CHAPTER 14

SAS Code: Joint Models for Continuous and Discrete Longitudinal Data

We show how models of a mixed type can be analyzed using standard statistical software. We mainly focus on the SAS procedures PROC NLMIXED and PROC GLIMMIX, and show how these programs can be used to jointly analyze a continuous and binary outcome. The models are illustrated using the Irwin’s toxicity study, as discussed in Chapter 14.

In the Irwin’s toxicity study, the dosage (‘dose’), time after dosing (‘time’) and the day of treatment (‘day’) are given. The continuous response (Pupil Size) and binary response (Toe pinch) are summarized in one variable (‘resp’). The variable ‘id’ is the subject’s indicator. The relevant variables for the analysis, for the first 5 subjects, are

```
Obs    id    Dose    time    Day        name          resp

  1     1      0       1      0     Toe pinch         1.000
  2     1      0       1      0     Pupil Size        0.700

  3     1      0       1      1     Toe pinch         1.000
  4     1      0       1      1     Pupil Size        0.675

  5     1      0       1      2     Toe pinch         1.000
  6     1      0       1      2     Pupil Size        0.800

  7     1      0       2      0     Toe pinch         1.000
  8     1      0       2      0     Pupil Size        0.675

  9     1      0       2      1     Toe pinch         1.000
 10     1      0       2      1     Pupil Size        0.725
 ...
```

We will refer to this dataset as ‘mcd’.

Since different distributions and link functions have to be used for the different outcomes, we use a special device available in the SAS procedure PROC GLIMMIX, i.e., the ‘byobs=(.)’ specification that can be used to specify both the distribution in the ‘dist=’ option and the link function in the ‘link=’ option. Thus, before we start with the main analysis, two variables need to be created to specify the outcome distribution and link function for each observation in the dataset. For example, assume ‘distvar’ is a variable denoting a Gaussian distribution for the continuous measurement and a Bernouilli one for the binary measurement. Then, the option ‘dist=byobs(distvar)’ specifies a normal distribution for the first measurement of every subject, and a Bernouilli distribution for the second one. The following code creates a distributional indicator for the continuous outcome ‘Pupil Size’ and the binary outcome ‘Toe Pinch’, as well as a link function indicator.

```
data mcd;
length dist $11;
set mcd;
if name="Pupil Size" then distvar="Normal";
if name="Toe pinch"  then distvar="Binary";
if name="Pupil Size" then linkvar="IDEN";
if name="Toe pinch"  then linkvar="LOGIT";
run;
```

14.1 Marginal Generalized Linear Model

As in the example, we will first focus on one single time point. For fitting the marginal generalized linear model, we can now use the program:

```
proc glimmix data=mcd (where=(Day=0 and Time=8)) method=rspl;
class id distvar;
nloptions maxiter=50 technique=newrap;
model resp(event='1') = distvar distvar*dose
                /noint s  distvar=byobs(distvar);
random _residual_/ subject=id type=unr;
output out=out predicted(blup ilink)=pred;
run;
```

Note that there is no link function specification in this program, implying that the default link functions corresponding with the Gaussian and Bernouilli distribution are used. Specifying the option ‘link=byobs(linkvar)’ would produce exactly the same model. Since we want a separate intercept (α_0 and β_0, respectively) and a separate treatment effect (α_1 and β_1) for the two outcomes, the variable ‘distvar’ is used as an interaction in the fixed effects structure, through the MODEL statement. Note that there is no need to estimate a general intercept, which can be done by including ‘noint’ into the MODEL statement options, since inclusion of the variable ‘distvar’ as a main effect in the model produces already a separate intercept for the two outcomes. The NLOPTIONS statement is included to control convergence. Since convergence can be an issue, the user may need to change the iterative technique, the maximum number of iterations and the convergence tolerance.

The estimates of the fixed effects parameters are

```
                        Solutions for Fixed Effects

                                        Standard
 Effect       dist      Estimate          Error    DF    t Value    Pr > |t|

 dist         Binary     -1.3863         1.1902    17      -1.16      0.2602
 dist         Normal      6.5500         1.3723    17       4.77      0.0002
 Dose*dist    Binary      1.3863         1.3396    17       1.03      0.3152
 Dose*dist    Normal      6.4708         1.6333    17       3.96      0.0010
```

These parameters present the two intercepts β_0 and α_0, and the treatment effect β_1 and α_1, respectively. The option ‘event='1”' in the MODEL statement, ensures that we are modeling the probability of an event. The estimates of the covariance parameters are

```
        Covariance Parameter Estimates
```

```
                                                Standard
   Cov Parm     Subject    Estimate         Error

   Var(1)       id           1.1333        0.4138
   Var(2)       id           9.4155        3.4380
   Corr(2,1)    id         -0.04337        0.2577
```

The parameter 'Var(1)' is the variance of the Binary outcome and as such merely is an overdispersion parameter, the parameter 'Var(2)' is the variance of the Normal outcome. Finally, 'Corr(2,1)' is the correlation between both. Equivalently, one could specify the structure option for the covariance matrix in the RANDOM statement to 'type=UN' rather than the 'type=UNR' structure used above. The above panel changes to:

```
       Covariance Parameter Estimates

                                                Standard
  Cov Parm      Subject     Estimate          Error

   UN(1,1)       id           1.1333        0.4138
   UN(2,1)       id          -0.1417        0.8442
   UN(2,2)       id           9.4155        3.4380
```

Here, 'UN(1,1)' and 'UN(2,2)' represent the variance of the Binary and Normal distribution, and 'UN(2,1)' is the covariance between the two outcomes. Of course, the two options are equivalent. The correlation can be calculated without problem from the two variances and covariance, and the standard error from the asymptotic covariance matrix of the variance parameter, and vice versa. However, depending on the parameter of interest, one could opt for one or the other option, yielding the estimates with standard error directly.

14.2 Shared Parameter Model

Next, let us change focus to the random-effects shared parameter model. While the procedure GLIMMIX can be used to model a random-effects model in case of two outcomes of the same type, incorporation of the scale parameter λ which is necessary in case of modeling two outcomes of a different type is not possible. The procedure NLMIXED can be used. The following code, using the general likelihood feature, is used for the random-effects model for a continuous and binary outcome:

```
proc nlmixed data=mcd qpoints=20 maxiter=100
              maxfunc=2000 technique=newrap;
parms beta11=6.4 beta12=2.6 beta21=-0.3
      beta22=-0.3 sigma=0.1 lambda=0.6 tau=0.5;
if dist = "Normal" then do;
   mean = beta11 + beta12*dose + lambda*b;
   dens = -0.5*log(3.14159265358) - log(sigma)
          -0.5*(resp-mean)**2/(sigma**2);
   ll = dens;
end;
if dist = "Binary" then do;
   eta = beta21 + beta22*dose  + b;
```

```
   p = exp(eta)/(1+exp(eta));
   ll = resp*log(p) + (1-resp)*log(1-p);
end;
model resp ~ general(ll);
random b ~ normal(0,tau*tau) subject=id;

v1=exp(beta21+beta22)/(1+exp(beta21+beta22))**2;
v0=exp(beta21)/(1+exp(beta21))**2;
estimate 'rho at dose=0'
     (v0*lambda*tau**2)/sqrt(((lambda*tau)**2+sigma**2)*((v0**tau)**2+v0));
estimate 'rho at dose=1'
     (v1*lambda*tau**2)/sqrt(((lambda*tau)**2+sigma**2)*((v1**tau)**2+v1));
estimate 'sigma^2' sigma*sigma;
estimate 'tau^2' tau*tau;
run;
```

In should be noted that for each line in the dataset a contribution to the likelihood function, either based on a normal distribution or on a Bernouilli distribution, is calculated. The correlation among the two endpoints is estimated using formula (**??**). This model often has convergence problems, due to the nonlinear parameter λ, and a careful selection of starting values, and fine tuning of the optimization method may be required.

14.3 Hierarchical models

Let us now switch to the hierarchical models, as presented in Section **??**.

A program for the marginal generalized linear mixed model, using the GLIMMIX procedure, is

```
proc glimmix data=mcd method=rspl;
class id dist day;
nloptions maxiter=50 technique=newrap;
model resp(event='1') = dist dist*dose dist*t
              dist*day dist*t*day dist*dose*t dist*dose*day
             /noint s  dist=byobs(dist);
random _residual_/ subject=id type=UNR;
run;
```

Again, reaching convergence is not straightforward because of the many parameters to estimate in the covariance matrix. Therefore, changing the optimization options might be necessary.

A straightforward extension of the previous model is the random-intercepts model with correlated residual error.

```
proc glimmix data=mcd method=rspl;
class id dist pair day_class;
nloptions maxiter=50 technique=newrap;
model resp(event='1') = dist dist*dose dist*t
              dist*day dist*t*day dist*dose*t dist*dose*day
              /noint s  dist=byobs(dist);
random int/ subject=id group=dist;
```

```
random _residual_/ subject=pair type=UNR;
run;
```

Here, two RANDOM statements are used. The first one is to specify the random intercept to account for the longitudinal structure in the data. The option 'group=dist' will produce two independent random effects for the two endpoints. The residual error structure is specified using the second RANDOM statement. Here, the subject 'pair' is an identification number for all pairs of observations (for all subject, and for all different time points).

Finally, we again use the procedure NLMIXED for the uncorrelated random-intercepts model. The following code can be used

```
proc sort data=mcd;
by pair;
run;
proc nlmixed data=mcd
parms beta11=0.7 beta13=0.9 beta14=0.0 beta15=-0.1 beta16=0.01
      beta17=0.02 beta18=-0.2 beta21=2.0 beta23=-0.6 beta24=-0.35
      beta25=-1.2 beta26=-0.03 beta27=0.3 beta28=0.6 sigma=2.55
      tau1=0.25 tau2=0.3;
if name = "Pupil Size" then do;
   mean = beta11 +beta13*dose_class + beta14*t + beta15*day + beta16*t*day +
          beta17*dose_class*t +beta18*dose_class*day + u1;
   dens = -0.5*log(3.14159265358) - log(sigma)
          -0.5*(resp-mean)**2/(sigma**2);
   ll = dens;
end;
if name = "Toe pinch" then do;
   eta = beta21 +beta23*dose_class + beta24*t + beta25*day + beta26*t*day +
         beta27*dose_class*t +beta28*dose_class*day + u2;
   p = exp(eta)/(1+exp(eta));
   ll = (resp)*log(p) + (1-resp)*log(1-p);
end;
model resp ~ general(ll);
random u1 u2 ~normal([0,0],[tau1**2,rho*tau1*tau2,tau2**2]) subject=id;
estimate 'tau1^2' tau1*tau1;
estimate 'tau2^2' tau2*tau2;
estimate 'sigma^2' sigma*sigma;
run;
```

This is a straightforward extension of previously described models, where ρ is included to account for the correlation among the binary and continuous outcomes.