

# Worked analysis of owl data

Ben Bolker

March 19, 2010

©2010 Ben Bolker



Licensed under the Creative Commons attribution-noncommercial license  
(<http://creativecommons.org/licenses/by-nc/3.0/>). Please share & remix non-commercially, mentioning its origin.

## 1 Introduction/preliminaries

This is a worked example of data on begging by owl nestlings, analyzed as an example in Zuur et al. (2009) and originally appearing in Roulin and Bersier (2007).

Get the data: either download/install AED ([http://www.highstat.com/Book2/AED\\_1.0.zip](http://www.highstat.com/Book2/AED_1.0.zip)) or ZuurDataMixedModelling.zip (<http://www.highstat.com/Book2/ZuurDataMixedModelling.zip>) and extract the Owls data set

For example:

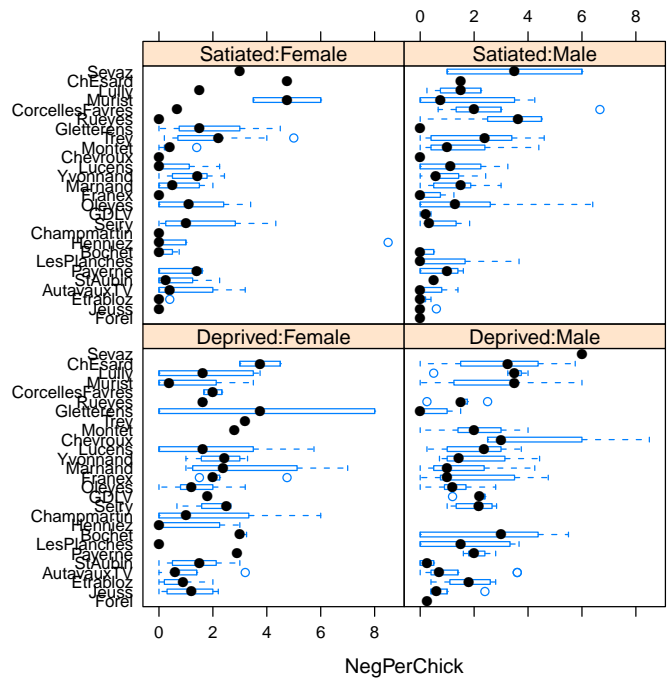
```
> download.file("http://www.highstat.com/Book2/ZuurDataMixedModelling.zip",  
               file="tmp.zip")  
> unzip("tmp.zip",files="Owls.txt")  
  
> Owls <- read.table("Owls.txt",header=TRUE)
```

or just `library(AED); Data(Owls)` if you've installed the AED package.

A quick look at the data with `lattice` (quicker than `ggplot`):

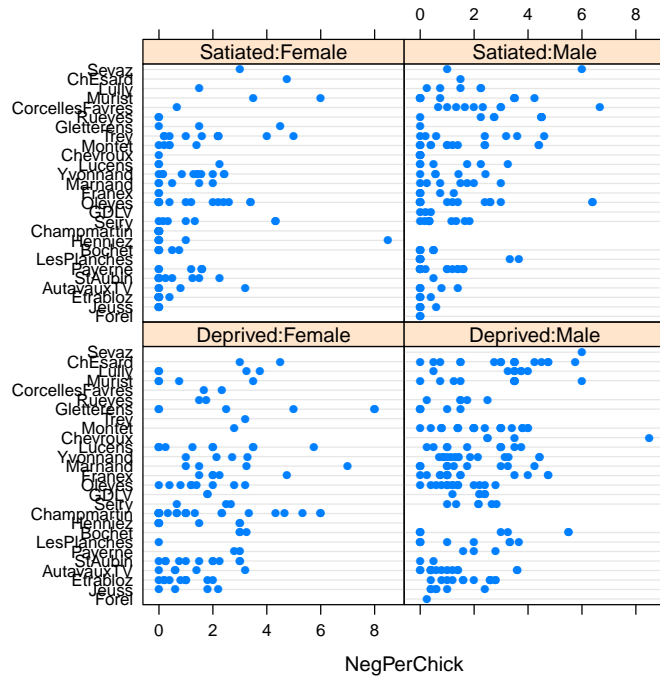
As box-whisker plot:

```
> library(lattice)  
> print(bwplot(reorder(Nest,NegPerChick)~NegPerChick|FoodTreatment:SexParent,  
              data=Owls))
```



As dotplot:

```
> print(dotplot(reorder(Nest, NegPerChick) ~ NegPerChick | FoodTreatment:SexParent,
  data=Owls))
```



## 2 Fitting I

Fit the data:

```
> library(lme4)
> g1 <- glmer(SiblingNegotiation~FoodTreatment*SexParent+offset(log(BroodSize))+
  (1|Nest),family=poisson,data=Owls)
> print(summary(g1))
```

Generalized linear mixed model fit by the Laplace approximation

Formula: SiblingNegotiation ~ FoodTreatment \* SexParent + offset(log(BroodSize)) + (1 | Nest)

Data: Owls

AIC	BIC	logLik	deviance
3532	3554	-1761	3522

Random effects:

Groups Name	Variance	Std.Dev.
Nest (Intercept)	0.20631	0.45421

Number of obs: 599, groups: Nest, 27

Fixed effects:

Estimate	Std. Error	z value	Pr(> z )
----------	------------	---------	----------

```

(Intercept)                0.65584    0.09564    6.857 7.03e-12 ***
FoodTreatmentSatiated      -0.65612    0.05612   -11.691 < 2e-16 ***
SexParentMale              -0.03705    0.04506    -0.822  0.4110
FoodTreatmentSatiated:SexParentMale  0.13130    0.07047    1.863  0.0624 .
---

```

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

```

Correlation of Fixed Effects:
      (Intr) FdTrtS SxPrnM
FdTrtmntStt -0.227
SexParentMl  -0.293  0.490
FdTrtmS:SPM  0.171 -0.768 -0.605

```

Check for overdispersion (Pearson residuals):

```

> rdev <- sum(residuals(g1)^2)
> mdf <- length(fixef(g1))
> rdf <- nrow(Owls)-mdf ## residual df [NOT accounting for random effects]
> rdev/rdf

```

```
[1] 5.630751
```

Overdispersion is quite a bit > 1 ...significance test:

```
> (prob.disp <- pchisq(rdev,rdf,lower.tail=FALSE,log.p=TRUE))
```

```
[1] -868.7967
```

Rather unlikely! (This is a *log* probability, corresponding to  $p \approx 10^{-377}$ .)

Here (with a hacked version of lme4 that allows per-observation random effects, i.e. a Poisson-lognormal distribution):

```

> Owls$obs <- 1:nrow(Owls) ## add observation number to data
> g2 <- glmer(SiblingNegotiation~FoodTreatment*SexParent+offset(log(BroodSize))+
              (1|Nest)+(1|obs),family=poisson,data=Owls)
> print(summary(g2))

```

Generalized linear mixed model fit by the Laplace approximation

Formula: SiblingNegotiation ~ FoodTreatment \* SexParent + offset(log(BroodSize)) + (1 |

Data: Owls

```

AIC BIC logLik deviance
1882 1908 -934.9 1870

```

Random effects:

```

Groups Name      Variance Std.Dev.
obs   (Intercept) 1.24111  1.11405
Nest  (Intercept) 0.22745  0.47692

```

Number of obs: 599, groups: obs, 599; Nest, 27

Fixed effects:

	Estimate	Std. Error	z value	Pr(> z )	
(Intercept)	0.2875	0.1518	1.894	0.0582	.
FoodTreatmentSatiated	-1.1106	0.1732	-6.411	1.45e-10	***
SexParentMale	0.0180	0.1518	0.119	0.9056	
FoodTreatmentSatiated:SexParentMale	0.1797	0.2206	0.815	0.4152	

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

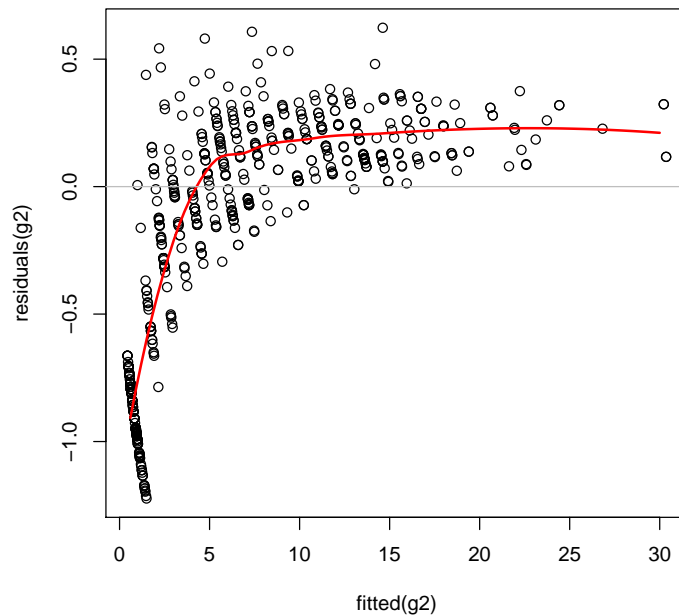
Correlation of Fixed Effects:

	(Intr)	FdTrtS	SxPrnM
FdTrtmntStt	-0.521		
SexParentMl	-0.624	0.527	
FdTrtmS:SPM	0.395	-0.766	-0.649

Considerable variation at both levels.

Examine residuals:

```
> plot(fitted(g2),residuals(g2))
> rvec <- seq(0,30,length=101)
> lines(rvec,predict(loess(residuals(g2)~fitted(g2)),newdata=rvec),
        col=2,lwd=2)
> abline(h=0,col="gray")
```



Oops ... the data didn't scream "zero-inflated" on first investigation, but now it seems as though they probably are (this is based also on a hint from Alain Zuur).

*is there a reasonably standard graphical diagnostic for zero-inflation? this graph seems pretty obvious, but maybe there's something clearer*

*Deal with zero-inflation: MCMCglmm, glmm.admb, ...*

*Plot residuals vs predictors (i.e. in this case by group (boxplot?)); plot random effects*

## 2.1 Plot predictions and confidence intervals

Proceeding as though the plot of residuals had not revealed a problem with the model ...

```
> library(ggplot2)
> G0 <- ggplot(Owls, aes(x=reorder(Nest, NegPerChick),
                          y=NegPerChick))+
  xlab("Nest")+ylab("Negotiations per chick")+coord_flip()+
  facet_grid(FoodTreatment~SexParent)
> ## boxplot display
> G1 <- G0+ geom_boxplot()
> ## dotplot display (I prefer this one)
> G2 <- G0+stat_sum(aes(size=factor(..n..)), alpha=0.5)+
  theme_bw()
```

Since there is no `predict` method for `glmer`, we'll do it by hand. (For `nest size=1` we have `offset=0` so prediction will produce negotiations/chick.)

```
> ## set up prediction frame
> pframe0 <- with(Owls, expand.grid(SexParent=levels(SexParent),
                                   FoodTreatment=levels(FoodTreatment)))
> ## construct model matrix
> mm <- model.matrix(~FoodTreatment*SexParent, data=pframe0)
> ## predictions from each model; first construct linear
> ## predictor, then transform to raw scale
> pframe1 <- data.frame(pframe0, eta=mm%*%fixef(g1))
> pframe1 <- with(pframe1, data.frame(pframe1, NegPerChick=exp(eta)))
> pframe2 <- data.frame(pframe0, eta=mm%*%fixef(g2))
> pframe2 <- with(pframe2, data.frame(pframe2, NegPerChick=exp(eta)))
```

We are using `exp(eta)` (and analogous code below) because we have used the default log link for the Poisson model. In general we will use the inverse-link function (e.g. `plogis` for logit link, the default for binomial data).

Confidence intervals: we already have the model matrix  $X$  for the points we want to predict, so we just need  $XVX^T$  to compute the per-point variances:

```
> pvar1 <- diag(mm %*% tcrossprod(vcov(g1),mm))
> pvar2 <- diag(mm %*% tcrossprod(vcov(g2),mm))
```

Add the variance due to among-nest variation. (This is intercept variation only, so we can just add the variance. If the among-nest variation affected more than the intercept, we would have to set up a design matrix and do a similar calculation to the one above.)

```
> tvar1 <- pvar1+VarCorr(g1)$Nest
> tvar2 <- pvar2+VarCorr(g2)$Nest
```

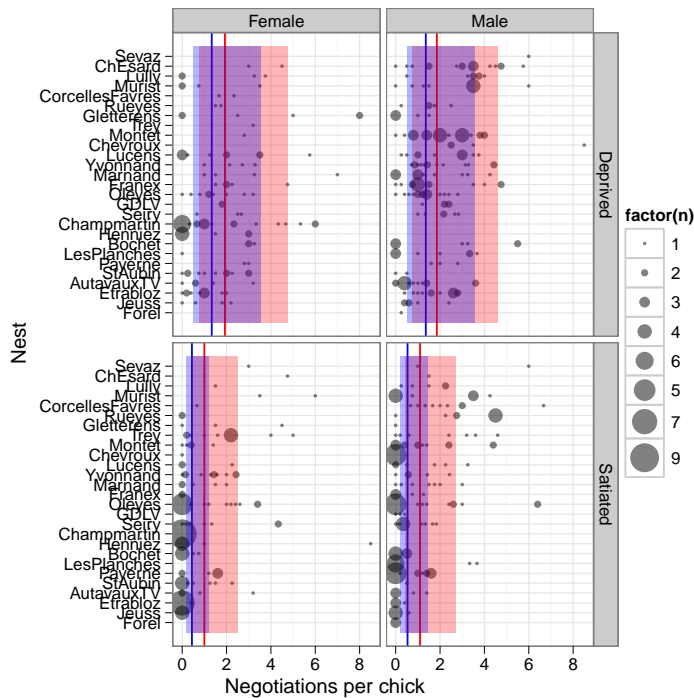
Attach standard errors, and computed confidence intervals, to prediction frames:

```
> pframe1 <- data.frame(pframe1,pse=sqrt(pvar1),tse=sqrt(tvar1))
> pframe1 <- with(pframe1,
  data.frame(pframe1,
    plo=exp(eta-1.96*pse),
    phi=exp(eta+1.96*pse),
    tlo=exp(eta-1.96*tse),
    thi=exp(eta+1.96*tse)))
> pframe2 <- data.frame(pframe2,pse=sqrt(pvar2),tse=sqrt(tvar2))
> pframe2 <- with(pframe2,
  data.frame(pframe2,
    plo=exp(eta-1.96*pse),
    phi=exp(eta+1.96*pse),
    tlo=exp(eta-1.96*tse),
    thi=exp(eta+1.96*tse)))
```

Basing confidence limits on  $\pm 1.96\sigma$  may be anticonservative in the finite-

Plot the results. Here I am plotting the predicted values for both models, as well as confidence intervals based on estimates of parameter error plus among-nest variance (`tlo` and `thi`). These are the confidence intervals on the means of a randomly selected nest in each category. I would use `plo` and `phi` to compute the confidence interval on the mean of an “average” nest, nest (i.e. not incorporating among-nest variation). If I wanted to compute prediction intervals I would probably have to do it by simulation, picking (multivariate) normally distributed values from the sampling distribution of the parameters and then simulating Poisson errors on top.

```
> print(G2 +
  geom_hline(data=pframe1,aes(yintercept=NegPerChick),col="red")+
  geom_hline(data=pframe2,aes(yintercept=NegPerChick),col="blue")+
  geom_rect(aes(xmin=0,xmax=28,ymin=tlo,ymax=thi,x=NULL),
    data=pframe1,fill="red",alpha=0.3)+
  geom_rect(aes(xmin=0,xmax=28,ymin=tlo,ymax=thi,x=NULL),
    data=pframe2,fill="blue",alpha=0.3))
```



```

> library(geepack)
> g3 <- geeglm(SiblingNegotiation~FoodTreatment*SexParent+offset(log(BroodSize)),
  corstr="exchangeable",id=Nest,
  family=poisson,data=Owls)

> g4 <- MASS::glmmPQL(SiblingNegotiation~FoodTreatment*SexParent+
  offset(log(BroodSize)),
  random=~1|Nest,
  family=poisson,data=Owls)

> g5 <- update(g4,family=quasipoisson)
> g6 <- update(g4,random=~1|Nest/obs)
> detach("package:nlme")

```

The values of the coefficients change, but the qualitative conclusion (we can detect a strong effect of satiation, but not too much else) remains the same.

```

> mtab1 <- modelTab(g1,g2,g3,hORIZ=TRUE)
> library("nlme")
> mtab2 <- modelTab(g4,g5,g6,hORIZ=TRUE)
> cbind(mtab1,mtab2)

```

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]
(Intercept)	0.656	0.288	0.659	0.654	0.654	0.654



```
FoodTreatmentSatiated          -0.656 -1.111 -0.599 -0.620 -0.620 -0.620
SexParentMale                   -0.037  0.018  0.001 -0.008 -0.008 -0.008
FoodTreatmentSatiated:SexParentMale 0.131  0.180  0.095  0.117  0.117  0.117

> detach("package:nlme")
```

### 3 To do

- Other packages.
- Confidence intervals
- Diagnostics: zero-inflation?
- QAIC etc.?

```
> library(MCMCglmm)
> MCMCglmm()
```

### References

- Roulin, A. and L. Bersier. 2007. Nestling barn owls beg more intensely in the presence of their mother than in the presence of their father. *Animal Behaviour* **74**:1099–1106. URL <http://www.sciencedirect.com/science/article/B6W9W-4PK8B6H-8/2/e43cfbaad4dc0bb2207adfc54a460c89>.
- Zuur, A. F., E. N. Ieno, N. J. Walker, A. A. Saveliev, and G. M. Smith. 2009. *Mixed Effects Models and Extensions in Ecology with R*. Springer. 1 edition.