CHAPTER

19

Binomial Mixed-Effects Model (Binomial GLMM)

OUTLINE

19.1 INTRODUCTION

As in a Poisson generalized linear mixed model (GLMM), we can also add into a binomial generalized linear model (GLM) random variation beyond what is stipulated by the binomial distribution. We show this for a slight modification of the Red-backed shrike example from Chapter 16. Instead of counting the number of pairs, which naturally leads to the adoption of a Poisson model, we now study the reproductive success (success or failure) of its much rarer cousin, the woodchat shrike (Fig. 19.1). We examine the relationship between precipitation during the breeding season and reproductive success; wet springs are likely to depress the proportion of successful nests. We assemble data from 16 populations studied over 10 years.

First, we write down the random-coefficients model (without intercept-slope correlation) for a binomial response. We model C_i, the number of successful pairs among N_i studied pairs in year i and study area j:

1. Distribution: $C_i \sim \text{Binomial}(p_i, N_i)$
2. Link function: logit, i.e., $\text{logit}(p_i) = \log\left(\frac{p_i}{1 - p_i}\right)$ = linear predictor

Introduction to WinBUGS for Ecologists
DOI: 10.1016/B978-0-12-378605-0.00019-3

FIGURE 19.1 Woodchat shrike (*Lanius senator*), Catalonia, 2008. (*Photo J. Rojals*)

3. Linear predictor: $\alpha_{j(i)} + \beta_{j(i)} * x_i$
4. Submodel for parameters: $\alpha_j \sim \text{Normal}(\mu_\alpha, \sigma_\alpha^2)$

$$\beta_j \sim \text{Normal}(\mu_\beta, \sigma_\beta^2)$$

Except for a different distribution and link function, the additional kind of data provided by the binomial totals N_i, and a different interpretation and indexing of the covariate, this model looks exactly like the Poisson GLMM in Chapter 16! The linear predictor, $\alpha_{j(i)} + \beta_{j(i)} * x_i$, specifies a population-specific, logit-linear relationship between breeding success and precipitation. Furthermore, populations are assumed to be related in the sense that both intercepts (α_j) and slopes (β_j) of those relationships come from two normal distributions whose hyperparameters we estimate.

19.2 DATA GENERATION

We generate data under the random-coefficients model, i.e., with both α_j and β_j assumed independent and random effects. We assume no correlation.

```
n.groups <- 16
n.years <- 10
n <- n.groups * n.years
pop <- gl(n = n.groups, k = n.years)
```

We create a uniform covariate as an index to spring precipitation: 0 denotes little rain and 1 much.

```
prec <- runif(n, 0, 1)
```

N_i, the binomial total, is the number of nesting attempts surveyed in year i.

```
N <- round(runif(n, 10, 50) )
```

We build the design matrix as before.

```
Xmat <- model.matrix(~pop*prec-1-prec)
print(Xmat[1:91,], dig = 2)           # Print top 91 rows
```

Next, we choose the parameter values from their respective normal distributions, but first, we need to select the associated hyperparameters.

```
intercept.mean <- 1       # Select hyperparams
intercept.sd <- 1
slope.mean <- –2
slope.sd <- 1
intercept.effects<-rnorm(n = n.groups, mean = intercept.mean, sd = intercept.sd)
slope.effects <- rnorm(n = n.groups, mean = slope.mean, sd = slope.sd)
all.effects <- c(intercept.effects, slope.effects)    # Put them all together
```

We assemble the counts C_i by first computing the value of the linear predictor, then applying the inverse-logit transformation, and finally integrating binomial noise (where we need N_i).

```
lin.pred <- Xmat %*% all.effects                 # Value of lin.predictor
exp.p <- exp(lin.pred) / (1 + exp(lin.pred))     # Expected proportion
```

For each population, we plot the expected (Fig. 19.2) and the observed breeding success (Fig. 19.3) of woodchat shrikes against standardized spring precipitation using a Trellis plot.

```
library("lattice")
xyplot(exp.p ~ prec | pop, ylab = "Expected woodchat shrike breeding success ",
xlab = "Spring precipitation index", main = "Expected breeding success")

C <- rbinom(n = n, size = N, prob = exp.p)         # Add binomial variation
xyplot(C/N ~ prec | pop, ylab = "Realized woodchat shrike breeding success", xlab
= "Spring precipitation index", main = "Realized breeding success")
```

19.3 ANALYSIS UNDER A RANDOM-COEFFICIENTS MODEL

We could assume that all shrike populations have the same relationship between breeding success and standardized spring precipitation, but at different levels, corresponding to a random-intercepts model

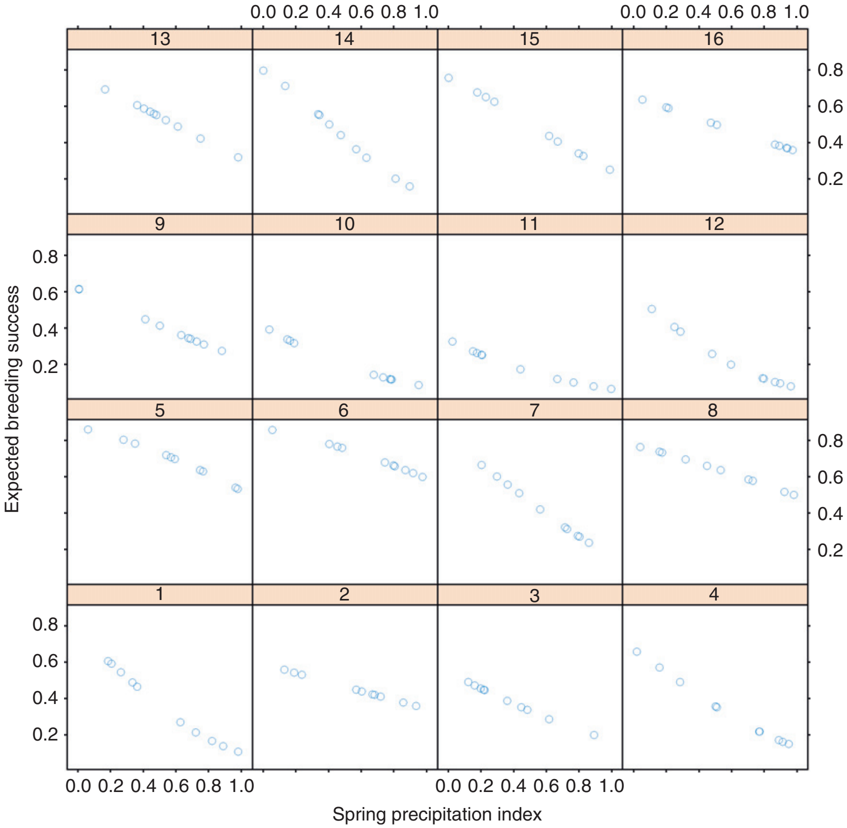

FIGURE 19.2 Trellis plot of the relationship between spring precipitation (standardized) and expected breeding success of woodchat shrikes in 16 populations over 10 years.

(see Section 12.3). However, we directly adopt the random coefficients model without correlation instead. This means assuming that every shrike population has a specific response to precipitation but that both intercept and slope are "similar" among populations.

19.3.1 Analysis Using R

```
library('lme4')
glmm.fit <- glmer(cbind(C, N-C) ~ prec + (1 | pop) + ( 0+ prec | pop), family =
binomial)
glmm.fit
```

```
> glmm.fit
Generalized linear mixed model fit by the Laplace approximation
Formula: cbind(C, N - C) ~ prec + (1 | pop) + (0 + prec | pop)
```

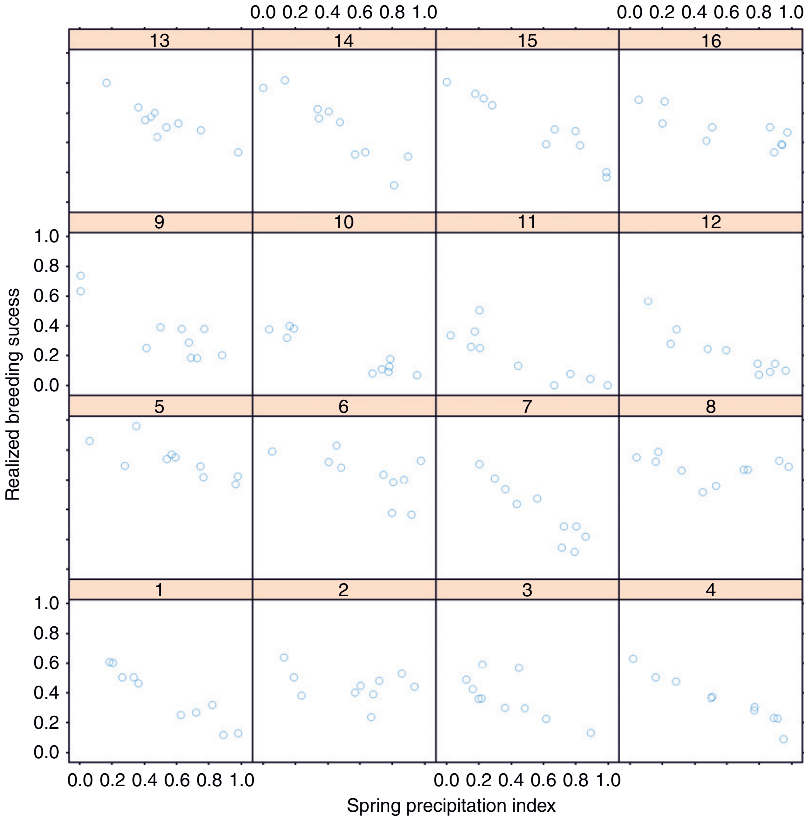

FIGURE 19.3 Trellis plot of the relationship between spring precipitation (standardized) and the realized breeding success of woodchat shrikes in 16 populations over 10 years. The difference between this and the previous plot is due to binomial sampling variation.

```
    AIC    BIC  logLik  deviance
  229.3  241.6  -110.6     221.3
Random effects:
 Groups Name        Variance  Std.Dev.
 pop    (Intercept)  0.38785   0.62277
 pop    prec         0.53258   0.72978
Number of obs: 160, groups: pop, 16

Fixed effects:
            Estimate  Std.Error  z value  Pr(>|z|)
(Intercept)   0.8006     0.1690    4.738  2.16e-06 ***
prec         -2.1290     0.2173   -9.796   < 2e-16 ***
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Correlation of Fixed Effects:
     (Intr)
prec -0.182
```

19.3.2 Analysis Using WinBUGS

```
# Define model
sink("glmm.txt")
cat("
model {

# Priors
 for (i in 1:n.groups){
    alpha[i] ~ dnorm(mu.int, tau.int)      # Intercepts
    beta[i] ~ dnorm(mu.beta, tau.beta)     # Slopes
}
 mu.int ~ dnorm(0, 0.001)     # Hyperparameter for random intercepts
 tau.int <- 1 / (sigma.int * sigma.int)
 sigma.int ~ dunif(0, 10)

 mu.beta ~ dnorm(0, 0.001      # Hyperparameter for random slopes
 tau.beta <- 1 / (sigma.beta * sigma.beta)
 sigma.beta ~ dunif(0, 10)

# Binomial likelihood
 for (i in 1:n) {
    C[i] ~ dbin(p[i], N[i])
    logit(p[i]) <- alpha[pop[i]] + beta[pop[i]]* prec[i]
  }
}
",fill=TRUE)
sink()

# Bundle data
win.data <- list(C = C, N = N, pop = as.numeric(pop), prec = prec, n.groups =
n.groups, n = n)

# Inits function
inits <- function(){ list(alpha = rnorm(n.groups, 0, 2), beta = rnorm(n.groups, 1,
1), mu.int = rnorm(1, 0, 1), mu.beta = rnorm(1, 0, 1))}

# Parameters to estimate
params <- c("alpha", "beta", "mu.int", "sigma.int", "mu.beta", "sigma.beta")
```

```
# MCMC settings
ni <- 2000
nb <- 500
nt <- 2
nc <- 3

# Start Gibbs sampling
out <- bugs(win.data, inits, params, "glmm.txt", n.thin=nt, n.chains=nc,
n.burnin=nb, n.iter=ni, debug = TRUE)
```

This standard GLMM converges nicely.

```
print(out, dig = 2)
> print(out, dig = 2)
Inference for Bugs model at "glmm.txt", fit using WinBUGS,
 3 chains, each with 2000 iterations (first 500 discarded), n.thin = 2
 n.sims = 2250 iterations saved
              mean    sd  2.5%    25%    50%    75%  97.5%  Rhat n.eff
[ ... ]
mu.int        0.80  0.19  0.44   0.68   0.80   0.92   1.17  1.00  2200
sigma.int     0.70  0.16  0.45   0.59   0.68   0.79   1.08  1.00  1600
mu.beta      -2.13  0.25 -2.64  -2.28  -2.12  -1.97  -1.64  1.00  2200
sigma.beta    0.84  0.23  0.48   0.67   0.81   0.96   1.38  1.00  1100
[ ... ]
DIC info (using the rule, pD = var(deviance)/2)
pD = 32.8 and DIC = 748.4
DIC is an estimate of expected predictive error (lower deviance is better).
>

# Compare with input values
intercept.mean ; intercept.sd ; slope.mean ; slope.sd
> intercept.mean ; intercept.sd ; slope.mean ; slope.sd
[1] 1
[1] 1
[1] -2
[1] 1
>
```

This seems to work, too. As is typical, the estimates of random-effects variances are greater in the Bayesian approach, presumably since inference is exact and incorporates all uncertainty in the modeled system. In contrast, the approach in `lmer()` or `glmer()` is approximate and may underestimate these variances (Gelman and Hill, 2007).

19.4 SUMMARY

As for the Poisson case, the introduction of random effects into a binomial GLM in WinBUGS is particularly straightforward and transparent. Fitting the resulting binomial GLMM in WinBUGS is very helpful for your general understanding of this class of models.

EXERCISES

1. *Predictions*: Produce a plot of the mean expected response of breeding success to the spring precipitation index. Try to overlay these estimates on the observed data.
2. *Fixed and random, shrinkage*: Fit a fixed-effects binomial analysis of covariance (ANCOVA) model to the data set we just assembled and compare the estimates of the population-specific intercepts and slopes under the fixed- and the random-effects models.