

SAS® Macro for Automated Model Selection Involving PROC GLIMMIX and PROC MIXED

Fan Pan, Jin Liu
University of South Carolina - Columbia

ABSTRACT

Generalized linear mixed model (GLMM) and linear mixed model (MIXED) deal with the situation where the mean conditional on the normal random effects is linearly related to the model effects. Assumptions for the MIXED are similar to GLMM including normally distributed random effects and residuals and independence among the random effects and model errors. GLMMs allow analysis of both normally distributed and certain types of non-normally distributed dependent variables when random effects are present, whereas MIXED is used for the normal distribution. Both PROC GLIMMIX and PROC MIXED can be used for the same dataset, but different results may be received by using different procedures. It will be of interest to compare the final model results between two procedures. This paper details a group of macros performing separate automated model selection using PROC GLIMMIX and PROC MIXED. The macro will use Output Delivery System (ODS) to save the resulting statistics from model fittings. Model selection indices, such as AICC (corrected Akaike Information Criterion) and Chi-square values, will be calculated basing on resulting statistic used in all possible model selection. Final GLMM and MIXED models include data exploration, influential diagnostics, and checking for model violations with the experimental ODS GRAPHICS option. The macros will also graphically compare the summaries of GLIMMIX and MIXED two best model selections, and give suggestion which procedure is better suit for the dataset. Two examples will be provided using real dataset to show the application of the macro. The macro will be significant for researchers who are interested in application of mixed models.

INTRODUCTION AND THEORETICAL FRAMEWORK

SAS 9.2 documentation for MIXED and GLIMMIX states Linear mixed models (MIXED procedure) are a special case in the family of generalized linear mixed models (GLIMMIX procedure), where the conditional distribution is normal and the link function is the identity function. GLIMMIX procedure model two things: mean component and variance component and their interrelations. By default, in case of normal distribution that is unit. Most models, which can be fitted with the MIXED procedure, can also be fitted with the GLIMMIX procedure, when applies to most types of normal theory linear models with random effects or repeated measures. Both PROC GLIMMIX and PROC MIXED can be used for the same dataset, but different results may be received by using different procedures. It will be of interest to compare the final model results between two procedures. However, in practical analysis, many people experience a real difference in the output with either PROC MIXED or GLIMMIX when applied to a dataset with random effects or repeated measures, assuming a normal distribution for the errors, for either a small or large dataset. Based on this similarity of two procedures and difference in real researches, some social workers who are not quite familiar with these two SAS commands may feel confused when they choose the most fitted model for their data.

Lots of suggestions based on how to choose between GLIMMIX and MIXED can be found. MIXED procedure assume dependent variable given random factor is normal but there are no assumptions about random factor, on the other hand GLIMMIX procedure assumes that random factors are normally distributed and all estimations are based on this assumption. This is an important indicator when many people choose between two procedures. Other statistical researchers suggest that if errors are normally distributed and random effects following a normal distribution, MIXED procedure is preferred, where as GLIMMIX will be used for count data or binary dependent variable. Some other requirements, such as marginal distribution if the marginal distribution marginal distribution $ZGZ' + R$ known is normal then use MIXED, if not then GLIMMIX was chosen. However, most of these suggestions are coming from the experience of people when they are doing analysis using two procedures. There is no systemic procedure

or requirement officially posted on the website, showing people how to make precise choices between two models.

Testing the distribution of the dependent variable usually is the basic step for most data analysis, which is also mainly used by some beginning researchers to choose statistical models. The GLIMMIX is more generally used than MIXED, because it analyzes both normally distributed and certain types of non-normally distributed dependent variables, such as the exponential family, whereas MIXED is restricted to the normal distribution. However if the dependent variable is normally distributed, it is hard to choose between two procedures, especially for some researchers who do not quite understand the SAS software. In this way, the comparison between two procedures is needed in order to get the most suitable model. When we analyze for real datasets, it is very hard make sure different situations to choose between was satisfied. Even though different conditions were tested before actually conducting a statistic model, it is still very difficult to confirm the chosen MIXED or GLIMMIX show better results for our data sets. Besides, in real situations, we always confront the analysis when our datasets are very small, which also contains random effects, both procedures will generally have some bias and estimation issues, because GLIMMIX uses penalized quasi-likelihood to fit its models, and that's known to produce biased results when you have small cluster sizes.

Our research goal is to develop a complete macro automated performing model selection using PROC GLIMMIX and PROC MIXED. The macro will use Output Delivery System (ODS) to save the resulting statistics from model fittings. Model selection indices, such as AICC (corrected Akaike Information Criterion) and Chi-square values, will be calculated based on resulting statistics used in all possible model selection. Final GLMMIX and MIXED models include data exploration, influential diagnostics, and checking for model violations with the experimental ODS GRAPHICS option. The macros will also graphically compare the summaries of GLIMMIX and MIXED two best model selections, and give suggestion which procedure is better suited for the dataset. Two examples will be provided using real dataset to show the application of the macro.

As the first step of our research, this paper details two groups of macros %GLMMIX and %MIXED. %GLMMIX is already provided by the SAS user groups, and Macro %MIXED has changed from %GLMMIX according to the difference of GLIMMIX procedure and MIXED procedure on the SAS website.

CODE

```
%macro mixstep(table=,selection=backward,n=,level5=,level4=,level3=,level2=,
effect5=intercept,effect4=intercept,effect3=intercept,effect2=intercept,method=ML, HYPE = ,
ddfm=satterth,varlist=,varcible=,threshold=0.05,type=vc,include=,varclass=);

%let error=0;

%let errormodal=0;

/*some errors messages*/

%if &table = %then %do;

    %put "ERROR: You didn't give the work table : Please enter the name of work table";

%let error=1;

%end;

%if &varlist = %then %do;

    %put "ERROR: You didn't give potentially variables to select : Please enter variables";%let error=1;

%end;

%if &varcible = %then %do;

    %put "ERROR: You didn't give the outcome variable : Please enter binary variables (0 or 1)";%let error=1;

%end;
```

```

%if &n = or &n<2 or &n>5 %then %do;
    %put "ERROR: You didn't give the maximal level of study : Please enter number between 2 and 5";%let
error=1;
%end;
%if &threshold<=0 or &threshold>0.25 %then %do;
    %put "ERROR: You didn't give a correct threshold : Please enter number between 0 and 0.25, preferably
0.05";
    %let error=1;
%end;

/*to test if the outcome variable is binary*/
%put "We test if the outcome variable is binary";
%if &table ne %then %do;
    ods output onewayfreqs=aaa(keep=&varcible);
    title "Test of variable &varcible";
    proc freq data=&table;tables &varcible;run; data aaaa;set aaa;id=1;run;
    data aaaaa;set aaaa;by id;retain errormodal 0;
    if first.id and &varcible ne 0 then errormodal=1;
    if last.id and &varcible ne 1 then errormodal=1;
    call symput ('errormodal',errormodal);
run;
%end;
%if &errormodal = 1 %then %do;
    %put "ERROR: You didn't give a correct outcome variable : Please enter a binary variable";%let error=1;
%end;
%if &error = 1 %then %do;
    %put "PLEASE CORRECT ERROR(S) DETAILED ABOVE";
%end;

/*if there is no error then we start the macro*/
%if &error ne 1 %then %do;
/*we declare some local variables*/
%let varselec=;%let notconv=0;%let stop=0; %let j=0;
/*we precise random effects in function of the number of levels in the model*/
%if &n=5 %then %do;
    %let random=%str(repeated &effect2/ subject=&level2 solution type=&type;
Repeated &effect3/ subject=&level3 solution type=&type;
Repeated &effect4/ subject=&level4 solution type=&type;
Repeated &effect5/ subject=&level5 solution type=&type;);
    %let class=%str(class &level5;class &level4;class &level3;class &level2 &varclass;);

```

```

%end;
%if &n=4 %then %do;
    %let random=%str(Repeated &effect2/ subject=&level2 solution type=&type;
    Repeated &effect3/ subject=&level3 solution type=&type;
    Repeated &effect4/ subject=&level4 solution type=&type;);
    %let class=%str(class &level4;class &level3;class &level2 &varclass;);
%end;
%if &n=3 %then %do;
    %let random=%str(Repeated &effect2/ subject=&level2 solution type=&type;
    Repeated &effect3/ subject=&level3 solution type=&type;);
    %let class=%str(class &level3;class &level2 &varclass;);
%end;
%if &n=2 %then %do;
    %let random=%str(Repeated &effect2/ subject=&level2 solution type=&type;);
    %let class=%str(class &level2 &varclass;); %end;
%put "WE SELECT VARIABLES WITH THE &selection PROCEDURE";
%put "GLIMMIX model with a &n.-level model random effects";
/*FIRST APPROCH FORWARD WE ADD ONE VARIABLE BY STEP*/
%if &selection=forward %then %do;
/*We do a loop corresponding at each step of variables selection*/
    %do %while (&stop=0);
        %let j = %eval(&j+1);
        %put "STEP &j";
data tested_var;format effect $50. probT 4.2;run;
/*we create an empty table*/

        %local i var;
        %let i=0;
        /*we keep one by one variables which remain to test*/
        %do %while(%scan(&varlist,&i+1,%str( )) ne %str( ));
            %let i = %eval(&i+1);
            %let var = %scan(&varlist,&i,%str( ));
            /*we run MIXED procedure on variables selected &varselec : variables
            already selected in the model &var : variables which remain to test */
            ods output
parameterestimates=parameterestimates(keep=effect probT);
            title "Model with the variable &i";
            proc mixed data=&table
                                method=&method CL;
                                ALPHA=. 05 COVTEST;

```

```

&class;
model &varcible (event='1')= &varselec &var /
ddfm=&ddfm;
&random/SOLUTION CL ALPHA=. 05;
ods exclude solutionr; run;
/*we keep values on each new variable to test*/
data parameterestimates;set parameterestimates;nobis=1;run;

data parameterestimates;set
parameterestimates;if last.nobis;by nobis;run;

data tested_var;set tested_var
parameterestimates;run;

/*we add at the table all precedent results*/

%end;

/*we keep the highest p value and then we compare it to the threshold given in the macro
option*/
data gtgt;set tested_var;if probT=. then delete;run;
proc sort data=gtgt;by probT;run;
data apport;set gtgt;by nobis;if first.nobis;run;
data fsdfds;set apport;retain stopper 0;
if probT>&threshold then stopper=1;else stopper=0;
call symput('stop',stopper);
call symput('var',effect);
call symput('thres',probT);
run;
/*if the variable is not significant, we stop the macro*/
%if &stop=1 %then %do;

%put "Algorithm tell us to stop looking for new variables in the model";
%put "Last variable proposed to be add was &var with a p-value of &thres";

%end; %else %do;

%put "Algorithm tell us to continue looking for new variables in the model";

%let newvar=&var;
%put "The new variable is &newvar";
/*we obtain the list of variables kept certain in the model*/
%let varselec=&varselec &newvar;
%put "Variables selected at step &j are &varselec";

/*part to give us the list of variables which remain to test at the next step j+1*/
%let varenplus=;
%let i=0;

```

```

%do %while(%scan(&varlist,&i+1,%str( )) ne %str( ));
%let i = %eval(&i+1);
%let variable= %scan(&varlist,&i,%str( ));
%if &variable ne &newvar %then %do;
    %let varselection=1;
    %end;
    %else %do;
    %let varselection=0;
    %end;
    %if &varselection=1 %then %do;
    %let varenplus= &varenplus &variable ;
    %end;
%end;

/*New list of variables to test at the next step*/
%let varlist=&varenplus;
%put "Variables to test at the next step are &varlist";
%end;
%end;

/* FINAL MODEL */
%if &varselec = %then %do;
    %put "There is no significant covariates at threshold &threshold to explain the variable &varcible";
    %put "Propose another variables in the model or increase the threshold";
    %end;
    %else %do;

title "Final model selected with &selection procedure";
title2 'Informative model : run also with the other type of selection';
title3 'Keep a critical sense about the final model';
ods output solutionR=solutionR
oddsratios=oddsratios (keep=estimate df lower upper)
parameterestimates=parameterestimates(keep=estimate df effect probT);
proc mixed data=&table method=&method CL;
ALPHA = .05 COVTEST;
&class;
model &varcible (event='1')=&include &varselec/
ddfm=&ddfm;
&random / SOLUTION CL ALPHA = .05; run;
%put "Final variables selected at the final step &j with the &selection selection are &varselec";
%put "The final model is available in the Results section";
/*fixed effects*/

```

```

data oddsratios2;set oddsratios;rename estimate=oddsratios;format lower upper 5.2;run;
data parameterestimates2;set parameterestimates;
if effect="Intercept" then delete;oddsratios=round(exp(estimate),.001);run;
/*we merge oddsratio results and parameters estimates*/
data effects;
merge parameterestimates2 oddsratios2;
run;
/*we add the value of Intercept*/
data intercept;set parameterestimates;if effect="Intercept" then output;run;
/*we obtain the final table with all fixed effects*/
data fixedeffects;
set intercept effects;
if oddsratios ne . then do;
    confidence_interval="['!!compress(round(lower,.01))!!' ; '!!compress(round(upper,.01))!!']";
end;
drop lower upper;
run;
title "Summary of fixed effects";
proc print data=fixedeffects;run;
%put "You can see the Results section to consult fixed effects";
%put "Summary of fixed effects are resumed in the table FIXED EFFECTS";
/*random effects*/
data randomeffects;set solutionr;if probT<=&threshold;run;
title "Summary of random effects";
proc print data=randomeffects;run;
%put "You can see the Results section to consult random effects";
%put "Summary of significant random effects are resumed in the table RAN DOMEFFECTS";
%put "All random effects are available in the table SOLUTIONR and in the final model results";
%end;

```

CONCLUSION

As the preliminary step of our whole research, the %MIXED was successfully development based on %GLIMMIX macro. In following study, Output Delivery System (ODS) to save the resulting statistics from model fittings will be added to the whole macro. Model selection indices, such as AICC (corrected Akaike Information Criterion) and Chi-square values will used to show researchers which procedure is better. Final GLMM and MIXED models will also check model violations with the experimental ODS GRAPHICS option.

REFERENCES

Francais Adrien (2007). Automatic selection macro of fixed effects with PROC GLIMMIX

RECOMMENDED READING

- SAS® *For %GLIMMIX*®

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Fan Pan
University of South Carolina - Columbia
Pan6@email.sc.edu

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

