

GLIMMIX

This is a new SAS procedure that fits GLMM's.

- The procedure had been available as a macro for some time.
- GLIMMIX essentially extends the MIXED procedure to GLM's, and in fact iteratively calls MIXED when fitting GLMM's.
- Only normal random effects are allowed.
- GLIMMIX uses an approximation when fitting models. The approximation in effect replaces the intractable integral that NLMIXED approximates (using quadrature) with a simple linear Taylor's expansion. It's crude, but can work and it's fast. See pp. 119–125 in SAS' GLIMMIX documentation for details on "Pseudo-likelihood Estimation Based on Linearization."

- The model is $E\{\mathbf{Y}|\boldsymbol{\gamma}\} = g^{-1}\{\mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\boldsymbol{\gamma}\}$ where $\boldsymbol{\gamma} \sim N_q(\mathbf{0}, \boldsymbol{\Sigma})$. Also, $var\{\mathbf{Y}|\boldsymbol{\gamma}\} = \mathbf{A}^{1/2}\mathbf{R}\mathbf{A}^{1/2}$ where $\mathbf{A}^{1/2}$ comes from the sampling model (e.g. Poisson, normal, binomial) and \mathbf{R} is a ‘marginal’ covariance matrix. For GLMM’s, $\mathbf{R} = \phi\mathbf{I}$ and so $var\{\mathbf{Y}|\boldsymbol{\gamma}\} = \phi\mathbf{A}$ where \mathbf{A} is a diagonal matrix. This is because for GLMM’s, the responses within a cluster are independent given the random effects $\boldsymbol{\gamma}$.
- GLIMMIX can also fit marginal models allowing for correlation within a cluster (like GENMOD), but uses a different estimation method than GENMOD with the repeated statement. Then \mathbf{R} has structure, e.g. exchangeability (called compound symmetry here), AR structure, spatial structures, and others found in PROC MIXED.
- The learning curve is steep, although it’s nice to be aware of alternative fitting procedures if necessary!

Ache monkey hunting

Data on the number of capuchin monkeys killed by 47 Ache hunters over several hunting trips were recorded. There were 363 total records. I'll describe the hunting process in class; it involves splitting into groups, chasing monkeys through the trees, and shooting arrows straight up.

Let Y_{ij} be the number of monkey's killed by hunter i , $i = 1, \dots, 47$ on trip j of length L_{ij} (the trip length serves as an 'offset' in the model fitting). Let λ_i be the hunter i 's kill rate (per day).

$$Y_{ij} \sim \text{Poisson}(\lambda_i L_{ij}),$$

where

$$\log \lambda_i = \beta_0 + \beta_1 a_i + \beta_2 a_i^2 + u_i,$$

$$u_1, \dots, u_{47} \stackrel{iid}{\sim} N(0, \sigma^2).$$

- Monkey hunting is dangerous.
- We include a quadratic effect because we expect a “leveling off” effect or possible decline in ability with age.
- Of interest is when hunting ability is greatest. Hunting prowess contributes to a man’s status within the group. a_i is hunter i ’s age-45 years.
- An individual’s kill rate is given by $\lambda = e^{\beta_0 + \beta_1 a + \beta_2 a^2} e^u$, where a is the individual’s age and u is their latent hunting ability.
- One can compare the effect of age within the span of, say, 20 to 60 years, to the spread of e^u to see which explains more of the variability in terms of hunting ability: age or innate ability.

Data sorted by trip number:

```
data ache1;
  input TRIP$ PID$ AGE nkills tripdays; ltripday=log(tripdays); age=age-45;
  datalines;
C082697A    3394    31  1  4
C082697A    3327    38  0  4
C082697A    3313    39  0  4
C082697A    3220    50  0  4
C082697A    3157    56  0  4
C082697A    3146    57  0  4
C082697A    3144    58  1  4
C082697A    7089    59  1  4
C082697A    3126    60  2  4
C082697A    7085    62  1  4
C102197A    3394    31  1  3
C102197A    3327    38  0  3
C102197A    3238    48  3  3
C102197A    3220    50  0  3
C102197A    3144    58  2  3
C102197A    3086    67  0  3

...et cetera...

T120997A    3182    53  0  5
T120997A    3094    65  0  5
T121597A    3254    46  0  4
T121597A    3128    60  0  4
;
```

With calls to genmod, nlmixed, and glimmix:

```
proc genmod data=ache1;
  class pid;
  model nkills=age age*age / dist=poisson link=log offset=ltripday;
  repeated subject=pid / type=exch;

proc glimmix data=ache1; class pid;
  model nkills = age age*age / dist=pois link=log offset=ltripday solution;
  random _residual_ / subject=pid type=cs;

proc sort; by pid; run; * need to sort by subject!;
proc nlmixed qpoints=100 data=ache1;
  parms b1=-2.3 b2=0.0251 b3=-0.002 v=1.0;
  eta=b1+b2*age+b3*age**2+u+ltripday;
  lambda=exp(eta);
  model nkills ~ poisson(lambda);
  random u ~ normal(0,v) subject=pid;

proc glimmix data=ache1; class pid;
  model nkills = age age*age / dist=pois link=log offset=ltripday solution;
  random intercept / subject=pid;
```

The GENMOD Procedure

Class	Levels	Values
PID	47	3086 3094 3111 3126 3128 3139 3144 3146 3157 3166 3172 3182 3217 3220 3238 3240 3254 3302 3313 3316 3322 3327 3349 3371 3378 3386 3390 3394 3401 3405 3414 3416 3434 3436 3450 3465 3480 3486 3495 3525 3529 3548 3572 7032 7085 7089 8024

GEE Model Information

Correlation Structure	Exchangeable
Subject Effect	PID (47 levels)
Number of Clusters	47
Correlation Matrix Dimension	28
Maximum Cluster Size	28
Minimum Cluster Size	1

Exchangeable Working
Correlation

Correlation 0.2180742191

Empirical Standard Error Estimates

Parameter	Estimate	Standard Error	95% Confidence Limits		Z	Pr > Z
Intercept	-2.2901	0.3266	-2.9303	-1.6500	-7.01	<.0001
AGE	0.0141	0.0257	-0.0363	0.0645	0.55	0.5840
AGE*AGE	-0.0019	0.0015	-0.0049	0.0010	-1.30	0.1937

The GLIMMIX Procedure

Model Information

Response Variable	nkills
Response Distribution	Poisson
Link Function	Log
Variance Function	Default
Offset Variable	ltripday
Variance Matrix Blocked By	PID

Dimensions

R-side Cov. Parameters	2
Subjects (Blocks in V)	47
Max Obs per Subject	28

Covariance Parameter Estimates

Cov Parm	Subject	Estimate	Standard Error
CS	PID	0.2730	0.1077
Residual		1.8348	0.1422

Solutions for Fixed Effects

Effect	Estimate	Standard Error	DF	t Value	Pr > t
Intercept	-2.3253	0.2836	46	-8.20	<.0001
AGE	0.01589	0.01657	314	0.96	0.3385
AGE*AGE	-0.00181	0.001374	314	-1.31	0.1898

The NL MIXED Procedure

Specifications

Dependent Variable	nkills
Distribution for Dependent Variable	Poisson
Random Effects	u
Distribution for Random Effects	Normal
Subject Variable	PID
Optimization Technique	Dual Quasi-Newton
Integration Method	Adaptive Gaussian Quadrature

Dimensions

Total Observations	363
Subjects	47
Max Obs Per Subject	28
Parameters	4
Quadrature Points	100

Parameter Estimates

Parameter	Estimate	Standard Error	DF	t Value	Pr > t	Alpha	Lower	Upper
b1	-2.6229	0.4515	46	-5.81	<.0001	0.05	-3.5317	-1.7141
b2	0.03385	0.02521	46	1.34	0.1859	0.05	-0.01689	0.08458
b3	-0.00491	0.002280	46	-2.16	0.0364	0.05	-0.00950	-0.00033
v	2.1081	0.8926	46	2.36	0.0225	0.05	0.3115	3.9048

The GLIMMIX Procedure

Model Information

Response Variable	nkills
Response Distribution	Poisson
Link Function	Log
Variance Function	Default
Offset Variable	ltripday
Variance Matrix Blocked By	PID

Dimensions

G-side Cov. Parameters	1
Subjects (Blocks in V)	47
Max Obs per Subject	28

Covariance Parameter Estimates

Cov Parm	Subject	Estimate	Standard Error
Intercept	PID	1.7965	0.6505

Solutions for Fixed Effects

Effect	Estimate	Standard Error	DF	t Value	Pr > t
Intercept	-2.4222	0.4113	46	-5.89	<.0001
AGE	0.02889	0.02307	314	1.25	0.2115
AGE*AGE	-0.00405	0.002079	314	-1.95	0.0525

- Notice the similarities in the GENMOD and GLIMMIX output fitting (the first two) marginal models.
- Notice the similarities in the GENMOD and GLIMMIX output fitting (the last two) conditional GLMM models.
- The quadratic effect is significant in the random effects models, but not the marginal models. This often happens when you focus on the individual.
- One benefit of fitting conditional random effects models: prediction is possible!

Question: How would I fit this data? **Answer:** Either in NLMIXED, WinBUGS, or using DPpackage in R:

```
d <- read.table("c:/tim/ron/Monkey2.txt")
ache <- d[,1]; age <- d[,2]; days <- d[,4]; kills <- d[,3];
ldays=log(days); age <- age-45; agesq <- age*age
int <- rep(1,47) # intercept term
prior <- list(alpha=1000,M=1,nu0=5,tinv=diag(6,1),mu=rep(0,1),
  beta0=rep(0,3),Sbeta0=diag(100,3),frstlprob=TRUE)
mcmc <- list(nburn=100,nsave=1000,nskip=20,ndisplay=100,tune2=0.5)
state <- list(beta=c(-2.6,0.04,-0.005),sigma=diag(0.1,1),alpha=0.1,b=matrix(rep(0,47),47,1),mu=rep(0,1),phi=0)
fit3 <- PTglmm(fixed=kills~int+age+agesq,offset=ldays,random=~1|ache,family=poisson(log),
  prior=prior,mcmc=mcmc,status=FALSE,state=state)
```

With condensed output (from `summary(fit3)`):

Bayesian semiparametric generalized linear mixed effect model

Model's performance:

Dbar	Dhat	pD	DIC	LPML
101.01	78.24	22.76	123.77	-71.00

Regression coefficients:

	Mean	Median	Std. Dev.	Naive Std.Error	95%HPD-Low	95%HPD-Upp
(Intercept)	0.000e+00	0.000e+00	0.000e+00	0.000e+00	0.000e+00	0.000e+00
int	-2.485e+00	-2.466e+00	4.803e-01	1.519e-02	-3.404e+00	-1.561e+00
age	3.814e-02	3.661e-02	2.588e-02	8.184e-04	-1.611e-02	8.459e-02
agesq	-6.160e-03	-5.898e-03	2.702e-03	8.544e-05	-1.109e-02	-7.913e-04

Baseline distribution:

	Mean	Median	Std. Dev.	Naive Std.Error	95%HPD-Low	95%HPD-Upp
mu-(Intercept)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
sigma-(Intercept)	2.33992	2.11703	0.97315	0.03077	0.85191	4.27001

Acceptance Rate for Metropolis Steps = 0.8327014 0.6898195 0 0.5627488

Number of Observations: 47

Number of Groups: 47

- NLMIXED estimates $(\hat{\beta}_0, \hat{\beta}_1, \hat{\beta}_2) = (-2.62, 0.034, -0.0049)$ and $\hat{\sigma}^2 = 2.11$.
GLIMMIX estimates $(\hat{\beta}_0, \hat{\beta}_1, \hat{\beta}_2) = (-2.42, 0.029, -0.0041)$ and $\hat{\sigma}^2 = 1.80$.
PTg1mm estimates $(\hat{\beta}_0, \hat{\beta}_1, \hat{\beta}_2) = (-2.47, 0.037, -0.0059)$ and $\hat{\sigma}^2 = 2.12$.
- From the Bayesian fit, we find the maximum kill rate occurs at 48.3 years with a 95% credible interval of (44.1, 60.4) years.
- Other variables (besides age) were not found to be significant.

- Why GLIMMIX? To easily handle crossed or nested random effects. To handle large dimensional random effects. To jointly model counts and continuous outcomes. To avoid waiting 3 hours for NLMIXED to converge. To fit spatial covariance and other complex covariance structures with GLM's that cannot be accommodated by GENMOD.
- Why not GLIMMIX? It uses approximations which can bias results. You don't know how biased your results actually are. However, most models are approximations to reality to begin with so maybe not that big of a deal.
- Bayesian approach also natural but not as fast or easy to implement. However, no approximations are used and inference is exact up to Monte Carlo error.
- There are other packages out there to perform similar analyses.