Multilevel modelling

Daniel Walther

29 juni 2015

Introduction

One of the key data manipulation tasks that must be accomplished prior to estimating several of the multilevel models (specifically contextual models and random coefficient models) is that group-level variables must be "assigned down" to the individual. To make a dataframe containing both individual and group-level variables, one typically begins with two separate dataframes. One dataframe contains individual-level data, and the other dataframe contains group-level data. By combining these two dataframes using a group identifying variable common to both, one is able to create a single data set containing both individual and group data.

Fixed effect in this context is the same coefficient for all subjects. *Random effect* is an effect that varies from group to group - this is usually just as simple as a specific intersept for the group.

The random effects give structure to the error term - if we know that errors are correlated in groups, and thus non-random, we should model this insight.

In R you write (1|group) to show that there should be a specific intercept for each group. You can add another, third, level in the same way: (1|group1) + (1|group2).

These are called mixed models since they have both fixed and random effects.

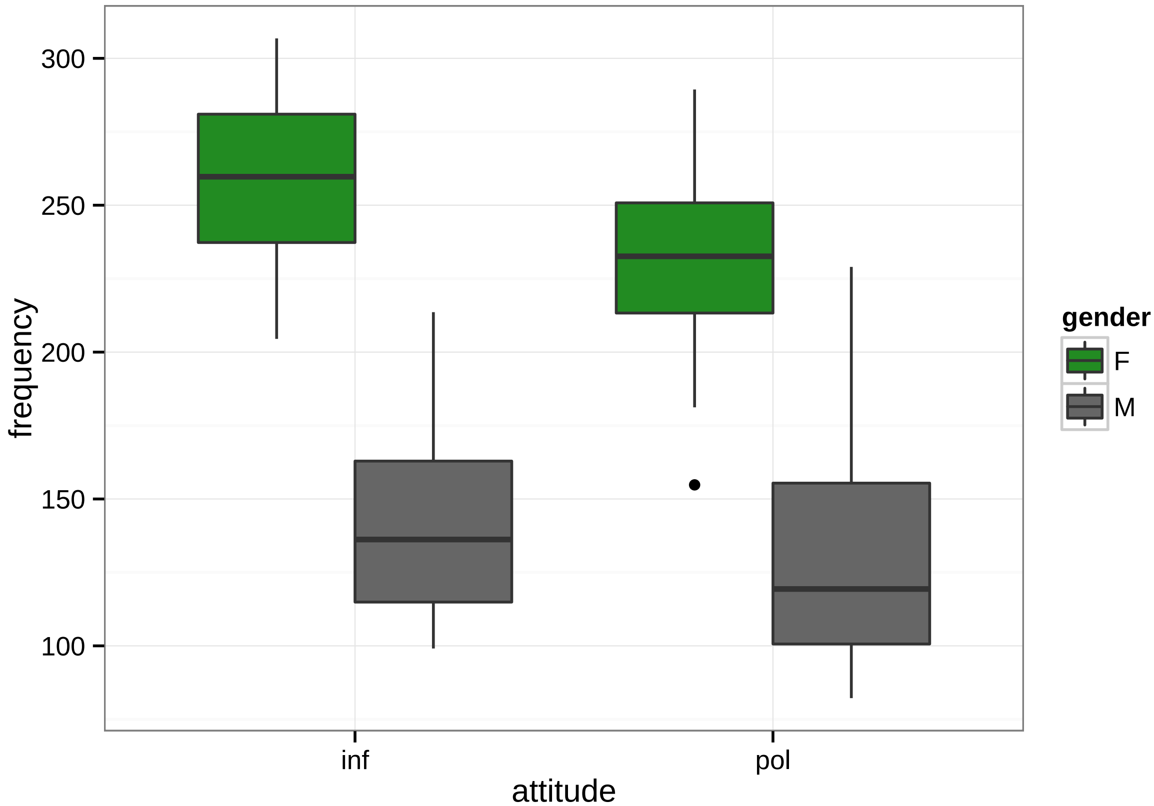
Examples in R

Look at data

```
library(dplyr)
library(ggplot2)

politeness = read.csv("http://www.bodowinter.com/tutorial/politeness_data.csv")

head(politeness)
```

```
##   subject gender scenario attitude frequency
## 1      F1      F        1      pol     213.3
## 2      F1      F        1      inf     204.5
## 3      F1      F        2      pol     285.1
## 4      F1      F        2      inf     259.7
## 5      F1      F        3      pol     203.9
## 6      F1      F        3      inf     286.9
```

```
ggplot(politeness, aes(x=attitude, y = frequency,
                       fill = gender)) +
  geom_boxplot() +
  theme_bw() +
  scale_fill_manual(values=c("forestgreen", "grey40"))
```



Specifying some simple models

```
library(lme4)
```

```
## Loading required package: Matrix
## Loading required package: Rcpp
```

```
model1 = lmer(frequency ~ attitude + (1|subject),
              data = politeness)

summary(model1)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: frequency ~ attitude + (1 | subject)
##    Data: politeness
##
## REML criterion at convergence: 804.7
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.2953 -0.6018 -0.2005  0.4774  3.1772
##
## Random effects:
```

```
##  Groups   Name        Variance Std.Dev.
##  subject  (Intercept) 3982     63.10
##  Residual              851     29.17
## Number of obs: 83, groups:  subject, 6
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  202.588     26.151   7.747
## attitudepol  -19.376      6.407  -3.024
##
## Correlation of Fixed Effects:
##             (Intr)
## attitudepol -0.121
```

```
# Compare this to
lm_model = lm(frequency ~ attitude, data = politeness)
summary(lm_model)
```

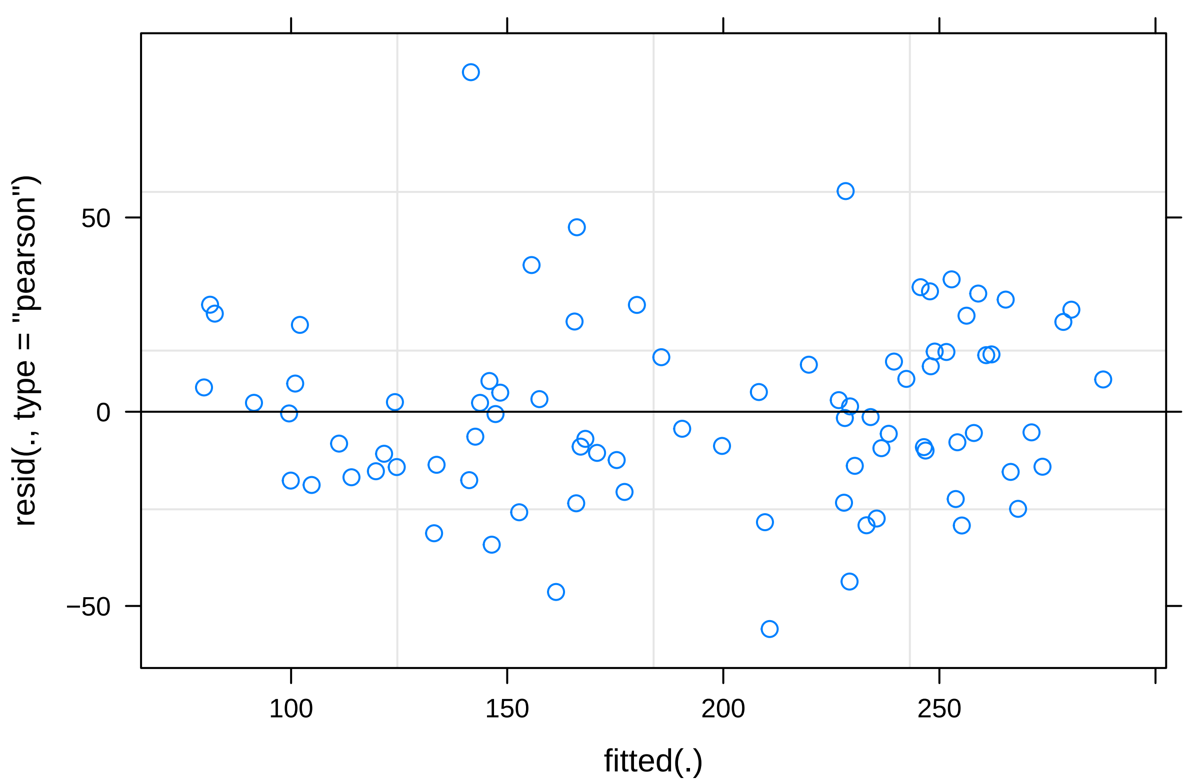
```
##
## Call:
## lm(formula = frequency ~ attitude, data = politeness)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -103.488  -62.122    9.044   51.178  105.044
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   202.59      10.08  20.107   <2e-16 ***
## attitudepol   -18.23      14.34  -1.272    0.207
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 65.3 on 81 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.01958,    Adjusted R-squared:  0.007475
## F-statistic: 1.618 on 1 and 81 DF,  p-value: 0.2071
```

```
# More complex model

model2 = lmer(frequency ~ attitude + (1|subject) +
                (1|scenario), data = politeness)

summary(model2)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: frequency ~ attitude + (1 | subject) + (1 | scenario)
##    Data: politeness
##
## REML criterion at convergence: 793.5
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
```

```
## -2.2006 -0.5817 -0.0639  0.5625  3.4385
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  scenario (Intercept)  219     14.80
##  subject  (Intercept) 4015     63.36
##  Residual              646     25.42
## Number of obs: 83, groups:  scenario, 7; subject, 6
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  202.588     26.754   7.572
## attitudepol  -19.695      5.585  -3.527
##
## Correlation of Fixed Effects:
##             (Intr)
## attitudepol -0.103
```

```
plot(model2)
```



The standard deviation in the random effects part shows how much variation there is between groups. In model2 we can see that there is a lot more variation between subjects than between the different scenarios in which the pitch was measured.

If we compare the standard linear model to the first lmer model we can see that attitude shifts from being non-significant to being clearly significant. This is both because the coefficient becomes stronger, -19.4 compared to -18.2, and because the standard error is far lower.

You can **plot** the result just like for normal lm models, at least to check that the residuals appear to lack structure.

```
# Get p-values for the fixed effects using normal approximation
coef = summary(model2) %>%
  coef() %>%
  data.frame() %>%
  mutate(pvalue = 1.96 * (1-pnorm(abs(t.value), mean=0,
                          sd=1)))
```

Baascially this is just a normal approximation of the t-value and then a check of how likely that result is using the cumulative normal distribution (with mean 0 and sd 1). The more observations we have, the better this approximation will be.

Random slope model

These models have been random intercept models, the variables are assumed to have the same effects but the baseline differs.

There are also random slope models. Here the same variable is assumed to have different effects in different groups - for example, the effect of unemployment on cabinet duration might be more pronounced in some countries.

```
model_rmslopes = lmer(frequency ~ attitude + gender +
                        (1+attitude|subject) +
                        (1+attitude|scenario),
                      data = politeness,
                      REML = FALSE)

summary(model_rmslopes)
```

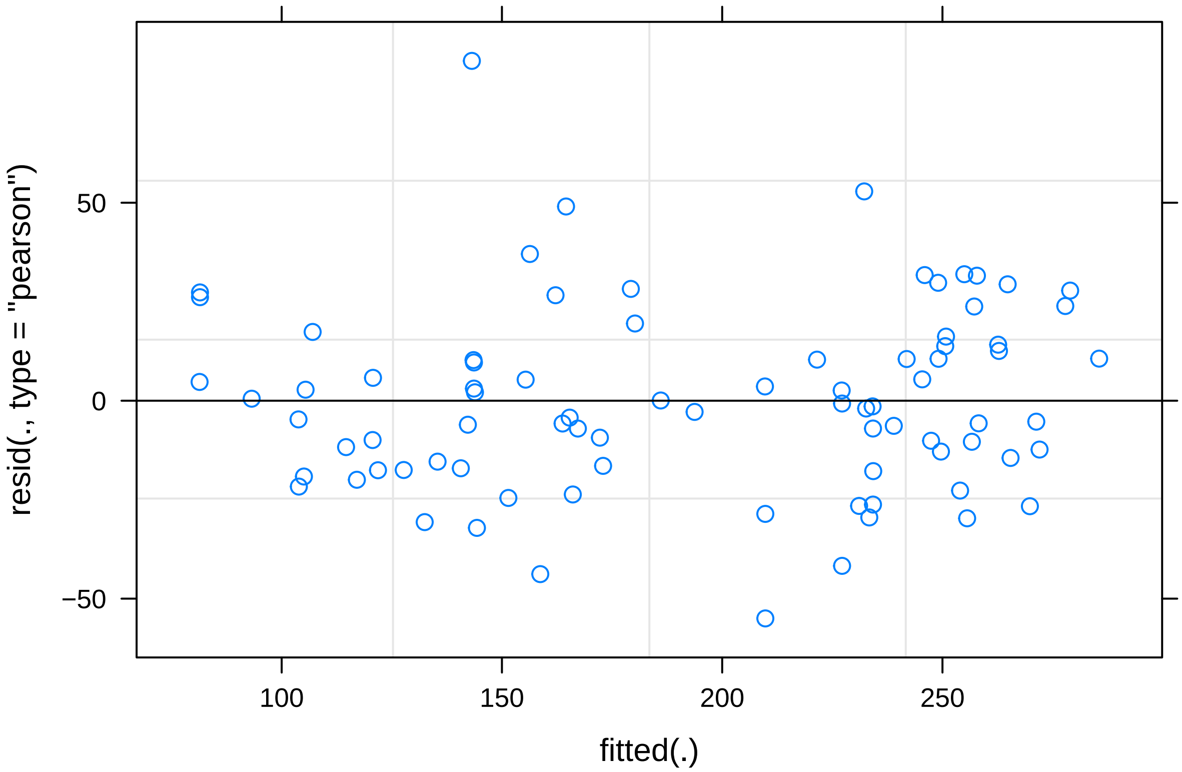
```
## Linear mixed model fit by maximum likelihood  ['lmerMod']
## Formula: frequency ~ attitude + gender + (1 + attitude | subject) + (1 +
##     attitude | scenario)
##    Data: politeness
##
##      AIC      BIC   logLik deviance df.resid
##    814.9    839.1   -397.4    794.9       73
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.1947 -0.6691 -0.0789  0.5256  3.4252
##
## Random effects:
##  Groups   Name        Variance Std.Dev. Corr
##  scenario (Intercept) 182.082  13.494
##           attitudepol  31.262   5.591   0.22
##  subject  (Intercept) 392.474  19.811
##           attitudepol   1.707   1.307   1.00
##  Residual             627.880  25.058
## Number of obs: 83, groups:  scenario, 7; subject, 6
##
## Fixed effects:
```

```
##             Estimate Std. Error t value
## (Intercept)  257.989     13.529  19.069
## attitudepol  -19.747      5.922  -3.334
## genderM     -110.802     17.512  -6.327
##
## Correlation of Fixed Effects:
##             (Intr) atttdp
## attitudepol -0.105
## genderM     -0.647  0.003
```

```r
coef(model_rmslopes)
```

```
## $scenario
##   (Intercept) attitudepol   genderM
## 1    245.2603   -20.43832 -110.8021
## 2    263.3012   -15.94386 -110.8021
## 3    269.1432   -20.63361 -110.8021
## 4    276.8309   -16.30132 -110.8021
## 5    256.0579   -19.40575 -110.8021
## 6    246.8605   -21.94816 -110.8021
## 7    248.4702   -23.55752 -110.8021
##
## $subject
##    (Intercept) attitudepol   genderM
## F1    243.8053   -20.68245 -110.8021
## F2    266.7321   -19.17028 -110.8021
## F3    260.1484   -19.60452 -110.8021
## M3    285.6958   -17.91951 -110.8021
## M4    264.1982   -19.33741 -110.8021
## M7    227.3551   -21.76744 -110.8021
##
## attr(,"class")
## [1] "coef.mer"
```

```r
plot(model_rmslopes)
```




```
# P-values

p_values = summary(model_rmslopes) %>%
  coef() %>%
  data.frame() %>%
  mutate(pvalue = 1.96 * (1 - pnorm(abs(t.value))))
```

The notation "(1+attitude|subject)" means that you tell the model to expect differing baseline-levels of frequency (the intercept, represented by 1) as well as differing responses to the main factor in question, which is "attitude" in this case.

You get p-values for the fixed effects but not for the random slopes.

Multilevel models without random slopes (but with random intercepts) have a too high false positive rate. For this reason you should include all random slopes that are warranted by the data.