ENV 710: Lecture 17

generalized linear mixed models

Duke | NICHOLAS SCHOOL OF THE ENVIRONMENT

generalized linear mixed models

Poisson regression

learning goals

- generalized linear models (GLM)
  - Poisson regression
- generalized linear mixed models (GLMM)

stuff you should know

generalized linear mixed models (GLMMs)

- GLMM's are GLM's with a random effect
- same as multilevel linear models, but with a response variable that is Poisson distributed – count data
- replace `glm()`, with `glmer()` from the `lme4` package or `glmmTMB()` from the `glmmTMB` package

example

salamanders and mining

Repeated samples of salamander counts were taken at 23 sites. Some of the sites were affected by mountain top removal coal mining, which is a categorical variable (yes/no) (Price et al. 2016).

let's start with a simple model with *mined* as a fixed effect and *site* as a random effect because the number of salamanders varies over the sites, but we are not specifically interested in any of the site effects

two potential functions: `glmer()` and `glmmTMB()`

```
glmer(count ~ mined + (1|site), Salamanders, family = poisson)

glmmTMB(count ~ mined + (1|site), Salamanders, family = poisson)
```

example

salamanders and mining

Repeated samples of salamander counts were taken at 23 sites. Some of the sites were affected by mountain top removal coal mining, which is a categorical variable (yes/no).

```
pm0 <- glmer(count ~ mined + (1|site), Salamanders,
       family = poisson)
 summary(pm0)

Formula: count ~ mined + (1 | site)

Random effects:
 Groups Name        Variance Std.Dev.
 site   (Intercept) 0.3313   0.5756
Number of obs: 644, groups:  site, 23

Fixed effects:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -1.5046     0.2211  -6.805 1.01e-11 ***
minedno       2.2637     0.2787   8.123 4.56e-16 ***
---
```

interpret the coefficients…

```
exp(fixef(pm0))

(Intercept)     minedno
  0.2220984   9.6189767
```

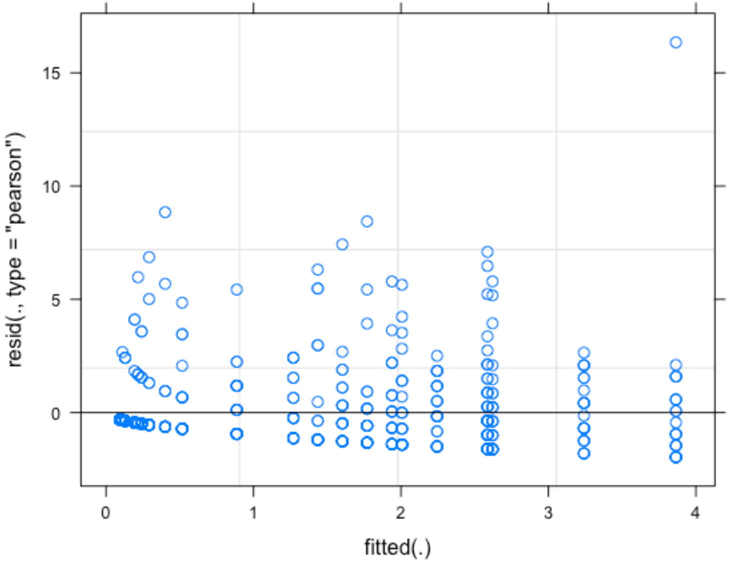
streams without mountain top mining have 9.6 times the number of salamanders!

example

salamanders and mining

Counts of salamanders in streams. Repeated samples of salamanders were taken at 23 sites. Some of the sites were affected by mountain top removal coal mining (Price et al. 2016).

```
Formula: count ~ mined + (1 | site)
Data: Salamanders

     AIC       BIC    logLik deviance df.resid
  2215.7    2229.1   -1104.8   2209.7      641
```



1 does the model fit the data well?

no! … need additional covariates to explain the data?

example

salamanders and mining

Counts of salamanders in streams. Repeated samples of salamanders were taken at 23 sites. Some of the sites were affected by mountain top removal coal mining (Price et al. 2016).

```
Formula: count ~ mined + (1 | site)
Data: Salamanders

    AIC      BIC   logLik deviance df.resid
 2215.7   2229.1  -1104.8   2209.7      641
```

2 is there overdispersion?

```
require(sjstats)
 overdisp(pm0)
```

```
dispersion ratio = 2.9207
Pearson's Chi-Squared = 1872.1541
p-value = 0.0000
```

```
deviance(pm0)/df.residual(pm0)
[1] 2.212839
```

3 coefficient of determination?

```
require(MuMIn)
 r.squaredGLMM(pm0)
```

```
                 R2m       R2c
delta      0.5931342 0.7465798
lognormal  0.6252014 0.7869430
trigamma   0.5485169 0.6904199
```

example

salamanders and mining

Counts of salamanders in streams. Repeated samples of salamanders were taken at 23 sites. Some of the sites were affected by mountain top removal coal mining (Price et al. 2016).

```
overdisp(pm4)

dispersion ratio =    0.262
  Pearson's Chi-Squared = 167.981
               p-value =       1
```

to account for overdispersion in GLMM's, use a negative binomial model or add an observation level random effect.

```
Salamanders$n <- c(1:nrow(Salamanders))
pm4 <- glmer(count ~ mined + (1|site) + (1|n), Salamanders, family = poisson)

Random effects:
 Groups Name        Variance Std.Dev.
 n      (Intercept) 1.4461   1.2026
 site   (Intercept) 0.1784   0.4224
Number of obs: 644, groups:  n, 644; site, 23

Fixed effects:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -2.1790     0.2217  -9.827   <2e-16 ***
minedno       2.3214     0.2552   9.097   <2e-16 ***
```

accounting for overdispersion doesn't change our inference.

example

stop and frisk

The ‘stops’ data on Sakai includes information on NYC police stop and frisk data (Gelman & Hill 2007). The dataset includes:

- numbers of stop and frisk data
- precinct id's, numbered 1-75
- ethnicity: 1=Black, 2=Hispanic, 3=White
- crime type: 1=violent, 2=weapons, 3=property, 4=drug
- past arrests: number of arrests of people of that ethnic group in that precinct

What factors determine the number of stops of people for frisking in NYC?

example

stop and frisk

Model stops by ethnicity

```
pois0 <- glm(stops ~ factor(eth),
             family = poisson, data = stops)
```

```
Call:
glm(formula = stops ~ factor(eth), family = poisson, data = stops)

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)   5.449936   0.003784 1440.11   <2e-16 ***
factor(eth)H -0.447714   0.006061  -73.87   <2e-16 ***
factor(eth)W -1.411119   0.008558 -164.89   <2e-16 ***
---
(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 181954  on 898  degrees of freedom
Residual deviance: 147566  on 896  degrees of freedom
AIC: 152908
```

example

stop and frisk

With the offset, stops by police are compared to the number of arrests in the previous year, so that the coefficient for the ‘Hispanic’ or ‘White’ indicator will be greater than 1 *if people in that group are stopped disproportionately to their rates of arrests, as compared to Blacks.*

```
pois1 <- glm(stops ~ factor(eth), offset = log(past.arrests),
             family = poisson, data = stops)
summary(pois1)

Call: glm(formula = stops ~ factor(eth), family = poisson, data = stops,
    offset = log(past.arrests))

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  -0.588086   0.003784 -155.40   <2e-16 ***
factor(eth)H  0.070208   0.006061   11.58   <2e-16 ***
factor(eth)W -0.161758   0.008558  -18.90   <2e-16 ***
---
(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 183981  on 898  degrees of freedom
Residual deviance: 183297  on 896  degrees of freedom
AIC: 188638
```

example

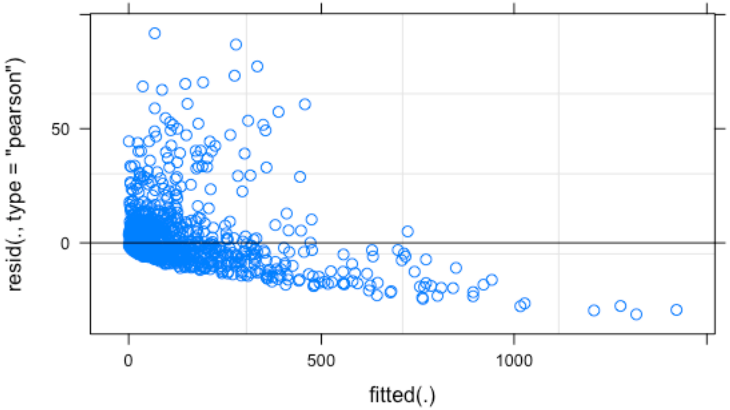
stop and frisk

With the offset, stops by police are compared to the number of arrests in the previous year, so that the coefficient for the 'Hispanic' or 'White' indicator will be greater than 1 *if people in that group are stopped disproportionately to their rates of arrests, as compared to Blacks.*

```
pois2 <- glmer(stops ~ factor(eth) + (1|precinct), offset = log(past.arrests),
               family = poisson, data = stops)
 summary(pois2)

     AIC       BIC    logLik deviance df.resid
147175.1 147194.3 -73583.5 147167.1      895

Random effects:
 Groups   Name        Variance Std.Dev.
 precinct (Intercept) 0.3653   0.6044
Number of obs: 899, groups:  precinct, 75

Fixed effects:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)  -0.445867   0.069938  -6.375 1.83e-10 ***
factor(eth)H  0.010316   0.006800   1.517    0.129
factor(eth)W -0.418563   0.009432 -44.376  < 2e-16 ***
---
```

```
pois2 <- glmer(stops ~ factor(eth) + (1|precinct), offset = log(past.arrests),
               family = poisson, data = stops)
```

```
plot(pois2)
```




```
overdisp(pois2)
```

```
# Overdispersion test

       dispersion ratio =    239.849
  Pearson's Chi-Squared = 214664.870
                p-value =    < 0.001
```

```r
stops$n <- seq(1, nrow(stops))

pois3 <- glmer(stops ~ factor(eth) + (1|precinct) + (1|n),
               offset = log(past.arrests),
               family = poisson, data = stops)
```

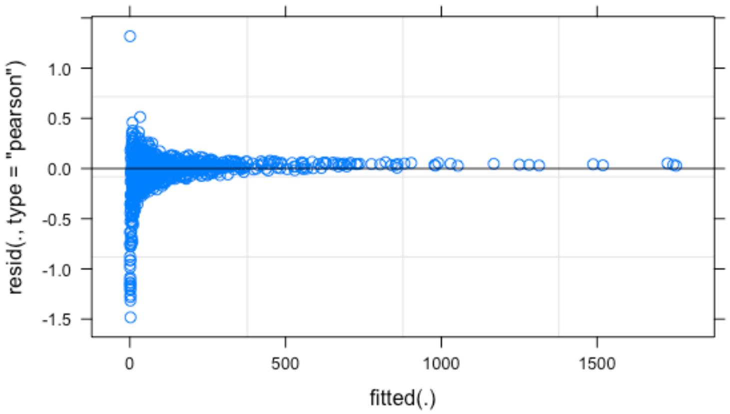
```r
summary(pois3)

     AIC       BIC    logLik deviance df.resid
 10315.1   10339.1   -5152.6  10305.1      894

Random effects:
 Groups   Name        Variance Std.Dev.
 n        (Intercept) 1.2319   1.1099
 precinct (Intercept) 0.2811   0.5302
Number of obs: 899, groups:  n, 899; precinct, 75

Fixed effects:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  -0.58471    0.08902  -6.568 5.10e-11 ***
factor(eth)H  0.05405    0.09154   0.590    0.555
factor(eth)W -0.39442    0.09269  -4.255 2.09e-05 ***
```

```r
overdisp(pois3)

# Overdispersion test

       dispersion ratio =  0.052
  Pearson's Chi-Squared = 46.727
                p-value =      1
```

```r
AIC(pois1, pois2, pois3)

      df      AIC
pois1  3 188638.5
pois2  4 147175.1
pois3  5  10315.1
```

```
stops$n <- seq(1, nrow(stops))

pois3 <- glmer(stops ~ factor(eth) + (1|precinct) + (1|n),
               offset = log(past.arrests),
               family = poisson, data = stops)
```

```
plot(pois3)
```




```
confint(pois3)
Computing profile confidence intervals ...
                    2.5 %     97.5 %
.sig01          1.0557622  1.1686892
.sig02          0.4226825  0.6615921
(Intercept)    -0.7602202 -0.4092334
factor(eth)H   -0.1256025  0.2336668
factor(eth)W   -0.5764995 -0.2127120
```

```
r.squaredGLMM(pois3)

                 R2m       R2c
delta      0.01611487 0.6262971
lognormal  0.01809103 0.7030998
trigamma   0.01320492 0.5132031
```

Questions?