A Zero-Inflated Overdispersed Hierarchical Poisson Model

Wondwosen Kassahun[1] **Thomas Neyens**[2] **Christel Faes**[2]
Geert Molenberghs[2,3] **Geert Verbeke**[3,2]

[1] *Department of Epidemiology and Biostatistics, Jimma University, Ethiopia*
[2] *I-BioStat, CenStat, Universiteit Hasselt, B-3590 Diepenbeek, Belgium*
[3] *I-BioStat, L-BioStat, Katholieke Universiteit Leuven, B-3000 Leuven, Belgium*

Supplementary Material: SAS Procedure NLMIXED Code

A Sample PROC NLMIXED Code

```
/*
Analyses for the epilepsy Data
treatment 0= placebo
treatment 1= treatment
y=nseizw
time=studyweek
*/

/*(P--) Model */
proc nlmixed data=epilepsy qpoints=20;
title 'Univariate analyse';
parms int0=0.5 slope0=-0.1 int1=1 slope1=0.1;
if (trt = 0) then eta = int0 + slope0*time;
else if (trt = 1) then eta = int1  + slope1*time;
lambda = exp(eta);
loglik=-lambda+y*eta-log(fact(y));
model y~ general(loglik);
estimate 'difference in slope' slope1-slope0;
estimate 'ratio of slopes' slope1/slope0;
predict exp(-lambda) out=P;
run;

/*Average predicted probability of zeros*/
proc means data=P;
var pred;
run;
```

```
/*ZI(P--) Model */
proc nlmixed data=epilepsy qpoints=20;
title 'Univariate analyse';
parms int0=0.5 slope0=-0.1 int1=1 slope1=0.1 a0=0 a1=0;
eta_prob = a0+ a1*time ;
p_0 = exp(eta_prob) / (1 + exp(eta_prob));
if (trt = 0) then eta = int0 + slope0*time;
else if (trt = 1) then eta = int1  + slope1*time;
lambda = exp(eta);
if y = 0 then loglik = log(p_0 + (1 - p_0) * exp(-lambda));
else loglik = log(1 - p_0) + y * log(lambda)- lambda  - lgamma(y+1);
model y~ general(loglik);
estimate 'difference in slope' slope1-slope0;
estimate 'ratio of slopes' slope1/slope0;
predict p_0 +(1-p_0)*exp(-lambda) out=ZIP;
run;

/*Average predicted probability of zeros*/
proc means data=ZIP;
var pred;
run;

/*(PN-) Model */
proc nlmixed data=epilepsy qpoints=20;
title 'Poisson-normal met general likelihood';
parms int0=0.5 slope0=-0.1 int1=1 slope1=0.1 sigma=1;
if (trt = 0) then eta = int0 + b + slope0*time;
else if (trt = 1) then eta = int1 + b + slope1*time;
lambda = exp(eta);
loglik=-lambda+nseizw*eta-log(fact(y));
model nseizw ~ general(loglik);
random b ~ normal(0,sigma**2) subject = id;
estimate 'difference in slope' slope1-slope0;
estimate 'ratio of slopes' slope1/slope0;
predict exp(-lambda) out=PN;
run;

/*Average predicted probability of zeros*/
proc means data=PN;
var pred;
run;

/*ZI(PN-) Model*/
```

```
proc nlmixed data=epilepsy qpoints=20;
title 'Poisson-normal met general likelihood';
parms int0=0.8179 slope0=-0.014 int1=0.647 slope1=-0.012 d11=0.98
       rho=0 d22=1.10 a0=-3 a1=0.1;
eta_prob = a0+ a1*time+b2 ;
p_0 = exp(eta_prob) / (1 + exp(eta_prob));
if (trt = 0) then eta = int0 + b1 + slope0*time;
else if (trt = 1) then eta = int1 + b1 + slope1*time;
lambda = exp(eta);
if y = 0 then loglik = log(p_0 + (1 - p_0) * exp(-lambda));
else loglik = log(1 - p_0) + y * log(lambda) - lambda - log(fact(y));
random b1 b2 ~ normal([0,0], [d11**2,rho*d11*d22,d22**2]) subject = id;
model y ~ general(loglik);
estimate 'difference in slope' slope1-slope0;
estimate 'ratio of slopes' slope1/slope0;
predict p_0+(1-p_0)*exp(-lambda) out=ZIPN;
run;

/*Average predicted probability of zeros*/
proc means data=ZIPN;
var pred;
run;

/*(P-G) Model*/
proc nlmixed data=epilepsy qpoints=20;
title 'Poisson-gamma == negative-binomial - alpha*beta=1';
parms int0=0.5 slope0=-0.1 int1=1 slope1=0.1 alpha=2;
if (trt = 0) then eta = int0 + slope0*time;
else if (trt = 1) then eta = int1 + slope1*time;
lambda = exp(eta);
beta=1/alpha;
loglik=lgamma(alpha+y)-lgamma(alpha)+y*log(beta)-(y+alpha)*log(1+beta*lambda)
       +y*eta-lgamma(y+1);
model y ~ general(loglik);
estimate 'difference in slope' slope1-slope0;
estimate 'ratio of slopes' slope1/slope0;
estimate 'beta=1/alpha' 1/alpha;
predict (1/(1+lambda/beta))**beta out=PG;
run;

/*Average predicted probability of zeros*/
proc means data=PG;
var pred;
```

```
run;

/*ZI(P-G) Model*/
proc nlmixed data=epilepsy qpoints=20;
title 'Poisson-gamma == negative-binomial - alpha*beta=1';
parms int0=0.5 slope0=-0.1 int1=1 slope1=0.1 alpha=0.05 a0=-1 a1=0.1;
if (trt = 0) then eta = int0 + slope0*time;
else if (trt = 1) then eta = int1 + slope1*time;
lambda = exp(eta);
eta_prob=a0+a1*time;
p_0=exp(eta_prob)/(1+exp(eta_prob));
m = 1/alpha;
p = 1/(1+alpha*lambda);
if y=0 then
ll = log(p_0+ (1-p_0)*(p**m));
else ll = log(1-p_0) + log(gamma(m + y)) - log(gamma(y + 1))
     - log(gamma(m)) + m*log(p) + y*log(1-p);
model y ~ general(ll);
estimate 'difference in slope' slope1-slope0;
estimate 'ratio of slopes' slope1/slope0;
estimate 'beta=1/alpha' 1/alpha;
predict p_0 +(1-p_0)*(1/(1+lambda/alpha))**alpha out=ZIPG ;
run;

/*Average predicted probability of zeros*/
proc means data=ZIPG;
var pred;
run;

/*(PNG) Model*/
proc nlmixed data=epilepsy  qpoints=20;
parms int0=0.5 slope0=-0.1 int1=1 slope1=0.1 sigma=1 alpha=1 ;
if (trt = 0) then eta = int0 + b + slope0*time;
else if (trt = 1) then eta = int1 + b + slope1*time;
lambda = exp(eta);
beta=1/alpha;
loglik=lgamma(alpha+y)-lgamma(alpha)+y*log(beta)-(y+alpha)*log(1+beta*lambda)
       +y*eta-lgamma(y+1);
random b ~ normal(0,sigma**2) subject = id ;
model y~ general(loglik);
estimate 'difference in slope' slope1-slope0;
estimate 'ratio of slopes' slope1/slope0;
estimate 'beta=1/alpha' 1/alpha;
```

```
predict (1/(1+lambda/alpha))**alpha out=PNG;
run;

/*Average predicted probability of zeros*/
proc means data=PNG;
var pred;
run;

/*ZI(PNG) Model*/
proc nlmixed data=epilepsy qpoints=20;
title 'Poisson-combined -  alpha*beta=1';
parms int0= 0.8511 slope0=-0.01048 int1=0.8165 slope1=-0.008 alpha=0.2937
d11=1.0810 rho=0 d22=3.19 a0=-1.78 a1=0.052;
if (trt = 0) then eta = int0 + b1 + slope0*time;
else if (trt = 1) then eta = int1 + b1 + slope1*time;
lambda = exp(eta);
eta_prob = a0+a1*time+b2 ;
p_0=exp(eta_prob)/(1+exp(eta_prob));
m = 1/alpha;
p = 1/(1+alpha*lambda);
if y=0 then
ll = log(p_0 + (1-p_0)*(p**m));
else ll = log(1-p_0) + log(gamma(m + y)) - log(gamma(y + 1))
     - log(gamma(m)) + m*log(p) + y*log(1-p);
model y ~ general(ll);
random b1 b2 ~ normal([0,0], [d11**2,rho*d11*d22,d22**2]) subject = id;
estimate 'difference in slope' slope1-slope0;
estimate 'ratio of slopes' slope1/slope0;
estimate 'beta=1/alpha' 1/alpha;
predict p_0+(1-p_0)*(1/(1+lambda/m))**m out=ZIPNG;
run;

/*Average predicted probability of zeros*/
proc means data=ZIPNG;
var pred;
run;
```