and

\[
\Sigma_u = \begin{pmatrix}
\sigma_{u0}^2 & \rho_u \sigma_{u0} \sigma_{u1} \\
\rho_u \sigma_{u0} \sigma_{u1} & \sigma_{u1}^2
\end{pmatrix}
\]

\[
\Sigma_w = \begin{pmatrix}
\sigma_{w0}^2 & \rho_w \sigma_{w0} \sigma_{w1} \\
\rho_w \sigma_{w0} \sigma_{w1} & \sigma_{w1}^2
\end{pmatrix}
\]

\[
\begin{pmatrix} u_0 \\ u_1 \end{pmatrix} \sim \mathcal{N} \left( \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \Sigma_u \right), \quad \begin{pmatrix} w_0 \\ w_1 \end{pmatrix} \sim \mathcal{N} \left( \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \Sigma_w \right)
\quad (6)
\]
Visualizing random effects
Visualizing random effects

These are degenerate estimates