GLMMs in R: analyzing overdispersed data

Ben Bolker

June 1, 2010

Looking for the simplest possible example that encapsulates overdispersion which can be sensibly modeled via lognormal-Poisson approaches (i.e., individual-level random effects).

Unfortunately I haven't yet found a good, non-problematic dataset that uses Poisson or binomial data, has overdispersion, but doesn't have other issues [zero-inflation, too small/messy for straightforward analysis, etc.] ... ("All [simple data sets] are alike. Every [messy data set] is [messy] in its own way.")

From `http://glmm.wikidot.com/faq`:

- quasilikelihood estimation: `MASS::glmmPQL` (the "quasi-" families may be unreliable in `lme4`, and may disappear; not clear whether there is a good theoretical foundation for extending quasilikelihood to the GLMM case); `geepack::geeglm`, `gee::gee`
- individual-level random effects (`MCMCglmm` or hacked `lme4`) [or WinBUGS or AD Model Builder or ...] [note that individual-level random effect estimation is probably dodgy for PQL approaches]
- alternative distributions
  - Poisson-lognormal (see above, "individual-level random effects")
  - negative binomial
    - `glmmADMB::glmm.admb` (off-CRAN: `http://otter-rsch.com/admbre/examples/glmmadmb/glmmADMB.html`)
    - `repeated::gnlmm` (off-CRAN: `http://www.commanster.eu/rcode.html`)
    - WinBUGS/JAGS (via R2WinBUGS/Rjags)
    - AD Model Builder (via R2ADMB?)
- beta-binomial: all of the above except (?) `MCMCglmm`, `glmm.admb`
- zero-inflated: all of the above except `gnlmm`

1 Examples (simulation)

It's easy enough to generate lognormal-Poisson-distributed "data" and show that a (hacked) version of lme4 recovers them appropriately, but it may not be very informative. This is a basic Poisson simulation with a single covariate (uniformly randomly distributed), random intercept differences among blocks, and random intercept differences among individuals.

```
> simfun <- function(ng = 20, nr = 100, fsd = 1, indsd = 0.5, b = c(1,
+     2)) {
+     ntot <- nr * ng
+     b.reff <- rnorm(ng, sd = fsd)
+     b.rind <- rnorm(ntot, sd = indsd)
+     x <- runif(ntot)
+     dd <- data.frame(x, f = factor(rep(LETTERS[1:ng], each = nr)),
+         obs = 1:ntot)
+     dd$eta0 <- model.matrix(~x, data = dd) %*% b
+     dd$eta <- with(dd, eta0 + b.reff[f] + b.rind[obs])
+     dd$mu <- exp(dd$eta)
+     dd$y <- with(dd, rpois(ntot, lambda = mu))
+     dd
+ }
```

Try it out:

```
> library(lme4)
> set.seed(1001)
> dd <- simfun()
> (m0 <- glmer(y ~ x + (1 | f), family = "poisson", data = dd))

Generalized linear mixed model fit by the Laplace approximation
Formula: y ~ x + (1 | f)
   Data: dd
   AIC   BIC logLik deviance
 12768 12785  -6381    12762
Random effects:
 Groups Name        Variance Std.Dev.
 f      (Intercept) 1.4459   1.2024
Number of obs: 2000, groups: f, 20

Fixed effects:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  1.08635    0.26935    4.03  5.5e-05 ***
x            2.08502    0.01914  108.92  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
Correlation of Fixed Effects:
  (Intr)
x -0.046
```

```
> (m1 <- glmer(y ~ x + (1 | f) + (1 | obs), family = "poisson",
+     data = dd))
```

```
Generalized linear mixed model fit by the Laplace approximation
Formula: y ~ x + (1 | f) + (1 | obs)
   Data: dd
  AIC  BIC logLik deviance
 4598 4620  -2295     4590
Random effects:
 Groups Name        Variance Std.Dev.
 obs    (Intercept) 0.23339  0.48311
 f      (Intercept) 1.42310  1.19294
Number of obs: 2000, groups: obs, 2000; f, 20

Fixed effects:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   0.9920     0.2686    3.69 0.000222 ***
x             2.0501     0.0498   41.17  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Correlation of Fixed Effects:
  (Intr)
x -0.102
```

A summary function to fit the full model and extract parameters:

```
> cfun <- function(d) {
+     m <- glmer(y ~ x + (1 | f) + (1 | obs), family = "poisson",
+         data = d)
+     c(sqrt(unlist(VarCorr(m))), fixef(m))
+ }
```

Run it 50 times:

```
> rr <- replicate(50, cfun(simfun()))
```

This works pretty well (Figure 1).

2 Examples (real)

- **Count data: sheep tick burdens on the heads of red grouse chicks** (Elston et al., 2001):

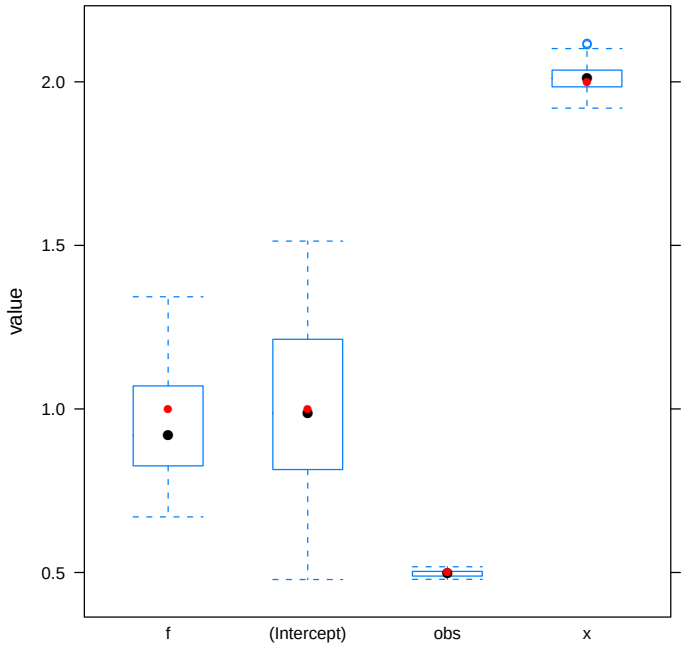

Figure 1: Basic results for Poisson-lognormal model

  - Originally analyzed with the "GLMM procedure in Genstat 5.4.1 (Genstat 5 Committee, 1997 ; Payne & Arnold, 1998) and the SAS GLIMMIX macro (Littell et al. 1996)". (Both of these are marginal [P/MQL] algorithms, individual-level effect estimation is supposed to be dodgy in this case ... although I don't have a peer-reviewed reference handy [see e.g. the paragraph headed "final remark" in `https://stat.ethz.ch/pipermail/r-sig-mixed-models/2008q4/001488.html`].
  - I can run `MASS::glmmPQL`, but don't get the same results as quoted in the paper — haven't looked into the details ...
  - When I try to run this analysis in a hacked version of `lme4` I get `Cholmod ... 'not positive definite'` and `mer_finalize ... false convergence` warnings ...
  - In `MCMCglmm`, I get `Mixed model equations singular: use a (stronger) prior` after 8000 iterations.

  In any case, this does not look like a straightforward/simple analysis.

- **Count data: owl nestling begging** (Roulin and Bersier, 2007), reproduced as an example in Zuur et al. (2009): data available from `http://www.highstat.com/Book2/ZuurDataMixedModelling.zip`
  - I have run this analysis in lme4, and the results are reasonably sensible. However, the residuals are a bit funny, and Alain Zuur has mentioned that he is going to use the data in a methods paper on zero-inflation.
  - could try this in `glmm.admb` or `MCMCglmm`, which both allow zero-inflation

- **Count data: gopher tortoise shell counts** (Ozgul et al., 2009): tried analysis in various ways, ended up coding in WinBUGS. Random effect (site) has quite limited sample sizes (only 10 sites), and `glmer` finds a best estimate of zero variance among sites (even among sites once we drop the overdispersion variation).

- **Binomial data: *Glycera* cell survival** I'm working on an analysis of a big factorial experiment on the response of *Glycera* (a marine worm) cells to various stressors. The data aren't (yet) mine to release. In addition, I had convergence problems with `glmer` — I ended up analyzing the data with `MCMCglmm`. (The version of `glmer` in `lme4a` gives slightly different results (more than round-off error), and does *not* produce convergence warnings ...

- I have various binary data sets, but these are not particularly good for exploring overdispersion, because overdispersion is unidentifiable in binary data.

References

Elston, D. A., R. Moss, T. Boulinier, C. Arrowsmith, and X. Lambin (2001). Analysis of aggregation, a worked example: numbers of ticks on red grouse chicks. *Parasitology 122*(5), 563–569.

Ozgul, A., M. K. Oli, B. M. Bolker, and C. Perez-Heydrich (2009, April). Upper respiratory tract disease, force of infection, and effects on survival of gopher tortoises. *Ecological Applications: A Publication of the Ecological Society of America 19*(3), 786–798. PMID: 19425439.

Roulin, A. and L. Bersier (2007, October). Nestling barn owls beg more intensely in the presence of their mother than in the presence of their father. *Animal Behaviour 74*(4), 1099–1106.

Zuur, A. F., E. N. Ieno, N. J. Walker, A. A. Saveliev, and G. M. Smith (2009, March). *Mixed Effects Models and Extensions in Ecology with R* (1 ed.). Springer.