A Comparison of Denominator Degrees of Freedom Approximation Methods in the Unbalanced Two-Way Factorial Mixed Model

Karl B. Gregory

Chair:	Dr. Michael Speed
Co-Chair:	Dr. Michael Longnecker
Committee Member:	Dr. Bruce Lowe

Abstract

This research assesses four methods of denominator degrees of freedom approximation for F-tests of fixed effects in the small-sample unbalanced two-way factorial mixed model ANOVA. Its purpose is to determine which of four methods, containment, Satterthwaite's, Kenward-Roger, and Kenward-Roger 1st-order, best preserves the type I error rate for tests of fixed effects under varying severities and patterns of unbalancedness. Three functions of the matrix of cell counts for a 3×5 design are suggested as indices of unbalancedness, and it is investigated whether these indices can provide a rule for choosing the best denominator degrees of freedom approximation method given any matrix of replication values. No such rule emerges; the relative performance of the four methods is found to be invariant to the proposed indices of unbalancedness. Moreover, the four methods do not exhibit any significant differences in performance, save for the Kenward Roger method, which is adversely affected when an AR(1) error covariance structure is introduced.

Introduction:

This research evaluates the performance of four denominator degrees of freedom approximation methods for F-tests of fixed effects in an unbalanced 3×5 factorial mixed model ANOVA. If there are a levels of a fixed factor A and b levels of a random factor B and n_{ij} replications at the i^{th} level of A and the j^{th} level of B, then the model can be expressed as

$$y_{ijk} = \mu + \alpha_i + b_j + (\alpha b)_{ij} + e_{ijk} \tag{1}$$

where y_{ijk} is the response value for the k^{th} replicate of the ij^{th} treatment combination. The α_is are the fixed factor effects, the b_js are the random factor effects, and μ is the overall mean. It is assumed that $b_j \sim N(0, \sigma_b^2)$, $(\alpha b)_{ij} \sim N(0, \sigma_{\alpha b}^2)$, and $e_{ijk} \sim N(0, \sigma_e^2)$, for $i = 1, \ldots, a$, $j = 1, \ldots, b$, and $k = 1, \ldots, n_{ij}$.

It is often of interest to test the hypothesis

$$H_0 : \alpha_1 = \alpha_2 = \cdots = \alpha_a \tag{2}$$

for which the usual test statistic is $F = MS_A/MS_{AB}$. When $n_{ij} = n$ for all ij, F statistic is equal to

$$F = \frac{\sum_{i=1}^{a} bn \left(\bar{y}_{i..} - \bar{y}_{...}\right)^2 / (a-1)}{\sum_{i=1}^{a}\sum_{j=1}^{b} n \left(\bar{y}_{ij.} - \bar{y}_{i..} - \bar{y}_{.j.} + \bar{y}_{...}\right)^2 / (a-1)(b-1)} \tag{3}$$

When the random effects and error term are independent and normally distributed, the F statistic follows an F distribution with $a-1$ and $(a-1)(b-1)$ degrees of freedom. However, when the n_{ij}s are not all equal to n, the F statistic becomes more complicated and will only approximately follow an $F_{a-1,\hat{\nu}}$ distribution where $\hat{\nu}$, the denominator degrees of freedom, must be chosen carefully. There are many procedures for choosing the denominator degrees of freedom, four of which are the containment, Satterthwaite, Kenward-Roger, and Kenward-Roger 1st-order procedures; the question arises of which should be used when. The preferred method will be that which best maintains the nominal rejection rate when H_0 is true (that is the nominal type I error rate) and preserves the most power when H_0 is false.

Following are descriptions of the four procedures (which are summarized from SAS documentation) and a discussion of claims as to their relative merits. The procedures will be referred to hereafter as "ddfm options."

Containment: The containment ddfm option chooses the denominator degrees of freedom for the F-test of fixed effects using a syntactical procedure. Let the sources of variation in the experiment be written in the form A, B,

$A \times B$, and $Error(A, B)$, where A represents the fixed factor and B the random factor. The containment method sets the denominator degrees of freedom equal to the smallest of the degrees of freedom values associated with the random effects syntactically containing A. Here, $A \times B$ and $Error(A, B)$ contain A, and $A \times B$ has $(a-1)(b-1)$ degrees of freedom, and $Error(A, B)$ has $(n-1)ab$ degrees of freedom. Thus, in the two-way factorial mixed model, the containment procedure will always choose $(a-1)(b-1)$ for the denominator degrees of freedom.

Satterthwaite: This option invokes a general Satterthwaite approximation of the denominator degrees of freedom which is detailed in the SAS PROC MIXED documentation. When the two-way factorial mixed design is balanced, the Satterthwaite ddfm option will yield the same result as containment, but not in the unbalanced case; in the unbalanced case, the approximation will usually lie between the lowest and highest degrees of freedom values associated with the random effects.

Kenward Roger: The Kenward Roger ddfm option employs the Satterthwaite method after inflating the variance-covariance matrix so as to remove its downward bias. SAS documentation warns that this method may not be suitable for covariance matrices having nonzero second derivatives, such as those with an AR(1) structure.

Kenward Roger 1st Order: This is a modified Kenward Roger method in which only first-order derivatives are used in the calculation of the adjusted variance-covariance matrix. This may be preferred under certain covariance structures such as the AR(1).[1]

A number of simulation studies have investigated the performance of these four methods in different situations.

Spilke et al. studied the performance of the containment, Satterthwaite, Kenward Roger, and residual ddfm options under RCB, split-plot, and strip-plot designs.[2] They generated datasets with small sample sizes, and introduced unbalancedness by eliminating two values at random from each dataset. All random effects were independent and normally distributed. Their conclusions favored the Kenward Roger method, as it best preserved the nominal type I error rate for tests of fixed effects, and when data were simulated under the alternate hypothesis, it maintained relatively high power compared with the other methods.

Waseem Alnosair presented a modification of the Kenward Roger method and compared its performance with the original Kenward Roger, Satterthwaite, and containment methods.[3] Simulating data for a partially balanced incomplete block design, a BIBD, and a complete block design with missing data, he found that both his version of the Kenward Roger method as well as the original outperformed the Satterthwaite and containment methods. He also found that the Satterthwaite method performed more poorly for smaller sample sizes.

Schaalje et al. compared the performance of the Satterthwaite and Kenward Roger methods in split-plot and repeated measures designs under various error covariance structures.[4] They identify three properties of experimental designs that affect the performance of the two methods: The covariance structure, the

sample size, and unbalancedness. They claim that these factors have a greater affect on the Satterthwaite method than on the Kenward Roger method, and further maintain that the latter works as well or better than the Satterthwaite method in all situations. They add the caveat, however, that under small sample sizes and complex covariance structures, the Kenward Roger method produces an inflated type I error rate. Specifically, they mention the ANTE(1) covariance structure, which is listed in the SAS documentation as one potentially affecting the performance of the Kenward Roger method.

Purpose of Current Study:

According to these studies, the relative performance of the ddfm options will depend on the covariance structure and the degree of unbalancedness, and differences in performance will be greater when sample sizes are small. The present study thus considers a 3×5 factorial mixed model with small sample sizes under both the Gauss-Markov and AR(1) error covariance stuctures, and for each covariance structure seeks a rule based on measures of unbalancedness for choosing the best ddfm option. Such a rule is sought under 6 situations. When H_0 as in (2) is true, three situations are simulated: In the first, the errors are independent. In the second, they follow an AR(1) structure, but the AR(1) structure is ignored in the PROC MIXED statement. In the third, the errors follow the same AR(1) structure, but it is specified in the PROC MIXED statement using the REPEATED statement. The same three situations are simulated for H_0 false, for a total of six situations.

This study proceeds under the notion that if the best choice of ddfm option depends on the nature and degree of unbalancedness in the replications, then there must be some index of their unbalancedness that can point to the best choice. In many studies, simulations have been run on a few fixed sets of n_{ij}s, which has not yielded a rule for determining which method is the best given *any* set of n_{ij}s. It is here investigated whether such a rule can be found in computing Simpson's Diversity, Shannon's Entropy, or an index called ImbB from the matrix of n_{ij}s. A rule might be, for example, to use Satterthwaite's when the n_{ij}s have high values for Shannon's index, but to use Kenward-Roger when the n_{ij}s have low values of Shannon's Index. The three indices of imbalance are here described:

Shannon's Entropy: This index is often used as a measure of species diversity in an ecosystem.[5] Let N be the total number of organisms in an ecosystem in which there are s different species. Let n_i be the number of organisms belonging to the i^{th} species, and define $p_i = n_i/N$. Then Shannon's diversity index is

$$H = -\sum_{i=1}^{s} p_i log(p_i)$$

Larger values of H correspond to greater diversity. Placing the ab different n_{ij}s into a vector and indexing them by k, Shannon's entropy can be computed on

the matrix of cell counts as $-\sum_{k=1}^{ab}(n_k/N)log(n_k/N)$, where $N = \sum_{k=1}^{ab} n_k$. As an index of the balancedness of an experimental design, greater diversity corresponds to greater uniformity of cell counts across treatment combinations. Thus larger values of H correspond to greater balance.

Simpson's Diversity: This index is also used to measure species diversity in an ecosystem.[6] Consider choosing 2 elements from a population of size N, in which there are s different types of elements and n_i elements of each type, so that $N = \sum_{i=1}^{s} n_i$. The probability of getting two elements of a different type is

$$D = 1 - \frac{\sum_{i=1}^{s} n_i(n_i - 1)}{N(N-1)}$$

which is the value of Simpson's Diversity index. Note that higher values of D correspond to greater diversity in the population. Again, placing the ab different n_{ij}s into a vector, indexing them by k, and letting $N = \sum_{k=1}^{ab} n_k$, Simpson's diversity can be computed on the matrix of cell counts. Note that larger values of the index correspond to greater "species evenness", or balance in the design.

ImbB: It is conceivable that the type I error rate for tests of fixed effects depends on the orientation of the unbalancedness with respect to the fixed and random factors. For example, the situation in which the same number of replications is observed at each level of the fixed factor, but different numbers are observed across the levels of the random factor, may have an effect on the type I error rate differing from that of the converse situation — when the cell counts are unbalanced across the levels of the fixed factor and balanced across the levels of the random factor. The ImbB index measures the unbalancedness in cell counts across the levels of the random factor by treating the matrix of cell counts as a matrix of responses in a 2-way ANOVA without replication. The value of the ImbB index is the proportion of the total variation in the "responses" occuring across factor B. It is computed thus:

$$\sum_{j=1}^{b} (\bar{n}_{\cdot j} - \bar{n}_{\cdot\cdot})^2 / \sum_{i=1}^{a}\sum_{j=1}^{b} (n_{ij} - \bar{n}_{\cdot\cdot})^2$$

Simulation:

It is of interest to find, for a given degree of unbalancedness, if any of the ddfm options is better than the others. This requires that the performance of each ddfm option be evaluated at various values of an unbalancedness index. A desirable simulation would consist of generating a large number of datasets having a fixed value of the index, and then randomly seperating them into four different groups for testing under the four ddfm options. This would be done for various values of the unbalancedness index. Then, a test for association

between correct decisions and ddfm option would be carried out at each index value. If no association were found, then no ddfm would be better than any other at that value of the index. However, the difficulty arises of constraining the random generation of sets of n_{ij}s such that they will produce a specific value of the unbalancedness index. To obviate this complication, the following simulation is carried out.

For each of the six aforementioned situations, 40,000 datasets are generated where for each dataset a 3×5 matrix of cell counts $\mathbf{N} = \{n_{ij}\}$ is randomly generated, where the n_{ij}s are random realizations of the discrete uniform distribution from 1 to 5. There are thus 1 to 5 replicates at each combination of the two factors. For each dataset the three indices of unbalancedness are computed on the random matrix $\mathbf{N}$ and these values are stored. The 40,000 datasets are generated 10,000 at a time, and each group of 10,000 is assigned to be tested using one of the four ddfm options (This is equivalent to generating 40,000 datsets and then randomly assigning 10,000 to each ddfm option). Then, for each of the three indices in turn, the 40,000 datasets are separated according to their index values into bins, where the bin walls occur at deciles of the index values for the 40,000 datasets. This results in ten bins, each containing approximately 4,000 datasets, of which there are approximately 1,000 tested under each of the four ddfm options.

Each dataset thus generated can be viewed as an experimental unit with three qualitative properties: i) The bin of the unbalancedness index into which it falls, ii) the ddfm option used for testing its fixed effects, and iii) whether the result of the test was a correct or an incorrect decision. The output will allow tests of association between ddfm option and correctness of decision within fixed bins of the unbalancedness index.

To describe the simulation in greater detail, it will be convenient to express the model in matrix form. Let

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}_1\mathbf{u_1} + \mathbf{Z_2}\mathbf{u_2} + \mathbf{e}$$

where $\mathbf{y}$ is an $N \times 1$ random vector of response values, $\mathbf{X}$ is an $N \times (a+1)$ design matrix for fixed effects $\boldsymbol{\beta}$, which is $(a+1) \times 1$. $\mathbf{Z_1}$ is an $N \times b$ design matrix for random main effects $\mathbf{u_1}$, which is $b \times 1$ with distribution $N_b(\mathbf{0}, \sigma_b^\mathbf{2}\mathbf{I}_b)$. $\mathbf{Z_2}$ is an $N \times ab$ design matrix for random interaction effects $\mathbf{u_2}$, which is $ab \times 1$ with distribution $N_{ab}(\mathbf{0}, \sigma_{\alpha b}^\mathbf{2}\mathbf{I}_{ab})$. Lastly, $\mathbf{e}$ is $N \times 1$ with distribution $N_N(\mathbf{0}, \sigma_\mathbf{e}^\mathbf{2}\mathbf{R})$. For each dataset, once the matrix $\mathbf{N}$ of discrete uniform n_{ij}s is generated, the design matrices $\mathbf{X}$, $\mathbf{Z}_1$, and $\mathbf{Z}_2$ are constructed from it. New realizations of $\mathbf{u_1}$, $\mathbf{u_2}$, and $\mathbf{e}$ are then generated, and all of these components are assembled to create the response vector $\mathbf{y}$. The SAS code is provided in the appendix.

On all simulated datasets, the parameter values were the following: There were $a = 3$ levels of the fixed factor and $b = 5$ levels of the random factor. The common mean μ was set to 10, and for the H_0-true data $\alpha_1 = \alpha_2 = \alpha_3 = 0$. The variances of the random effects and the error term were, respectively, $\sigma_b^2 = 9$,

$\sigma^2_{\alpha b} = 4$ and $\sigma^2_e = 1$. For the Gauss-Markov models the error covariance was $cov(\mathbf{e}) = \sigma^2_e \mathbf{I}_N$, and for the AR(1) data, $cov(\mathbf{e})$ was block diagonal with ab blocks, where each block was equal to $\sigma^2_e \left\{\rho^{|k-k'|}\right\}_{k,k'}$ with $k, k' = 0, \ldots, n_{ij}$. Simulations were done at $\rho = .5$. For the H_1 data, the fixed effects were set to $\alpha_1 = -2.5$, $\alpha_2 = 0$, and $\alpha_3 = 2.5$, and the variances of the random effects were unchanged.

Results:

First consider the cases in which H_0 is true. Table 1 displays partial results for the H_0-true simulation under the Gauss-Markov model. It is of interest to determine, within a given bin of an unbalancedness index, if the four ddfm options produce different type I error rates. This amounts to testing for an association between the decision outcome and the ddfm option of the datasets within each bin of the index. Note that the outcome takes on the values FP, for false positive in the case of a type I error and TN, for true negative in the case of a correct failure to reject H_0. If there is no significant association between outcome and ddfm option, then no ddfm option is better than the others. To carry out the test for association between ddfm option and outcome while conditioning on the index bins, the Cochran-Mantel-Haenzel (CMH) test for general assocation is used.

Figure 1 visually represents the results given above for the Shannon Index bins. The bins lie on the horizontal axis, and the four lines trace the type I error rates produced by the four ddfm options across the ten bins. Each point on the graph thus represents around 1,000 simulated datasets. Recall that higher bins of the index correspond to greater balance in the design.

Shannon Index	Bin	1		2		...	10	
	Outcome	FP	TN	FP	TN	...	FP	TN
	Con	49	896	65	879	...	51	958
	Satterth	57	988	49	959	...	54	952
	KR	49	950	60	947	...	54	909
	KR 1st	39	933	44	935	...	47	934
Simpson Index	Bin	1		2		...	10	
	Outcome	FP	TN	FP	TN	...	FP	TN
	Con	51	881	59	870	...	49	973
	Satterth	58	988	49	982	...	53	969
	KR	50	979	69	947	...	52	997
	KR 1st	43	938	38	957	...	47	930
ImbB Index	Bin	1		2		...	10	
	Outcome	FP	TN	FP	TN	...	FP	TN
	Con	46	914	45	921	...	61	940
	Satterth	52	949	52	963	...	48	901
	KR	53	973	46	946	...	51	970
	KR 1st	62	940	39	990	...	52	983

Table 1: Simulation Results for Gauss-Markov model with H_0 true.

Situation	Index	CMH p-value
Gauss-Markov model with H_0 true	Shannon	.2736
	Simpson	.2831
	ImbB	.2730
AR(1) model with H_0 true	Shannon	.8661
(AR(1) ignored in PROC MIXED)	Simpson	.8652
	ImbB	.8713
AR(1) model with H_0 true	Shannon	$< .0001$
(AR(1) specified in PROC MIXED)	Simpson	$< .0001$
	ImbB	$< .0001$

Table 2: Results of CMH tests for general association between ddfm and outcome controlling for bin (H_0 true).

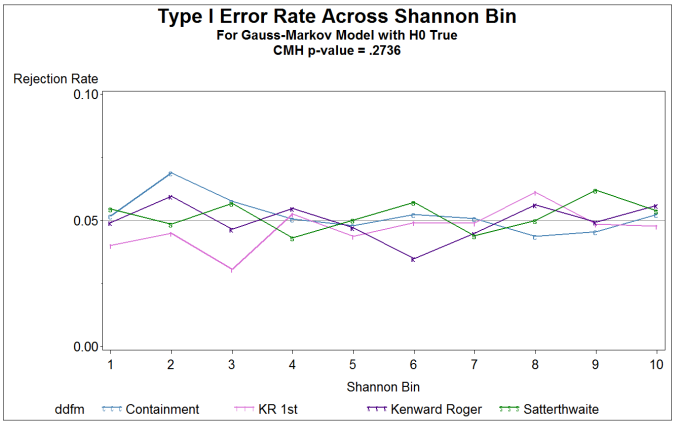


Figure 1: Gauss-Markov model with H_0 true.

Table 2 shows all the H_0-true results of the CMH test for general association between ddfm option and outcome controlling for bin. Recall that for each situation, 40,000 datasets were generated; the three different p-values given are those that result when the 40,000 datasets are binned in three different ways—according to the three different unbalancedness indices. Note that for the first two situations no significant association emerges. For the third, however, when the AR(1) covariance structure is specified using the REPEATED statement in PROC MIXED, ddfm option and outcome do exhibit an association. Figure 2 plots the type I error rates produced by each of the four ddfm options against the bins of the Simpson index, and shows the type I error rates produced by the Kenward Roger method to be higher than the others across all the bins.

The insight herein gained is twofold: That the Kenward Roger type I error rate was higher than the others validates the warning in the SAS documentation that that method may perform poorly with covariance structures having nonzero second derivatives, such as the AR(1) structure. One of the other three should be used in this situation. Secondly, from the plot it is observed that the Simpson bin is of no help in choosing the best ddfm option. The Kenward Roger type I

error rates are above the others across all the bins, and the lines corresponding to the other ddfm options stay very close together. Indeed, when the Kenward-Roger datasets are removed, and the CMH test is performed for the remaining three ddfm options, the p-value becomes .3210.

The type I error rate plots across the bins of the other two indices, Shannon's and ImbB, which are provided in the appendix, exhibit similar patterns. Thus it seems that if there exists a rule for choosing the best ddfm option based on the degree of unbalancedness in the design, it is not found in any of these three indices.

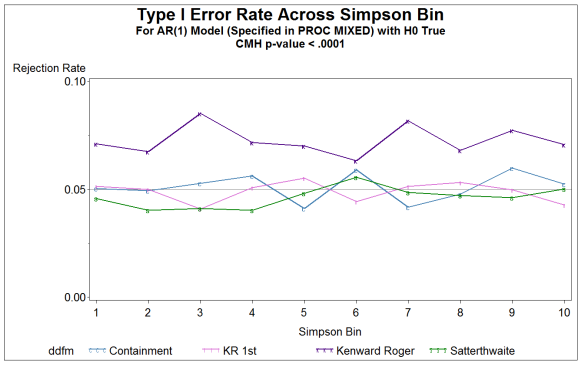


Figure 2: AR(1) model with H_0 true (AR(1) specified in PROC MIXED).

Consider now the cases in which H_0 is false. Here, $\alpha_1 = -2.5, \alpha_2 = 0, \alpha_3 = 2.5$. Now that H_0 is false, The ddfm options are compared in terms of the power they acheive rather than their type I error rates, but the analysis only differs in the decision outcomes for each dataset; there are now either correct rejections, denoted by TP for true positive, or incorrect failures to reject, denoted by FN for false negative.

Table 3 displays partial results for the H_1 simulation under the Gauss-Markov model, and the power acheived by the four ddfm options in each of the ten Simpson bins is plotted in Figure 3.

Table 4 displays the CMH tests for general association between ddfm option and outcome controlling for bin in the three different situations simulated under a false null hypothesis. The results are similar to those of the H_0-true simulations; a significant association between ddfm option and outcome only emerges when the AR(1) covariance structure is specified with a REPEATED statement in PROC MIXED. Figure 4 plots the power of the four ddfm options against the bins of the Shannon index for the AR(1)-specified simulation.

Again, the line for the Kenward-Roger method lies above the others. It seems that the AR(1) covariance structure renders the Kenward-Roger method more likely to reject the null hypothesis than the other methods. The other three methods seem evenly matched in terms of power; there is no discernable pattern in them across the bins, and none consistently exceeds the others. Indeed, the

Shannon Index	Bin	1		2		...	10	
	Outcome	FN	TP	FN	TP	...	FN	TP
	Con	227	762	210	799	...	201	777
	Satterth	205	779	204	805	...	198	810
	KR	228	755	221	788	...	223	828
	KR 1st	232	779	203	796	...	195	774

Simpson Index	Bin	1		2		...	10	
	Outcome	FN	TP	FN	TP	...	FN	TP
	Con	229	761	212	800	...	199	817
	Satterth	192	780	212	780	...	197	774
	KR	217	736	222	832	...	235	818
	KR 1st	231	763	201	804	...	192	802

ImbB Index	Bin	1		2		...	10	
	Outcome	FN	TP	FN	TP	...	FN	TP
	Con	223	786	185	814	...	204	800
	Satterth	180	798	196	797	...	187	834
	KR	205	798	217	803	...	194	792
	KR 1st	196	813	228	757	...	213	776

Table 3: Simulation Results for Gauss-Markov model with H_0 false.

Situation	Index	CMH p-value
Gauss-Markov model with H_0 false	Shannon	.2892
	Simpson	.2946
	ImbB	.3055
AR(1) model with H_0 false	Shannon	.1772
(AR(1) ignored in PROC MIXED)	Simpson	.1762
	ImbB	.1732
AR(1) model with H_0 false	Shannon	$< .0001$
(AR(1) specified in PROC MIXED)	Simpson	$< .0001$
	ImbB	$< .0001$

Table 4: Results of CMH tests for general association between ddfm and outcome controlling for bin (H_0 false).

CMH test performed on just the containment, Satterthwaite, and Kenward-Roger 1st-order methods binned according to the Shannon index results in a p-value of .5613 for a general association between ddfm option and outcome. Producing these plots with respect to the Simpson and ImbB bins reveals similar patterns. This further confirms that the Kenward-Roger method is affected by covariance structures having nonzero second derivatives, and again casts doubt on whether a rule for determining the best ddfm option can be based on an index of unbalancedness.

Conclusions:

It has been investigated whether a rule for determining which of four ddfm options in PROC MIXED will be best can be based on three indices of unbalancedness in the design. No such rule emerged; the values of the three indices of unbalancedess proved uninformative as to the best ddfm option. In

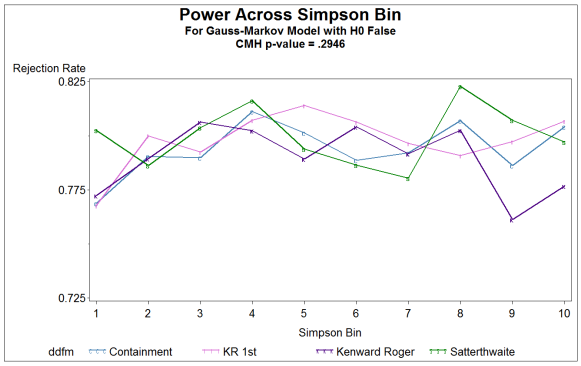


Figure 3: Gauss-Markov model with H_0 false.

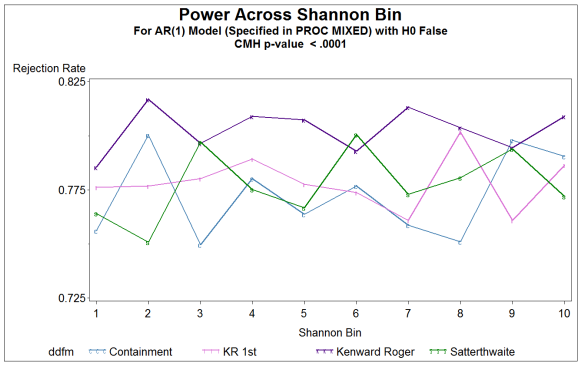


Figure 4: AR(1) model with H_0 false (AR(1) specified in PROC MIXED).

the case of a specified AR(1) error covariance structure, the Kenward-Roger method produced inflated type I error rates and acheived greater power in the H_1 simulations than the other three methods. This coincides with the warning given in the PROC MIXED documentation that the Kenward-Roger method may be affected by covariance structures with nonzero second derivatives. The Kenward-Roger 1st-Order method, however, maintained a close-to-nominal type I error rate and a power closer to that of the containment and Satterthwaite methods in the AR(1) simulations.

In both the Gauss-Markov and AR(1)-unspecified models, no difference in the performance of the four methods was found. The CMH tests for association between ddfm option and outcome conditional on index bin failed to reject the null hypothesis of non-assocation. Thus in no bin was one method conclusively superior or inferior to the others. This casts doubt on whether there be any differences in the performance of the ddfm options traceable to the degree or nature of imbalance in the design. Unless in the case of a covariance structure with nonzero second derivatives, it appears not to matter which method is

chosen. In sum, the present study has yielded no reason to use any other method than the default, which is containment, in all cases; of the alleged superiority of the Kenward-Roger and Satterthwaite methods to containment, this research finds no support.

Admittedly, large-scale simulation on fixed sets of n_{ij}s has, as in the studies earlier cited, has revealed some differences in the four methods; they do exist, however slight. Perhaps the best way to determine which is the best method, once an experiment has been run and its set of n_{ij}s established, is to simulate the performance of all competing ddfm options on exactly that set of n_{ij}s, setting the parameter values for the simulation equal to their observed values. Tailoring the simulation to the case at hand in this way would seem the most reliable means of determining the best ddfm option. Such case-by-case simulation, however, would be burdensome and impractical for lay users of PROC MIXED. The final advice is thus to use the default setting of containment unless there be some demonstrable reason for doing otherwise.

References

[1] Jill Tao and Catherine Truxillo. *Mixed Models Analyses Using SAS*. Cary, NC, 2009.

[2] Joachim Spilke, Hans-Peter Piepho, and Xiyuan Hu. A simulation study on tests of hypotheses and confidence intervals for fixed effects in mixed models for blocked experimentswith missing data. *Journal of Agricultural, Biological and Environmental Statistics*, 10(3):374–389, Sep 2005.

[3] Waseem S. Alnosaier. *Kenward-Roger Approximate F Test for Fixed Effects in Mixed Linear Models*. PhD thesis, Oregon State University, Corvallis, Oregon, 2007.

[4] G. Bruce Schaalje, Justin B. McBride, and Gilbert W. Fellingham. Adequacy of approximations to distributions of test statistics in complex mixed linear models. *Journal of Agricultural, Biological, and Environmental Statistics*, 7(4):512–524, Dec 2002.

[5] Robert K. Peet. Relative diversity indices. *Ecology*, 56(2):496–498, Early Spring 1975.

[6] E. H. Simpson. Measurement of diversity. *Nature*, 163:688–688, April 1949.

Appendix:

I.

The SAS code for simulating the the Gauss-Markov data under H_0 tested under the containment method is shown here. The code is modified slightly to simulate under the other situations.

```
ods select none;
options nomprint;

libname output "\\tsclient\C\Users\kbgregory\Documents\
Masters Research\Simulation Output\H_0 cov 0 data";
run;

%MACRO builddata(a,b,maxn,mu,Bvar,ABvar,Errorvar,rhoAB,rhoB,rhoError);

proc iml;

/* Simulation parameters are specified here. */

a=&a;
b=&b;
maxn=&maxn;

mu=&mu;
rhoAB=&rhoAB;
rhoB=&rhoB;
rhoError=&rhoError;

Bvar=&Bvar;
ABvar=&ABvar;
Errorvar=&Errorvar;

/* Here the numbers of replications per cell are randomly generated.  */

Nij=J(a,b,0);
call randgen(Nij,'uni');
Nij=int((maxn)*(Nij))+1;
Nijrow=shape(Nij,1,a*b);
totN=sum(Nij);

/* The following creates a table of indices for the factor levels that
   match the randomly generated sample sizes. */

F1base=J(a,1,1);
F1=J(totN,1,0);
pF1=1||J(1,a,0);

do i = 1 to a;
pF1[i+1]=sum(Nij[1:i,])+1;
F1[pF1[i]:pF1[i+1]-1,1]=F1base[i,1]@J(pF1[i+1]-pF1[i],1,i);
end;

F2base=J(a*b,1,1);
F2=J(totN,1,0);
pF2=1||J(1,a*b,0);
```

```
do i = 1 to a;
do j = 1 to b;
l = (i-1)*b+j;
pF2[l+1]=sum(Nijrow[1:l])+1;
F2[pF2[l]:pF2[l+1]-1,1]=F2base[l,]@J(pF2[l+1]-pF2[l],1,j);
end;
end;

Rep=J(totN,1,0);

do i = 1 to a;
do j = 1 to b;
l=(i-1)*b+j;
pF2[l+1]=sum(Nijrow[1:l])+1;
do k = 1 to Nijrow[l];
Rep[pF2[l]-1+k]=k;
end;
end;
end;

Indices=F1||F2||Rep;

/*  The following constructs appropriate design matrices given
the matrix of replication values generated above.  */

Xbase=J(a,1,1)||I(a);
X=J(totN,a+1,0);

n=J(1,a,0);
px=1||J(1,a,0);

do i=1 to a;
n[i]=sum(Nij[i,]);
px[i+1]=1+sum(n[1:i]);
X[px[i]:px[i+1]-1,1:a+1]=Xbase[i,]@J(n[i],1,1);
end;

pz1=1||J(1,a*b,0);
Z1base=J(a,1,1)@I(b);
Z1=J(totN,b,0);

do i = 1 to a;
  do j = 1 to b;
l=(i-1)*b+j;
pz1[l+1]=sum(Nijrow[1:l])+1;
Z1[pz1[l]:pz1[l+1]-1,1:b]=Z1base[j,]@J(pz1[l+1]-pz1[l],1,1);
end;
end;
```

```
Z2base=I(a*b);
Z2=J(totN,a*b,0);
pz2=1||J(1,a*b,0);

do l = 1 to a*b;
pz2[l+1]=sum(Nijrow[1:l])+1;
Z2[pz2[l]:pz2[l+1]-1,1:a*b]=Z2base[l,]@J(pz2[l+1]-pz2[l],1,1);
end;

Z=Z1||Z2;

/*  The following produces the variance-covariance matrices
G1 and G2 for the random effects and R for the error term
and uses them to generate values for the random effects and
the errors. */

Bcov=J(b,b,0);

if rhoB =0 then Bcov=Bvar#I(b);
if rhoB ^= 0 then ;
do i=1 to b;
do j=1 to b;
Bcov[i,j]=Bvar#rhoB##abs(i-j);
end;
end;

print Bcov;

ABcov=J(a*b,a*b,0);

if rhoAB =0 then ABcov=ABvar#I(a*b);
if rhoAB ^=0 then;
do i=1 to a*b;
do j=1 to a*b;
ABcov[i,j]=ABvar#rhoAB##abs(i-j);
end;
end;

R=J(totN,totN,0);

if rhoError = 0 then R=Errorvar#I(totN);
if rhoError ^= 0 then;
do i = 1 to a;
do j = 1 to b;
l=(i-1)*b+j;
pF2[l+1]=sum(Nijrow[1:l])+1;
Errorcovij=J(Nijrow[l],Nijrow[l],0);

do h = 1 to Nijrow[l];
do k = 1 to Nijrow[l];
```

```
Errorcovij[h,k]=Errorvar#rhoError##abs(h-k);
end;end;
R[pF2[l]:pF2[l+1]-1,pF2[l]:pF2[l+1]-1]=Errorcovij;
end;
end;

G = block(Bcov,ABcov);
V = Z*G*Z‘ + R;

/* Below the response values are generated and the
 final dataset exported.*/

Beta=(mu||J(1,a,0))‘;
Gamma1=randnormal(1,J(b,1,0),Bcov)‘;
Gamma2=randnormal(1,J(a*b,1,0),ABcov)‘;
Error=randnormal(1,J(totN,1,0),R)‘;

Y = X*beta + Z1*Gamma1 + Z2*Gamma2 + Error;

dataset = Indices||X||Z1||Z2||Y;

create factorial var {F1 F2 Rep Y };
append var {F1 F2 Rep Y };

/* The following code produces the three indices of unbalancedness.  */

SSB=a*(Nij[:,]-Nijrow[:])*(Nij[:,]-Nijrow[:])‘;
SSA=b*(Nij[,:]‘-Nijrow[:])*(Nij[,:]‘-Nijrow[:])‘;

AB=J(1,a*b,0);
do i = 1 to a;
do j = 1 to b;
l=(i-1)*b+j;
AB[l]=Nij[i,j]-Nij[i,:]-Nij[:,j]+Nij[:,:];
end;
end;

SSAB=AB*AB‘;

SST=(Nijrow-Nijrow[:])*(Nijrow-Nijrow[:])‘;

imbB=SSB/SST;
imbA=SSA/SST;
imbAB=SSAB/SST;

Simpson=1-sum(Nijrow#(Nijrow-1))/((totN*(totN-1)));
Shannon=-sum((Nijrow/totN)#log(Nijrow/totN));

create replications var {Nijrow imbB imbA imbAB Simpson Shannon};
append var {Nijrow imbB imbA imbAB Simpson Shannon};
```

```
quit;

proc means data=replications median range var mean;
ods output means.summary=repstats;
run;

%MEND builddata;

/*  The following MACRO runs the builddata MACRO and performs the
mixed model ANOVA on the dataset.  It then records whether a
correct or incorrect decision was made. It then appends each
new iteration of the MACRO onto the dataset pvalues_con_01. */

%MACRO iterate(iterations,a,b,maxn,mu,Bvar,ABvar,
Errorvar,rhoAB,rhoB,rhoError);

%do i=1 %to &iterations;
%builddata(&a,&b,&maxn,&mu,&Bvar,&ABvar,&Errorvar,&rhoAB,&rhoB,&rhoError);

proc mixed data=factorial;
class F1 F2;
model y = F1 /ddfm=contain;
random F2 F1*F2;
ods output  Mixed.Tests3=con_results
Mixed.CovParms=covparms
Mixed.ConvergenceStatus=convstat;

proc transpose data=covparms out=cov
(rename=(Col1=B Col2=AB Col3=Error) drop=_name_);
run;

data iteration_con_01;
merge  con_results (rename=(ProbF=pvalue))
cov
convstat
repstats (rename=( NIJROW_Median=Median_nij
   NIJROW_Range=Range_nij
NIJROW_Var=Var_nij
NIJROW_Mean=Mean_nij
IMBB_Median=imbB
IMBA_Median=imbA
IMBAB_Median=imbAB
SIMPSON_Median=Simpson
SHANNON_Median=Shannon));
label  pvalue='pvalue';

keep  pvalue B AB Error Median_nij Range_nij Var_nij Mean_nij
imbB imbA imbAB Simpson Shannon pdG;
```

```
run;
proc append base=iterations_con_01 data=iteration_con_01;
run;

dm "log;clear; odsresults;clear;";

%end;
data pvalues_con_01;
set iterations_con_01;
if pvalue lt .05 then Type_I =1; else Type_I=0;

label Median_nij='Median_nij'
   Range_nij='Range_nij'
Var_nij='Var_nij'
Mean_nij='Mean_nij'
imbB='imbB'
imbA='imbA'
imbAB='imbAB'
Shannon='Shannon'
Simpson='Simpson';
ddfm='Containment';
run;

%MEND iterate;


%iterate(iterations=5,a=3,b=5,maxn=5,mu=10,Bvar=9,ABvar=4,
Errorvar=1,rhoAB=0,rhoB=0,rhoError=0);


data output.pvalues_con_01;
set pvalues_con_01;
run;
```

The code above produces 10,000 datasets and tests for the significance of fixed effects using the containment ddfm. The same is run using the other three ddfm options, yielding 40,000 datasets in total. These 40,000 datasets are binned according the three unbalancedness indices with the following SAS code.

```
libname output "C:\Users\kbgregory\Documents\
Masters Research\Simulation Output\H_0 cov 0 data";
run;

data output.pvalues_all_01;
set  output.pvalues_kr_1st_01
 output.pvalues_con_01
 output.pvalues_satt_01
 output.pvalues_kr_01 ;
```

```
run;

proc univariate data=output.pvalues_all_01 ;
var Simpson Shannon ImbB ;
output out=Shannon_Simpson_Imb_01 pctlpts=10 20 30 40 50 60 70 80 90
pctlpre=Simpson_ Shannon_ ImbB_
pctlname=P10 P20 P30 P40 P50 P60 P70 P80 P90;
run;

data q;
set Shannon_Simpson_Imb_01;
i=1;
run;

data p;
set output.pvalues_all_01;
i=1;
run;

data bins_01;
merge p q (obs=1);
by i;

if  ImbB < ImbB_P10  then ImbB_bin=1;
else if ImbB >= ImbB_P10  and ImbB < ImbB_P20  then ImbB_bin=2;
else if ImbB >= ImbB_P20  and ImbB < ImbB_P30  then ImbB_bin=3;
else if ImbB >= ImbB_P30  and ImbB < ImbB_P40  then ImbB_bin=4;
else if ImbB >= ImbB_P40  and ImbB < ImbB_P50  then ImbB_bin=5;
else if ImbB >= ImbB_P50  and ImbB < ImbB_P60  then ImbB_bin=6;
else if ImbB >= ImbB_P60  and ImbB < ImbB_P70  then ImbB_bin=7;
else if ImbB >= ImbB_P70  and ImbB < ImbB_P80  then ImbB_bin=8;
else if ImbB >= ImbB_P80  and ImbB < ImbB_P90  then ImbB_bin=9;
else if ImbB >= ImbB_P90  then ImbB_bin=10;

if  Shannon < Shannon_P10  then Shannon_bin=1;
else if Shannon >= Shannon_P10  and Shannon < Shannon_P20  then Shannon_bin=2;
else if Shannon >= Shannon_P20  and Shannon < Shannon_P30  then Shannon_bin=3;
else if Shannon >= Shannon_P30  and Shannon < Shannon_P40  then Shannon_bin=4;
else if Shannon >= Shannon_P40  and Shannon < Shannon_P50  then Shannon_bin=5;
else if Shannon >= Shannon_P50  and Shannon < Shannon_P60  then Shannon_bin=6;
else if Shannon >= Shannon_P60  and Shannon < Shannon_P70  then Shannon_bin=7;
else if Shannon >= Shannon_P70  and Shannon < Shannon_P80  then Shannon_bin=8;
else if Shannon >= Shannon_P80  and Shannon < Shannon_P90  then Shannon_bin=9;
else if Shannon >= Shannon_P90  then Shannon_bin=10;

if  Simpson < Simpson_P10  then Simpson_bin=1;
else if Simpson >= Simpson_P10  and Simpson < Simpson_P20  then Simpson_bin=2;
else if Simpson >= Simpson_P20  and Simpson < Simpson_P30  then Simpson_bin=3;
else if Simpson >= Simpson_P30  and Simpson < Simpson_P40  then Simpson_bin=4;
else if Simpson >= Simpson_P40  and Simpson < Simpson_P50  then Simpson_bin=5;
```

```
else if Simpson >= Simpson_P50  and Simpson < Simpson_P60  then Simpson_bin=6;
else if Simpson >= Simpson_P60  and Simpson < Simpson_P70  then Simpson_bin=7;
else if Simpson >= Simpson_P70  and Simpson < Simpson_P80  then Simpson_bin=8;
else if Simpson >= Simpson_P80  and Simpson < Simpson_P90  then Simpson_bin=9;
else if Simpson >= Simpson_P90  then Simpson_bin=10;

bin_the_first=compress(var_bin||mean_bin||ImbB_bin||ImbA_bin);

bin=compress(ImbB_bin||Shannon_bin||Simpson_bin);

exp='01';

keep bin pvalue Type_I ddfm var_nij mean_nij ImbB ImbA var_bin
mean_bin ImbB_bin ImbA_bin bin  Shannon_bin Simpson_bin exp;

run;

proc sort data=bins_01;
by ddfm imbB_bin;
run;

data output.imbB_binstats_01;
set bins_01;
by ddfm imbB_bin;
retain FP 0 Total 0;

if first.imbB_bin then do ;
Total=0;
FP=0;
end;

Total+1;
FP=FP+type_I;
TN=Total-FP;

if ddfm = 'Kenward Roger 1st order' then ddfm='KR 1st';
if last.imbB_bin ;

type_I_rate = FP / Total;

keep imbB_bin FP TN Total type_I_rate ddfm exp;

run;

proc sort data=bins_01;
by ddfm Shannon_bin;
run;

data output.Shannon_binstats_01;
set bins_01;
```

```
by ddfm Shannon_bin;
retain FP 0 Total 0;

if first.Shannon_bin then do ;
Total=0;
FP=0;
end;


Total+1;
FP=FP+type_I;
TN=Total-FP;

if ddfm = 'Kenward Roger 1st order' then ddfm='KR 1st';
if last.Shannon_bin ;

type_I_rate = FP / Total;

keep Shannon_bin FP TN Total type_I_rate ddfm exp;

run;

proc sort data=bins_01;
by ddfm Simpson_bin;
run;

data output.Simpson_binstats_01;
set bins_01;
by ddfm Simpson_bin;
retain FP 0 Total 0;

if first.Simpson_bin then do ;
Total=0;
FP=0;
end;

Total+1;
FP=FP+type_I;
TN=Total-FP;

if ddfm = 'Kenward Roger 1st order' then ddfm='KR 1st';
if last.Simpson_bin ;

type_I_rate = FP / Total;

keep Simpson_bin FP TN Total type_I_rate ddfm exp;
run;
```

The following SAS code restructures the simulation output and performs

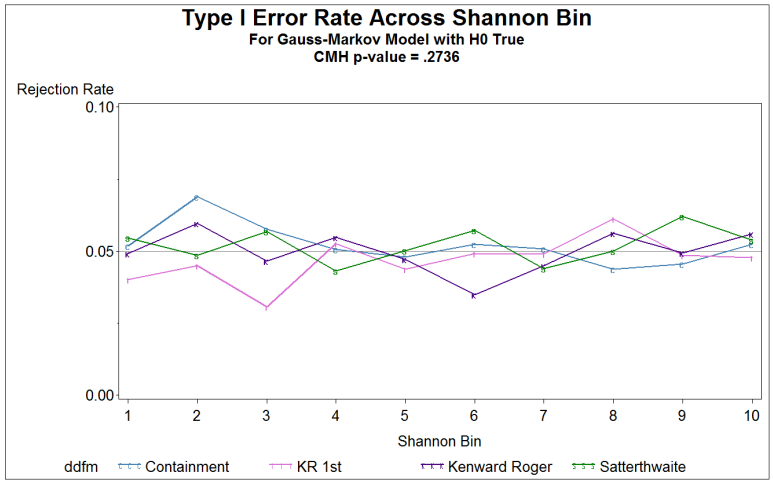
the CMH test for association between outcome and ddfm option conditioning on the index bin.

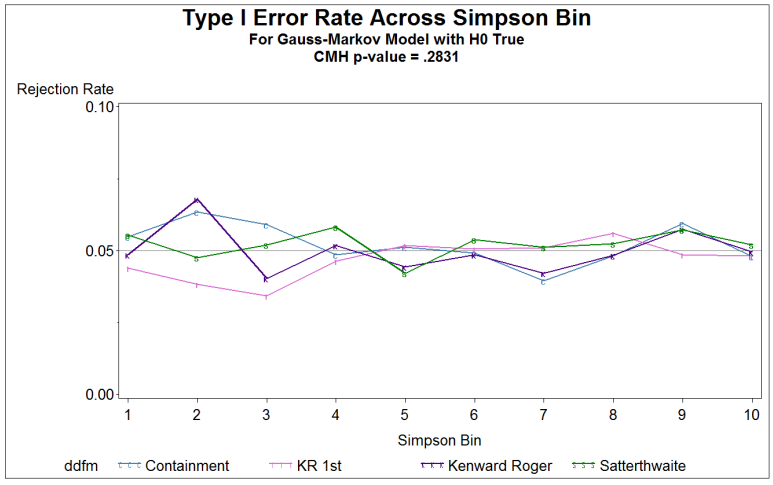
```
ods  listing;

libname output "C:\Users\kbgregory\Documents\
Masters Research\Simulation Output\H_0 cov 0 data";
run;


data simpson_CMH_01;
set output.Simpson_binstats_01;
do j = 1 to 2;
if j = 1 then do; freq= FP; Outcome='FP';end;
else if j=2 then do; freq = TN; Outcome='TN';end;
if ddfm = 'Kenward Roger 1st order' then ddfm='KR 1st';
output;
end;

keep ddfm simpson_bin outcome freq;
run;

/* Controlling for "Simpson Bin," is there any
 association between ddfm and outcome?  */

proc freq data=simpson_cmh_01 order=data;
tables simpson_bin*ddfm*outcome / CMH;
weight freq;
run;


data shannon_CMH_01;
set output.Shannon_binstats_01;
do j = 1 to 2;
if j = 1 then do; freq= FP; Outcome='FP';end;
else if j=2 then do; freq = TN; Outcome='TN';end;
if ddfm = 'Kenward Roger 1st order' then ddfm='KR 1st';
output;
end;
keep ddfm shannon_bin outcome freq;
run;

/* Controlling for "Shannon Bin," is there any
 association between ddfm and outcome?  */

proc freq data=shannon_cmh_01 order=data;
tables shannon_bin*ddfm*outcome / CMH;
weight freq;
run;
```

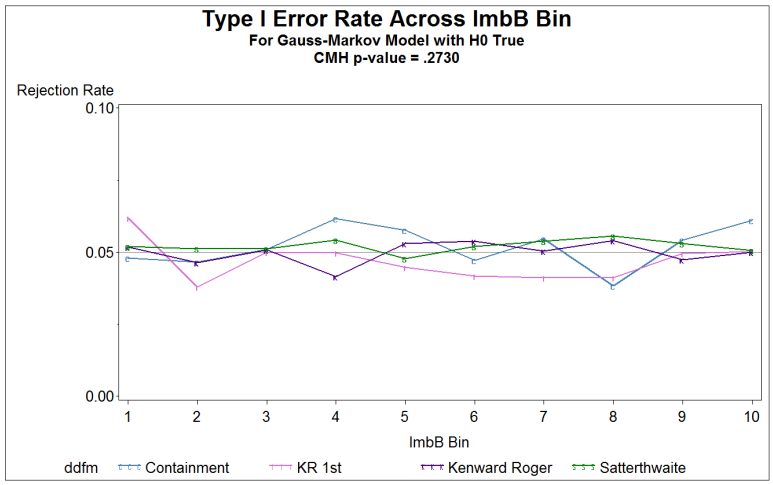
```
data ImbB_CMH_01;
set output.ImbB_binstats_01;
do j = 1 to 2;
if j = 1 then do; freq= FP; Outcome='FP';end;
else if j=2 then do; freq = TN; Outcome='TN';end;
if ddfm = 'Kenward Roger 1st order' then ddfm='KR 1st';
output;
end;
keep ddfm ImbB_bin outcome freq;
run;

/* Controlling for "ImbB Bin," is there any
 association between ddfm and outcome?  */

proc freq data=ImbB_cmh_01 order=data;
tables ImbB_bin*ddfm*outcome / CMH;
weight freq;
run;
```

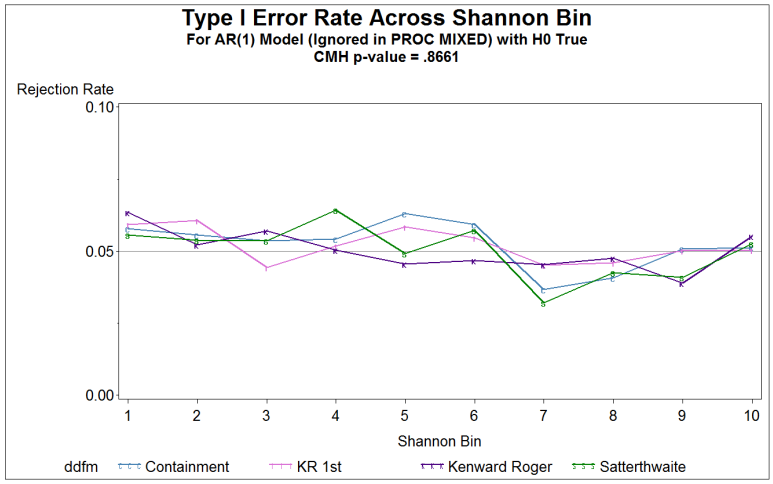
II.

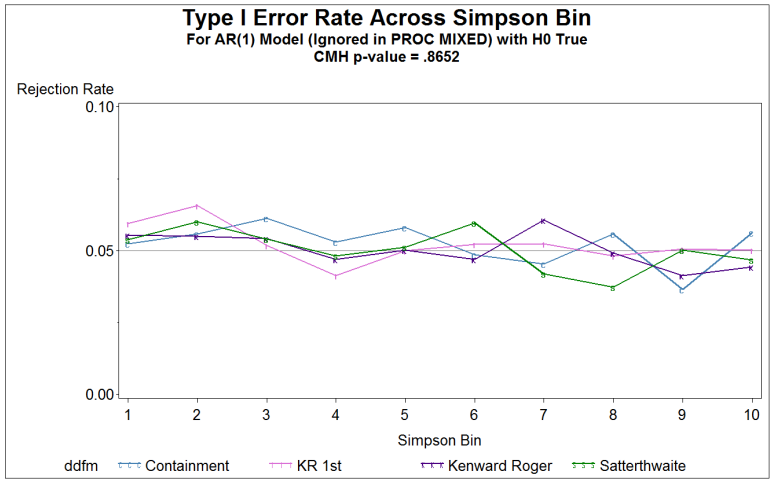
The graphical output for all simulations is provided here. There are 18 plots, 4 of which already appeared in the body of this paper but are given again for the sake of completeness. For each of the 6 simulated situations, there are three graphs, which represent the binning of the 40,000 randomly generated datasets according to the three different indices of unbalancedness. The p-value for the CMH test for association between ddfm option and outcome conditioned on bin is given with each plot.

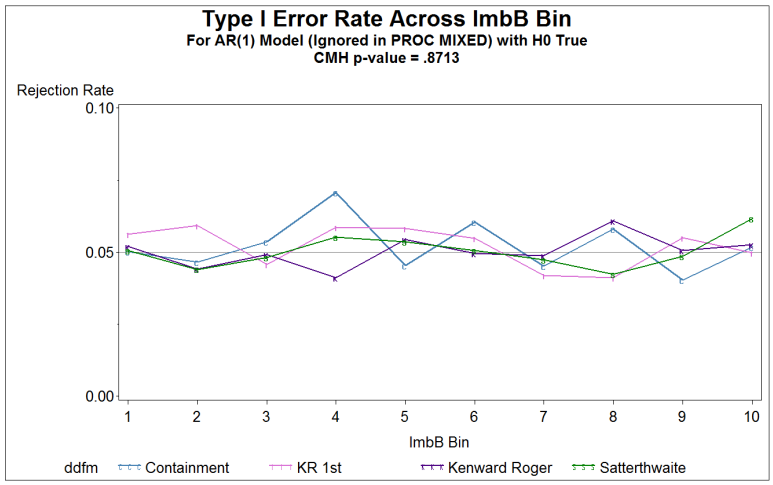

Type I Error Rate Across Shannon Bin
For Gauss-Markov Model with H0 True
CMH p-value = .2736
Rejection Rate
0.10
0.05
0.00
1
2
3
4
5
6
7
8
9
10
Shannon Bin
ddfm
Containment
KR 1st
Kenward Roger
Satterthwaite

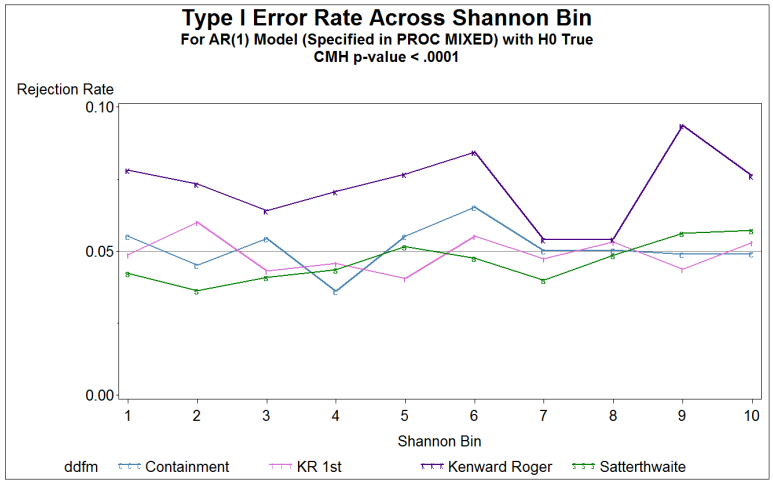

Type I Error Rate Across Simpson Bin
For Gauss-Markov Model with H0 True
CMH p-value = .2831
Rejection Rate
0.10
0.05
0.00
1
2
3
4
5
6
7
8
9
10
Simpson Bin
ddfm
Containment
KR 1st
Kenward Roger
Satterthwaite

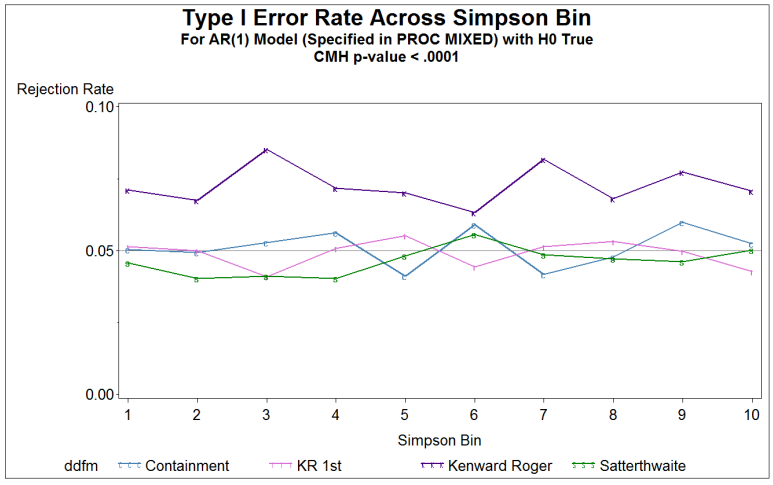

Type I Error Rate Across ImbB Bin
For Gauss-Markov Model with H0 True
CMH p-value = .2730
Rejection Rate
0.10
0.05
0.00
1
2
3
4
5
6
7
8
9
10
ImbB Bin
ddfm
Containment
KR 1st
Kenward Roger
Satterthwaite

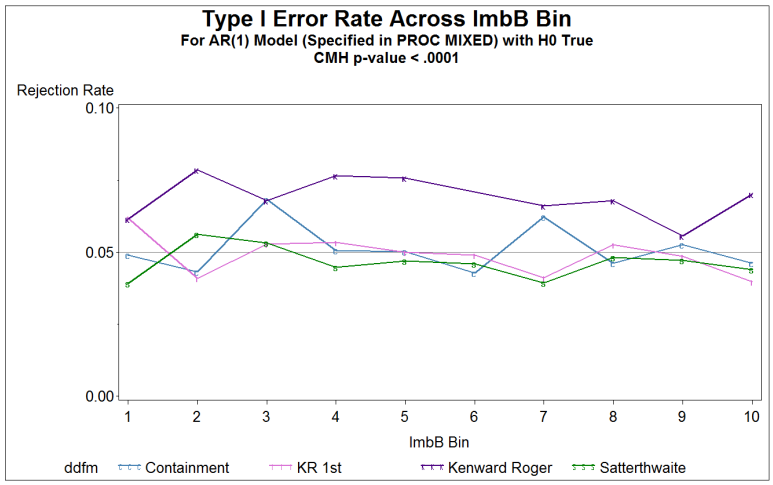

Type I Error Rate Across Shannon Bin
For AR(1) Model (Ignored in PROC MIXED) with H0 True
CMH p-value = .8661
Rejection Rate
0.10
0.05
0.00
1
2
3
4
5
6
7
8
9
10
Shannon Bin
ddfm
Containment
KR 1st
Kenward Roger
Satterthwaite

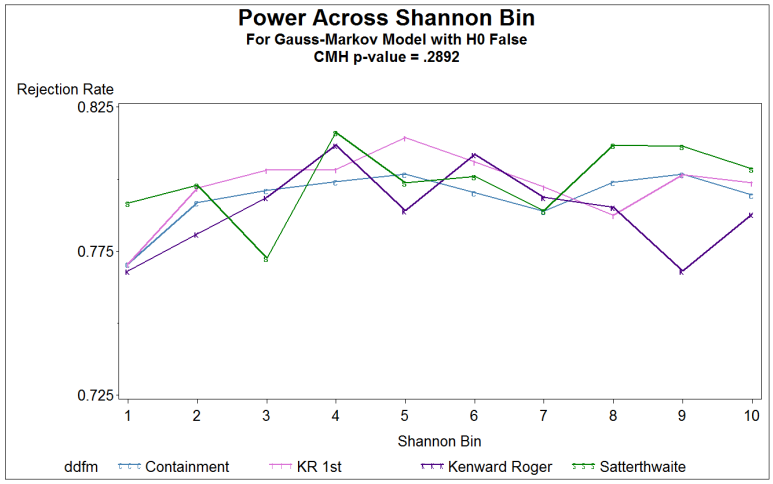

Type I Error Rate Across Simpson Bin
For AR(1) Model (Ignored in PROC MIXED) with H0 True
CMH p-value = .8652
Rejection Rate
0.10
0.05
0.00
1
2
3
4
5
6
7
8
9
10
Simpson Bin
ddfm
Containment
KR 1st
Kenward Roger
Satterthwaite

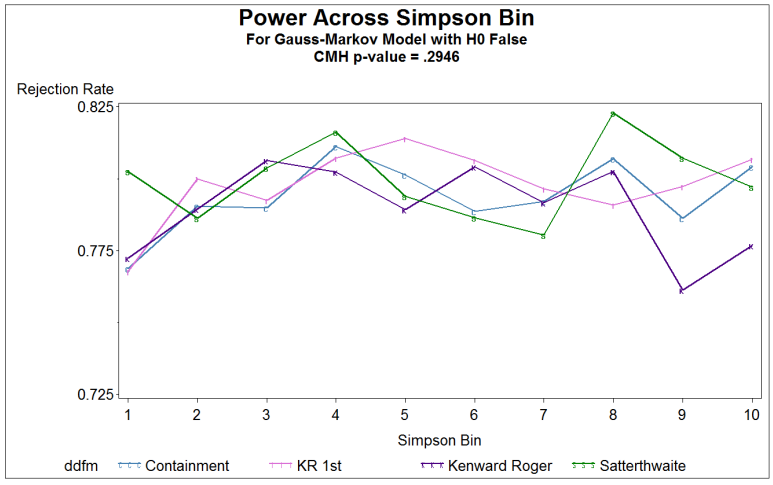

Type I Error Rate Across ImbB Bin
For AR(1) Model (Ignored in PROC MIXED) with H0 True
CMH p-value = .8713
Rejection Rate
0.10
0.05
0.00
1
2
3
4
5
6
7
8
9
10
ImbB Bin
ddfm
Containment
KR 1st
Kenward Roger
Satterthwaite

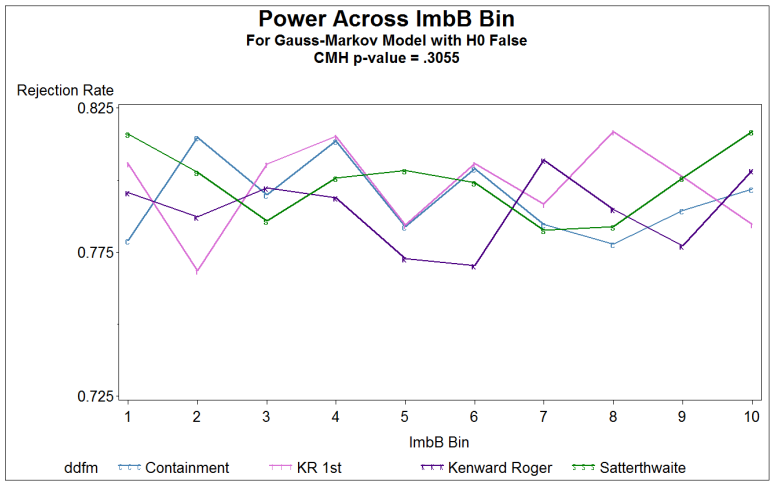

Type I Error Rate Across Shannon Bin
For AR(1) Model (Specified in PROC MIXED) with H0 True
CMH p-value < .0001
Rejection Rate
0.10
0.05
0.00
1
2
3
4
5
6
7
8
9
10
Shannon Bin
ddfm
Containment
KR 1st
Kenward Roger
Satterthwaite

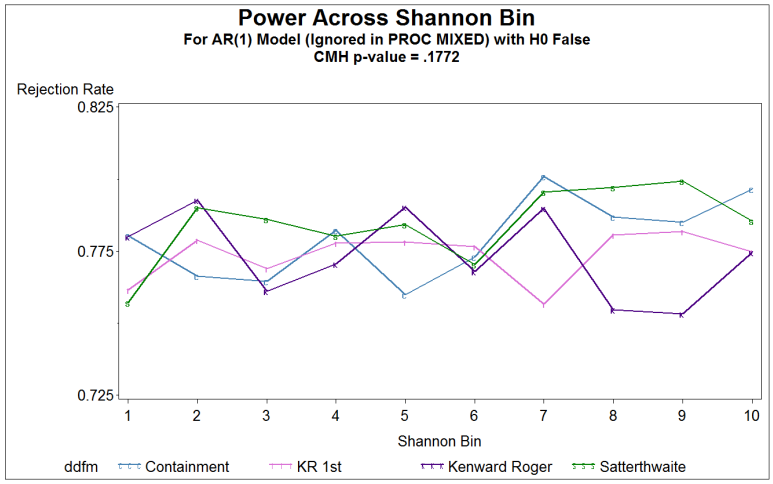

Type I Error Rate Across Simpson Bin
For AR(1) Model (Specified in PROC MIXED) with H0 True
CMH p-value < .0001
Rejection Rate
0.10
0.05
0.00
1
2
3
4
5
6
7
8
9
10
Simpson Bin
ddfm
Containment
KR 1st
Kenward Roger
Satterthwaite

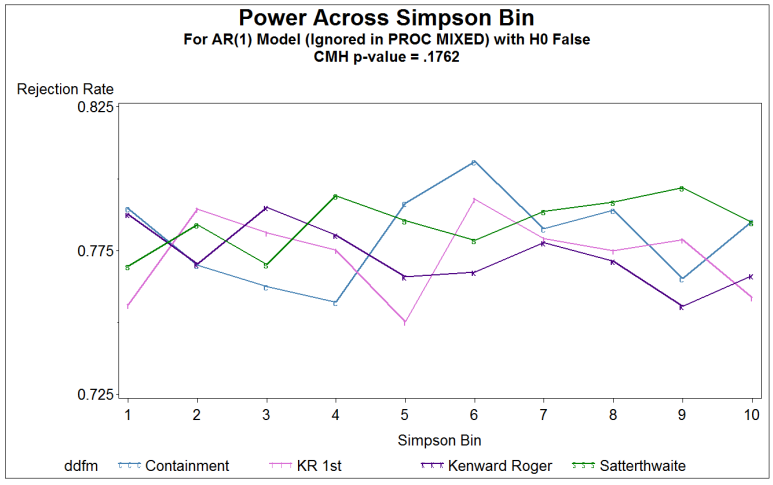

Type I Error Rate Across ImbB Bin
For AR(1) Model (Specified in PROC MIXED) with H0 True
CMH p-value < .0001
Rejection Rate
0.10
0.05
0.00
1
2
3
4
5
6
7
8
9
10
ImbB Bin
ddfm
Containment
KR 1st
Kenward Roger
Satterthwaite

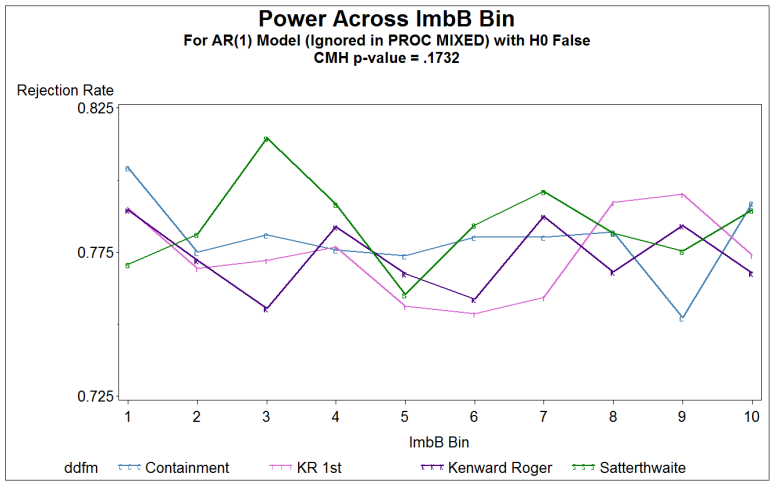

Power Across Shannon Bin
For Gauss-Markov Model with H0 False
CMH p-value = .2892
Rejection Rate
0.825
0.775
0.725
1
2
3
4
5
6
7
8
9
10
Shannon Bin
ddfm
Containment
KR 1st
Kenward Roger
Satterthwaite

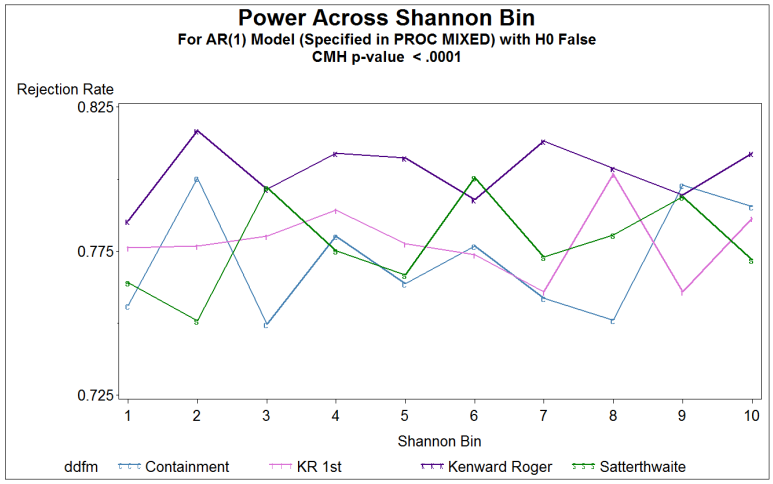

Power Across Simpson Bin
For Gauss-Markov Model with H0 False
CMH p-value = .2946
Rejection Rate
0.825
0.775
0.725
1
2
3
4
5
6
7
8
9
10
Simpson Bin
ddfm
Containment
KR 1st
Kenward Roger
Satterthwaite

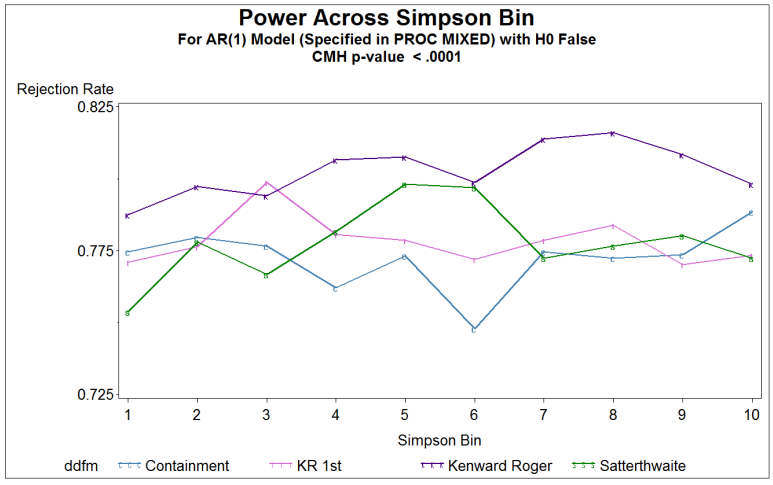

Power Across ImbB Bin
For Gauss-Markov Model with H0 False
CMH p-value = .3055
Rejection Rate
0.825
0.775
0.725
1
2
3
4
5
6
7
8
9
10
ImbB Bin
ddfm
Containment
KR 1st
Kenward Roger
Satterthwaite

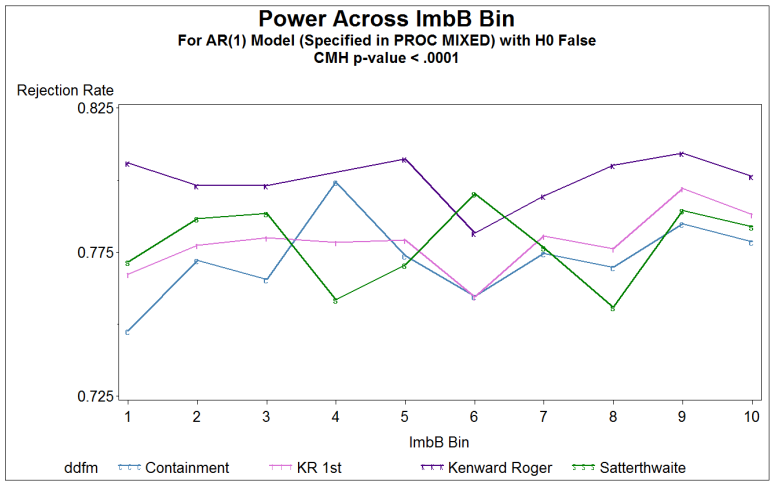

Power Across Shannon Bin
For AR(1) Model (Ignored in PROC MIXED) with H0 False
CMH p-value = .1772
Rejection Rate
0.825
0.775
0.725
1
2
3
4
5
6
7
8
9
10
Shannon Bin
ddfm
Containment
KR 1st
Kenward Roger
Satterthwaite


Power Across Simpson Bin
For AR(1) Model (Ignored in PROC MIXED) with H0 False
CMH p-value = .1762
Rejection Rate
0.825
0.775
0.725
1
2
3
4
5
6
7
8
9
10
Simpson Bin
ddfm
Containment
KR 1st
Kenward Roger
Satterthwaite


Power Across ImbB Bin
For AR(1) Model (Ignored in PROC MIXED) with H0 False
CMH p-value = .1732
Rejection Rate
0.825
0.775
0.725
1
2
3
4
5
6
7
8
9
10
ImbB Bin
ddfm
Containment
KR 1st
Kenward Roger
Satterthwaite


Power Across Shannon Bin
For AR(1) Model (Specified in PROC MIXED) with H0 False
CMH p-value < .0001
Rejection Rate
0.825
0.775
0.725
1
2
3
4
5
6
7
8
9
10
Shannon Bin
ddfm
Containment
KR 1st
Kenward Roger
Satterthwaite


Power Across Simpson Bin
For AR(1) Model (Specified in PROC MIXED) with H0 False
CMH p-value < .0001
Rejection Rate
0.825
0.775
0.725
1
2
3
4
5
6
7
8
9
10
Simpson Bin
ddfm
Containment
KR 1st
Kenward Roger
Satterthwaite


Power Across ImbB Bin
For AR(1) Model (Specified in PROC MIXED) with H0 False
CMH p-value < .0001
Rejection Rate
0.825
0.775
0.725
1
2
3
4
5
6
7
8
9
10
ImbB Bin
ddfm
Containment
KR 1st
Kenward Roger
Satterthwaite