Nonlinear Mixed Effects Models, a Tool for Analyzing Repeated Measurements A Brief Tutorial Using SAS® Software

Peter Gaccione, Harvard School of Public Health, Boston, MA
M.S. Blanchard, Frontier Science & Technology Research Foundation, Inc. Chestnut Hill, MA

Abstract

The exponential increase in computing power has led analysts to consider statistical techniques which until recently were not even conceived of. One area of exciting work is the use of nonlinear models to explain biologically derived data, particularly with the inclusion of random, individual effects. Programs for analyzing nonlinear mixed effects models are now becoming more prevalent and available to the general user, and for more general uses. This paper will outline this methodology, and describe an application using SAS Institute's latest contribution to the field, the NLINMIX macro.

Introduction

Repeated measurements are data where individuals have multiple measurements over time or space. Analyzing these data requires recognizing and estimating variability both between and within individuals. Further, it is not uncommon for the relationship between an explanatory variable (e.g., time) and a response variable (e.g., drug concentration or growth) to be nonlinear in the parameters. Nonlinear mixed effects models provide a tool for analyzing repeated measurements data by taking into consideration these two types of variability as well as the nonlinear relationship between the explanatory variable and the response variable.

The Model

Let y_{ij} be the j = 1 to n responses for individual i, and let there be i = 1 to m individuals. The nonlinear function f (x_{ij}, β_i) models the relationship between the response variable y_{ij} and the explanatory variable x_{ij}. The variability between individuals is included in the model by allowing the p x 1 parameter vector β_i to vary between individuals. Thus the mean response for the individual i at time j, given the parameter vector β_i is

$$E(y_{ij}|\beta_i) \;=\; f\,(x_{ij},\beta_i) \qquad (1)$$

As described by Davidian and Giltinan (1993a; 1993b; 1995), the variability in y_{ij} may be a function of f (x_{ij},β_i). For example, the within patient variability may be proportional to the power of the mean and modeled as

$$\mathrm{Var}(y_{ij}|\beta_i) = \sigma^2 f^{\,2\theta}\,(x_{ij},\beta_i), \qquad (2)$$

with scale parameter σ and power parameter θ, or more generally as

$$\mathrm{Var}(y_{ij}|\beta_i) = \sigma^2 g^2\,(f\,(x_{ij},\beta_i),\theta). \qquad (3)$$

Thus, the jth measurement on individual i can be written as

$$y_{ij} \;=\; f\,(x_{ij}, \beta_i) + \sigma g(f\,(x_{ij}, \beta_i),\theta)\epsilon_{ij} \qquad (4)$$

where ϵ_{ij} is a random error term with mean zero and variance one.

The between individual variation is modeled based on the assumption of individual specific sets of regression parameters. This variability in the parameters can be the result of "systematic dependence on individual attributes" (Davidian and Giltinan, 1993a; 1993b), e.g., weight or age. Parameters may also vary due to unexplained random variation, (e.g., individual variation or run to run variation in an assay). To account for this we can write the following model

$$\beta_i = A_i\beta + B_i b_i, \qquad (5)$$

where β is the vector of typical values for the population parameters, A_i is the design matrix corresponding to the systematic portion of the models for β_i, b_i is a random vector with mean 0 and covariance matrix D, and B_i is the design matrix associated with b_i defining the noise portion of the model for β_i (Davidian and Giltinan, 1995; Lindstrom and Bates, 1990). Note the model for β_i does not have to be linear.

Inference

When using nonlinear mixed effects models, two types of inference may be distinguished, population and individual inference. With population inference, the interest focuses on estimation of typical values for the population, β and D. Individual inference focuses on the parameters associated with a particular individual, in which case β_i would be the statistic of interest. The mixed effects models methodology is implemented here to improve the inference by utilizing common information across subjects. For example, when considering clearance as a possible parameter in describing a drug concentration time curve, the population inference would be the estimation of the mean and variance of clearance for the population being studied, while the individual inference would center on predicting the value of clearance for a particular individual.

Methods of Estimation

Two types of estimation methods are quickly noted here: individual estimation methods, and methods based on linearization using a Taylor series expansion. Individual estimation methods depend on the ability to estimate β_i from the data from each subject; the estimated β_i are then the data used to estimate β, D, σ, and θ. Two main linearization methods are popular: they use Taylor series expansions in the random effect b_i. One is a first order method proposed by Beal (1984) and Sheiner, Rosenberg and Melmon (1972), in which a Taylor series is taken about b_i set to 0 (the expected value). The second is a first order conditional method presented by Lindstrom and Bates (1990), in which a Taylor series is taken about b_i set to the conditional estimate of the between individual random effects (Roe, 1997). These methods are appealing in that they are applicable to sparse data problems and the estimation methods used parallel those of linear hierarchical models (Davidian and Giltinan, 1995).

NLINMIX macro

"NLINMIX was originally written to implement the algorithm of Lindstrom and Bates" (Wolfinger, 1997). Their estimation scheme iterates to convergence between a nonlinear least squares step and linear mixed model step, corresponding to steps in PROC NLIN and in PROC MIXED. The NLINMIX macro, however, only uses PROC NLIN to compute starting values, since "solving the mixed models equations corresponds to taking one Gauss-Newton step in a nonlinear least squares optimization" (Wolfinger, 1997). The NLINMIX macro also includes two other methods of estimation, the first order method, and a GEE method (Littell, Milliken, Stroup, and Wolfinger, 1996; Wolfinger, 1993). It is listed and documented in Littell, Milliken, Stroup and Wolfinger, 1996, and is available by anonymous ftp (ftp.sas.com), on the World Wide Web, (http://www.sas.com/) in the SAS Supplemental Library, or from the SAS Technical Support Division.

Application

To date most of the applications of nonlinear mixed effects modeling have been in the area of pharmacokinetics, where the relationship between drug concentration and time depends upon an overall biological mechanism as well as individual subject characteristics. This creates an ideal setting for the consideration of both fixed (population attributable) and random (subject specific) parameters. As mentioned above, software developed to analyze such data have taken several approaches. The application for this paper is an analysis of data collected on AIDS patients receiving triple combination antiviral treatment. Their viral load (V(t)) - is quantified on days 2, 7, 10, 14, 21, 48, 84, 168, and 336 after initiation of treatment.

Model, Data Setup and Implementation

The problem involves modeling viral load over time as a function of four fixed and random effects. In particular, the model is

$$\log_{10} V(t) = \exp(p_1)\exp(-d_1\, t) + \exp(p_2)\exp(-d_2 t) \ .$$

p_1 and p_2 are viral loads in two compartments, d_1 and d_2 are their rates of decline, and t is the time in days after initiation of therapy. The program expects data to be set up as repeated measures. For example, data for the first four patients would look as follows:

patid	lgcopy	day
1	4.36173	2
1	3.53148	7
1	3.07918	8
1	2.97772	16
1	2.57978	22
1	2.64345	29
1	2.11394	57
1	2.00000	91
2	3.38021	2
2	2.00000	7
3	5.20412	2
3	4.34242	7
3	4.14613	14
4	4.11394	7
4	3.84510	9
4	3.23045	14
4	2.90309	21
4	2.77815	28
4	2.43136	56

When the SAS program for the analysis is run the response variable (**lgcopy)** will be referenced in the NLINMIX macro by the statement **response=lgcopy**, and t in the model will be the **day** variable. The model function itself is given to the macro in the **pred=** statement. Note that subject 2 has only two observations. If there are many subjects with few observations estimation and convergence problems could result during an analysis.

The NLINMIX macro implements a two step process for the estimation of the variance components of the model and the estimation of the fixed and random parameters. One call is made to PROC NLIN to perform standard nonlinear least squares estimation, in effect getting default starting values for the regression parameters. (This call is optional and is skipped in this paper's example. The "**parms=**" statement allows the user to supply starting values.) From then on successive runs are made with PROC MIXED on a linearized approximation of the model function until convergence involving both regression and variance-covariance parameters is reached.

There are many options available as macro parameters, and they are outlined at the beginning of the NLINMIX macro and in the hardcopy documentation. Some of those chosen for this paper's analysis are bolded in the program listing below. Since the program is essentially a series of runs of PROC MIXED the most comprehensive documentation is found in the PROC MIXED chapter in the manual SAS/STAT Software: Changes and Enhancements through Release 6.12**.** (1997) **(**This NLINMIX macro is specific to Version 6.12. Other versions of the macro are supported by earlier SAS versions)

Before making a first run many isues about estimation method, variance structures, and linearization approach should probably be considered. The analysis here implements the model stated above with the four parameters p_1, p_2, d_1, and d_2 employed as both fixed and random effects, within patient variance-covariance modeled as unstructured, appropriate starting values supplied for the fixed effects, and maximum likelihood methods of objective function optimization.

A zero order analysis should probably be run first (**expand=zero**). That is, the linearization of the nonlinear function via a Taylor Series expansion about the expected values of the random effects (b_i=0). These analyses usually result in accurate estimates with few convergence problems. (For the problem below a zero order expansion converged in 15 iterations). When the number of observations for many subjects is adequately large, a second analysis using a Taylor Series expansion around the EBLUPS may result in improved estimates. This was the analysis done for this paper.

Reference should be made to the manuals cited earlier for description and action of the parameters of the SAS macro, as well as output shown below. The SAS data set viral contains the variables lgcopy, day, and patid in observations as noted previously. Note how programming statements are allowed within the macro call, and how the connection is made of the random effects (u1,u2,u3,u4) to the fixed effects (p1,d1,p2,d2).

```
**BEGIN PROGRAM STATEMENTS *******;
data viral; set a.moddata;
/*if lgcopy>1.99 then lgcopy=log10(50);*/

%nlinmix(data=viral,
  response=lgcopy,
  subject=patid,
  model=%str(

ep1=exp(p1+u1);
ed1=exp((-1) * (d1+u2) * day);
ep2=exp(p2+u3);
ed2=exp((-1) * (d2+u4) * day);
function=ep1*ed1 + ep2*ed2;
pred=log10(function);
),
parms=p1=12.0   d1=0.4  p2=7  d2=0.035,
derivs=%str(
denom=function*log(10);
d_p1  = ep1*ed1/denom;
d_d1  = -day*d_p1;
d_p2 = ep2*ed2/denom;
d_d2  = -day * ep2*ed2/denom;
d_u1=d_p1;
d_u2=d_d1;
d_u3=d_p2;
d_u4=d_d2;
),
random=u1 u2 u3 u4,
type=un,
expand=eblup,   /*expand=zero*/
procopt=,
modopt=p pm,
method=ml,
maxit=25,
/*note order of covparms:
       p1      d1      p2     d2
  p1    1       2       4      7
  d1            3       5      8
  p2                    6      9
  d2                          10  and
     residual is 11th in line.
So, below, 1.23 is a starting val for the
var of p1, -0.03 is a starting val for
covar of p1 and d1, etc., and 0.04 is a
starting val for est of the residual.*/
/*
covparms=(1.23) (-0.03) (0.02) (1.23) (-
0.003) (2.2) (0.004) (0.001) (0.019)
(0.0004) (0.04)/noiter,
*/
add=%str(
    id patid day lgcopy;
    make 'covparms' out=covparms;
    make 'predmeans' out=predmean noprint;
    make 'predicted' out=predict noprint;
    title "Expansion=zero";
),
options=skipnlin
);

proc print data=solnr;/*print random eff*/
run;
quit;

**END PROGRAM STATEMENTS**;
```

A partial listing of output follows. Note that it is basically the same as that for a run of PROC MIXED. However, the names of all the parameters are in the form of the derivative names. In the sections listing the Parameter Search (COVP1,...,COVP11), **G** Matrix, and Covariate Parameter Estimates - UN(1,1),...,UN(4,4) - the values are the same: COVP1 is the updated estimate used in the last iteration for the variance of the random effect p1, and UN(1,1) matches the element in the first row and column of **G**. This similarity is because of the unstructured G matrix modeling chosen. (For other macro options chosen the values in these sections might differ.) The fixed effects are listed as the names of the derivative values also (i.e., the effect of p_1 is given by D_P1, etc.) as are the random effects, which are printed via a PROC PRINT of the PROC MIXED created data set **solnr** (output not shown).

```
********OUTPUT LISTING*****EXPANSION= EBLUP
Parameter Search
COVP1   COVP2   COVP3   COVP4   COVP5   COVP6
1.3237 -0.0226  0.01271.3650    -0.0264  2.2457
COVP7   COVP8   COVP9  COVP10  COVP11  Variance
0.0088 0.0005  0.0191  0.0003  0.0405   0.0405
G Matrix
Eff  SU Row     COL1        COL2      COL3      COL4
```

```
D_U1  1 1 1.32373701 -0.02257844 1.36496792 0.00875276
D_U2  1 2 -0.0225784  0.01269719 -0.0264311 0.00054472
D_U3  1 3 1.36496792 -0.02643111 2.24568117 0.01911465
D_U4  1 4 0.00875276  0.00054472 0.01911465 0.00031971

Covariance Parameter Estimates (MLE)
Cov Parm   Subject     Estimate
UN(1,1)    _SUBJECT     1.32373701
UN(2,1)    _SUBJECT    -0.02257844
UN(2,2)    _SUBJECT     0.01269719
UN(3,1)    _SUBJECT     1.36496792
UN(3,2)    _SUBJECT    -0.02643111
UN(3,3)    _SUBJECT     2.24568117
UN(4,2)    _SUBJECT     0.00054472
UN(4,3)    _SUBJECT     0.01911465
UN(4,4)    _SUBJECT     0.00031971
Residual                0.04051991

Model Fitting Information for _RESID
 Description                        Value
Observations                     328.0000
Log Likelihood                   -100.845
Akaike's Information Criterion   -111.845
Schwarz's Bayesian Criterion     -132.706
-2 Log Likelihood                201.6897
PARMS Model LRT Chi-Square         0.0000
PARMS Model LRT DF                10.0000
PARMS Model LRT P-Value            1.0000

Solution for Fixed Effects
Effect  Estimate   Std Error  DF    t   Pr > |t|
D_P1  12.40909309 0.19302141 141 64.29   0.0001
D_D1   0.45500091 0.02364094 141 19.25   0.0001
D_P2   7.87568362 0.23245659 141 33.88   0.0001
D_D2   0.03345429 0.00327884 141 10.20   0.0001
```

Discussion -: Nonlinear mixed effects are usually very complicated numerical problems, and as a result convergence issues commonly arise. When using the NLINMIX macro there are two places where this can occur: within PROC MIXED (when the parameters for the linearized model are being estimated), and from the NLINMIX macro's algorithm itself, which decides when all parameters are sufficiently estimated. The first problem requires modifications one would make in fixing regular linear mixed model analysis problems. The second follows from the first; i.e., the parameter estimates at each iteration are not converging. Following are suggestions for addressing the latter problems.

A first fix to try is to increase the number of iterations of the algorithm via the **maxit=** option. For situations where an effect is close to 0 and some negative-definite matrix notes show up on the log a different type structure for estimating **G** such as **fa0(n)**, where n is the number of random effects, is helpful. A third and often critical aid is to change or vary the starting values for the regression estimates in the **parms=** option, possibly even skipping the PROC NLIN run. Yet another technique would be would be to recognize where the program is taking the estimates of the variance-covariance parameters, and to use these as starting values. An example of this is shown above using the **covparms** statement with the **noiter** option.

There are two bolded commented sections in the program listing above. The first is "/* if lgcopy<1.99 then lgcopy=2 */ " and the other begins with "/* note order of covparms*/". These comments were removed, and another EBLUP analysis was run. This resulted in many values in the viral data set being changed, and when these data were fit to the original model without the covparms statement convergence problems resulted. (As it was the original program took over 200 algorithmic iterations to converge.) Some bolded snippets of the log of this new run are listed below, noting what is happening at each iteration.

At the tenth iteration the program is using 12.4 for p1, 0.45 for d_2, etc. for the fixed effects and 1.23, -0.03, etc. for the variance-covariance parameters. Note that convergence is not changing, and certainly not decreasing. At this point the program can be stopped and the program run again using these values (taken from the log, or from the list file via the **printall** option to the **options** parameter of the macro, or from the list file using the **itdetails** option to the **procopt** parameter, or from a data set of the **covparms** using the **make** statement) as starting estimates. The **noiter** option is critical because it keeps the estimates in the range of the desired starting values and prevents PROC MIXED from performing Newton-Raphson iterations which might drift them away. Analysts for this project have utilized this technique to "make the data fit," because of the non-statistical considerations to which the data must conform. Again, reliance on subsequent diagnostics (residual plots, plots of predicted versus observed data, etc.) should always be relied upon to determine if desired and accurate results have been attained. Output from this subsequent run is not reproduced here (convergence occurred after 114 iterations). These are just some of the many methods of dealing with convergence problems which are available. The PROC MIXED and NLINMIX documentation provide many others .

```
**Parts Of Log For Second EBLUP Run*****
iteration=1 convergence criterion=0.232604019
iteration=2 convergence criterion=0.16681315
iteration=9 convergence criterion=0.178689353
iteration=10convergence criterion=0.178837376
P1=12.414653735 D1=0.453718315
P2=7.9570118519 D2=0.0384494646
COVP1=1.238061911 COVP2=-0.038193406
COVP3=0.0201254384 COVP4=1.2318131698
COVP5=0.0038266397 COVP6=2.2131889679
COVP7=0.0049772397 COVP8=0.0018343721
COVP9=0.0191337297 COVP10=0.0004672305
COVP11=0.0405251261
```

Conclusion

Nonlinear mixed effects models provide a tool for analyzing repeated measurements data in which the relationship between the explanatory and response variables can be modeled as a single function, allowing the parameters to differ between individuals. In addition, these techniques recognize that the variability associated with the response variable for a given individual may depend on the response value in a way that is similar for all individuals. This could be due, for example, to properties associated with measurement error. The NLINMIX macro uses a linearization method to allow for estimation of both individual and population parameters.

References

[1] Beal,S.L. (1984). Population pharmacokinetic data and parameter estimation based on their first two statistical moments. *Drug Metabolism Reviews*. *15*(1&2), 173-193.

[2] Davidian, M. & Giltinan, D.M. (1993a). Analysis of repeated measurement data using nonlinear mixed effects models. *Chemometrics and Intelligent Laboratory Systems. 20*, 1-24.

[3] Davidian, M. & Giltinan, D.M. (1995). *Nonlinear Models for Repeated Measurement Data.* Chapman & Hall, New York.

[4] Davidian, M. & Giltinan, D.M. (1993b). Some general estimation methods for nonlinear mixed-effects models. *Journal of Biopharmaceutical Statistics*, *3*(1), 23-55.

[5] Lindstrom, M.J., Bates, D.M. (1990). Nonlinear mixed effects models for repeated measures data. *Biometrics. 46,* 673-687.

[6] Littell, R.C., Milliken, G.A., Stroup, W.W., Wolfinger,R.D. (1996). *SAS® System for Mixed Models.* SAS Institute, Inc, Cary, N.C.

[7] Roe, D.J. (1997). Comparison of population pharmacokinetic modeling methods using simulated data: results form the population modeling workgroup. *Statistics in Medicine. 16,* 1241-1262.

[8] SAS/STAT Software: Changes and Enhancements through Release 6.12 (1997). SAS Institute Inc., Cary, N.C.

[9] Sheiner, L.B., Rosenberg, B., Melmon, K.L. (1972). Modelling of Individual Pharmacokinetics for Computer-Aided Drug Dosage. *Computers and Biomedical Research. 5,* 441-459.

[9] Wolfinger, R. (1993). Laplace's approximation for nonlinear mixed models. *Biometrika. 80,* 4, 791-795.

[10} Wolfinger R. (1997). Comment: experience with the SAS macro NLINMIX. *Statistics in Medicine.* 16, *1258-1259.*