From LMMs to GLMMs

Bayesian statistics 5 – mixed effects models and background for GLMMS

Frédéric Barraquand (CNRS, IMB)

28/11/2023

Things that we learned the last time / will learn today

Session 4

- Random effects are random because we estimate their variances
- Some people have proposed to call them *variance component effects*
- In a Bayesian setup, the difference between fixed and random effects manifests in priors: `alpha[i] ~ dnorm(0,0.01)` or `alpha[i] ~ dnorm(0,tau.alpha)`?
- A prior on a quantity that could be considered a prior is a hyperprior

Today (TD/practical on birds)

- Specifying mixed models with fixed and random effects (although the distinction between the two is tenuous in a Bayesian setup)

Things that we learned the last time / will learn today

Session 4

- Random effects are random because we estimate their variances
- Some people have proposed to call them *variance component effects*
- In a Bayesian setup, the difference between fixed and random effects manifests in priors: `alpha[i] ~ dnorm(0,0.01)` or `alpha[i] ~ dnorm(0,tau.alpha)`?
- A prior on a quantity that could be considered a prior is a hyperprior

Today (TD/practical on birds)

- Specifying mixed models with fixed and random effects (although the distinction between the two is tenuous in a Bayesian setup)
- Some more shrinkage → partial pooling of random effects coefficients that get pulled towards their mean

Things that we learned the last time / will learn today

Session 4

- Random effects are random because we estimate their variances
- Some people have proposed to call them *variance component effects*
- In a Bayesian setup, the difference between fixed and random effects manifests in priors: `alpha[i] ~ dnorm(0,0.01)` or `alpha[i] ~ dnorm(0,tau.alpha)`?
- A prior on a quantity that could be considered a prior is a hyperprior

Today (TD/practical on birds)

- Specifying mixed models with fixed and random effects (although the distinction between the two is tenuous in a Bayesian setup)
- Some more shrinkage → partial pooling of random effects coefficients that get pulled towards their mean
- Devise a model with a hierarchical structure

Things that we learned the last time / will learn today

Session 4

- Random effects are random because we estimate their variances
- Some people have proposed to call them *variance component effects*
- In a Bayesian setup, the difference between fixed and random effects manifests in priors: `alpha[i] ~ dnorm(0,0.01)` or `alpha[i] ~ dnorm(0,tau.alpha)`?
- A prior on a quantity that could be considered a prior is a hyperprior

Today (TD/practical on birds)

- Specifying mixed models with fixed and random effects (although the distinction between the two is tenuous in a Bayesian setup)
- Some more shrinkage → partial pooling of random effects coefficients that get pulled towards their mean
- Devise a model with a hierarchical structure
- Technical: Pairs of parameters can be correlated in the joint posterior, we should check whether they are so with pairwise posterior plots

What we will cover next today

Correlation structures = advanced mixed modelling

- Between different parameters (intercept, slope)
- Between the same parameters but between/within groups

Moving towards GLMs and GLMMs

- Dealing with counts: Poisson distribution and Poisson GLMs
- If time, Poisson GLMMs

Next time

Poisson GLMMs on the crested tit example

- Do our variance partitioning results on hold on the full dataset?
- Do our covariate results hold on the full dataset?

More GLMMs on counts

Correlation between random effects I

Back to Kéry (2010, chapter 12) where Y_i is mass and x_i is length, so that the regression gives body condition:

$$Y_i = \alpha_{j[i]} + \beta_{j[i]} x_i + \epsilon_i$$

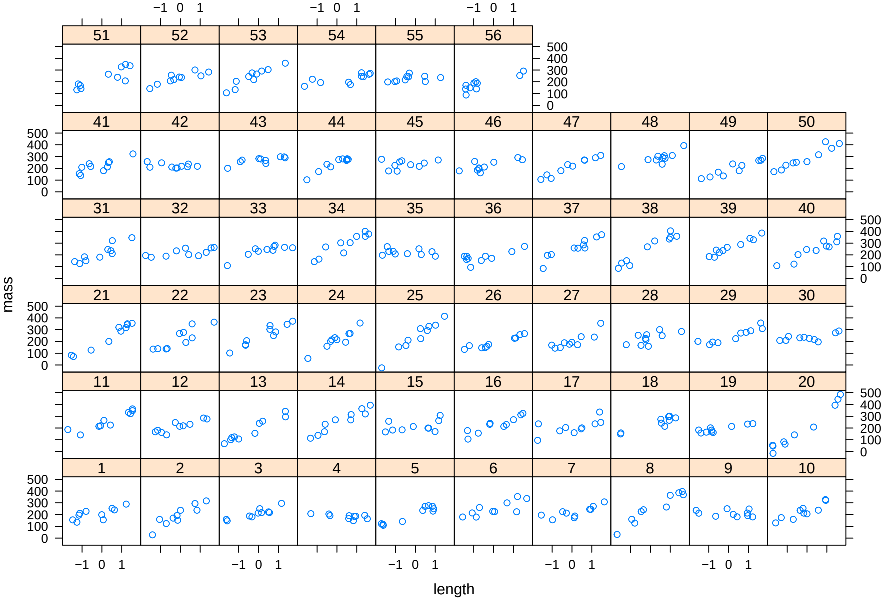
$\epsilon_i \sim \mathcal{N}(0, \sigma^2)$ i.i.d. and $(\alpha_j, \beta_j) \sim \mathcal{N}([\mu_\alpha \mu_\beta], \Sigma)$.

with $\Sigma = \begin{pmatrix} \sigma_\alpha^2 & \sigma_{\alpha\beta} \\ \sigma_{\alpha\beta} & \sigma_\beta^2 \end{pmatrix}$ and $\sigma_{\alpha\beta} = \rho \sigma_\alpha \sigma_\beta$

Let's plot this with $\mu_\alpha = 230, \mu_\beta = 60, \sigma_\alpha = 20, \sigma_\beta = 30, \rho = 33\%$.

```r
library("lattice")
xyplot(mass ~ length | pop)
```

Correlation between random effects II



Code for random effects model I

```
### 12.5.3. REML analysis using R
library('lme4')
lme.fit3 <- lmer(mass ~ length + (length | pop))

### 12.5.4. Bayesian analysis using JAGS

# Bundle data
str(bdata <- list(mass = as.numeric(mass), pop = as.numeric(pop),
length = length, ngroups = max(as.numeric(pop)), n = n))
```

```
## List of 5
##  $ mass   : num [1:560] 195 227 199 289 155 ...
##  $ pop    : num [1:560] 1 1 1 1 1 1 1 1 1 1 ...
##  $ length : num [1:560] -1.15378 -0.79988 -0.00749 1.22131 0.067
##  $ ngroups: num 56
##  $ n      : num 560
```

Code for random effects model II

```
# Specify model in BUGS language
cat(file = "lme.model3.txt", "
model {

# Priors
 for (j in 1:ngroups){
    alpha[j] <- B[j,1]
    beta[j] <- B[j,2]
    B[j,1:2] ~ dmnorm(B.hat[j,], Tau.B[,])
    B.hat[j,1] <- mu.int
    B.hat[j,2] <- mu.slope
}

 mu.int ~ dnorm(0, 0.001)       # Hyperpriors for random intercepts
 mu.slope ~ dnorm(0, 0.001)     # Hyperpriors for random slopes

 Tau.B[1:2,1:2] <- inverse(Sigma.B[,])
 Sigma.B[1,1] <- pow(sigma.int,2)
 sigma.int ~ dunif(0, 100)      # SD of intercepts
 Sigma.B[2,2] <- pow(sigma.slope,2)
 sigma.slope ~ dunif(0, 100)        # SD of slopes
 Sigma.B[1,2] <- rho*sigma.int*sigma.slope
 Sigma.B[2,1] <- Sigma.B[1,2]
 rho ~ dunif(-1,1)
 covariance <- Sigma.B[1,2]

 tau <- 1 / ( sigma * sigma)        # Residual
 sigma ~ dunif(0, 100)          # Residual standard deviation

# Likelihood
 for (i in 1:n) {
    mass[i] ~ dnorm(mu[i], tau)     # The 'residual' random variable
    mu[i] <- alpha[pop[i]] + beta[pop[i]]* length[i]  # Expectation
 }
}
```

Results of RE model with correlation I

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 560
##    Unobserved stochastic nodes: 62
##    Total graph size: 3045
##
## Initializing model
```

```
## Inference for Bugs model at "lme.model3.txt", fit using jags,
##  3 chains, each with 2000 iterations (first 1000 discarded), n.thin = 2
##  n.sims = 1500 iterations saved
##              mu.vect sd.vect     2.5%      25%      50%      75%    97.5% Rhat
## covariance   188.825 107.495    0.485  117.719  178.447  247.301  423.031 1.004
## mu.int       224.208   3.029  218.130  222.185  224.242  226.296  230.106 1.002
## mu.slope      52.584   4.541   43.724   49.477   52.621   55.703   61.262 1.005
## rho            0.299   0.145    0.001    0.202    0.302    0.400    0.568 1.002
## sigma         30.302   0.975   28.510   29.604   30.269   30.945   32.242 1.001
## sigma.int     19.585   2.590   15.089   17.711   19.422   21.112   25.394 1.001
## sigma.slope   31.799   3.440   26.000   29.306   31.495   33.996   39.137 1.005
## deviance    5407.877  17.241 5376.933 5395.611 5407.430 5419.134 5442.478 1.002
##             n.eff
```

Results of RE model with correlation II

```
## covariance    970
## mu.int       1300
## mu.slope      420
## rho          1500
## sigma        1500
## sigma.int    1500
## sigma.slope   370
## deviance     1100
##
## For each parameter, n.eff is a crude measure of effective sample size,
## and Rhat is the potential scale reduction factor (at convergence, Rhat=1).
##
## DIC info (using the rule, pD = var(deviance)/2)
## pD = 148.6 and DIC = 5556.4
## DIC is an estimate of expected predictive error (lower deviance is better).
```

Results of RE model with correlation III

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: mass ~ length + (length | pop)
## REML criterion at convergence: 5611.447
## Random effects:
##  Groups   Name        Std.Dev. Corr
##  pop      (Intercept) 18.89
##           length      30.75    0.31
##  Residual             30.19
## Number of obs: 560, groups:  pop, 56
## Fixed Effects:
## (Intercept)       length
##      226.40        54.39
```

The mixed model – theory to understand correlations

In matrix notation, the mixed model writes

$$\mathbf{y} = \mathbf{X}\beta + \mathbf{Z}\mathbf{a} + \boldsymbol{\epsilon}$$

Convention here: simple bold for vectors, capital bold for matrices. Cov($\boldsymbol{\epsilon}$) and Cov($\mathbf{a}$) are called covariance matrices, for the residuals and the random effects, respectively.

Possibility for correlations between random effects, as we've seen before, but random effects also generate correlations within groups.

Genetics example: who's your mommy?

Let's assume that Y_i is the phenotype of individual i. Like, size. In the population, we have cows that descend from their mommies $m = 1, 3, ..., M$. Let's say $M = 10$, we have $\mathbf{a} = (A_1, ..., A_{10})$.

$$Y_i = \mu + bx_i + A_{m[i]} + \epsilon_i$$

- We assume that residuals are Gaussian iid. In math, $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$ with $\text{Cov}(\epsilon_i, \epsilon_j) = \mathbb{V}(\epsilon_i) = \sigma^2$ for $j = i$ and $\text{Cov}(\epsilon_i, \epsilon_j) = 0$ for $j \neq i$.
- We assume that x_i is the fixed effect of how much food the cow receives on average.
- We assume that mother phenotypes have themselves negligible correlation, so that $\text{Cov}(a_i, a_j) = 0$ for $j \neq i$ (possible to add multiple levels with grandma effects etc.)

Some background on quantitative genetics

What is the covariance matrix for the overall random effect?

$$Y_i = \mu + bx_i + \eta_i$$

with $\eta_j = A_{m[i]} + \epsilon_i$.

From rules on covariances, we can compute that

$$\mathrm{Cov}(\eta_i, \eta_j) = \sigma^2 \mathbf{1}_{i=j} + \sigma_A^2 \mathbf{1}_{m[i]=m[j]}$$

Let's write this on the board to get a better look.

Modelling residual covariance and adding random effects are usually two equivalent ways to add structure to your model. (but often we add random effects because variance-covariance matrices are tricky beasts).

Autocorrelated noise and distance-decay correlation structures

Let's imagine that we have a temporal effect $B_t = \varphi B_{t-1} + \zeta_t, \zeta_t \sim \mathcal{N}(0, \sigma^2)$. In this model, $\text{Corr}(B_t, B_{t-k}) = \varphi^k$ due to the properties of the AR(1) process.

Decay of correlation with distance in time. Seal population dynamics example in *Mixed effects models and extensions in ecology with r* by Zuur et al. (2009)

You can have distance-decay of correlation in space too [advanced class].

Are my random effects nested or crossed?

This is more of a *question about data design than it is about modelling*. A good post on Cross-Validated on that issue.

More references:

Nested by design: model fitting and interpretation in a mixed model era by Schielzeth and Nakagawa

Data Analysis Using Regression and Multilevel/Hierarchical Models, by Gelman and Hill

A brief introduction to mixed effects modelling and multi-model inference in ecology, by Harrison et al.

Poisson GLMMs – beyond the Gaussian realm

We want to be able to model our counts (of birds, cells, mechanical failures,. . .) including the many zeroes, ones and twos in them. \

Data for Poisson, Negative Binomial GLMMs and the likes:

- `data=c(0,0,1,2,3,0,8,2,3,...)`
- and not `data=c(146,827,22,38,49,167,0,332,92,...)`

Before having complex models with random effects

- Let's go back to simple GLMs
- And for that to the Poisson distribution

The Poisson(λ) distribution, $Y_i \sim \mathcal{P}(\lambda)$ I

Mean and variance are linked in the Poisson distribution.

- Expectation $\mathbb{E}(Y_i) = \lambda$
- Variance $\mathbb{V}(Y_i) = \lambda$

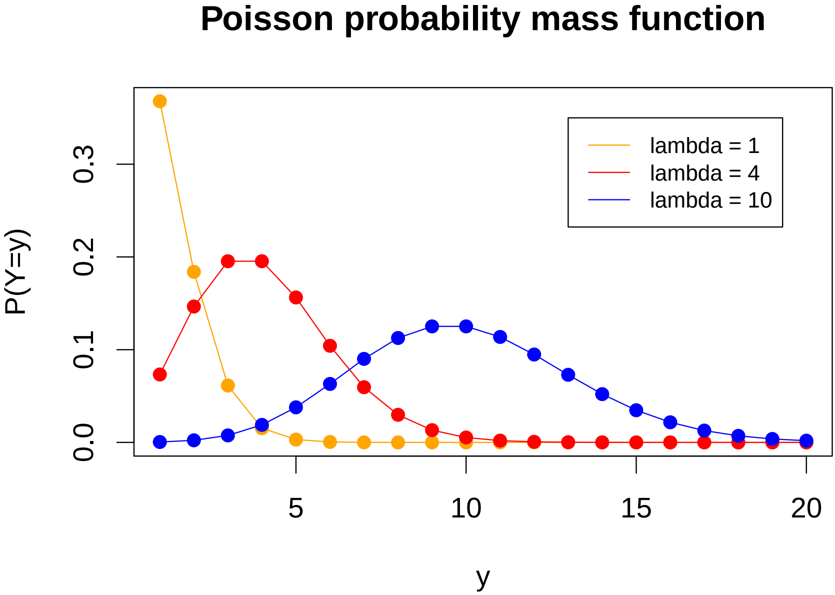
In general counts have usually $\mathbb{V}(Y_i) \propto \mathbb{E}(Y_i)$. That's why when we look at residuals of GLMs we use $\frac{y_i - \hat{y}_i}{\sqrt{\hat{y}_i}}$. Since $\mathrm{SD}(Y_i) \propto \sqrt{\mathbb{E}(Y_i)}$.

Reminder, distribution for positive quantities:

In the Gamma distribution, $\mathbb{V}(Y_i) = a\theta^2 = \frac{1}{a}(\mathbb{E}(Y_i))^2$.

Log-normal, $\mathbb{V}(Y_i) = \text{stuff}(\sigma) \times \mathbb{E}(Y_i)^2$.

The Poisson(λ) distribution, $Y_i \sim \mathcal{P}(\lambda)$ II



Poisson and other count distributions

Counts are special. Different mean-variance scaling for Poisson.

$CV = SD/mean = \frac{1}{\sqrt{\text{mean}}}$

When counts are *overdispersed*, we use distributions in-between the Poisson and those shown before like the Negative Binomial. There are other options.

GLM, the basic regression model

$$Y_i \sim \mathcal{P}(\lambda_i)$$

with

$$\lambda_i = g^{-1}(a + bx_i)$$

or again

$$g(\lambda_i) = a + bx_i$$

GLM, the basic regression model

$$Y_i \sim \mathcal{P}(\lambda_i)$$

with

$$\lambda_i = g^{-1}(a + bx_i)$$

or again

$$g(\lambda_i) = a + bx_i$$

Typically the *link function* $g = \log$ and

$$\lambda_i = \exp(a + bx_i)$$

or again

$$\log(\lambda_i) = a + bx_i$$

GLM, the basic regression model

$$Y_i \sim \mathcal{P}(\lambda_i)$$

with

$$\lambda_i = g^{-1}(a + bx_i)$$

or again

$$g(\lambda_i) = a + bx_i$$

Typically the *link function* $g = \log$ and

$$\lambda_i = \exp(a + bx_i)$$

or again

$$\log(\lambda_i) = a + bx_i$$

Thus this is by design close to log-transforming the data, but you have the extra Poisson variation.

GLM, the basic regression model

$$Y_i \sim \mathcal{P}(\lambda_i)$$

with

$$\lambda_i = g^{-1}(a + bx_i)$$

or again

$$g(\lambda_i) = a + bx_i$$

Typically the *link function* $g = \log$ and

$$\lambda_i = \exp(a + bx_i)$$

or again

$$\log(\lambda_i) = a + bx_i$$

Thus this is by design close to log-transforming the data, but you have the extra Poisson variation.

Extra question: should you add noise $\log(\lambda_i) = a + bx_i + \epsilon_i$?

Heavy debates in ecology and evolution

on whether one should log-transforming count data or use GLMs:

- O'Hara and Kotze said in 2010 that we always should use GLMs.

Heavy debates in ecology and evolution

on whether one should log-transforming count data or use GLMs:

- O'Hara and Kotze said in 2010 that we always should use GLMs.
- Then Ives showed that if you just want to know which coefficients have an effect, you might do better with log-transformation.

Heavy debates in ecology and evolution

on whether one should log-transforming count data or use GLMs:

- O'Hara and Kotze said in 2010 that we always should use GLMs.
- Then Ives showed that if you just want to know which coefficients have an effect, you might do better with log-transformation.
- Warton et al. further showed that it depends and provided advice on how to choose. (if you have very very small counts so $E(y) < 1$ you can't get away with it)

Heavy debates in ecology and evolution

on whether one should log-transforming count data or use GLMs:

- O'Hara and Kotze said in 2010 that we always should use GLMs.
- Then Ives showed that if you just want to know which coefficients have an effect, you might do better with log-transformation.
- Warton et al. further showed that it depends and provided advice on how to choose. (if you have very very small counts so $E(y) < 1$ you can't get away with it)
- Finally in 2020 Morrissey and Ruxton showed a caveat in the paper that started the debate.

Heavy debates in ecology and evolution

on whether one should log-transforming count data or use GLMs:

- O'Hara and Kotze said in 2010 that we always should use GLMs.
- Then Ives showed that if you just want to know which coefficients have an effect, you might do better with log-transformation.
- Warton et al. further showed that it depends and provided advice on how to choose. (if you have very very small counts so $E(y) < 1$ you can't get away with it)
- Finally in 2020 Morrissey and Ruxton showed a caveat in the paper that started the debate.
- Literature auto-correct time = 10 years. Beware!

So what should I pick?

- You have very small counts: GLMs
- You don't, depends on what you want to do
  1. You want to estimate coefficients to find effects. You can log-transform.
  2. You want to simulate from the model → log-transformed data won't give you counts with the right variance. You need the GLMs.

So what should I pick?

- You have very small counts: GLMs
- You don't, depends on what you want to do
  1. You want to estimate coefficients to find effects. You can log-transform.
  2. You want to simulate from the model → log-transformed data won't give you counts with the right variance. You need the GLMs.
- Bayesian models are *extra practical to simulate data under the fitted model*, because you can simulate at the same time as you fit
- This is even how you diagnose your model fit! *Posterior predictive checks*.

We will see this next time.