

Assessing Variabilities for Bioanalytical Methods

By
Binal Patel
May 2010

A Community Based Master's Project presented to the faculty of Drexel University School of Public Health in partial fulfillment of the Requirement for the Degree of Master of Public Health.

Dr. Jason Liao, Preceptor, Merck & Co., Inc.
Dr. Issa Zakeri, Advisor, Drexel School of Public Health

Acknowledgements:

I would like to thank Jane Liao, a programmer, at Merck & Co. for some guidance with graphing SAS code. I would also like to give a special thanks to Dr. Jason Liao, the director of Non-Clinical Statistics at Merck & Co., and my preceptor and to Dr. Issa Zakeri, my advisor at Drexel School of Public Health. They both have guided me along this process and been very helpful.

Table of Contents

Abstract.....	vi
Introduction.....	1
Background and Significance.....	1
Research Design and Methods.....	10
Results.....	14
Discussion and Conclusions.....	22
References.....	24

List of Appendices

Appendix A.....	Bias Simulation Distributions – for bias within, bias between, and bias total
Appendix B.....	MSE Simulation Distributions – for bias within, bias between, and bias total
Appendix C.....	SAS Code for Simulation
Appendix D.....	SAS code for bias/MSE graphs
Appendix E.....	SAS code for Histograms
Appendix F.....	SAS code for DATA

List of Tables

i. Simulation Values for Variables.....	pg 11
ii. CV (RSD) Values for Sigma Between and Within.....	pg 12
iii. Table of Variances for Each Test Type.....	pg 22

List of Illustrations

Figure 1. Accuracy and Precision.....	pg 14
Figure 2. Simulation graph one – Final Bias for Certain Parameters.....	pg 15
Figure 3. Simulation graph two – Final MSE for Certain Parameters.....	pg 16
Figure 4. Simulation graph three – Final Bias for Certain Parameters.....	pg 17
Figure 5. Simulation graph four – Final MSE for Certain Parameters.....	pg 18
Figure 6. Simulation graph five – Final Bias for Certain Parameters.....	pg 19
Figure 7. Simulation graph six – Final MSE for Certain Parameters.....	pg 19
Figure 8. Simulation graph seven – Final Bias for Certain Parameters.....	pg 20
Figure 9. Simulation graph eight – Final MSE for Certain Parameters.....	pg 21

Abstract

Assessing Variabilities for Bioanalytical Methods

Binal Patel

Dr. Jason Liao, Preceptor, Merck & Co., Inc.

Dr. Issa Zakeri, Advisor, Drexel School of Public Health

Variability arises from many different sources. In the pharmaceutical industry, it is important that these variabilities are estimated and controlled in order to provide a good quality and safe product. The FDA has certain guidelines dealing with precision that need to be followed. This analysis tested five different methods from SAS[®] to assess which test should be used. These five tests include methods from PROC VARCOMP – ML, REML, TYPE1 and MIVQUE0 and from PROC MIXED – REML. This analysis simulated data to get an estimate of precision under true values. The simulation only dealt with two parameters where variability could arise and it had four possible values for each. The simulation also took into account the number of labs and number within, which also had four values each, in total created 252 different scenarios. Bias and mean square error (MSE) means were used to evaluate these five methods. Data was also obtained from Merck & Co. and CV (RSD) values were found to check precision of various tests. Generally speaking, the bias and MSE decrease as the sample size increases for all methods. MIVQUE0 was concluded to be the best test for checking variability. If the sample size is high, the testing method does not matter because all will produce the same or similar results, however, the pharmaceutical industry does not have the luxury of having many lab assays available. Therefore, these results are based on providing the best recommendation for the current situation and not an ideal situation.

Introduction:

In a biologics and vaccine product development, the process is the product. In other words, a well characterized and controlled process is the key in developing a biologics/vaccine product. Due to the structure complicity with high dimension heterogeneity, the process is very complicated and it is hard to characterize and control. Thus, it is very important to assess and control the variability and identify the sources of variability so that with a well defined goal and cost, an acceptable product is achievable.

The purpose of this study is to assess different statistical methods in estimating the variability and find one with good statistical properties such as small bias and mean squared error (MSE). *In Vitro* data will be provided from Merck & Co. that will be used for analysis, and data will be simulated in order to assess different scenarios.

By comparing different statistical methods, using the SAS[®] program, and finding a proper way to estimate variabilities, will allow for a method to be used that is both accurate and consistent. This will then be shared with Merck & Co., Inc. so it can be taken into account in their product operation strategy.

Background and Significance:

There are many different bioanalytical methods used in drug development. Some of these are “gas chromatography (GC), high-pressure liquid chromatography (LC), combined GC and LC mass spectrometric (MS) procedures such as LC-MS, LC-MS-MS, GC-MS, and GC-MS-MS performed for the quantitative determination of drugs and/or metabolites in biological matrices such as blood, serum, plasma or urine (FDA, 2001).” There are other methods that use tissue and skin samples as well. These different methods need to be controlled for variability in order to produce a drug that is the most beneficial. Variability can arise from many difference sources

such as lack of consistency within an observer when carrying out numerous recordings or lack of consistency between different observers (Brennan & Silman, 1992). This affects the reliability of the findings, which is very important in clinical applications.

The FDA has procedures that must be followed in the pharmaceutical industry when it comes to analyzing bioanalytical methods. “Selective and sensitive analytical methods for the quantitative evaluation of drugs and their metabolites (analytes) are critical for the successful conduct of preclinical and/or biopharmaceutics and clinical pharmacology studies (FDA, 2001, p. 2).” This is necessary because it ensures that the particular biological matrix in question is reliable and is reproducible. There are three types of validation that the FDA characterizes: Full Validation, Partial Validation and Cross Validation.

Full validation is needed when there is a new drug or a particular bioanalytical method is being used for the first time (FDA, 2001). Partial validations are “modifications of already validated bioanalytical methods. Partial validation can range from as little as one intra-assay accuracy and precision determination to a nearly full validation (FDA, 2001, p. 2).” Some of the changes that may be considered to fall into this category, according to the FDA are:

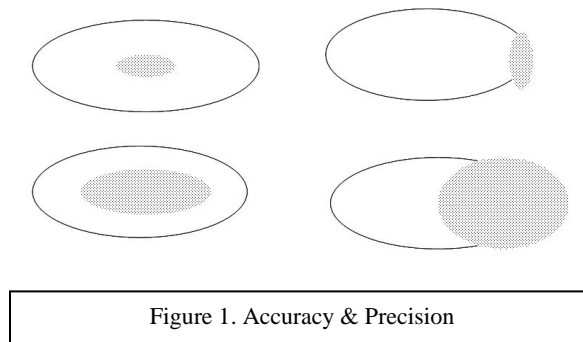
- Bioanalytical method transfers between laboratories or analysts
- Change in analytical methodology (e.g., change in detection systems)
- Change in anticoagulant in harvesting biological fluid
- Change in matrix within species (e.g., human plasma to human urine)
- Change in sample processing procedures
- Change in species within matrix (e.g., rat plasma to mouse plasma)
- Change in relevant concentration range
- Changes in instruments and/or software platforms
- Limited sample volume (e.g., pediatric study)
- Rare matrices
- Selectivity demonstration of an analyte in the presence of concomitant medications
- Selectivity demonstration of an analyte in the presence of specific metabolites

Cross validation is “a comparison of validation parameters when two or more bioanalytical methods are used to generate data within the same study or across different studies (FDA, 2001, p. 3).”

When validating these bioanalytical methods, it must include processes that show that the quantitative methods used are reproducible with low variability. The FDA has 6 parameters the bioanalytical method validation is based on: accuracy, precision, selectivity, sensitivity, reproducibility and stability (FDA, 2001).

The FDA defines selectivity as “the ability of an analytical method to differentiate and quantify the analyte in the presence of other components in the sample (FDA, 2001, p. 4).” Stability is the “chemical stability of an analyte in a given matrix under specific conditions for given time intervals (FDA, 2001, p. 21).” Reproducibility is the “precision between two laboratories. It also represents precision of the method under the same operating conditions over a short period of time (FDA, 2001, p. 21).”

Precision and accuracy can sometimes be mistaken for the same thing because in common language, they are synonyms. In an analytical standpoint, these two words have very different meanings and therefore some issues may arise in determining one or the other or both. The best way to differentiate between these two terms is to think of them with darts. If a person has four darts and hits all darts close to the bulls’ eye but in different spots, then the person has accuracy since it is close to the intended target. However, if the same person hits the dart far away from the bulls’ eye, but manages to get all four darts relatively close together, then the person is precise. In order to have both precision and accuracy, the intended target should be reached, the same way every time. The following figure depicts four difference scenarios:



The first scenario shows a case where there is both good accuracy and precision. The second scenario shows a case where there is good precision but bad accuracy. The third scenario shows a case where there is good accuracy but bad precision and the fourth scenario shows a case where there is neither good accuracy nor precision.

Precision can be further broken up into repeatability, within-laboratory reproducibility and reproducibility (Menditto, Patriarca, & Magnusson, 2007). Repeatability is “when the least changes are allowed (e.g. assays carried out over a short period of time, by the same analyst using the same instrument, etc.) (Menditto, et al, 2007, p. 46).” Within-in laboratory reproducibility, also known as intermediate precision, is “when, within the same laboratory, any relevant influence factor is allowed to vary (e.g. assays carried out over a longer period of time, by different analysts, using different reagent lots, in different environmental conditions and even using different instruments of the same specifications) (Menditto, et al, 2007, 46).”

Precision can be explained and assessed on a “within-batch and a between-batch basis (Karnes, Shiu & Shah, 1991, p. 424).” The within batch assessment is “considered as a measure of the precision of a method under optional conditions, where as the between batch assessment is considered a better representation of the precision (Karnes et al., 1991, p. 424).” The between batch is a better representation because this measure assesses the data that is more likely to be

subjected to different sources of variability (Karnes et al., 1991). Karnes, Shiu, and Shah (1991) state that the total error can also be shown as:

$$E = |x - \mu| + 2.58 \text{ SD}$$

This equation shows that accuracy calculations need to be performed on mean values (Karnes et al., 1991).

Repeated measures are multiple responses in one experimental unit, which are usually taken over time. The purpose of repeated measures is to analyze the response trends over time (Littell, Henry, & Ammerman, 1998). “The covariance structure refers to variances at individual times and to the correlation between measures at different times on the same unit (Littell et al., 1998, p. 1222).” The variation between the unit deals with measures that are correlated because of similar characteristics. The variation within the unit deals with responses that are closer in time than to the responses that are further away from time. The means of the bias within and the bias between will be calculated as well as their respective mean square error in order to analyze the different statistical methods.

A sequence of repeated independent measurements that have no change in the procedure may not have the same outcome values as discussed before and the fluctuations would be modeled as random. However, there may be errors that have the same effect on every measurement (Rice, 2007). “If the true value of the quantity being measured is denoted by x_0 , the measurement, X , is modeled as

$$X = x_0 + \beta + \varepsilon$$

where β is the constant, or systematic, error and ε is the random component of the error; ε is a random variable with $E(\varepsilon) = 0$ and $\text{Var}(\varepsilon) = \sigma^2$. We then have

$$E(X) = x_0 + \beta$$

and

$$\text{Var}(X) = \sigma^2$$

β is often called the bias of the measurement procedure (Rice, 2007, p. 135).” The bias and variance are two factors that affecting the size of the error. A perfect measurement would have both bias and variance equal to zero (Rice, 2007). An overall measure of the measurement error is called the mean squared error and is given by

$$MSE = E[(X - x_0)^2]$$

Since the MSE is the expected squared of the difference between X and x_0 , it can be decomposed into squared bias and variance, as the following states (Rice):

$$MSE = \beta^2 + \sigma^2$$

Assays need to be validated and part of it is to determine how precise they are so there can be confidence of the finding. What actually ends up being tested is imprecision and in a normal distribution, the measure of imprecision would end up being the standard deviation (Chesher, 2008). Sometimes the variance is used, which is the standard deviation squared. Holding this true with a normal distribution then the total variance is equal to the sum of the variability from all different sources. For example, the following shows the equation in a simple case, for total variance from a between and within run:

$$\sigma^2_{Total} = \sigma^2_{Within-run} + \sigma^2_{Between-run}$$

Usually when measuring analytes in a matrix, the mean and standard deviation are unknown (Chesher, 2008). Therefore, the following equations are use to find estimates of the mean, variance and CV (RSD):

$$\bar{x} = \frac{\sum x}{n}$$

$$S^2 = \frac{\sum(x - \bar{x})^2}{n - 1} = \frac{\left(\sum x^2 - \frac{(\sum x)^2}{n}\right)}{n - 1}$$

$$CV(RSD) = \frac{s}{\bar{x}} \times 100 \%$$

where CV is the coefficient of variation (Chesher, 2008) . The CV is also called the Relative standard deviation (RSD) (Findlay, 2000). This number is a way to quantitatively measure precision.

Data will be provided from Merck & Co. that was used to check the precision of relative IC50 values. IC50 values, also known as half maximal inhibitory concentration values, are important in the pre-vaccine production process. These values are a measure of how a substance can inhibit a biological function by half, hence the 50. The FDA states that “IC50 represents the concentration of a drug that is required for 50% inhibition of viral replication *in vitro*.” These values were used in a binding assay. Assays are used to measure the amount of activity in any biological organism or in this case, a drug. The data provided was collected by experiments that were conducted by two different analysts, who used different number of labs and were performed on different days. In this particular case, when trying to measure for the same thing, variability will be present.

To find the variance, a mixed model will be used. A linear mixed effects model is a variation of a standard linear model that generalizes correlation and variability that is not constant. A mixed effects model contains both fixed effects as well as random effects. Fixed effects occur when levels of an effect are from an entire population, while random effects are when you make an inference about the population and it is only a sample and not the entire population (SAS Institute Inc., 2002-2005). It is important to use a model that considers random effects because these could affect the variability of the data. The formula for the linear mixed

effects model is:

$$y = X\beta + Z\gamma + \varepsilon$$

where β is a vector of fixed effects, X is a matrix of covariates, Z is a matrix of covariates whose columns are a subset of the columns of X , γ is a vector of random effects and assumed to have a multivariate normal distribution with mean vector zero covariance matrix G , and ε is the error term. Some advantages of using this model is that it can take into account missing data, allow random effects to be properly computed and specified, has more flexibility in modeling the error, and it can allow the error terms to show non-constant variability, which helps in modeling the dependent variable (Yaffe, 2001). A key assumption for this model is that γ and ε are normally distributed with

$$E \begin{bmatrix} \gamma \\ \varepsilon \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$Var \begin{bmatrix} \gamma \\ \varepsilon \end{bmatrix} = \begin{bmatrix} G & 0 \\ 0 & R \end{bmatrix}$$

The PROC MIXED procedure in SAS[®] allows a modeling of this mixed effects linear model as the means of the data as well as their variances and covariances (SAS Institute Inc., 2002-2005). According to the SAS onlinedoc, there are three assumptions about this procedure:

1. The data are normally distributed
2. The means or expected values, of the data are linear in terms of a certain set of parameters
3. The variances and covariances of the data are in terms of a different set of parameters, and they exhibit a structure matching of those available in PROC MIXED.

“Since Gaussian data can be modeled entirely in terms of their means and variances/covariances, the two sets of parameters in a mixed linear model actually specify the complete probability distribution of the data. The parameters of the mean model are referred as fixed-effects parameters, and the parameters of the variance-covariance model are referred to as covariance

parameters (SAS Institute Inc., 2002-2005).” The covariance estimates are what makes the mixed linear model different from a standard linear model. There are certain instances where covariance parameters are needed (SAS Institute Inc., 2002-2005):

1. The experimental units on which the data are measured can be grouped into clusters, and the data from a common cluster are correlated
2. Repeated measurements are taken on the same experimental unit, and these repeated measurements are correlated or exhibit variability that changes.

Vaccine production deals with repeated measurements on the same experimental unit, therefore PROC MIXED will be used for analysis. The PROC MIXED procedure can also take into account randomness – it can test a variable for random effects that may affect the variability of the data.

The PROC MIXED model also includes the PROC VARCOMP procedure in SAS, which deals with linear models that have random effects (SAS Institute Inc., 2002-2005). This procedure estimates the random effects contribution to the variance of the dependent variable (SAS Institute Inc., 2002-2005). PROC VARCOMP deals with simple random effects, while PROC MIXED, in addition to analyzing variance components, can take into account many different analyses as well as other covariate structures. PROC VARCOMP can actually give values as negative; however, with PROC MIXED, the values will be shown as zero. There are four different types of methods that can be used in this procedure.

The Type I method (type1) “computes the Type I sum of squares for each effect, equates each mean square involving only random effects to its expected value, and solves the resulting system of equations (SAS Institute Inc., 2002-2005).” The “mean square is the sum of squares divided by the degrees of freedom and the expected mean square is the expected value of the mean square in the mixed model” – the type 1 estimates result from solving the equations by

equating the observed mean square to the expected values (SAS Institute Inc., 2002-2005). The MIVQUE0 method “produces unbiased estimates that are invariant with respect to the fixed effects of the model and that are locally best quadratic unbiased estimates given that the true ratio of each component to the residual error component is zero (SAS Institute Inc., 2002-2005).” A sum of squares matrix is produced and the estimates stem from solving equations presented by this matrix. This only varies from the Type I method by adjusting the random effects only for the fixed effects. The Maximum Likelihood method (ML) “computes maximum-likelihood estimates of the variance components (SAS Institute Inc., 2002-2005).” The Restricted Maximum-Likelihood method (REML) “is similar to the maximum likelihood method, but it first separates the likelihood into two parts: one that contains the fixed effects and one that does not (SAS Institute Inc., 2002-2005).” These four methods along with the PROC MIXED REML will be analyzed.

Research Design and Methods

- A. Overview of study design: This study will have two analyses done. One used simulated data and another used the data provided by Merck & Co., Inc. These two analyses will be investigated to find the best statistical method for estimating variabilities.

- B. Subjects: Subjects will be simulated in one part of the analysis. The other part will have subjects that are non-human, lab-based samples. These samples will be obtained from Merck & Co., Inc.

- C. Data Collection Methods and Procedures:

Source of data: A simulation will create data for one part of the analysis by using SAS. For another part of the analysis, data will come from Merck & Co., Inc. This data will be non-human, non-identifiable data.

Procedure – Part 1: A simulation program was created to assess the bias and MSE for different variance estimating procedures. Since the mean and standard deviation are unknown, a number of variables were given to see different scenarios for the five different SAS methods. The mean, or μ , was set to 1 with rep set to 1000. Depending on the number of labs, a value was found by the following variance component model:

$$y = \mu + \delta + \varepsilon$$

where for between values, $\delta \sim N(0, \sigma_b^2)$ and for within values, $\varepsilon \sim N(0, \sigma_w^2)$.

The sigma's for both between and within values had four distinct values. The number of labs as well as the number within also had distinct values. The following table shows the values for each.

TABLE 1. SIMULATION VALUES FOR VARIABLES

Sigma Between	Sigma Within	Number of Labs	Number Within
.05	.05	2	2
.1	.1	3	5
.2	.2	10	10
.5	.5	30	30

TABLE 2. CV(RSD) VALUES FOR SIGMA BETWEEN AND WITHIN

CV Values	0.05	0.1	0.2	0.5
0.05	0.07079916	0.11215370	0.20836520	0.53594729
0.1	0.1121537	0.1421314	0.2264312	0.5449129
0.2	0.2083652	0.2264312	0.2885950	0.5800237
0.5	0.5359473	0.5449129	0.5800237	0.8054324

The model with these values was used to create a dataset. That dataset was then processed through the five different statistical methods from PROC MIXED and PROC VARCOMP using SAS[®]. The VARCOMP procedure in SAS is especially designed to obtain estimates of the variance components in a mixed model. Y was modeled against the number of labs. This data was then combined and outputted into a dataset that unique variables (i.e. sigma between, sigma within, number of labs, number within). There were five different datasets for each method, which then were combined into one, which also contained the variance component of lab and the variance component of error. The variance component of lab is also known as the variance between and the variance component of error is also known as the variance within. Once these datasets were created, the bias and MSE were calculated. The bias between and within was found by:

$$\text{simulated bias}(\hat{\alpha}) = \frac{1}{1000} \sum_{s=1}^{1000} (\hat{\alpha}_s - \alpha)$$

The MSE between and within was found by:

$$\text{simulated MSE}(\hat{\alpha}) = \frac{1}{1000} \sum_{s=1}^{1000} (\hat{\alpha}_s - \alpha)^2$$

The bias total and MSE total was also calculated. Distributions of the simulated bias and simulated MSE for between, within and total are attached in the appendix for a particular combination. Using the PROC MEANS function in SAS, the mean for the bias between, within and total, as well as the mean for the MSE between, within and total were calculated. Since there were four varying variables, each with four different possibilities, there were 252 datasets that were produced. From these datasets, a program was written to graphically show the means of the bias and MSE by each test type. The x-axis was split into four different constants in order to show the trend when three of the variables are held constant.

Procedure – Part 2: Data from Merck & Co., Inc. was used to conduct a similar procedure. The data contained IC50 values and the precision of these values needed to be calculated. The lab data contained various factors that could influence variability, such as analyst, date, lot and concentration. This data was processed through the five different PROC MIXED and PROC VARCOMP procedures and then combined into one dataset. The new dataset contained total variance, which was simply obtained by adding the variances of all the factors together. Since the bias and MSE cannot be calculated due to not knowing the true mean and variance, CV (RSD) values were be calculated instead to measure variability.

D: Analysis Plan

Part 1: The analysis was be done by using SAS[®] version 9.1. The statistical methods that were used to analyze these data were the PROC VARCOMP procedure and the PROC MIXED procedure. The PROC VARCOMP procedure has four different types of tests that can be used: REML, MIVQUE0, TYPE1 and ML. The PROC MIXED procedure uses the REML test type. Therefore, there were a total of five different test outputs that will be considered. These outputs were transformed into graphs, which showed a comparison of the bias and the MSE for both between and within.

Part 2: The analysis was also done by using SAS[®] version 9.1. The same PROC VARCOMP and PROC MIXED procedures were used. The difference is that with the actual data, bias and MSE cannot be calculated so graphs were not produced for comparison. Instead, the CV (RSD) values were calculated for each factor by test type in order to see which test produces the least amount of variability.

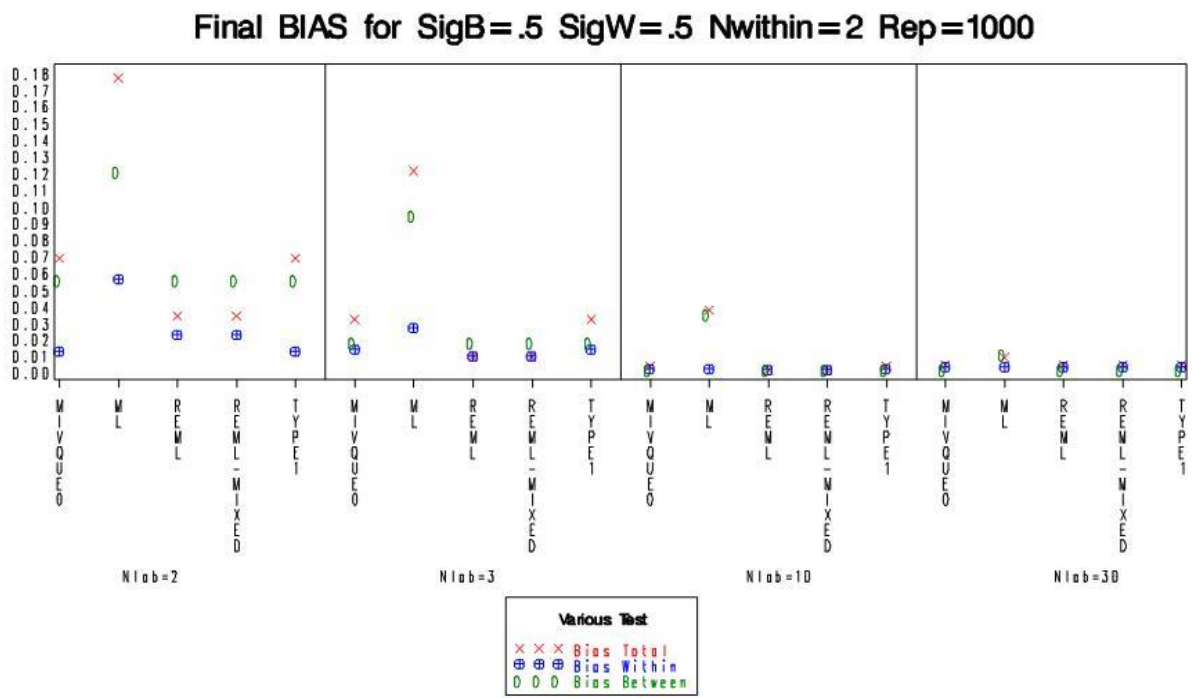
E: Human Subjects Consideration: There was no involvement of human subjects. The data that was used was non-human and that data was not identifiable. All data was either simulated or will come from Merck & Co., Inc. The data has already been obtained by Merck & Co., Inc. therefore there was no collection process.

Results

A. Part 1 – Simulation

Simulated data were graphed producing 252 mean bias graphs as well as 252 mean MSE graphs. A certain constant was picked which had values that were closest to reality – sigma between = 0.5, sigma within = .5, number within = 2, with the number of labs varying. Choosing the number of labs to vary, will allow us to notice any trends present. The following is the bias graph simulated:

FIGURE 2. Simulation graph one – Final Bias for Certain Parameters

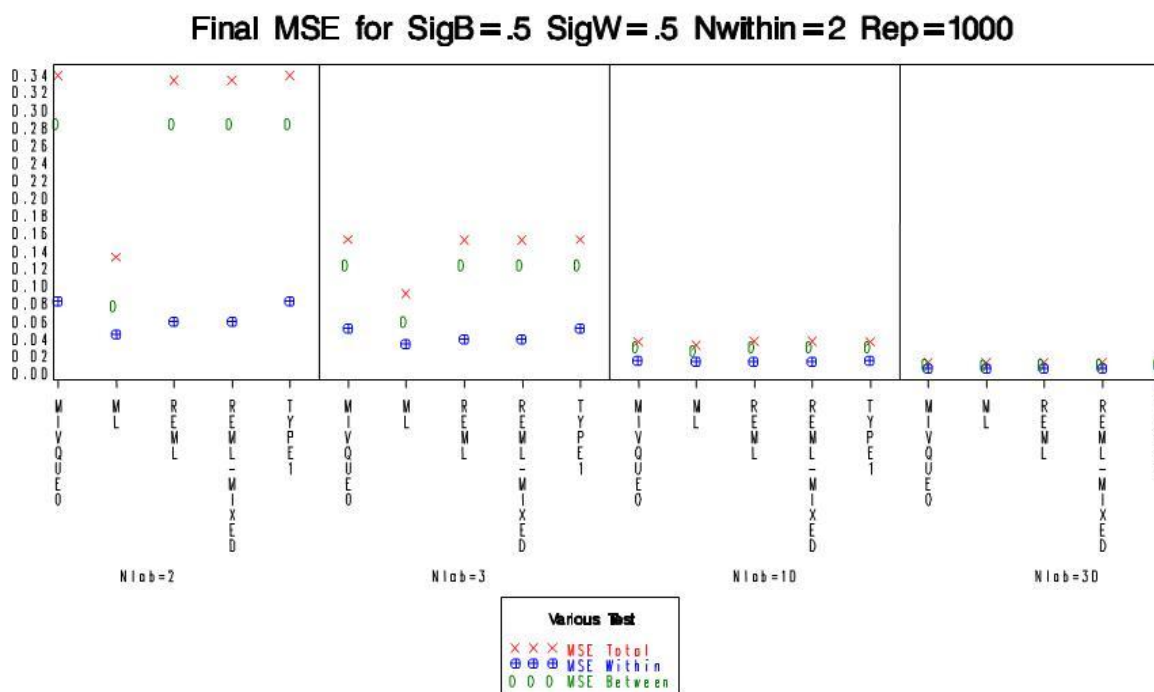


By looking at this graph, it can be shown that as the number of labs increases, the bias decreases. Also, both REML methods from PROC VARCOMP and PROC MIXED,

output the same result, which should be expected since both use the same method. The ML test is the test that varies the most – it has a very large bias compared to the other tests, however, once the number of labs increases, the bias of ML gets closer to the values of the others. MIVQUE0 and TYPE1 tests also have similar results, which is to be expected since both procedures are similar.

The following is the MSE graph simulated:

FIGURE 3 Simulation graph two – Final MSE for Certain Parameters



This graph depicts the same trend that as the number of labs increases, the MSE decreases. One noticeable difference is that the MSE between values are a lot higher than the MSE within values, when the number of labs is small. As the number of labs increases, the difference between the two decreases. As the REML from both PROC

MIXED and PROC VARCOMP both have the same results as before as well as MIVQUE0 and TYPE1.

Looking at Figure 4, we are now checking the trends of bias and MSE across the number within values. MIVQUE0 and TYPE1 both have the smallest bias when the number within is small. ML on the other hand, has the highest bias. The REML procedures both have bias values in between those of ML and MIVQUE0/TYPE1. As the number within increases, we see the same trend that all the procedures end up outputting the same results, however, this was at a lot faster rate than compared to the number of labs. ML is still the only one to have a high bias, regardless of the increase in number within.

FIGURE 4. Simulation graph three – Final Bias for Certain Parameters

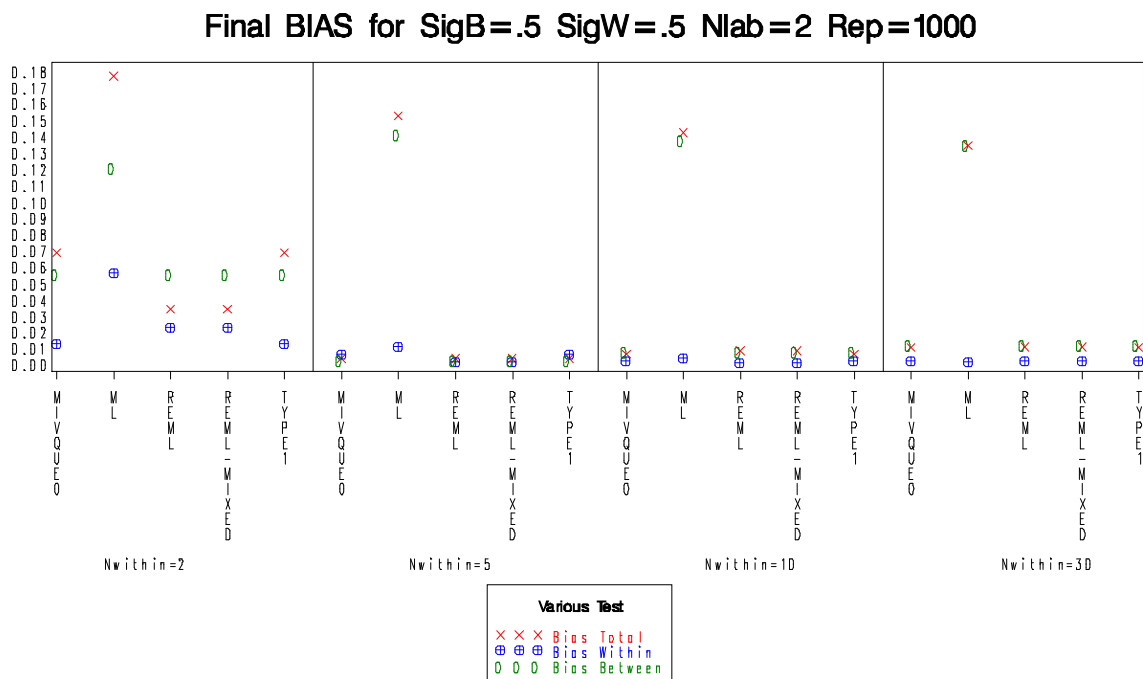
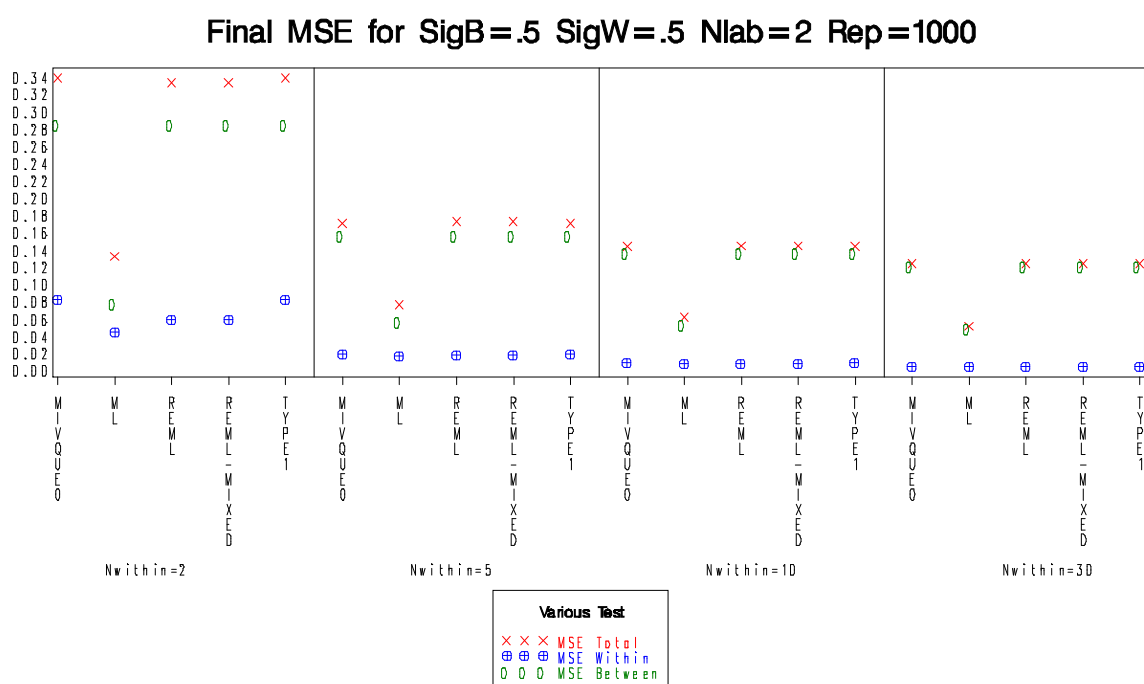


Figure 5, showing the MSE trends across the number within, shows that for a low number within, the MSE varies depending on the test type, but as the number within increases, the MSE values become similar to each other across all tests. For the lowest number within, two, we can see that the MSE within is lower than the MSE between. The difference between these two is large in all tests except ML. By looking at this particular graph, we could say that ML performs the best for MSE.

FIGURE 5 Simulation graph four – Final MSE for Certain Parameters



When looking at the sigma within parameter, Figures 6 and 7, we can see that all tests seem to have the same results until the highest sigma within, being 0.5. ML has a similar bias when compared with other tests, but it has a higher BIAS between than the others. When looking at the MSE, the trend is similar. All tests have similar MSE's up to

the highest sigma within, and similar MSE between. Looking at the highest sigma within, 0.5, we can see that ML has the lowest MSE.

FIGURE 6 Simulation graph five – Final Bias for Certain Parameters

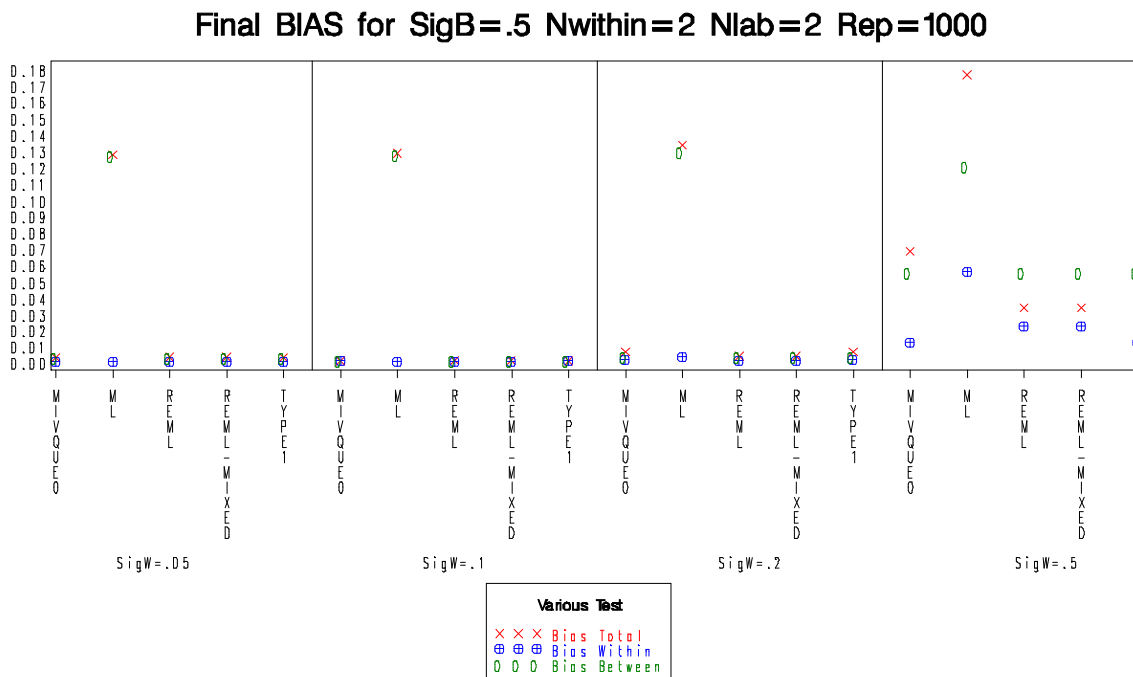
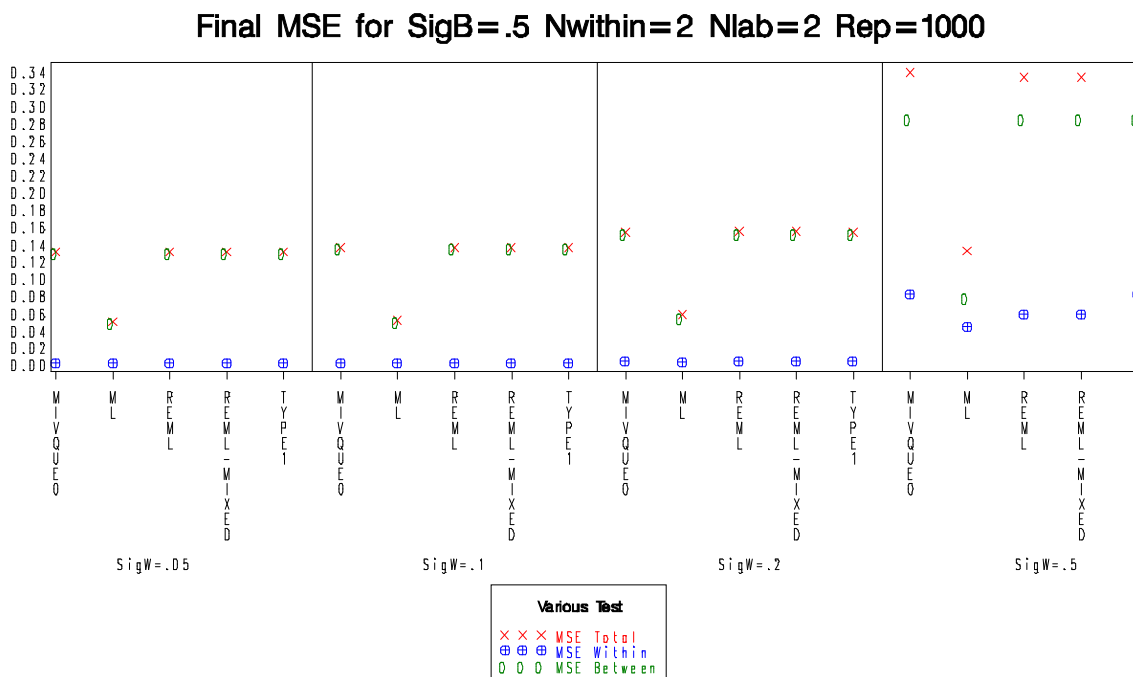


FIGURE 7 Simulation graph six – Final MSE for Certain Parameters



Looking at Figure 8, which depicts the trend across sigma between, we can see a varying pattern. All tests have different values for bias, but each test is similar to itself as sigma between increases. When looked at the highest sigma between, we can see that MIVQUE0 and TYPE1 both have the smallest bias values and ML has the highest bias values.

FIGURE 8 Simulation graph seven – Final Bias for Certain Parameters

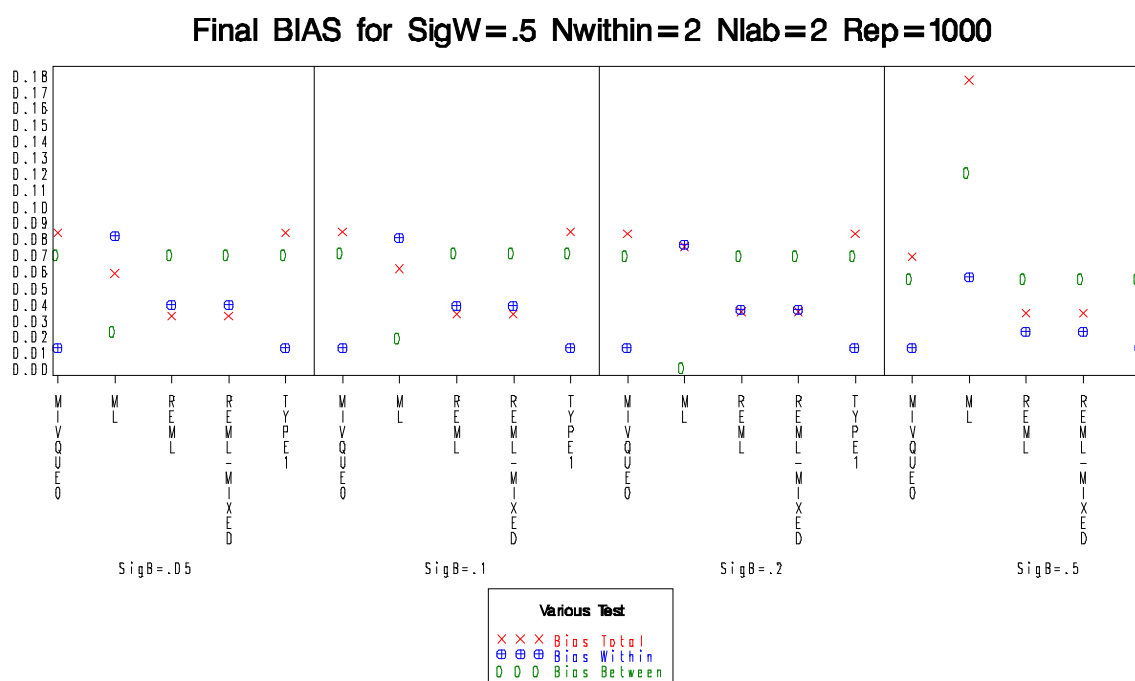
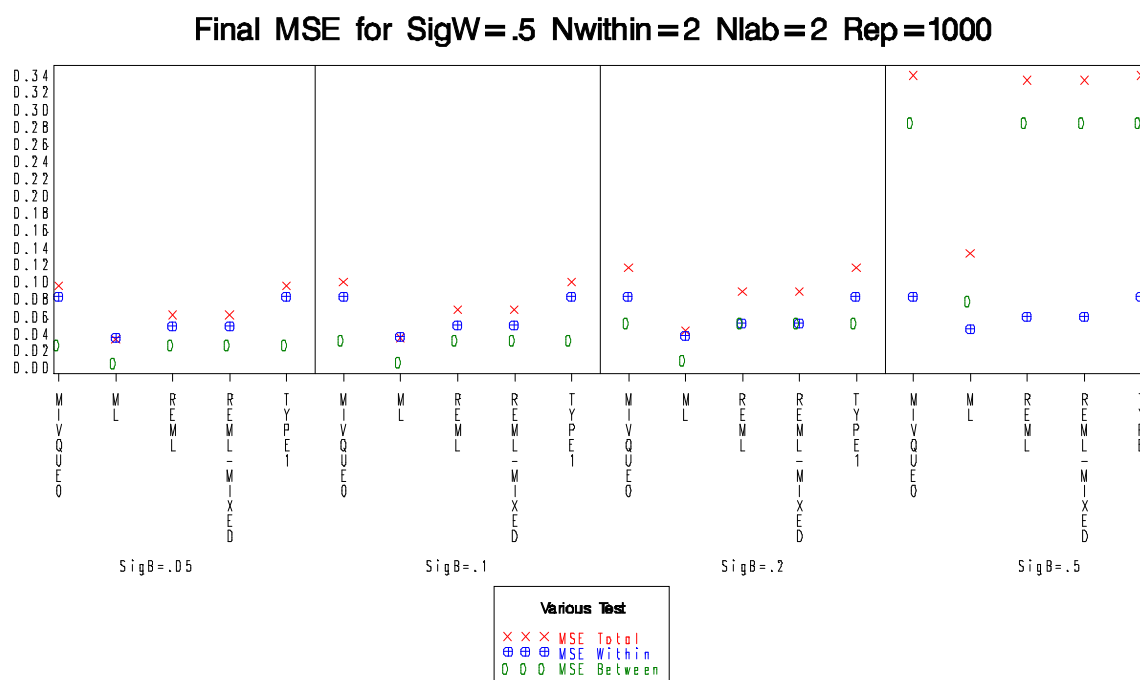


Figure 9 shows MSE trends for sigma between and it starts off similar to bias. All tests have different values, but when compared to itself, the values are similar. As sigma between increases, the bias between value seems to get closer to the bias within value. Looking at when sigma between is 0.2, we can see that these two values for the REML

procedures become the same. When looking at the highest parameter, sigma between being 0.5, we can see that ML has the lowest.

FIGURE 9 Simulation graph eight – Final MSE for Certain Parameters



Since the actual data will not look at bias and MSE, CV (RSD) values were also calculated for between and within values for each of the sigma's and the CV (RSD) % ranges from 7% to 81%, which covers most practical scenarios.

B. Part 2 – Data

In this case, the following CV/RSD values were found for each factor as well as the total.

The following table depicts these results by test type and shows the variance as well as the RSD %.

TABLE 3. TABLE OF VARIANCES FOR EACH TEST TYPE

	VARCOMP								MIXED	
	MIVQUE0		ML		TYPE 1		REML		REML	
	Variance	RSD %	Variance	RSD %	Variance	RSD %	Variance	RSD %	Variance	RSD %
Analyst	0.004375	6.6216283	0.0023205	4.8199618	0.0039263	6.2721667	0.0059361	7.7160811	0.0059361	7.7160793
Concentration	0.0000885	0.9405379	0.0031846	5.6476996	0.0111747	10.600627	0.0042038	6.4905025	0.0042037	6.4904494
Date	0.002404	4.9060554	0.0120427	11.007066	0.0073462	8.5867701	0.0121704	11.065603	0.0121704	11.065601
Error	0.0195677	14.057176	0.0131651	11.511807	0.0131243	11.49384	0.0131104	11.487718	0.0131105	11.487721
Lot	0	0	1.9624E-6	0.1400858	0	0	0	0	0	0
Total	0.0259319	16.208335	0.0307149	17.661112	0.0317053	17.948045	0.0354208	18.988308	0.0354207	18.988289

Discussion and Conclusions

A. Part 1- Simulation

By looking at the bias graph, the test that would produce the least amount of bias would be MIVQUE0 or TYPE1. However, when looking at the MSE graph, the test type that would produce the least MSE would be ML. It is important that both bias and MSE be considered. Since this is a simulated data set, the results will not be the same every-time. This knowledge should be taken into consideration when looking at the actual data. From the simulation, it can be concluded to use either the MIVQUE0 or TYPE1 test because ML seems to have a lot of variation, especially when the number of labs is low. The pharmaceutical industry does not have the luxury of having a large sample size – most of the time, the number of labs is very low. Since this is the case, it is best to go with a stable test.

B. Part 2- Data

By calculating the CV/RSD values, we are able to see which test performed the best. By individual factors, the test with the lowest variance differs. For analyst, the test

with the lowest variance is ML, however when looked at the concentration, the test with the lowest variance is then MIVQUE0. Each factor plays a part in the variability, therefore, the total variance was also calculated. By looking at the total, we can see that MIVQUE0 has the lowest value.

This finding can be supported by the finding in the simulation data because we found that this was one of the tests that had a low bias. In the pharmaceutical industry, if they have the luxury of having a high sample size, it would not matter which test they used – they would all get similar results. However, in this particular case, which is similar to most cases, MIVQUE0 is the test that should be used to calculate variance. It is important to use a consistent test – with this additional variability coming from the methods can be eliminated.

LIST OF REFERENCES

LIST OF REFERENCES

- Brennan, P. & Silman, A. (1992). Statistical methods for assessing observer variability in clinical measures. *BMJ*, *304*, 1491-1494.
- Cheser, D. (2008). Evaluating Assay Precision. *Clin Biochem Rev*, *29*, S23-S26.
- FDA (2001). Guidance for Industry – Bioanalytical Method Validation.
<http://www.fda.gov/downloads/Drugs/GuidanceComplianceRegulatoryInformation/Guidances/UCM070107.pdf>
- Findlay, J.W.A, Smith, W.C., Lee, J.W., Nordblom, G.D., Das, I., DeSilva, B.S., Khan, M.N., & Bowsher, R. R. (2000). Validation of immunoassays for bioanalysis: a pharmaceutical industry perspective. *Journal of Pharmaceutical and Biomedical Analysis*, *21*, 1249-1273.
- Karnes, H. T., Shiu. G., & Shah, V. P. (1991). Validation of Bioanalytical Methods. *Pharmaceutical Research*, *8*, 421-426.
- Littell, R. C., Henry, P. R. & Ammerman, C. B. (1998). Statistical analysis of repeated measures data using SAS procedures. *Journal of Animal Science*, *76*, 1216-1231.
- Menditto, A., Patriarca, M. & Magnusson, B. (2007) Understanding the meaning of accuracy, trueness and precision. *Accred Qual Assur*, *12*, 45-47.

Peters, F. T. & Maurer, H. H. (2002). Bioanalytical method validation and its implications for forensic and clinical toxicology – A review. *Accreditation and Quality Assurance*, 7, 441-449.

Rice, J. A. (2007). *Mathematical Statistics and Data Analysis*. 3rd edition. Duxbury Press.

SAS Institute Inc., SAS OnlineDoc® 9.1.3, Cary, NC: SAS Institute Inc., 2002-2005.

Yaffe, R. A. *Mixed Analysis of Variance Models with SPS*, New York University.

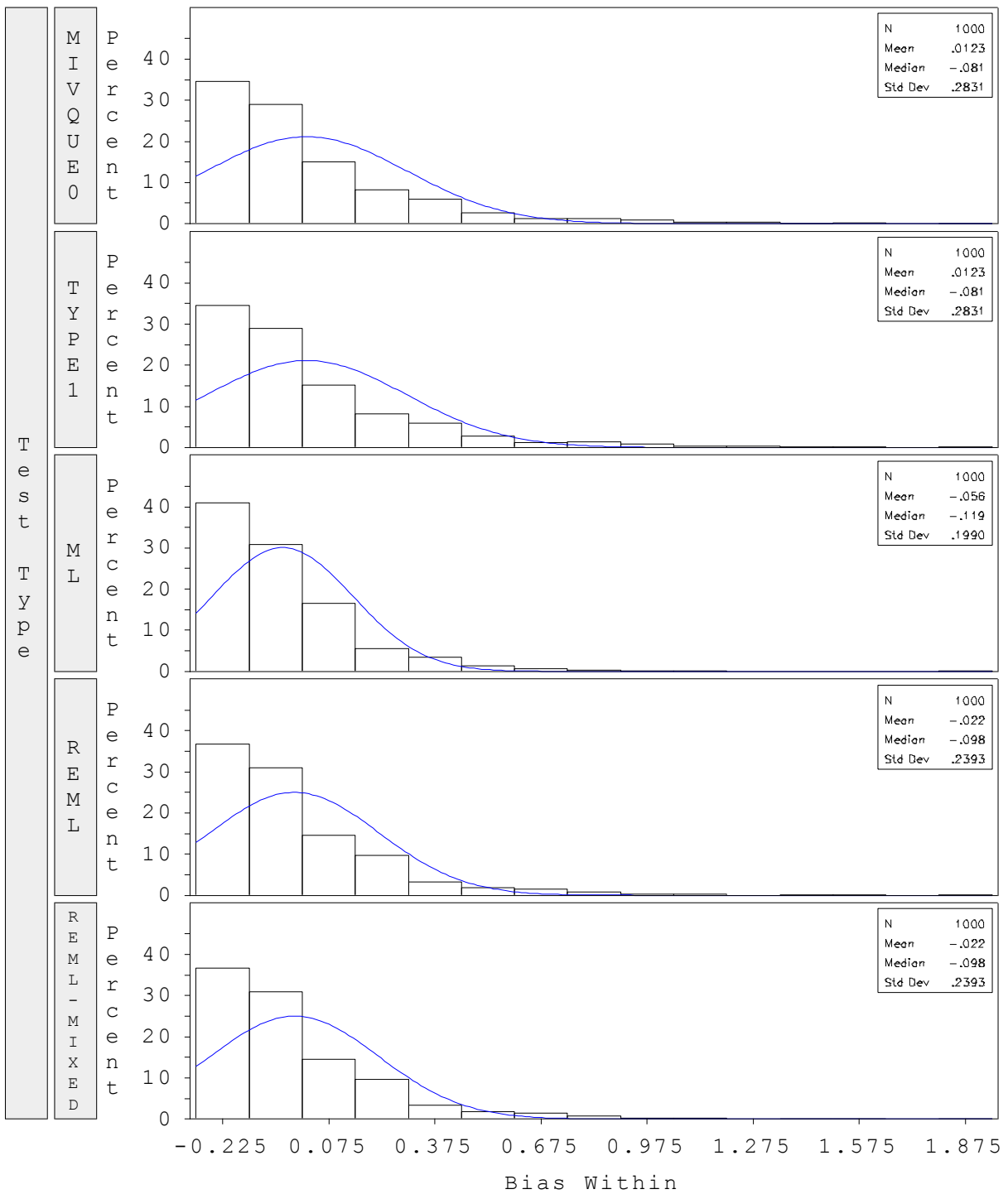
www.nyu.edu/its/socsci/Docs/SPSSMixed.ppt

APPENDICES

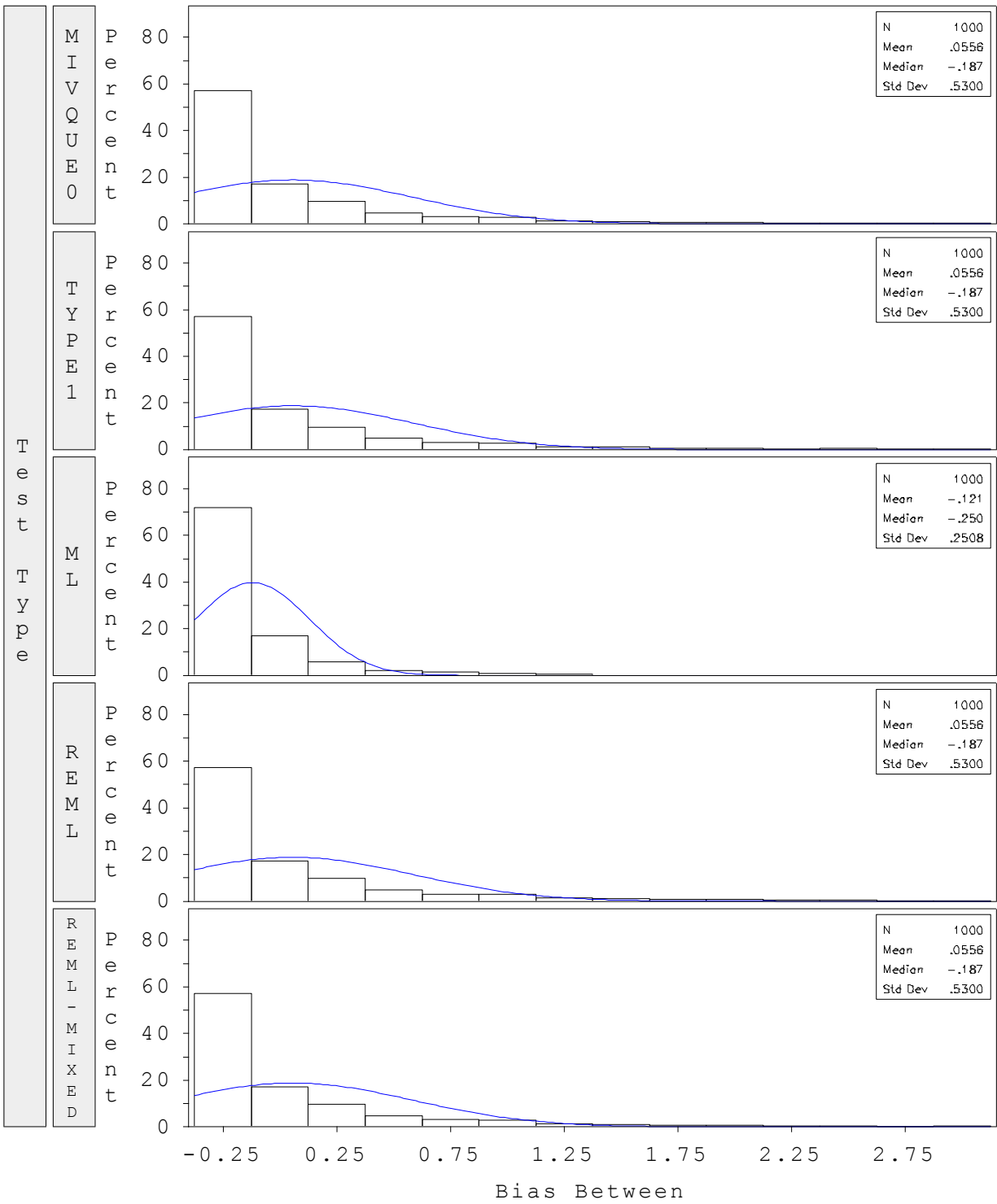
APPENDICES

Appendix A. Bias Simulation Distributions – for bias within, bias between and bias total

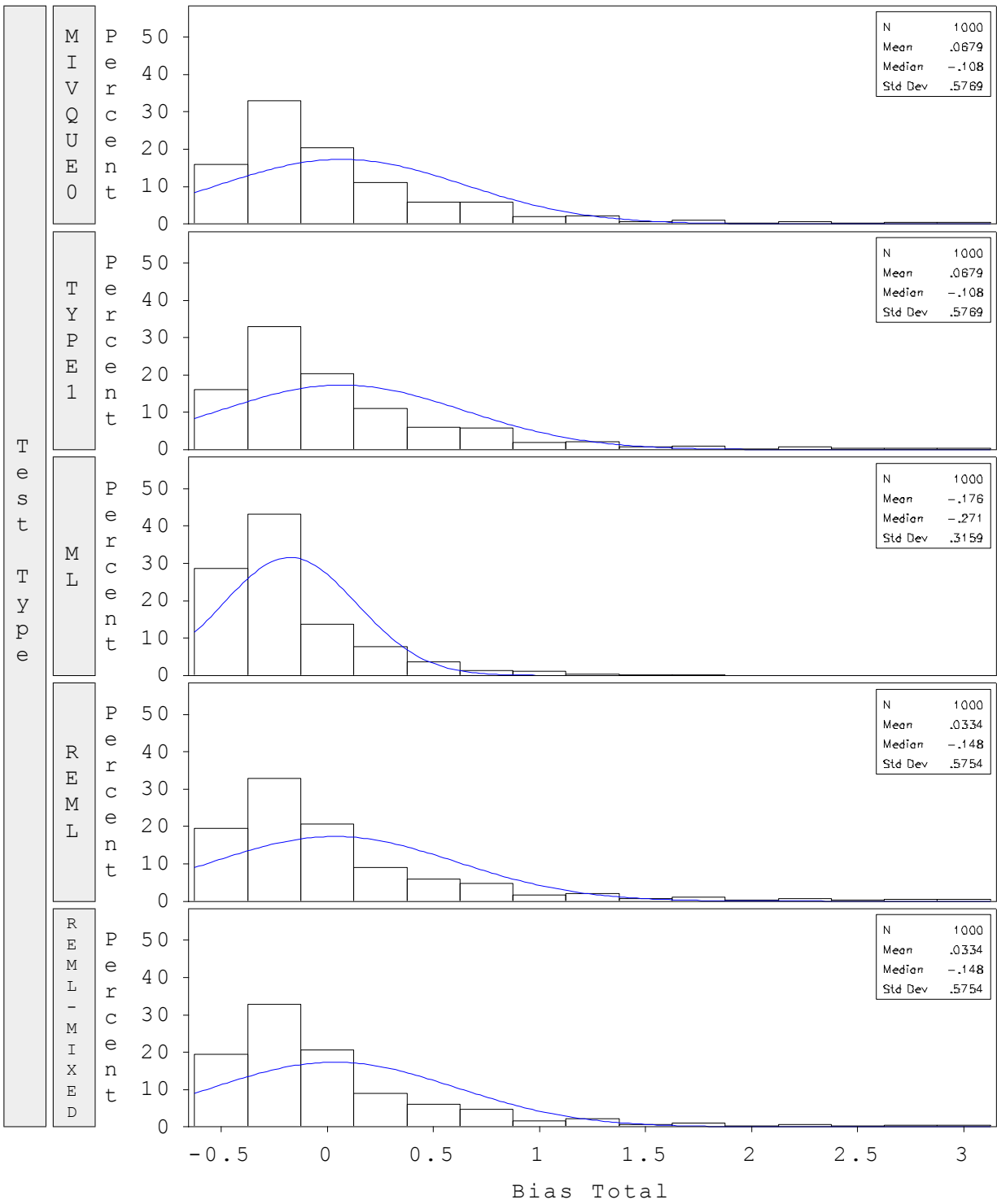
DISTRIBUTION OF BIAS for SigB=.5 SigW=.5 Nwithin=2 Nlab=2



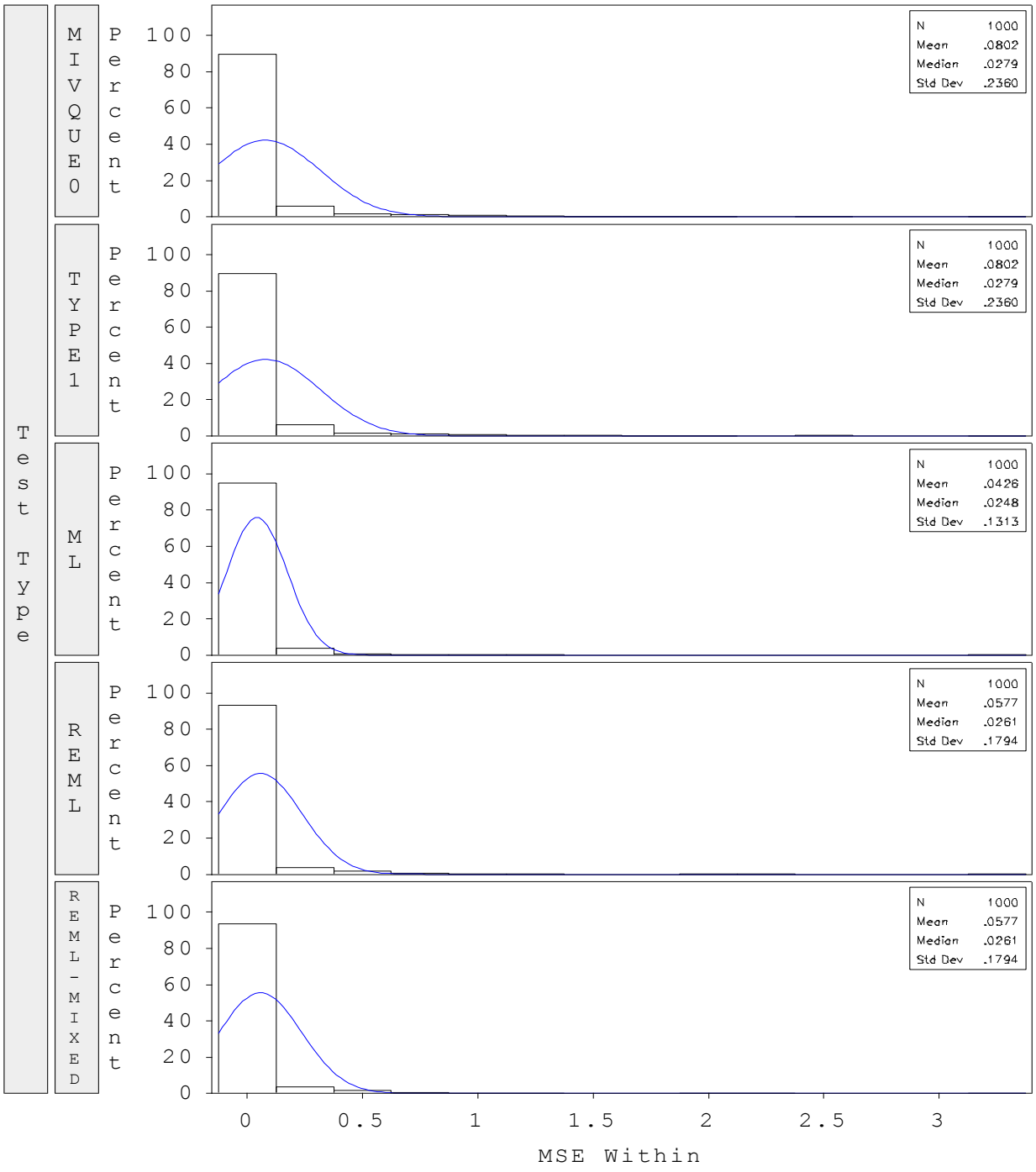
DISTRIBUTION OF BIAS for SigB=.5 SigW=.5 Nwithin=2 Nlab=2



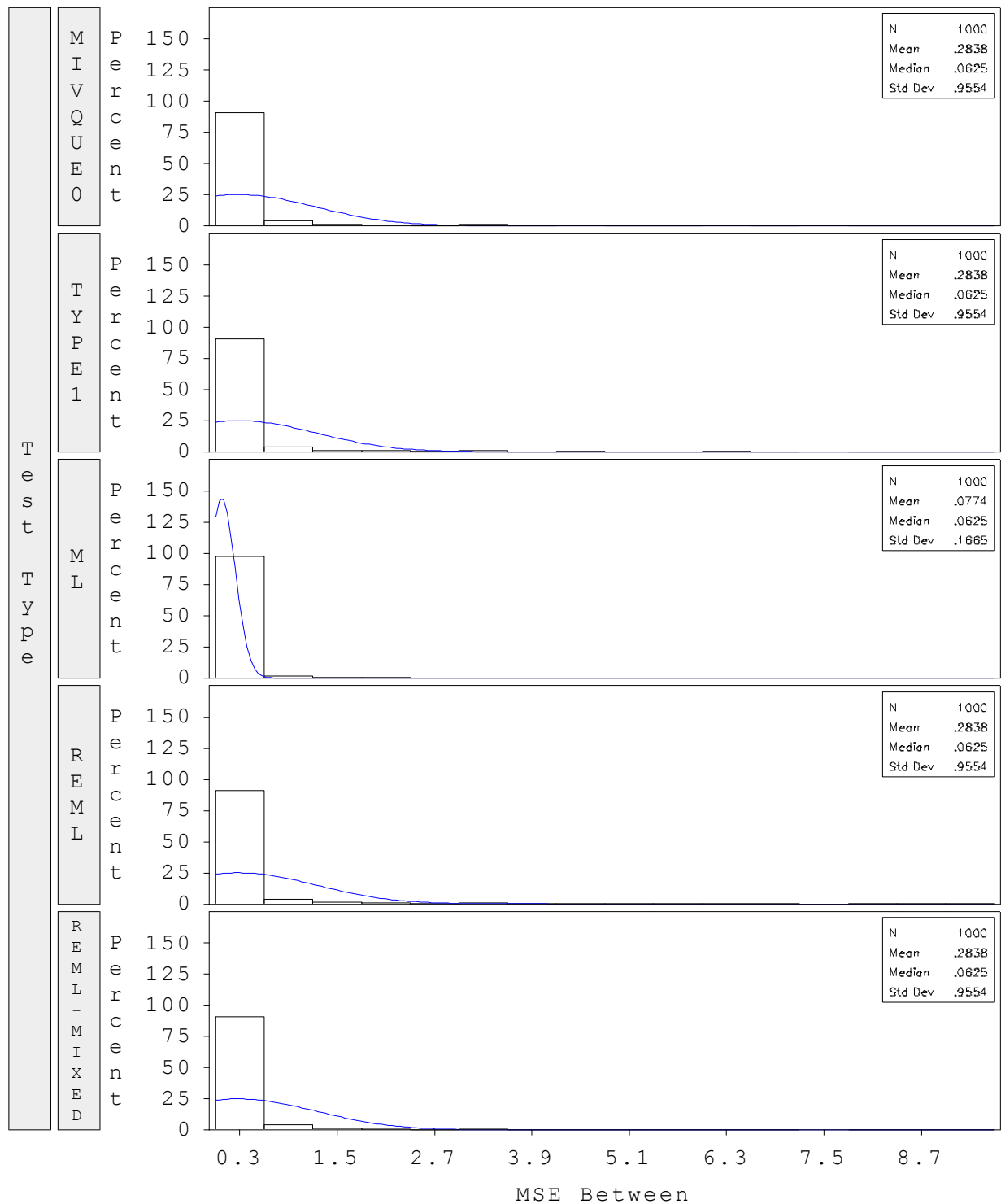
DISTRIBUTION OF BIAS for SigB=.5 SigW=.5 Nwithin=2 Nlab=2



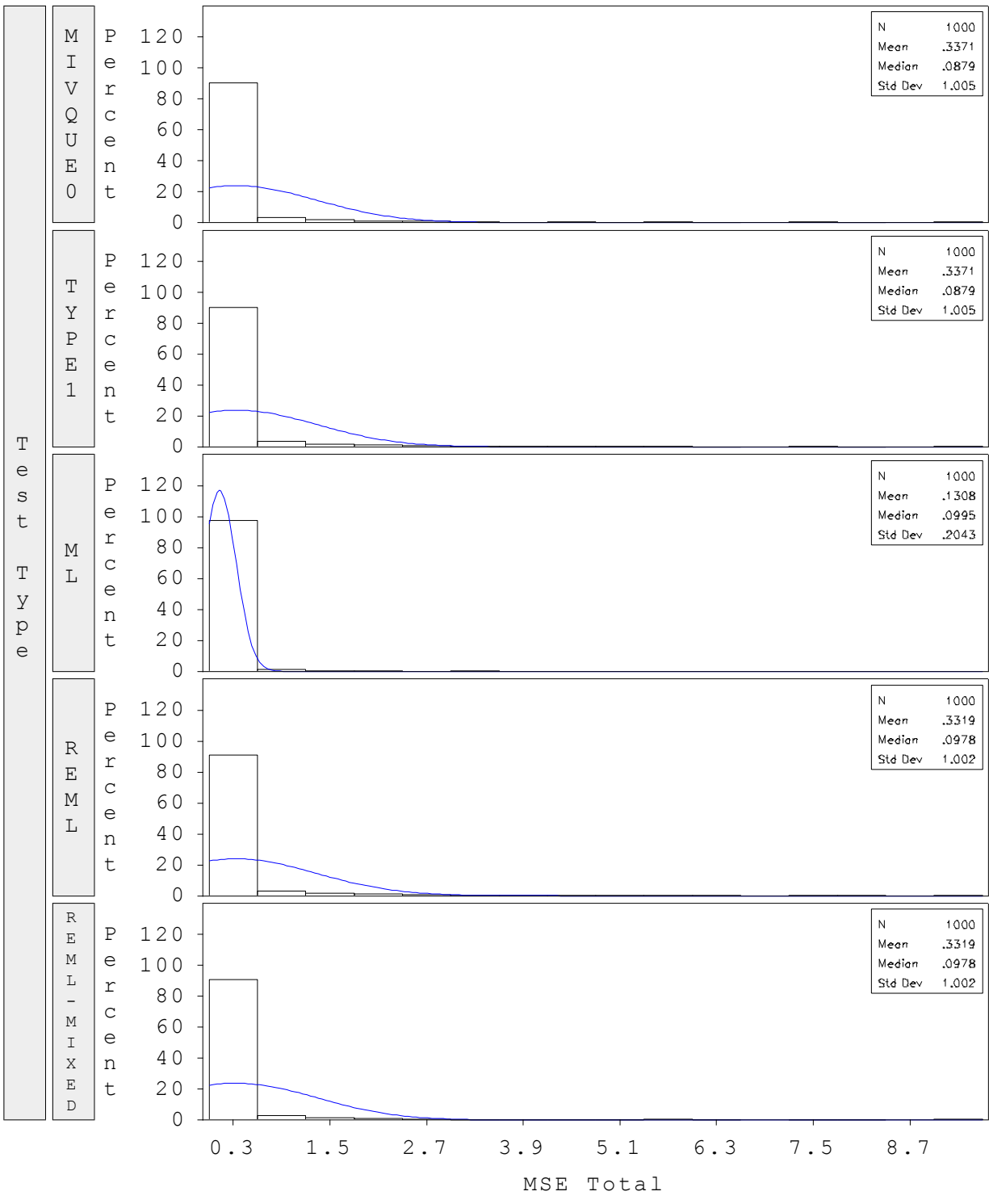
Appendix B. MSE Simulation Distributions –for MSE within, MSE between and MSE total DISTRIBUTION OF MSE for SigB=.5 SigW=.5 Nwithin=2 Nlab=2



DISTRIBUTION OF MSE for SigB=.5 SigW=.5 Nwithin=2 Nlab=2



DISTRIBUTION OF MSE for SigB=.5 SigW=.5 Nwithin=2 Nlab=2



Appendix C. SAS Code for Simulation

```

%let seed1= 45667;
%let seed2= 72352;
%let mu=1;
%let rep=1000;

%macro simulate (sigb=, sigw=, nwithin=, nlab=);
***SIMULATION***;
data simulate;
do rep=1 to &rep;
do lab = 1 to &nlab ;
ylab=&sigb*normal(&seed1*lab+100);
do i=1 to &nwithin;
y=&mu+ylab+&sigw*normal(&seed2*lab*i+100);
output;
end;
end;
end;
run ;

proc print data = simulate;
title "DATA SIMULATED" ;
run ;

***REML - MIXED*** ;
ods listing close ;
ods output covparms=remlmix;
proc mixed data=simulate noclprint noitprint method = reml ;
by rep;
class lab;
model y=;
random lab;
run ;

data remlmix ;
length varcomp $50 ;
set remlmix ;
if covparm = "Residual" then covparm = "Error" ;
varcomp = compress(left("Var("||covparm||")")) ;
run ;

proc transpose data = remlmix out=remlmixt (drop=_name_);
id varcomp ;
by rep ;
var estimate ;
run ;

proc print data = remlmixt ;
title "PROC MIXED REML - TRANSPOSED" ;
run ;

***VARCOMP - MIVQUE0*** ;
ods output estimates=mivque0;
proc varcomp data=simulate method = mivque0 ;

```

```

        by rep ;
        class lab ;
        model y=lab;
run ;

proc transpose data = mivque0 out=mivque0t (drop=_name_);
    id varcomp ;
    by rep ;
    var y ;
run ;

proc print data = mivque0t ;
    title "PROC VARCOMP - MIVQUE TRANSPOSED" ;
run ;

***VARCOMP - REML *** ;
ods output estimates=reml ;
proc varcomp data=simulate method = reml ;
    by rep ;
    class lab ;
    model y=lab ;
run ;

proc transpose data = reml out=remlt (drop=_name_);
    id varcomp ;
    by rep ;
    var estimate ;
run ;

proc print data = remlt ;
    title "PROC VARCOMP - REML TRANSPOSED" ;
run ;

***VARCOMP - TYPE 1 *** ;
ods output estimates=typel ;
proc varcomp data=simulate method = typel ;
    by rep ;
    class lab;
    model y=lab ;
run ;

proc transpose data = typel out=typelt (drop=_name_);
    id varcomp ;
    by rep ;
    var estimate ;
run ;

proc print data = typelt ;
    title "PROC VARCOMP - TYPE 1 TRANSPOSED" ;
run ;

***VARCOMP - ML *** ;
ods output estimates=ml ;
proc varcomp data=simulate method = ml ;
    by rep ;
    class lab ;

```

```

        model y=lab ;
run ;

proc transpose data = ml out=mlt (drop=_name_);
    id varcomp ;
    by rep ;
    var estimate ;
run ;

proc print data = mlt ;
    title "PROC VARCOMP - ML TRANSPOSED" ;
run ;

data combine (drop=i) ;
    length testtype $50 ;
    set mivque0t (in=inm)
        mlt      (in=inmi)
        remlt    (in=inre)
        typelt   (in=int)
        remlmixt (in=inmx) ;
    **CREATE LABEL VARIABLE FOR EACH TEST**;
    if inm      then testtype = "MIVQUE0" ;
    if inmi     then testtype = "ML" ;
    if inre     then testtype = "REML" ;
    if int      then testtype = "TYPE1" ;
    if inmx     then testtype = "REML - MIXED" ;

    label testtype="Test Type" ;

    array test {2} var_lab_ var_error_ ;
    do i = 1 to 2 ;
        if test {i} lt 0 then test{i} = 0 ;
    end ;

run ;
proc sort data = combine ;
    by testtype ;
run ;

ods listing ;
proc print data = combine ;
    title "COMBINATION OF DIFFERENT METHODS" ;
run ;

proc sql;
    create table sasuser.simulate as
        select testtype, rep, var_lab_, var_error_
        from combine;
quit ;

data biasmse;
    set sasuser.simulate (rename=(var_lab_=v_between var_error_=v_within));

    bias_b=v_between-&sigb**2;
    mse_b=bias_b**2;
    bias_w=v_within-&sigw**2;
    mse_w=bias_w**2;
    bias_t=bias_b+bias_w ;

```

```

    mse_t = bias_t**2 ;
run;

proc sort data=biasmse;
    by testtype;
run;

proc means data=biasmse noprint;
    by testtype;
    var v_between v_within bias_b bias_w mse_b mse_w bias_t mse_t;
    output out=final mean=vb_m vw_m bb_m bw_m mb_m mw_m bt_m mt_m
           std=vb_s vw_s bb_s bw_s mb_s mw_s bt_s mt_s;
    title 'summary results';
run;

data final (drop=i) ;
    set final ;
    array values {16} vb_m vw_m bb_m bw_m mb_m mw_m bt_m mt_m vb_s vw_s
    bb_s bw_s mb_s mw_s bt_s mt_s ;
    do i = 1 to 16 ;
        values{i} = abs(values{i}) ;
    end ;
    label vb_m = "Mean-Variance Between"
          vw_m = "Mean-Variance Within"
          bb_m = "Mean-Bias Between"
          bw_m = "Mean-Bias Within"
          mb_m = "Mean-MSE Between"
          mw_m = "Mean-MSE Within"
          bt_m = "Mean-Bias Total"
          mt_m = "Mean-MSE Total"
          vb_s = "STD-Variance Between"
          vw_s = "STD-Variance Within"
          bb_s = "STD-Bias Between"
          bw_s = "STD-Bias Within"
          mb_s = "STD-MSE Between"
          mw_s = "STD-MSE Within"
          bt_s = "STD-Bias Total"
          mt_s = "STD-MSE Total"
          ;
run ;

dm 'clear log'

%mend simulate ;

%macro dostuff ;
    %do i = 1 %to 4 ;
        %if &i = 1 %then %let sb = .5 ;
        %else %if &i = 2 %then %let sb = .2 ;
        %else %if &i = 3 %then %let sb = .1 ;
        %else %if &i = 4 %then %let sb = .05 ;

        %do j = 1 %to 4 ;
            %if &j = 1 %then %let sw = .1 ;
            %else %if &j = 2 %then %let sw = .05 ;
            %else %if &j = 3 %then %let sw = .2 ;
            %else %if &j = 4 %then %let sw = .5 ;
        %end ;
    %end ;
%macro dostuff ;

```

```

%do k = 1 %to 4 ;
    %if &k = 1 %then %let nw = 2 ;
    %else %if &k = 2 %then %let nw = 5 ;
    %else %if &k = 3 %then %let nw = 10 ;
    %else %if &k = 4 %then %let nw = 30 ;

    %do l = 1 %to 4 ;
        %if &l = 1 %then %let nl = 2 ;
        %else %if &l = 2 %then %let nl = 3 ;
        %else %if &l = 3 %then %let nl = 10 ;
        %else %if &l = 4 %then %let nl = 30 ;

        %if &sb = .05 %then %let sb1 = 005 ;
        %if &sb = .1 %then %let sb1 = 01 ;
        %if &sb = .2 %then %let sb1 = 02 ;
        %if &sb = .5 %then %let sb1 = 05 ;
        %if &sw = .05 %then %let sw1 = 005 ;
        %if &sw = .1 %then %let sw1 = 01 ;
        %if &sw = .2 %then %let sw1 = 02 ;
        %if &sw = .5 %then %let sw1 = 05 ;

        %simulate(sigb=&sb, sigw=&sw, nwithin=&nw, nlab=&nl) ;

proc sql;
    create table sasuser.graph&sb1&sw1&nw&nl as
        select testype, vb_m, vw_m, bb_m, bw_m, mb_m, mw_m, bt_m, mt_m
        from final;
quit ;

    %end ;
%end ;
%end ;
%mend ;
%do stuff ;

```

Appendix D. SAS code for bias/MSE graphs

```

options mlogic nodate ;
%macro graph
  (sb1=, sb2=, sb3=, sb4=, sw1=, sw2=, sw3=, sw4=, nw1=, nw2=, nw3=, nw4=, nl1=, nl2=, nl3=, n
  l4=,
          tit=, name=, label=);
data nlabgraph ;
  set sasuser.graph&sb1&sw1&nw1&nl1 (in=in1)
      sasuser.graph&sb2&sw2&nw2&nl2 (in=in2)
      sasuser.graph&sb3&sw3&nw3&nl3 (in=in3)
      sasuser.graph&sb4&sw4&nw4&nl4 (in=in4);
  **if obs ne . ;
  if testype = "MIVQUE0" then test = 1 ;
  else if testype = "ML" then test = 2 ;
  else if testype = "REML" then test = 3 ;
  else if testype = "REML - MIXED" then test = 4 ;
  else if testype = "TYPE1" then test = 5 ;

  if in1 then test = test ;
  if in2 then test = test + 5 ;
  if in3 then test = test + 10 ;
  if in4 then test = test + 15 ;

run ;

/*proc print data = nlabgraph ;
title &tit ;
run ;*/

ods rtf file="c:\users\benal\graph&name..rtf";
goptions reset = all ;
options orientation = landscape;
axis1 label=none minor = none major = none ;
axis3 label=(&label)
MINOR = NONE major = (number=20) value = ('MIVQUE0' 'ML' 'REML' 'REML-MIXED'
'TYPE1' 'MIVQUE0' 'ML' 'REML' 'REML-MIXED' 'TYPE1' 'MIVQUE0' 'ML' 'REML'
'REML-MIXED' 'TYPE1' 'MIVQUE0' 'ML' 'REML' 'REML-MIXED' 'TYPE1' );
axis2 label=none major = none minor = none ;

symbol1 value='x' height=1 l=1 c=red /*i=join*/;
symbol2 value='+' height=1 l=2 c=blue /*i=join*/;
symbol3 value='o' height=1 l=3 c=green /*i=join*/;

legend1 position = (bottom center outside) mode=reserve across=1 frame
label=(f=swiss h=1 c=black 'Various Test' position=top justify=center)
value=(t=1 c=red 'MSE Total' h=.75 t=2 c=blue 'MSE Within' h=.75 t=3 c=green
'MSE Between' h = .75);

legend2 position = (bottom center outside) mode=reserve across=1 frame
label=(f=swiss h=1 c=black 'Various Test' position=top justify=center)
value=(t=1 c=red 'Bias Total' h=.75 t=2 c=blue 'Bias Within' h=.75 t=3
c=green 'Bias Between' h = .75);

proc gplot data = nlabgraph ;
plot mt_m*test mw_m*test mb_m*test/ overlay href=(5.5 10.5 15.5)
chref=(black) vminor=1 haxis=axis3 vaxis = axis1 legend=legend1;

```

```
title "Final MSE for &tit" ;  
run ;  
quit ;  
proc gplot data = nlabgraph ;  
plot bt_m*test bw_m*test bb_m*test/ overlay href=(5.5 10.5 15.5)  
chref=(black) vminor=1 haxis=axis3 vaxis = axis2 legend=legend2;  
title "Final BIAS for &tit" ;  
run ;  
quit ;  
  
ods rtf close ;  
%mend graph ;
```

Appendix E. SAS code for Histograms

```

ods listing close ;
goptions hsize=7in vsize =9in ;
options orientation=portrait nodate;
**ods rtf file = "C:\Documents and Settings\patebini\My
Documents\histogram.rtf" ;
**ods rtf file = "C:\Users\Benal\Documents\CBMP\graphs\histogram&tit..rtf" ;
ods rtf file= "G:\CBMP\graphs\histograms\histogram&tit..rtf" ;
proc univariate data = biasmse noprint;
    class testtype ;
    var bias_t bias_w bias_b;
    histogram bias_t bias_w bias_b/  normal height=2 ctext=black
chref=cx888888 cframeside=cxeeeeeee nrows=5 ;
    inset n="N" (5.0)
        mean="Mean" (5.4)
        median="Median" (5.4)
        std="Std Dev" (5.4)/pos = ne height=1 ;
    title "DISTRIBUTION OF BIAS for SigB=&sigb SigW=&sigw Nwithin=&nwithin
Nlab=&nlab" ;
    label bias_t = "Bias Total"
        bias_w = "Bias Within"
        bias_b = "Bias Between" ;
    format testtype test. ;
run ;
ods rtf close ;

ods rtf file= "G:\CBMP1\graphs\histograms\histomse&tit..rtf" ;
proc univariate data = biasmse noprint;
    class testtype ;
    var mse_t mse_w mse_b;
    histogram mse_t mse_w mse_b/  normal height=2 ctext=black
chref=cx888888 cframeside=cxeeeeeee nrows=5 ;
    inset n="N" (5.0)
        mean="Mean" (5.4)
        median="Median" (5.4)
        std="Std Dev" (5.4)/pos = ne height=1 ;
    title "DISTRIBUTION OF MSE for SigB=&sigb SigW=&sigw Nwithin=&nwithin
Nlab=&nlab" ;
    label mse_t = "MSE Total"
        mse_w = "MSE Within"
        mse_b = "MSE Between" ;
    format testtype test. ;
run ;
ods rtf close ;
ods listing ;

```


Appendix F. SAS code for DATA

```

options nodate ;
%let printme=0 ;

data roman;
  length analyst $5. date $10. lot $12.;
  input date $ lot $ conc ric501 ric502 ric503 ric504 ric505 ric506;
  %*logric=log(ric50);
  analyst="Roman";
  cards;
23042009 277147 500 1.36 1.38 1.17 . . .
05092008 277147 500 0.79 0.72 0.78 . . .
15052008 0282650 100 0.77 1.05 1.04 . . .
03072008 0282650 100 0.99 0.97 1.17 1.01 0.94 1.00
10092008 0282650 500 1.08 1.14 0.76 . . .
16122008 0282650 500 1.11 0.87 0.82 . . .
03072008 MK2578PR0802 500 1.08 1.15 1.10 1.03 1.12 .
25092008 MK2578PR0802 500 0.97 0.95 0.92 . . .
13012009 MK2578PR0802 500 0.81 0.96 0.69 . . .
03122008 MK2578PR0803 500 1.14 1.01 0.83 . . .
29072008 WL00030113 40 0.85 0.93 0.99 0.93 1.00 .
29072008 WL00030114 500 0.97 1.08 0.95 1.03 1.09 0.98
16122008 WL00030114 500 0.94 0.90 1.03 . . .
;
run;

data meng;
  length analyst $5. date $10. lot $12.;
  input date $ lot $ conc ric501 ric502 ric503;
  %*logric=log(ric50);
  analyst="Meng";
  cards;
08122008 WL00030114 500 1.25 1.12 1.18
08122008 WL00030113 40 1.01 0.97 0.91
27012009 WL00030114 500 1.37 1.31 1.12
27012009 WL00030113 40 1.16 0.93 0.93
06022009 MK2578PR0803 500 1.17 1.30 1.11
06022009 MK2578PR0803 500 0.8 1.25 1.07
04032009 MK2578PR0803 500 0.94 1.03 0.89
04032009 MK2578PR0803 500 1.12 1.24 1.05
;
run;

data ric50 (drop=ric501 ric502 ric503 ric504 ric505 ric506);
  set roman meng;

  array res{6} ric501 ric502 ric503 ric504 ric505 ric506;

  do rep=1 to 6;
    ric50=res{rep};
    logric50=log(ric50);
    if ric50=. then delete;
  output;
  end;
run;

```

```

proc sort data = ric50 ;
    by rep ;
run ;

***REML - MIXED*** ;
ods listing close ;
ods output covparms=remlmix;
proc mixed data=ric50 noclprint noitprint method = reml ;
    class analyst date conc lot ;
    model logric50=;
    random analyst date conc lot ;
run ;

data remlmix ;
    length varcomp $50 ;
    set remlmix ;
    if covparm = "Residual" then covparm = "Error" ;
    varcomp = compress(left("Var("||covparm||")")) ;
run ;

proc transpose data = remlmix out=remlmixt (drop=_name_);
    id varcomp ;
    var estimate ;
run ;

proc print data = remlmixt (where=(&printme));
    title "PROC MIXED REML - TRANSPOSED" ;
run ;

***VARCOMP - MIVQUE0*** ;

proc varcomp data=ric50 method = mivque0 ;
    ods output estimates=mivque0;
    class analyst date conc lot ;
    model logric50=analyst date conc lot ;
run ;

proc transpose data = mivque0 out=mivque0t (drop=_name_);
    id varcomp ;
    var logric50 ;
run ;

proc print data = mivque0t (where=(&printme)) ;
    title "PROC VARCOMP - MIVQUE TRANSPOSED" ;
run ;

***VARCOMP - REML *** ;

proc varcomp data=ric50 method = reml ;
    ods output estimates=reml ;
    class analyst date conc lot ;
    model logric50=analyst date conc lot ;
run ;

proc transpose data = reml out=remlt (drop=_name_);
    id varcomp ;

```

```

        var estimate ;
run ;

proc print data = remlt (where=(&printme));
    title "PROC VARCOMP - REML TRANSPOSED" ;
run ;

***VARCOMP - TYPE 1 *** ;
proc varcomp data=ric50 method = typel ;
    ods output estimates=typel ;
    class analyst date conc lot ;
    model logric50=analyst date conc lot ;
run ;

proc transpose data = typel out=typelt (drop=_name_);
    id varcomp ;
    var estimate ;
run ;

proc print data = typelt ;
    title "PROC VARCOMP - TYPE 1 TRANSPOSED" ;
run ;

***VARCOMP - ML *** ;
proc varcomp data=ric50 method = ml ;
    ods output estimates=ml ;
    class analyst date conc lot ;
    model logric50=analyst date conc lot ;
run ;

proc transpose data = ml out=mlt (drop=_name_);
    id varcomp ;
    var estimate ;
run ;

proc print data = mlt (where=(&printme));
    title "PROC VARCOMP - ML TRANSPOSED" ;
run ;

data combine (drop=i) ;
    length testtype $50 ;
    set mivque0t (in=inm)
        mlt      (in=inmi)
        remlt   (in=inre)
        typelt  (in=int)
        remlmixt (in=inmx) ;
    **CREATE LABEL VARIABLE FOR EACH TEST**;
    if inm      then
        do ;
            testtype = "MIVQUE0" ;
            var_total = var_lot_ + var_error_ + var_analyst_ +
var_date_ + var_conc_ ;
            end ;
        if inmi then
            do ;
                testtype = "ML" ;

```

```

                var_total = var_lot_ + var_error_ + var_analyst_ +
var_date_ + var_conc_ ;
                end ;
                if inre then
                    do ;
                        testtype = "REML" ;
                        var_total = var_lot_ + var_error_ + var_analyst_ +
var_date_ + var_conc_ ;
                        end ;
                    if int then
                        do ;
                            testtype = "TYPE1" ;
                            var_total = var_lot_ + var_error_ + var_analyst_ +
var_date_ + var_conc_ ;
                            end ;
                    if inmx then
                        do ;
                            testtype = "REML - MIXED" ;
                            var_total = var_lot_ + var_error_ + var_analyst_ +
var_date_ + var_conc_ ;
                            end ;

                array test {6} var_lot_ var_error_ var_analyst_ var_date_ var_conc_
var_total;
                array rsd {6} lotrsd errsd analrsd datersd concrsd totrsd ;
                do i = 1 to 6 ;
                    if test {i} lt 0 then test{i} = 0 ;
                    rsd{i} = sqrt(exp(test{i})-1) * 100 ;
                end ;

                label testtype                = 'Test Type'
                    var_total                 = 'Total'
                    var_lot_                  = 'Lot'
                    var_error_                 = 'Error'
                    var_analyst_              = 'Analyst'
                    var_date_                 = 'Date'
                    var_conc_                 = 'Concentration'
                    lotrsd                    = 'Lot'
                    errsd                     = 'Error'
                    analrsd                   = 'Analyst'
                    datersd                   = 'Date'
                    concrsd                   = 'Concentration'
                    totrsd                    = 'Total'

                ;

run ;
proc sort data = combine ;
    by testtype ;
run ;
proc transpose data = combine (keep=testtype lotrsd errsd analrsd datersd
concrsd totrsd) out=combinet prefix=rsd;
    id testtype ;
run ;

```

```

proc sort data = combinet ;
    by _label_ ;
run ;

proc transpose data = combine (drop=lotrsd errsd analrsd datersd concrsd
totrsd) out=combinet2 ;
    id testtype ;
run ;

proc sort data = combinet2 ;
    by _label_ ;
run ;

data final (rename=_label_=testtype);
    merge combinet
        combinet2 ;
    by _label_ ;
run ;

ods listing ;
proc print data = final ;
    title "COMBINATION OF DIFFERENT METHODS" ;
run ;

options nodate nonumber orientation=landscape ;
ods listing close ;
ods escapechar='^' ;
**ods rtf style=TStyleRTF file="C:\Users\Benal\table.rtf" ;
ods rtf style=TstyleRTF file="C:\Documents and Settings\patebini\My
Documents\table2.rtf" ;
**----- REPORT TITLES -----** ;
title1 justify=center "TABLE OF VALUES FOR EACH TEST TYPE" ;

**----- REPORT DEFINITION -----** ;

proc report data=final missing nowindows center split='|' ;
    column testtype ("VARCOMP" ("MIVQUE0" MIVQUE0 rsdmivque0) ("ML" ml rsdml)
("TYPE 1" typel rsdtypel) ("REML" reml rsdreml)
        ("MIXED" ("REML" reml___mixed rsdreml___mixed));

    define testtype          / display ' '          style = [cellwidth =
.9in] ;
    define MIVQUE0/ display 'Variance'          style = [cellwidth = .8in] ;
    define rsdmivque0      / display 'RSD %'          style = [cellwidth = .8in] ;
    define ml              / display 'Variance'          style =
[cellwidth = .8in] ;
    define rsdml          / display 'RSD %'          style =
[cellwidth = .8in] ;
    define typel          / display 'Variance'          style = [cellwidth =
.8in] ;
    define rsdtypel      / display 'RSD %'          style = [cellwidth = .8in] ;
    define reml          / display 'Variance'          style =
[cellwidth = .8in] ;
    define rsdreml      / display 'RSD %'          style = [cellwidth =
.8in] ;
    define reml___mixed / display 'Variance'          style = [cellwidth = .8in] ;
    define rsdreml___mixed / display 'RSD %'          style = [cellwidth = .8in] ;

```

```
run ;  
**----- CLOSE RTF AND RESET TITLES/FOOTNOTES -----**;  
ods rtf close ;  
ods listing ;
```

