Paper 1558-2014

%COVTEST: A SAS® Macro for Hypothesis Testing in Linear Mixed Effects Models via Parametric Bootstrap

Peter K. Ott. Forest Analysis and Inventory Branch. BC Ministry of Forests, Lands and Natural Resource Operations. Victoria, BC Canada.

ABSTRACT

Inference of variance components in linear mixed effect models (LMEs) is not always straightforward. I introduce and describe a flexible SAS® macro (%COVTEST) that uses the likelihood ratio test (LRT) to test covariance parameters in LMEs by means of the parametric bootstrap. Users must supply the null and alternative models (as macro strings), and a data set name. The macro calculates the observed LRT statistic and then simulates data under the null model to obtain an empirical p-value. The macro also creates graphs of the distribution of the simulated LRT statistics. The program takes advantage of processing accomplished by PROC MIXED and some SAS/IML® functions. I demonstrate the syntax and mechanics of the macro using three examples.

INTRODUCTION

Mixed models are a popular tool among statistical modellers. This is evidenced by the fact that there are five SAS/STAT® procedures available to fit mixed models in SAS® 9.3: PROC GLM (sort of), PROC MIXED, PROC HPMIXED, PROC GLIMMIX and PROC NLMIXED. To keep on track, this presentation will focus entirely on linear mixed models even though the 'problem' described below also affects linear, generalized linear, and nonlinear mixed effects models.

Inference of variance components in linear mixed effect models (LMEs) is generally straightforward for simple designs. When the data are balanced and the expected mean squares can be arranged in such a way as to isolate the effect of interest, the ratio of these mean squares will have an exact F distribution.

However, inference for variance components can be challenging for more complicated cases (e.g. unusual designs, data that involve imbalance, missing cells, incompletely crossed factors, etc.) because an exact F-test cannot be constructed. Instead the Wald, score or likelihood ratio test (LRT) are usually turned to. But when variance components are constrained to be nonnegative and the null hypothesis lies on the boundary of the parameter space (e.g. $H_0\colon \sigma^2 = 0$), the classical null distribution for all of the above tests no longer holds - see for example Stram and Lee (1994, 1995).

Under these constrained conditions, the distribution of the LRT statistic has been given much attention, so I will focus on it in particular. At best the distribution of the LRT statistic is approximated by a mixture of chi-square distributions with different degrees of freedom (Stram and Lee *ibid*, Verbeke and Molenberghs 2000). At worst it cannot be described analytically. What then?

One solution is to approximate the sampling distribution of the LRT statistic using simulation – namely the parametric bootstrap (Efron 1979). The idea is to first calculate the LRT statistic using the observed data. That is, we calculate the value of -2logL_{ML} (or -2logL_{REML}) for the reduced (i.e. null) and full (i.e. alternative) models, and then take the difference. We then simulate many datasets using the estimated parameters from the reduced model, and for each simulated dataset, estimate both models and calculate the LRT statistic. Naturally the distribution of the simulated LRT statistic forms the required reference distribution because the datasets were generated under the null hypothesis. The p-value for the LRT is estimated using the proportion of the simulated LRT statistics that exceed the observed LRT statistic.

Taking advantage of PROC MIXED, ODS and SAS/IML®, I present a macro program to conduct the parametric bootstrap in SAS®. I use three linear model examples to demonstrate the mechanics and capabilities of macro.

THE %COVTEST MACRO

The %covtest macro has three components:

First, the %str() macro function is utilized to define the full model (m1) that represents the model under the alternative hypothesis. Usually this model would have already been fit successfully and the user was content with its fit, model diagnostics, etc. To define m1, the user copies the relevant lines from some successful PROC MIXED code, pastes it between the parentheses of %str(), and removes any unnecessary options. Since the %covtest macro only requires the final value of the -2logL$_{\text{ML}}$ (or -2logL$_{\text{REML}}$), most procedure options are unnecessary and will only slow (or potentially crash) macro processing.

Second, the user must define the reduced model m0 that represents the model with the null hypothesis imposed. It is very important that m0 is a nested version of m1 because (a) it is a requirement for the LRT, and (b) little error checking exists in the macro to verify it. Here nested means that m1 can be transformed into m0 by imposing a set of linear constraints on the parameters of m1. The null model may have one, some or all of the random (and/or repeated) statement terms removed from full model. Fixed terms can be removed too, but only when using maximum likelihood as the estimation method.

Third, the user must call the %covtest macro which includes m1 and m0 as arguments, and it also requires the user to specify a dataset name. I recommend including only those variables in the dataset that are absolutely necessary because any observations with at least one missing value get removed near the start of the macro. Optionally the user may also supply the estimation method (REML or ML), the number of bootstrap simulated samples, a starting seed (so results can be repeated), and a particular graphing option (described below).

Now let’s look at a few examples...

EXAMPLE 1

Consider the simple hierarchical linear model taken from Example 3.4 of Littell et. al. (2006). The “semiconductor data” involve measurements on the thickness of the oxide layer on silicon wafers. The wafers come from 8 different randomly chosen lots, with 3 random wafers selected per lot, and 3 measurements taken at random sites on each wafer.

The model is: $y_{ijk} = \mu + a_i + w_{j(i)} + \varepsilon_{k(ij)}$ where y_{ijk} is the thickness of the oxide layer, and $i = 1,2,\dots,8 \;\; j = 1,2,3 \;\; k = 1,2,3$ index the lots, wafers, and sites on the wafers respectively. Also, $a_i \sim N(0,\sigma_a^2), w_{j(i)} \sim N(0,\sigma_w^2)$, and $\varepsilon_{k(ij)} \sim N(0,\sigma^2)$, with all three of these random variables being independent and identically distributed.

Say we wish to test that the wafer-to-wafer variation is zero: $H_0\colon \sigma_w^2 = 0$ versus $H_1\colon \sigma_w^2 > 0$.

Because the design is simple and balanced, an exact F-test is available to test H_0. So if we fit PROC GLM or PROC MIXED (with method=type1 or type3), we see that $\hat{\sigma}_w^2 = 35.87$, and that F=9.56 and p<0.0001 (i.e. the wafer-to-wafer variation is significant). A comparison with the %covtest macro now follows.

First, somewhere early in the SAS® code we need to point to the location where the macro is stored:

```
%include 'G:\...\covtest macro v2013.1.SAS';
```

Then we need to set-up the full and null models:

```
*full model;
%let m1=%str(
  class lot wafer site;
  Model Thick=;
  random lot wafer(lot);
);

*null model;
%let m0=%str(
  class lot wafer site;
  Model Thick=;
  random lot;
);
```

The dataset name is called “semicon”, so here’s how to invoke the macro:

```
%covtest(&m1, &m0, ds_name=semicon, estm=REML, nsim=1000, seed=123, mixture=T);
```

The output is in four parts. The first part (Figure 1.1) shows the distribution of the LRT statistics (denoted lambda) simulated under the null hypothesis, and it also charts the observed LRT statistic as a pink vertical line. We see from Figure 1.1 that the observed LRT is far out in the right tail of the reference distribution (i.e. an unusually large value).

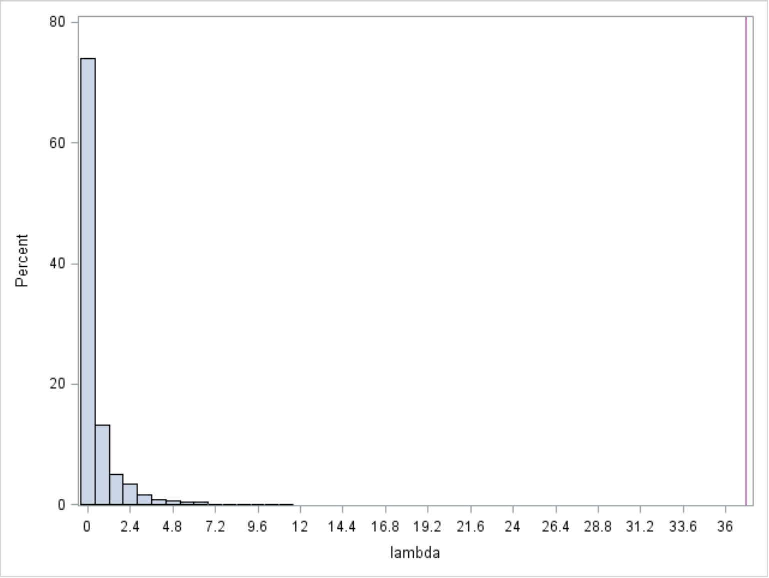

Figure 1.1. Distribution of the Simulated LRT Statistic (lambda), and the Observed LRT Statistic (vertical line) for Example 1

The second part (Figure 1.2) shows the empirical cumulative distribution function (CDF) for the simulated LRT statistics. We see that about 50% of them are zero, which is not unusual for cases where the variance component(s) in the alternative model are constrained to be nonnegative.

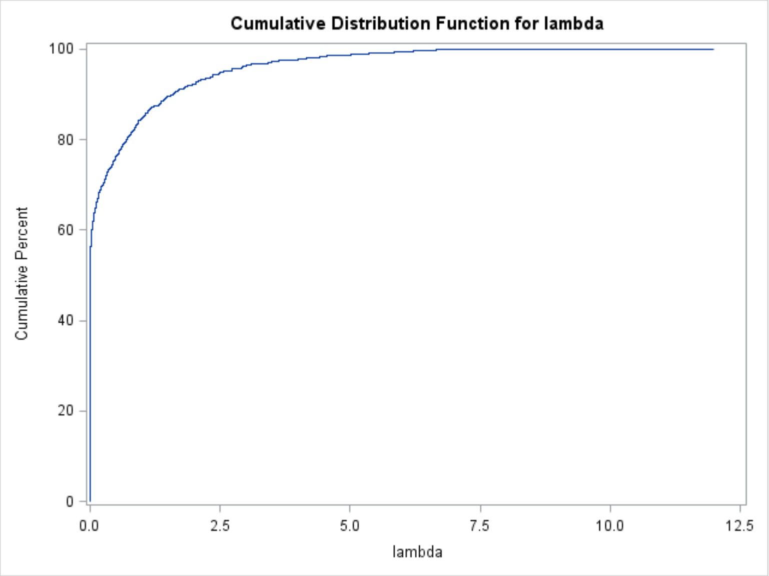

Figure 1.2. Empirical CDF for the Simulated LRT Statistic (lambda) for Example 1

The third part (Figure 1.3) is a probability-probability plot on the p-value scale (Pinheiro and Bates 2000). The nominal p-values for the simulated LRT statistics are plotted against the empirical p-values. The purpose of this plot is to show how well a nominal LRT distribution matches the empirical one. The nominal p-values in the plot can be calculated two different ways, which is what the last argument of the macro controls. With mixture=T (the default), the macro uses a 50:50 mixture of chi-square distributions having q0 and q1 degrees of freedom (where q is the number of covariance parameters in each model). With mixture=F, the macro uses a simple chi-square distribution with k df (where k is the difference in the total number of parameters (fixed + covariance) between the two models). Often neither will be perfect but the latter situation would be more appropriate for cases where the covariance parameter being tested is unconstrained (e.g. a correlation coefficient). For this example using a chi-square mixture of 1 and 2 df for the nominal distribution, we see from Figure 1.3 that the nominal p-values are consistently too large (i.e. conservative).

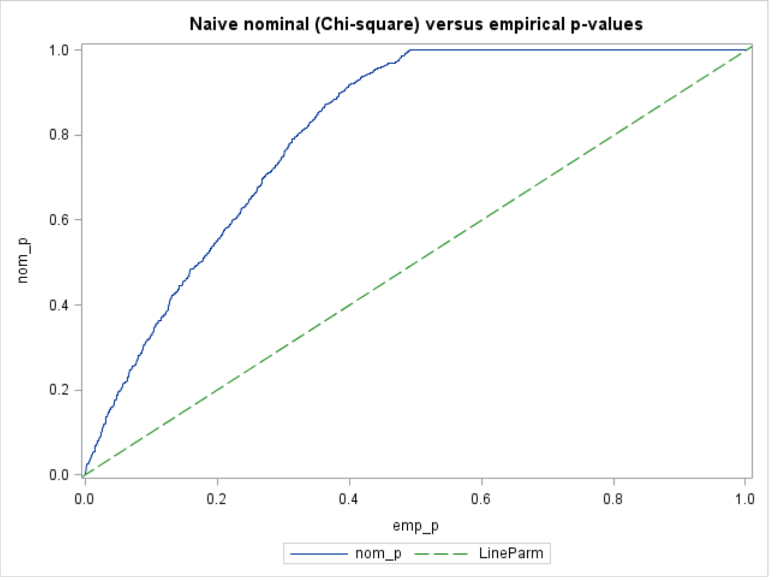

Figure 1.3. Nominal p-values (nom_p) for the Simulated LRT Statistic Plotted Against the Empirical p-values (emp_p) for Example 1

The final part of output shows the observed LRT statistic and the empirical p-value for testing H_0. From Output 1.1 we see that none of the simulated test statistics exceed the observed one, which is consistent with the exact F-test.

```
The observed LRT statistic and associated simulated/empirical p-value via
parametric bootstrap

Observed LRT statistic          Empirical p-value
37.1096                         0
```

Output 1.1. Observed LRT Statistic and Empirical p-value for Example 1

EXAMPLE 2

This next illustration is another type of hierarchical model, but this time taken from example 58.5 of the online help for PROC MIXED.

The observed responses are replicate assay results, expressed in percent of label claim (y), at various shelf ages, expressed in months (x). The desired mixed model involves three batches of product that differ randomly in intercept (initial potency) and slope (degradation rate).

The model is: $y_{ij} = (\alpha + a_i) + (\beta + b_i) \cdot x_{ij} + \varepsilon_{ij}$ where $i = 1,2,3 \quad j = 1,2, \dots, n_i$ index the batches and observations within each batch.

Also $\begin{bmatrix} a_i \\ b_i \end{bmatrix} \sim N\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \mathbf{G}\right)$ where $\mathbf{G} = \begin{bmatrix} \sigma_a^2 & \sigma_{ab} \\ \sigma_{ab} & \sigma_b^2 \end{bmatrix}$ and $\varepsilon_{ij} \sim N(0, \sigma^2)$.

Say we wish to test whether the among-batch variation is negligible: $H_0 : \sigma_b^2 = 0$ versus $H_1 : \sigma_b^2 > 0$. Note also that with $\sigma_b^2 = 0$ there is no need for σ_{ab}. So m1 has three covariance parameters and m0 has only one.

We could fit the above model and test H_0 using PROC GLIMMIX:

```
proc glimmix data=rc;
 class Batch;
 model Y = Month / s;
 random int Month / type=un sub=Batch;
 covtest . 0 0 / df=1,3; *testing both sigma2b and sigmaab zero;
run;
```

Using PROC GLIMMIX, $\hat{\mathbf{G}} = \begin{bmatrix} 0.98 & -0.10 \\ -0.10 & 0.04 \end{bmatrix}$ and the observed LRT statistic is 8.67 with p=0.0186 using a 50:50 mixture of χ_1 and χ_3 as the null reference distribution. Another possibility is to try a 50:50 mixture of χ_1 and χ_2 - it yields p=0.0082.

Let’s compare the PROC GLIMMIX LRT results with the %covtest macro. Setting up m1 and m0, and then invoking the macro is straightforward, but notice how the /s option is removed from both m1 and m0:

```
*full model;
%let m1=%str(
  class Batch;
  model Y = Month;
  random Int Month / type=un sub=Batch;
);

*null model;
%let m0=%str(
  class Batch;
  model Y = Month;
  random Int / type=un sub=Batch;
);

%covtest(&m1, &m0, ds_name=rc, estm=REML, nsim=1000, seed=123, mixture=T);
```

For this example, during implementation the log window indicates a warning that approximately 50 of the model fits to the simulated data failed to converge, which is generally not a problem unless they represent a large fraction of the targeted number of simulations. The output in this case will be based on approximately 950 simulations instead of the requested 1000. Note also that the empirical p-value remains valid at only three significant digits with n ≈ 1000.

The output illustrates that:

(i) the observed LRT statistic falls in the tail of the simulated distribution (Figure 2.1),

(ii) very few (maybe none) of the simulated LRT statistics are zero (Figure 2.2),

(iii) the nominal (1,3) mixture-based p-value is slightly too low for the full range of empirical p-values (Figure 2.3), and

(iv) the p-value is (correctly) higher than the nominal p-value provided by PROC GLIMMIX (Output 2.1).

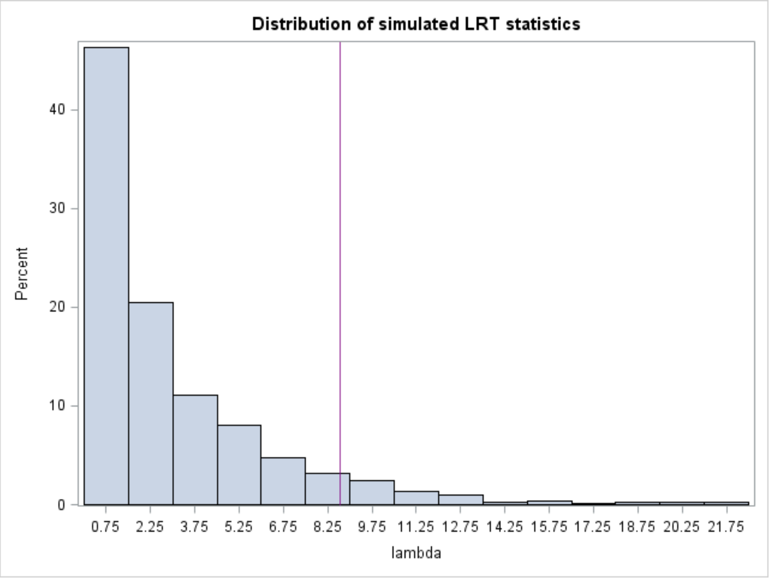

Figure 2.1. Distribution of the Simulated LRT Statistic (lambda), and the Observed LRT Statistic (vertical line) for Example 2

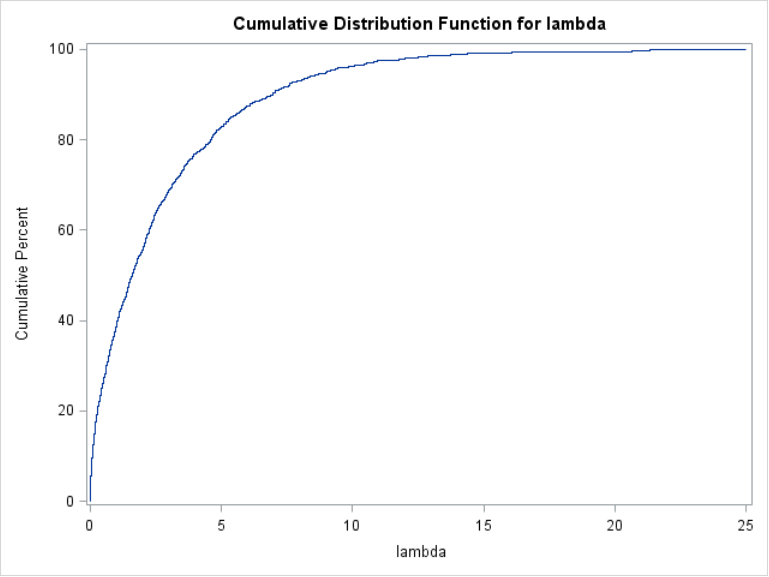

Figure 2.2. Empirical CDF for the Simulated LRT Statistic (lambda) for Example 2

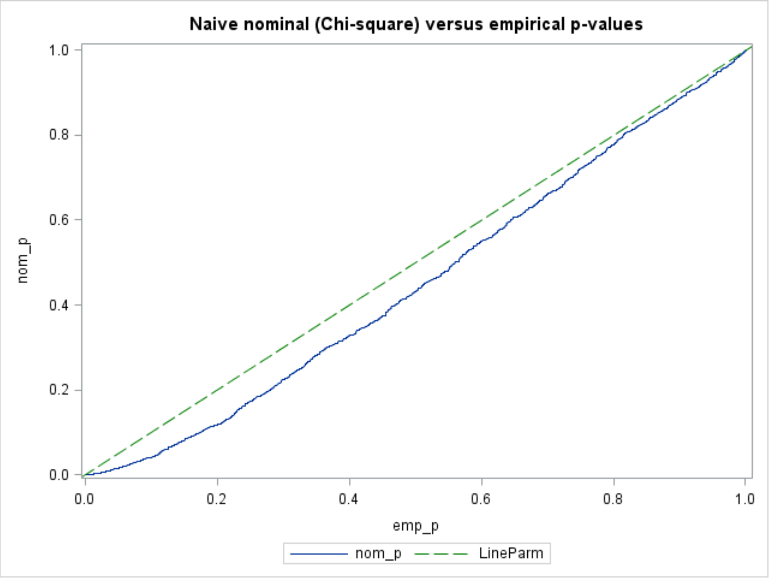

Figure 2.3. Nominal p-values (nom_p) for the Simulated LRT Statistic Plotted Against the Empirical p-values (emp_p) for Example 2

```
The observed LRT statistic and associated simulated/empirical p-value via
parametric bootstrap

Observed LRT statistic          Empirical p-value
8.66894                         0.054679
```

Output 2.1. Observed LRT statistic and empirical p-value for Example 2

Given the choice, I would be more comfortable using the empirical p-value (0.055) rather than one based on a nominal distribution that is less reliable.

EXAMPLE 3

This final example, taken from Stroup (1989), is outlined in example 58.1 from the PROC MIXED online help. The data arise from a balanced split-plot design with the whole plots arranged in a randomized complete block design. The whole-plot factor A has three levels, and the split-plot factor B has two levels. I will also pretend that A and hence A×B are random effects. The model is: $y_{ijk} = \mu + R_i + A_j + (RA)_{ij} + B_k + (AB)_{jk} + \varepsilon_{ijk}$ where $i = 1,2,3,4$ $j = 1,2,3$ $k = 1,2$ index the blocks, and the levels for factors A and B. As usual the random effects are mutually independent and normally distributed: $R_i \sim N(0, \sigma_R^2)$, $A_j \sim N(0, \sigma_A^2)$, $(RA)_{ij} \sim N(0, \sigma_{RA}^2)$, $(AB)_{jk} \sim N(0, \sigma_{AB}^2)$ and $\varepsilon_{ijk} \sim N\left(0, \sigma^2\right)$.

The main objective is to test whether the variance component associated with A is significant: $H_0\colon \sigma_A^2 = 0$ versus $H_1\colon \sigma_A^2 > 0$. Unlike the situation when A is fixed, an exact test no longer exists, so we'll use the %covtest macro. I'll use two random statements, just to show that the macro can handle up to three:

```
*full model;
%let m1=%str(
  class A B block;
  model Y = B;
  random int A / subject=block;
  random int B / subject=A;
);

*null model;
%let m0=%str(
  class A B block;
  model Y = B;
  random int A / subject=block;
  random B / subject=A;
);

%covtest(&m1, &m0, ds_name=sp, estm=REML, nsim=1000, seed=125, mixture=T);
```

The results (graphs not shown) reveal that the variance component σ_A^2 is non-significant (Output 3.1).

```
The observed LRT statistic and associated simulated/empirical p-value via
parametric bootstrap

Observed LRT statistic         Empirical p-value
0.56458                        0.179
```

Output 3.1. Observed LRT Statistic and Empirical p-value for Example 3

The normal care should be taken when, as in this example, testing factors in the presence of an interaction. Note that in the construction of m0 we dropped only factor A from m1 and kept the A×B interaction. What has been achieved? The interaction fills-in for the absence A, and acts as a nested effect. We are not testing the "main effect" A (i.e. all levels of A have the same expected y) because the influence of A continues to operate (albeit via a different mechanism) through the A×B interaction. In fact, because the parameterization of PROC MIXED is not full rank, the total model sums of squares (had we calculated them) is no different from before. What has changed is the covariance structure imposed upon y, and the fitted -2logL$_{ML}$ (or -2logL$_{REML}$). When the true goal is to test for the "main effect" of A for a model involving both A and A×B, an estimable function (i.e. contrast statement) is more appropriate. See McLean et.al. (1991) and Lencina et.al (2005) for further discussion.

I have one last comment that is particularly relevant to experimental designs (i.e. controlled experiments), where "unit structure" such as blocks, plots, etc. are often modelled as random effects. The %covtest macro can be used to test these, but keep in mind that (a) doing so is not the norm, and more importantly (b) these effects should be retained in the model (significant or not) when testing other factors to ensure that the model faithfully portrays the randomization of the experiment.

CONCLUSION

The %covtest macro is an easy-to-use macro for SAS/STAT® users familiar with fitting linear mixed models. Its main benefit is that it avoids reliance on a null reference distribution for the LRT statistic that may not be suitable. This type of situation arises for data that are "messy" or when trying to fit complex models.

Although not shown in the examples above, the macro will work for testing combinations of covariance parameters (i.e. removing several effects from the alternative model), and for simultaneously testing combinations of fixed effects and covariance parameters (just be careful to use method=ML for the latter case).

In addition to normal bug fixing, future work on the macro may include improving the overall coding efficiency and enhancing the handling of missing values. Fitting complex models and/or models to large data sets is the chief challenge because of the iterative nature of the macro. One potential solution involves approximating the LRT and estimating the parameters with quadratic programming as outlined in Shaw and Geyer (1997). Another solution involves allowing the macro to switch to PROC HPMIXED for difficult problems. To accommodate generalized linear mixed models (GLMMs), a slightly different macro would be necessary because of how the %covtest programming simulates data under the null hypothesis, and the oft-inaccessible true log likelihood for GLMMs.

My hope is that in the not-too-distance future, the parametric bootstrap will be a built-in component to an existing SAS/STAT® procedure (PROC GLIMMIX?, PROC PLM?), making the %covtest macro unnecessary. But until then, it may prove helpful the next time you find yourself testing the significance of variance components in a linear mixed effects model.

Those wishing to use the %covtest macro can download it here: http://www.for.gov.bc.ca/hts/strat_analysis.htm

REFERENCES

- Efron, B. 1979. Bootstrap methods: Another look at the jackknife. The Annals of Statistics, 7 (1): 1–26.
- Lencina, V.B., Singer, J.M. and E.J. Stanek. 2005. Much Ado About Nothing: the Mixed Models Controversy Revisited. International Statistical Review, 73 (1): 9–20.
- Littell, R.C., Milliken, G.A., Stroup, W.W., Wolfinger, W.D. and O. Schabenberger. 2006. SAS® for Mixed Models, Second Edition. Cary NC: SAS Institute Inc. 813 pp.
- Pinheiro, J.C. and D.M. Bates. 2000. Mixed-Effects Models in S and S-PLUS. Springer-Verlag, New York, Inc. 528 pp.
- SAS Institute Inc. 2011. SAS/STAT® 9.3 User’s Guide. Cary, NC: SAS Institute Inc.
- Shaw, F.H. and C.J Geyer. 1997. Estimation and testing in constrained covariance component models. Biometrika, 84 (1): 95-102.
- Stram, D.O. and J.W. Lee. 1994. Variance components testing in the longitudinal mixed effects model. Biometrics, 50: 1171-1177.
- Stram ,D.O. and J.W. Lee. 1995. Correction to: Variance components testing in the longitudinal mixed effects model. Biometrics, 51: 1196.
- Stroup, W. W. 1989. Predictable Functions and Prediction Space in the Mixed Model Procedure. In: Applications of Mixed Models in Agriculture and Related Disciplines, Southern Cooperative Series Bulletin No. 343, Louisiana Agricultural Experiment Station, Baton Rouge, 39–48.
- Verbeke, G. and G. Molenberghs. 2000. Linear Mixed Models for Longitudinal Data. Springer-Verlag, New York, Inc. 569 pp.

CONTACT INFORMATION

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

Peter Ott
Forest Analysis and Inventory Branch
BC Ministry of Forests Lands and Natural Resource Operations
PO Box 9512 Stn Prov Govt
Victoria, BC V8W 9C2 Canada
Email: peter.ott@gov.bc.ca
Web: http://www.for.gov.bc.ca/hts/strat_analysis.htm