How to fit longitudinal data in PROC MCMC ?

Hennion Maud, maud.hennion@arlenda.com
Rozet Eric, eric.rozet@arlenda.com

Arlenda

What’s the purpose of this presentation

- Due to continuous monitoring, longitudinal data are generally available during clinical development. A lack of understanding concerning appropriate longitudinal data analysis methods can lead to inappropriate or inefficient analysis (inaccurate results, wrong interpretations, wrong conclusion). **Using simplistic analytic methods can scuttle detection of important signal and effects, even in well-designed and -conducted studies.**

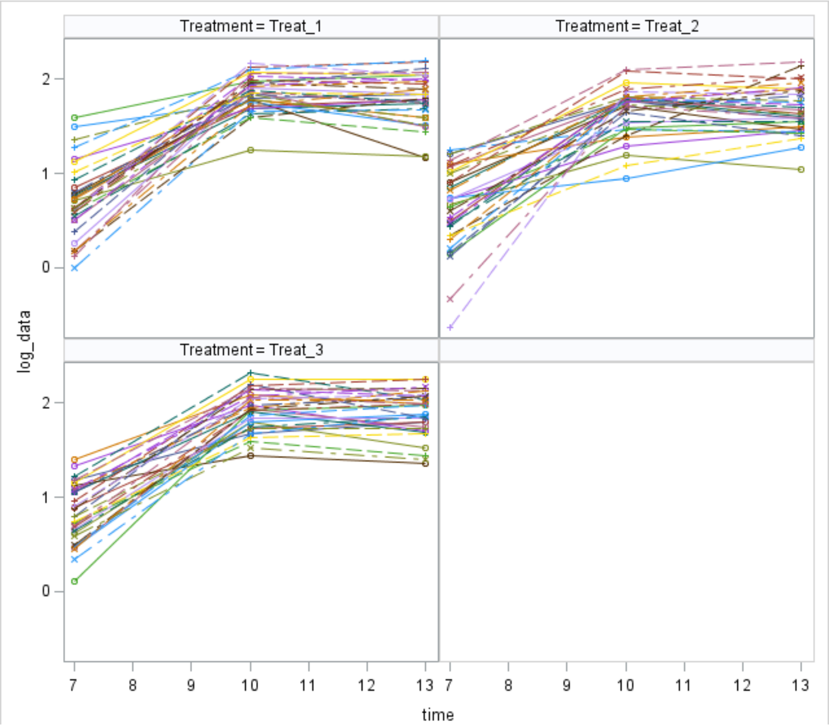
- What I will show you in this presentation:
  - An easy way to fit longitudinal data for a large pattern of models and variance-covariance structures
  - How to do it with PROC MCMC

- What I won’t show you in this presentation:
  - A course on Bayesian statistics
  - A course on longitudinal model

An example of longitudinal data

- Three treatments are tested on 35 subjects.

- For each subject, we have
  - 3 repeated measures across time of Treat_1 (+ 1 baseline measure)
  - 3 repeated measures across time of Treat_2 (+ 1 baseline measure)
  - 3 repeated measures across time of Treat_3 (+ 1 baseline measure)



Assumption on the longitudinal data

- Repeated measures of a treatment j ($j = 1, 2, 3$) within a subject i are correlated.

$$R_{ij} = \begin{pmatrix} \boldsymbol{\sigma^2} & \boldsymbol{\rho\sigma^2} & \boldsymbol{\rho^2\sigma^2} \\ \boldsymbol{\rho\sigma^2} & \boldsymbol{\sigma^2} & \boldsymbol{\rho\sigma^2} \\ \boldsymbol{\rho^2\sigma^2} & \boldsymbol{\rho\sigma^2} & \boldsymbol{\sigma^2} \end{pmatrix}$$

- The treatments are independent.

$$R_i = \begin{pmatrix} R_{\boldsymbol{i}1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & R_{i2} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & R_{i3} \end{pmatrix}$$

- The observations of a patient are correlated (random effect on the intercept)

$$V(Y_i) = \begin{pmatrix} \sigma_g^2 & \sigma_g^2 & \sigma_g^2 \\ \sigma_g^2 & \sigma_g^2 & \sigma_g^2 \\ \sigma_g^2 & \sigma_g^2 & \sigma_g^2 \end{pmatrix} + \begin{pmatrix} R_{\boldsymbol{i}1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & R_{i2} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & R_{i3} \end{pmatrix}$$

$$\sigma_g^2 = \begin{pmatrix} \sigma_g^2 & \sigma_g^2 & \sigma_g^2 \\ \sigma_g^2 & \sigma_g^2 & \sigma_g^2 \\ \sigma_g^2 & \sigma_g^2 & \sigma_g^2 \end{pmatrix}$$

Model on longitudinal data – proc mixed

- Let Y_i , the vector of the observations of patient i

$$Y_i = \boxed{X_i\beta} + \boxed{Z_i\, b_i} + \boxed{\epsilon_i}$$

```
proc mixed data=dataa;
class time_(ref="C") treatment subjid;

model log_data = treatment time_ treatment*time_ log_baseline/ solution;

random int /subject=subjid;

repeated time_ / subject=treatment(subjid) type=AR(1) R Rcorr;
run;
```

Modeling Joint Likelihood

- PROC MCMC assumes that the input observations are independent
  ➔ the joint log-likelihood is the sum of individual log-likelihood functions
- In case of longitudinal data, measures of a patient are not independent.
  ➔ Measures of a patient follows a multivariate distribution and the joint log-likelihood is the sum of the patient log-likelihood.

- Two options
  - Store all observations from one subject in one row to model the within-subject covariance structure.
  - Create arrays and store all relevant variables (response and covariates) in the arrays. Then, construct the joint log-likelihood for the entire data set instead of for each observation.

How to do it in proc MCMC?

1/ Define the model matrix of the fixed effects

$$\boldsymbol{Y_i} = \enclose{circle}{\boldsymbol{X_i \beta}} + \boldsymbol{Z_i\, b_i} + \boldsymbol{\epsilon_i}$$

```
proc transreg
```

2/ Specify the matrix of variance

$$\boldsymbol{V(Y_i)} = \begin{pmatrix} \mathbf{1} \\ \vdots \\ \mathbf{1} \end{pmatrix} \boldsymbol{\sigma_g^2} (\mathbf{1} \quad \dots \quad \mathbf{1}) + \boldsymbol{R_i}$$

```
beginnodata;

endnodata;
```

3/ Specify the log-likelihood by incrementing

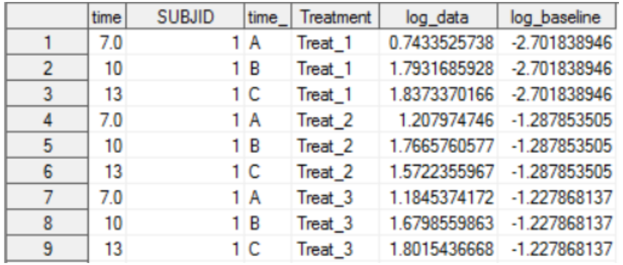
```
model general(ljointpdf);
```

Determine the model matrix X

- As PROC MCMC does not support a CLASS statement, the categorical variables should be changed into dummy variables (0/1).

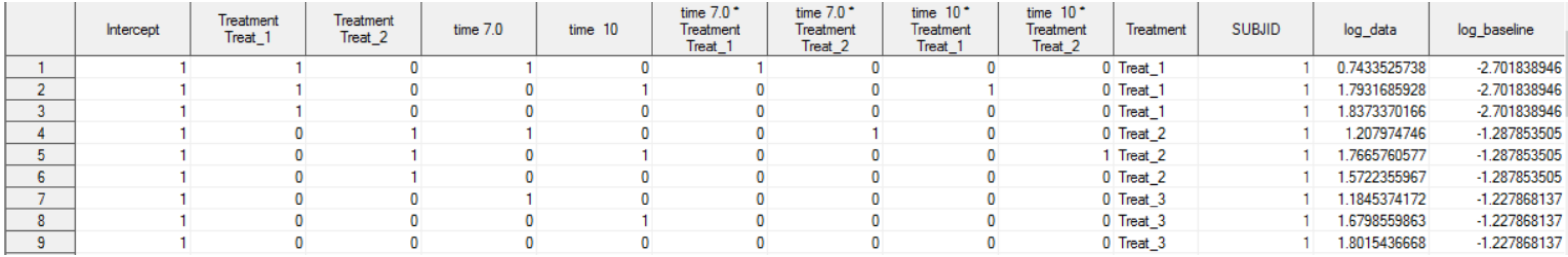
- In order to simplify the writing of the model equation, the model matrix can be directly computed and used in PROC MCMC.

Determine the model matrix X



	time	SUBJID	time_	Treatment	log_data	log_baseline
1	7.0	1	A	Treat_1	0.7433525738	-2.701838946
2	10	1	B	Treat_1	1.7931685928	-2.701838946
3	13	1	C	Treat_1	1.8373370166	-2.701838946
4	7.0	1	A	Treat_2	1.207974746	-1.287853505
5	10	1	B	Treat_2	1.7665760577	-1.287853505
6	13	1	C	Treat_2	1.5722355967	-1.287853505
7	7.0	1	A	Treat_3	1.1845374172	-1.227868137
8	10	1	B	Treat_3	1.6798559863	-1.227868137
9	13	1	C	Treat_3	1.8015436668	-1.227868137

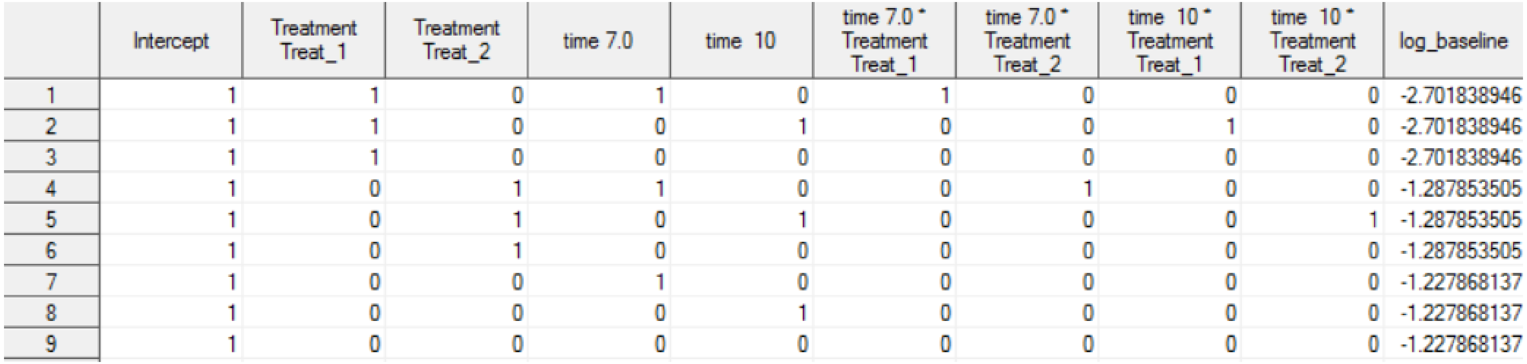
```
proc transreg data=dataa design;
model class(treatment time time*treatment / zero=last);
id subjid log_data log_baseline;
output out=model_matrix_X(drop=time _TYPE_ _NAME_);
run;
```



	Intercept	Treatment Treat_1	Treatment Treat_2	time 7.0	time 10	time 7.0 * Treatment Treat_1	time 7.0 * Treatment Treat_2	time 10 * Treatment Treat_1	time 10 * Treatment Treat_2	Treatment	SUBJID	log_data	log_baseline
1	1	1	0	1	0	1	0	0	0	Treat_1	1	0.7433525738	-2.701838946
2	1	1	0	0	1	0	0	1	0	Treat_1	1	1.7931685928	-2.701838946
3	1	1	0	0	0	0	0	0	0	Treat_1	1	1.8373370166	-2.701838946
4	1	0	1	1	0	0	1	0	0	Treat_2	1	1.207974746	-1.287853505
5	1	0	1	0	1	0	0	0	1	Treat_2	1	1.7665760577	-1.287853505
6	1	0	1	0	0	0	0	0	0	Treat_2	1	1.5722355967	-1.287853505
7	1	0	0	1	0	0	0	0	0	Treat_3	1	1.1845374172	-1.227868137
8	1	0	0	0	1	0	0	0	0	Treat_3	1	1.6798559863	-1.227868137
9	1	0	0	0	0	0	0	0	0	Treat_3	1	1.8015436668	-1.227868137

Determine the model matrix X

- In the following, we can just keep the covariates of the model. We store them in the covariates dataset.



	Intercept	Treatment Treat_1	Treatment Treat_2	time 7.0	time 10	time 7.0 * Treatment Treat_1	time 7.0 * Treatment Treat_2	time 10 * Treatment Treat_1	time 10 * Treatment Treat_2	log_baseline
1	1	1	0	1	0	1	0	0	0	-2.701838946
2	1	1	0	0	1	0	0	1	0	-2.701838946
3	1	1	0	0	0	0	0	0	0	-2.701838946
4	1	0	1	1	0	0	1	0	0	-1.287853505
5	1	0	1	0	1	0	0	0	1	-1.287853505
6	1	0	1	0	0	0	0	0	0	-1.287853505
7	1	0	0	1	0	0	0	0	0	-1.227868137
8	1	0	0	0	1	0	0	0	0	-1.227868137
9	1	0	0	0	0	0	0	0	0	-1.227868137

Structure of PROC MCMC

- **Parameters block:** All the parameters of the model have to be listed with their optional initial values. If a vector of parameters is used, an array has to be defined before with the array statement.

- **Prior block:** Prior distributions have to be defined for each model parameter defined in the parameters block with the prior statement.

- **Programming statements:** Programming statements to define new parameters or for computation of the likelihood.

- **Model statements:** Specify the likelihood.

Structure of PROC MCMC

- **Parameters block:** All the parameters of the model have to be listed with their optional initial values. If a vector of parameters is used, an array has to be defined before with the array statement.

- **Prior block:** Prior distributions have to be defined for each model parameter defined in the parameters block with the prior statement.

- **Programming statements:** Programming statements to define new parameters or for computation of the likelihood

- **Model statements:** Specify the likelihood.

How to write the model in PROC MCMC

```
     beginnodata;
         call Fillmatrix(VCV,sig2g);

         do k= 0 to 2;
                  do i = 1 to 3;
                     do j = 1 to 3;
1                      VCV[i+(k*3),j+(k*3)] = sig2e * rho**abs(j-i) + sig2g ;
                      end;
              end;
         end;

2        call mult(covar,beta,mu);
     endnodata;

         ljointpdf = 0;

         do irec = 1 to &nrec;
          if (first1[irec] = 1) then counter=0;
3         counter = counter + 1;
          ytemp[counter] = data[irec];
          mutemp[counter] = mu[irec];
          if (last1[irec] = 1) then  ljointpdf = ljointpdf + lpdfmvn(ytemp, mutemp, VCV);
         end;

     model general(ljointpdf);
     run;
```

How to write the model in PROC MCMC: Variance-Covariance matrix

- STEP 1: Specify the Variance-Covariance matrix for a subject i.
- Reminder, each subject has 9 measures (3 by treatment)

$$\boldsymbol{V(Y_i)} = \begin{pmatrix} \mathbf{1} \\ \vdots \\ \mathbf{1} \end{pmatrix} \boldsymbol{\sigma_g^2} (\mathbf{1} \quad \dots \quad \mathbf{1}) + \boldsymbol{R_i}$$

$$= \begin{pmatrix} \sigma_g^2 & \sigma_g^2 & \sigma_g^2 \\ \sigma_g^2 & \sigma_g^2 & \sigma_g^2 \\ \sigma_g^2 & \sigma_g^2 & \sigma_g^2 \end{pmatrix} + \begin{pmatrix} \boxed{R_{i1}} & 0 & 0 \\ 0 & R_{i2} & 0 \\ 0 & 0 & R_{i3} \end{pmatrix}$$

$$\text{Variance matrix for the three measures for treatment 1} = \begin{pmatrix} \sigma^2 & \sigma^2\rho & \sigma^2\rho^2 \\ \sigma^2\rho & \sigma^2 & \sigma^2\rho \\ \sigma^2\rho^2 & \sigma^2\rho & \sigma^2 \end{pmatrix}$$

How to write the model in PROC MCMC: Variance-Covariance matrix

- The matrix of Variance-Covariance VCV is filled up with σ_g^2

```
call Fillmatrix(VCV,sig2g);
```

- Each element of the block diagonal is specified in a loop

```
do k= 0 to 2;
    do i = 1 to 3;
        do j= 1 to 3;
                VCV[i+(k*3),j+(k*3)] =
                        sig2e * rho**abs(j-i) + sig2g ;
        end;
    end;
end;
```

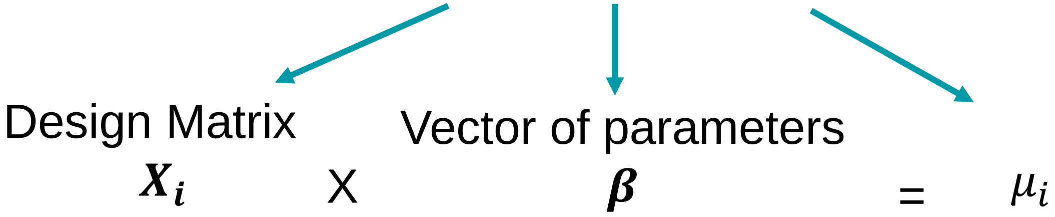
$$\begin{pmatrix} \sigma^2 & \sigma^2\rho & \sigma^2\rho^2 \\ \sigma^2\rho & \sigma^2 & \sigma^2\rho \\ \sigma^2\rho^2 & \sigma^2\rho & \sigma^2 \end{pmatrix}$$

How to write the model in PROC MCMC: Fixed effects

- STEP 2: Compute the fixed part of the model (i.e. the mean profile)

$$\boldsymbol{Y_i} = \boxed{\boldsymbol{X_i\beta}} + \boldsymbol{Z_i}\,\boldsymbol{b_i} + \boldsymbol{\epsilon_i}$$

```
call mult(covar,beta,mu);
```



Design Matrix $\boldsymbol{X_i}$ X Vector of parameters $\boldsymbol{\beta}$ = μ_i

How to write the model in PROC MCMC: Log-Likelihood

- STEP 3: Compute the log-likelihood

```
ljointpdf = 0;

    do irec = 1 to &nrec;
          if (first1[irec] = 1) then counter=0;
          counter = counter + 1;
          ytemp[counter] = data[irec];
          mutemp[counter] = mu[irec];
          if (last1[irec] = 1) then  ljointpdf = ljointpdf + lpdfmvn(ytemp, mutemp, VCV);
    end;
```

$$Y_i \sim mN(\mu_i, VCV_i)$$

To allow the specification of the likelihood, the **jointmodel** option has to be added in the proc mcmc statement

Before fitting the model

- Define arrays and store the data (response variable and covariates)

```
array covar[1] /nosymbols ;
array data[1]/nosymbols;
array first1[1]/nosymbols;
array last1[1]/nosymbols;

begincnst;
     rc = read_array("recodedsplit1",data,"log_data");
     rc = read_array("recodedsplit1",first1,"first");
     rc = read_array("recodedsplit1",last1,"last");
     rc = read_array("covariates",covar);
endcnst;
```

Before fitting the model

- List of the model parameters

```
array beta[&nvar] ;
array VCV[&nrep,&nrep];

parms sig2e 0.1  ; /*residual*/
parms rho   0.1 ;  /*correlation ar1*/
parms sig2g 0.1  ; /*random intercept*/
parms (beta1-beta&nvar) 1;
```

- Prior distributions of the parameters

```
prior beta: ~ normal(0,var=1e6);
prior sig2g ~ general(0,lower=0);
prior sig2e ~ general(0,lower=0);
prior rho   ~ general(0,lower=0,upper=1);
```

Conclusion

- Longitudinal data are widely used in clinical development.
  ➔ The appropriate statistical tools have to be used to ensure an efficient analysis of the data.
- In a frequentist approach, popular procedures as PROC MIXED in SAS/STAT software allow to fit longitudinal model.

- In a Bayesian approach, longitudinal data can be fitted with PROC MCMC. The correlation within patient need to handled.
- The methodology shown today allows to fit longitudinal model in a Bayesian way with the following advantages:
  - Large panel of model
  - Structure for the variance within-subject
  - Ease of model writing (Model Matrix)
  - Ease of adaptation
  - Prior distributions on variance components (not on a variance matrix)

Thank you for your attention !

Do you have questions?

Maud Hennion,
maud.hennion@arlenda.com