An Evaluation of Fit Statistics in the Identification of Spurious Classes in Finite Mixture Models

A Dissertation

Presented to

The Faculty of the Curry School of Education

University of Virginia

In Partial Fulfillment

of the Requirements for the Degree

Doctor of Philosophy

by

Michael Fox Hull, B.A,

December 2016

Copyright page

Abstract

J. Patrick Meyer

Finite mixture modeling is a popular tool for model based clustering. Research has shown that some data conditions or model misspecification can lead to the identification of spurious classes. However, the majority of research has focused on identifying the correct number of classes when the true number of classes is two or greater rather than the most basic hypothesis of a true one class distribution. The purpose of this study is to more fully explore the extent to which finite mixture models identify spurious classes when the true number of classes is one.

Data were simulated to form single-component normal and nonnormal distributions. Mixture models with one to four components were fit and log likelihood based, classification based, and likelihood ratio based fit statistics were employed to identify the best fitting model. The eleven fit statistics evaluated 72 analysis by data conditions with 250 replications when using multivariate normal and multivariate skew normal component distributions and 125 replications when using the more computationally intensive multivariate skew t component distributions.

The results showed that type of fit statistic, degree of data nonnormality, and type of component distribution accounted for the most variance in identifying the correct model. The ICL-BIC outperformed all other fit statistics and as data nonnormality increased, so did the identification of spurious classes. However, allowing the shape of component distributions to vary reduced spurious class identification and, when paired with the best performing fit statistics, eliminated the identification of spurious classes to within a reasonable statistical probability.

This study did not examine the degree of inaccuracy in identifying the correct model – i.e. examining which model was preferred rather than identification of the correct model. Additionally, this study did not examine conditions where the correct model had more than one class and nonnormal components were used to fit the models. Further, this study made no attempt to evaluate other statistical considerations in the identification of the correct model such as the separation of class means and the proportion or number of cases in the classes.

Key words: finite mixture model, spurious classes, fit statistics, nonnormal component distribution, simulation

Department of Leadership, Foundations & Policy
Curry School of Education
University of Virginia
Charlottesville, Virginia

APPROVAL OF THE DISSERTATION

This dissertation, ("An Evaluation of Fit Statistics in the Identification of Spurious Classes in Finite Mixture Models"), has been approved by the Graduate Faculty of the Curry School of Education in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

____
Name of Chair (J. Patrick Meyer)

____
Committee Member Name (Timothy Konold)

____
Committee Member Name (Ji Hoon Ryoo)

____
Committee Member Name (Stanley Trent)

______________________Date

Dedication

Acknowledgement

This research was partially funded by a Curry Doctoral Student Dissertation IDEAs (Innovative, Developmental, Exploratory Awards) Grant sponsored by the Curry School Research and Development Fund.

TABLE OF CONTENTS

LIST OF TABLES

Chapter 1 Introduction

Finite mixture modeling (FMM) is a popular tool for model based clustering. It provides a number of advantages over traditional cluster analysis methods like hierarchical and K-means clustering. For one, FMM draws upon statistical theory for postulating a population model and deriving sample estimates of model parameters (Vermunt & Magidson, 2002). It is also insensitive to the scaling of observed variables, and it allows cases to be probabilistically assigned to clusters. Another advantage is that FMM provides for the development of nonarbitrary clustering criteria. These advantages make FMM an attractive alternative to traditional clustering methods. However, FMM has problems of its own that may hinder its efficacy in clustering observations.

Bauer and Curran (2003a) write that a nonnormally distributed composite distribution is a necessary and perhaps even sufficient condition for the identification of nontrivial component distributions (i.e. clusters) giving rise to the data. The problem is that the converse is not true. Subpopulations and clusters represented by multiple component distributions are not a necessary condition for data to be nonnormally distributed. Data may be nonnormally distributed for a variety of valid reasons that have nothing to do with the presence of clusters in the data. For example, response times are known to be positively skewed and test scores are commonly platykurtic and negatively skewed (Keats & Lord, 1962). The result is that FMM will identify spurious classes when applied to data generated from a single component nonnormal distribution (Bauer & Curran, 2003a). Consequently, researchers may arrive at erroneous conclusions about a substantive area of study. The problem of identifying spurious classes is not limited to FMMs under the conditions of classical test theory.

Alexeev, Templin, and Cohen (2011) showed that an item response theory mixture Rasch model will identify spurious classes when it is fit to data generated as a three parameter logistic (3PL) or a two parameter logistic (2PL) model. Further, they showed that the likelihood of identifying spurious classes increases with the number of test items, the number of subjects, and the deviation of the slope parameter from the slope parameter assumed by the Rasch model—i.e. one.

Bauer and Curran (2003a) demonstrated the influence of nonnormal distributions on spurious classes by testing one and two component mixture models. Others also limited their work to one and two component distributions (Muthen, 2003; Rindskopf, 2003). Consequently, the number of spurious classes identified in nonnormal distributions was not fully appreciated. Bauer and Curran (2003b) later suggested that the problem of spurious classes may be even more serious than their initial analysis indicated where researchers could identify far more than two classes. They demonstrated that a four-class normal mixture model fit the single-component nonnormal data even better than the two-class solution in their original paper.

Mixture Modeling Applications

FMM is a popular choice for cluster analysis. In education, it has been applied to test motivation theory (Pastor, Barron, Miller, & Davis, 2007), academic performance standards (Brown, 2007; Alexeev, Templin, & Cohen, 2011; Kelava & Brandt, 2014), deficiency in mathematics understanding (Chan, Leu, & Chen, 2007), performance decline (Jin & Wang, 2014), identifying groups based on guessing behavior (Leong, Mahdi, & Ling, 2013) and test speededness (Bolt, Cohen, & Wollack, 2002; Meyer, 2010). Additionally, FMM has been applied to differential item functioning (Cohen & Bolt, 2005; Frederickx, Tuerlinckx, De Boeck, & Magis, 2010; Cho & Cohen, 2010; Finch & Finch, 2013; Lee & Beretvas, 2014), as a follow up to identify causes of differential item functioning (Cohen & Bolt, 2005; Cho, Lee, & Kingston, 2012), and latent transition analysis following an intervention (Cho, Cohen, Bottge, 2013). In the social sciences, it has been applied to aggression and harassment (Giang & Graham, 2008), quality of life (Punzo, 2014), and profiling child maltreatment perpetrators (Yampolskaya, Greenbaum, & Berson, 2009). In addition, Hoijtink and Notenboom (2004) used mixture modeling to evaluate the stage model theory of the development of spelling ability in children. Büsch, Hagemann, and Bender (2010) evaluated a questionnaire assessing handedness and Freund, Tietjens, and Strauss (2013) examined the effects of using different Likert scale ranges for an assessment of physical self-concept. McCrae, Chapman, and Christ (2006) searched for latent groups among children who had been sexually abused and Gebhardt, Rose, and Mitte (2013) used mixture models to evaluate the efficacy a median split procedure in correctly classifying persons who repress negative affect experiences. Other research used latent class cluster analysis to create profiles of eating disorders (Wade, Crosby, & Martin, 2006), develop four appraisal/coping styles used in stressful situations (Larsson, Kempe, & Starrin, 1988), describe the psychosocial adjustment of adolescents (Ding, 2006), examine latent classes from personality data (Maij-de Meij, Kelderman, & van der Flier, 2008), heterogeneity of people who took the Chinese version of the Beck Depression Inventory—II (Wu & Huang, 2010), and assessment of lifetime prevalence of mental disorders (Almansa, Vermunt, Forero, & Alonso, 2014). Mixture modeling analysis has also been applied to data from fields such as business (Tuma & Decker, 2013), medicine (Bozdogan, 1994; Sawatzky, Ratner, Kopec, & Zumbo, 2012; Howe & Bozdogan, 2013; Hesser & Andersson, 2014; Lee & McLachlan, 2014; Muthén & Asparouhov, 2014), ornithology (McCrea, Morgan, & Cole, 2012), production / mechanical engineering (Yu, 2012), and transportation safety (Kim & Mahmassani, 2014; Park, Lord, & Lee, 2014). These applications of mixture modeling not only inform substantive areas of research, but also have implications for cluster members. Researchers must have confidence that clusters are accurately identified.

In the context of cluster analysis, the purpose of finite mixture modeling is to estimate the model parameters and compute the probability that each observation belongs to each of the classes. The model is fit several times using a different number of classes and fit statistics allow the researcher to identify the model with the correct number of classes. Fit statistics can be broadly categorized into three types: information criteria based, classification based (entropy and entropy penalty), and likelihood ratio test based.

Evaluating Competing Models

Information criteria are a function of the log-likelihood and some other penalty imposed for model complexity. These fit statistics differ on the way the penalty is computed, but the penalty is necessary given that statistical models with more parameters fit data better than similar models with fewer parameters. In the mixture modeling context, this means that models with more component distributions will fit better than those with fewer components. The penalty helps prevent over identifying the number of classes. The information criteria fit statistics employed in this study are: Akaike's Information Criterion (AIC; Akaike, 1973), the Bayesian Information Criterion (BIC; Schwartz, 1978), the consistent AIC (CAIC; Bozdogan, 1987), and versions of the BIC and CAIC that are adjusted for sample size (Sclove, 1987). Although information criteria are the type of fit statistics most frequently used, classification type fit statistics are also considered in this study.

Classification fit statistics use entropy, a measure of classification accuracy, as penalty when determining the fit of competing models. Essentially, the posterior probabilities of class membership are estimated and clearer assignment of cases to latent classes results in a better fit. When the separation among multivariate means of the component distributions for the classes is large, entropy, and the entropy penalty, should be smaller than when the multivariate means of the component distributions are closer together and assignment of cases to classes becomes less clear. This also implies that as the number of component distributions fitted to the data increase, the less clear the assignment of cases to classes becomes and, therefore, the larger entropy and entropy penalty becomes. Thus, the entropy and entropy penalty based fit statistics act to encourage the selection of a more parsimonious model. The entropy and entropy penalty based fit statistics employed in this study are: the Normalized Entropy Criterion (NEC; Celeux & Soromenho, 1996), the Classification Likelihood Criterion (CLC; Biernacki & Govaert, 1997), and the Integrated Classification Likelihood Criterion-BIC (ICL-BIC;

Biernacki, Celeux, & Govaert, 1998). The most recently developed type of fit statistic in this study are the likelihood ratio based fit statistics.

The likelihood ratio test (LRT) is a nested model test. The LRT statistic is a widely used method for testing nested models that is chi-square distributed with degrees of freedom equal to the difference in the number of estimated parameters for the competing models. However, in the mixture modeling context, the LRT cannot be used to test models where the number of classes is nested. The reason for this is that the parameters of the k class model must be set to zero to specify the $k – 1$ model. That is to say the probability of being in the kth class must be set to zero. Since probabilities range from zero to one, the parameter is being set at the border of admissible space. Consequently, the LRT statistic is no longer asymptotically chi-square distributed (McLachlan & Peel, 2000). To overcome this distributional dilemma, researchers have proposed alternative methods for employing the LRT in evaluating the fit of models with nested component structures and two of these fit statistics are employed in this study: the bootstrap likelihood ratio test (BLRT; McLachlan, 1987) and the Lo, Mendell, and Rubin likelihood ratio test (LMR-LRT; 2001).

The different formulations and methods of penalizing models that estimate a larger number of parameters can often lead to markedly different decisions based upon which fit statistic is used. To date there is no consensus on the “best” fit statistic and comparing the ability of the various fit statistics in identifying the correct number of classes provides important information for researchers seeking to uncover the true characteristics of the membership in their sample.

Efficacy of Fit Statistics

Identifying the number of clusters in the data depends on the researcher's choice of fit statistic. Two standards are of interest when evaluating the efficacy of a fit statistic. One standard is like statistical power; how often the correct number of clusters is identified when there really are clusters in the data. The second standard is like Type I error; how often multiple clusters are identified in the data when no clusters actually exist. A large portion of research on the efficacy of fit statistics in mixture modeling has focused on the first criteria while only a few have considered the second.

Tofighi and Enders (2007) in a growth mixture modeling format examined a number of fit statistics and found that across conditions, the SABIC and the LMR-LRT outperformed all other tests. Henson, Reise, and Kim (2007) in a structural equation mixture modeling format examined a similar set of fit statistics and found that across conditions, the ICL-BIC and the CLC outperformed all other tests in selecting the true two class model. However, Nylund, Asparouhov, and Muthén (2007) found that the BLRT outperformed the other likelihood tests and the BIC outperformed the information criteria tests. McLachlan and Ng (2000) reported the results of three simulation studies and found that the ICL-BIC, the CLC, and the matrix based Laplace-Empirical Criterion (LEC; Roberts, Husmeier, Rezek, & Penny, 1998) identified the true model and, of these three, the ICL-BIC is the easiest to implement. While these studies examine the effectiveness of a variety of fit statistics in their role of determining the correct number of classes in a mixture distribution, they do not examine the most basic condition of fitting more than one normal distribution to a true single-class distribution.

Recently, Peugh and Fan (2013) compared the performance of fit statistics in a true one class condition. The results of their simulation, with 48 conditions and 200 replications, found that the CLC, ICL – BIC, sample size adjusted ICL – BIC, CAIC, sample size adjusted CAIC, BIC, and D – BIC (Draper, 1995) all performed well. Oddly, Peugh and Fan (2013) did not use the NEC, aLMR – LRT, and the BLRT for the one-class true model citing the undefined nature of the NEC and the boundary limitations associated with the aLMR – LRT and the BLRT. However, each of these statistics has been shown to be useful in identifying a correct one class model (Biernacki, Celeux, & Govaert, 1999; Lo, Mendell, & Rubin, 2001; McLachlan, 1987). Additionally, Peugh and Fan (2013) did not examine what effect different levels of skew and kurtosis of the overall distribution would have on the accuracy of the fit statistics. Li, Cohen, Kim, and Cho (2009) investigated the efficacy of fit statistics with a true one class condition using Bayesian estimation of IRT models. They found that the BIC correctly identified the true number of classes on all replications. While their findings support the use of BIC for determining the correct number of classes, the results are based on only 30 replications and it is unclear if the results could be replicated with maximum likelihood estimation of the IRT models. Bauer and Curran (2003a) examined the performance of fit statistics using a true one class distribution compared with a two class alternative mixture model under various conditions of nonnormality. Their findings showed that the ICL-BIC outperformed the other fit statistics. However, correct model selection became worse as sample size increased and as nonnormality became more extreme.

Given the very recent implementation of fitting multivariate skew t distributions, there are no studies examining the efficacy of fit statistics in identifying the correct number of classes when skew t component distributions are fit to an overall distribution. Additionally, of the three studies that focus on fitting skew t distributions, two (Asparouhov & Muthen, 2014; Muthen & Asparouhov, 2014) focus on growth mixture models rather than finite mixture models and limit the fits statistics to the AIC and BIC. Lee and McLachlan (2014) also limited the fit statistics to the AIC and BIC, but within a finite mixture context. They fit models to real data with known two class components and did not address the issue of a true one class distribution. Researchers who employ factor FMMs also tend to limit the selection of fit statistic to one of the information criteria statistics—AIC, BIC, etc. and the one study that specifically examined the identification of spurious classes within IRT—Alexeev, Templin, and Cohen (2011)—used only the BIC to determine model selection.

The purpose of this study is to explore the extent to which finite mixture models identify spurious classes. This simulation study examines conditions when the observed data forms a single-component nonnormal distribution. Finite mixture models with one to four components were fitted and log likelihood based, classification (entropy and entropy penalty) based, and likelihood ratio based fit statistics were employed to identify the best fitting model.

Overview, Design & Procedures

Each data condition was replicated 250 times and one to four components were fitted to each condition’s distribution. Because of the amount of time required to calculate estimates from models fitting multivariate skew t component distributions, these conditions were limited to 125 replications. Log likelihood based, classification (entropy and entropy penalty) based, and likelihood ratio based fit statistics were employed to identify the best fitting model. Additionally, for all models, each analysis is conducted using 500 random starts of 20 iterations to help avoid the problem of the EM algorithm resolving on local maxima as well as improving the probability of obtaining model convergence, particularly when fitting the four component models. The percentage of times each fit statistic identified a model with *k* number of components is computed, where k = 1, 2, 3, and 4. The outcome of interest is the percentage of times that the correct one class model fit the data best. For each of the 72 analysis by data conditions the percent number of replications that each fit statistic identified each of the one through four class models as the correct model was calculated. These were then combined into a single data set for analysis. The statistics program R was used to generate the data, the R package Mplus Automation was used to create, run, and extract summaries from the models run in Mplus 7.3, and R was used to create the analysis data set and conduct analyses using it.

Chapter 2 Literature Review

Finite mixture modeling (FMM) is a popular tool for model based clustering. It provides a number of advantages over traditional cluster analysis methods like hierarchical and K-means clustering. For one, FMM draws upon statistical theory for postulating a population model and deriving sample estimates of model parameters (Vermunt & Magidson, 2002). It is also insensitive to the scaling of observed variables, and it allows cases to be probabilistically assigned to clusters. Another advantage is that FMM provides for the development of nonarbitrary clustering criteria. These advantages make FMM an attractive alternative to traditional clustering methods. However, FMM has problems of its own that may hinder its efficacy in clustering observations.

Bauer and Curran (2003a) write that a nonnormally distributed composite distribution is a necessary and perhaps even sufficient condition for the identification of nontrivial component distributions (i.e. clusters) giving rise to the data. The problem is that the converse is not true. Subpopulations and clusters represented by multiple component distributions are not a necessary condition for data to be nonnormally distributed. Data may be nonnormally distributed for a variety of valid reasons that have nothing to do with the presence of clusters in the data. For example, response times are known to be positively skewed and test scores are commonly platykurtic and negatively skewed (Keats & Lord, 1962). The result is that FMM will identify spurious classes when applied to data generated from a single component nonnormal distribution (Bauer & Curran, 2003a). Consequently, researchers may arrive at erroneous conclusions about a substantive area of study. The problem of identifying spurious classes is not limited to FMMs.

Alexeev, Templin, and Cohen (2011) showed that an item response theory mixture Rasch model will identify spurious classes when it is fit to data generated as a three parameter logistic (3PL) or a two parameter logistic (2PL) model. Further, they showed that the likelihood of identifying spurious classes increases with the number of test items, the number of subjects, and the deviation of the slope parameter from the slope parameter assumed by the Rasch model—i.e. one. In fact, a single item in a 30-item test with a slope parameter of 2.5 was sufficient to cause the identification of spurious classes in all replications when the sample size was 10,000. However, with a smaller sample size (4,000), four items with slope parameters of 1.75 caused the identification of spurious classes in all replications, but four items with slope parameters of 1.5 did not result in the identification of any spurious classes. Although, it should be noted, this part of their examination of the problem spurious classes was limited to ten replications for each condition.

Bauer and Curran (2003a) demonstrated the influence of nonnormal distributions on spurious classes by testing one and two component mixture models. Others also limited their work to one and two component distributions (Muthen, 2003; Rindskopf, 2003). Consequently, the number of spurious classes identified in nonnormal distributions was not fully appreciated. Bauer and Curran (2003b) later suggested that the problem of spurious classes may be even more serious than their initial analysis indicated where researchers could identify far more than two classes. They demonstrated that a four-class normal mixture model fit the single-component nonnormal data even better than the two-class solution in their original paper.

Taken together, these results suggest that the full extent of the problem with finite mixture models identifying spurious classes has not been fully explored. The purpose of this study is to explore the extent to which finite mixture models identify spurious classes. This simulation study examines conditions when the observed data forms a single-component nonnormal distribution. Finite mixture models with one to four components will be fitted and log likelihood based, classification (entropy and entropy penalty) based, and likelihood ratio based fit statistics will be employed to identify the best fitting model.

Mixture Modeling Applications

FMM is a popular choice for cluster analysis. In education, it has been applied to test motivation theory (Pastor, Barron, Miller, & Davis, 2007), academic performance standards (Brown, 2007; Alexeev, Templin, & Cohen, 2011; Kelava & Brandt, 2014), deficiency in mathematics understanding (Chan, Leu, & Chen, 2007), performance decline (Jin & Wang, 2014), identifying groups based on guessing behavior (Leong, Mahdi, & Ling, 2013) and test speededness (Bolt, Cohen, & Wollack, 2002; Meyer, 2010). Additionally, FMM has been applied to differential item functioning (Cohen & Bolt, 2005; Frederickx, Tuerlinckx, De Boeck, & Magis, 2010; Cho & Cohen, 2010; Finch & Finch, 2013; Lee & Beretvas, 2014), as a follow up to identify causes of differential item functioning (Cohen & Bolt, 2005; Cho, Lee, & Kingston, 2012), and latent transition analysis following an intervention (Cho, Cohen, Bottge, 2013). In the social sciences, it has been applied to aggression and harassment (Giang & Graham, 2008), quality of life (Punzo, 2014), and profiling child maltreatment perpetrators (Yampolskaya, Greenbaum, & Berson, 2009). In addition, Hoijtink and Notenboom (2004) used mixture modeling to evaluate the stage model theory of the development of spelling ability in children. Büsch, Hagemann, and Bender (2010) evaluated a questionnaire assessing handedness and Freund, Tietjens, and Strauss (2013) examined the effects of using different Likert scale ranges for an assessment of physical self-concept. McCrae, Chapman, and Christ (2006) searched for latent groups among children who had been sexually abused and Gebhardt, Rose, and Mitte (2013) used mixture models to evaluate the efficacy a median split procedure in correctly classifying persons who repress negative affect experiences. Other research used latent class cluster analysis to create profiles of eating disorders (Wade, Crosby, & Martin, 2006), develop four appraisal/coping styles used in stressful situations (Larsson, Kempe, & Starrin, 1988), describe the psychosocial adjustment of adolescents (Ding, 2006), examine latent classes from personality data (Maij-de Meij, Kelderman, & van der Flier, 2008), heterogeneity of people who took the Chinese version of the Beck Depression Inventory—II (Wu & Huang, 2010), and assessment of lifetime prevalence of mental disorders (Almansa, Vermunt, Forero, & Alonso, 2014). Mixture modeling analysis has also been applied to data from fields such as business (Tuma & Decker, 2013), medicine (Bozodogan, 1994; Sawatzky, Ratner, Kopec, & Zumbo, 2012; Howe & Bozodogan, 2012; Hesser & Andersson, 2014; Lee & McLachlan, 2014; Muthen & Asparouhov, 2014), ornithology

(McCrea, Morgan, & Cole, 2012), production / mechanical engineering (Yu, 2012), and transportation safety (Kim & Mahmassani, 2014; Park, Lord, & Lee, 2014). These applications of mixture modeling not only inform a substantive area of research, but also have implications for cluster members. Researchers must have confidence that clusters have been accurately identified. However, the question of whether the correct number of clusters has been identified has been a concern since the first attempts to uncover unknown subpopulations within an overall distribution.

Historical Perspective

Pearson (1894) was one of the first to apply a method of examining a distribution to determine its component parts. W.F.R. Weldon (1893) noted an asymmetrical distribution in his data on the ratio of forehead to body length in crabs and speculated whether this could indicate that the crabs were evolving into two subspecies. Weldon, realizing the limitations of his ability, asked his colleague, Pearson, to tackle the problem. Pearson employed a method-of-moments approach to create three models, each fitting two univariate normal distributions to the data; a process which included solving a ninth degree polynomial. Model 1 provided the best representation of the data with each normal distribution accounting for .4145 and .5855 respectively. After calculating the sixth moment, Pearson concluded that there was insufficient evidence to support the hypothesis of the existence of two subspecies and that the asymmetric distribution was the result of a naturally occurring variation in the population (Pearson, 1894). However, in a paper the following year, Pearson acknowledges the problem associated with identifying the correct number subpopulations in the data:

> The question may be raised, how are we to discriminate between a true curve of skew type and a compound curve, supposing we have no reason to suspect our statistics a priori of mixture. I have at present been unable to find any general condition among the moments, which would be impossible for a skew curve and possible for a compound, and so indicate compoundedness. I do not, however, despair of one being found (Pearson, 1895, p. 394).

The moments method employed by Pearson presented a daunting, complex time consuming task; for this reason, the use of mixture modeling was impractical and frequently beyond the capability of many researchers. However, with the advent of accessible computing resources, researchers were once again able to address the question of potential unknown subpopulations within an overall population.

Thorndike (1953), Cox (1957), and Fisher (1958) were among the first to propose methods of cluster analysis to identify classes within a population. However, it was MacQueen (1967) who formalized the *K*-means clustering method. *K*-means is the most popular form of cluster analysis and it is closely related to one of the simplest forms of FMM; both which postulate discrete latent classes with continuous indicator variables (see Kogan, 2007; Steinley, 2006). *K*-means clustering is an iterative procedure that begins with the researcher selecting the number of clusters to test. The process uses an algorithm developed independently by Lloyd (1957, 1982) and Forgy (1965). (Lloyd initially proposed his method at the Institute of Mathematical Statistics Meeting in 1957, but it was not published outside of Bell Laboratories, where he worked, until 1982.) The algorithm estimates cluster means and each data point is assigned to a cluster based on its distance from the cluster means—i.e. each person is assigned to a cluster based on the nearness of their score to the cluster mean. Cluster means are then estimated again based on the cluster assignments. This process continues until a stable solution is achieved—i.e. people are no longer switching between clusters. Similarly FMM uses the expectation-maximization (EM) algorithm (Dempster, Laird, & Rubin, 1977) to calculate maximum likelihood estimates. The EM algorithm also employs an iterative procedure where prior parameter estimates permit the calculation of the probability of class membership in the E-step and this information is used to find parameter estimates that maximize the log-likelihood in the M-step. Parameter estimates from the M-step serve as prior estimates for the E-step in the next iteration. This process continues until convergence on a stable solution—i.e. changes in parameter estimates become very small.

Although the two approaches are similar mixture modeling provides advantages over *K*-means clustering. Magidson and Vermunt (2002a, 2002b) and Vermunt (2011) showed the superiority of mixture modeling for cluster extraction and determining cluster membership for an individual set of scores. Stienley and Brusco (2011) had mixed results dependent upon the parameters imposed on the model; however, both Stienley and Brusco (2011) and McLachlan (2011) caution about generalization beyond this study. Further, Vermunt (2011) argues that many of the models investigated by Stienley and Brusco are rarely, if ever, seen in practical applications. Nonetheless, mixture modeling provides researchers many advantages over the traditionally used clustering approaches. It can model within and between class variance. It produces probability-based classification that allows the use of fit indices to evaluate the correct number of classes. It does not require the standardization of variables. It can be used with either continuous, categorical, or a mixture of both types of indicator variables. It allows for the inclusion of other variables—i.e. demographics—to simultaneously describe the clustering solution rather than having to perform a separate discriminant analysis. Because of these advantages and the increased availability of high speed computing resources necessary for the complex calculations involved in the method, mixture modeling has grown in popularity as a method of cluster analysis.

Regardless of whether method-of-moments, *K*-means clustering or FMM is used, determining the correct number of classes with a high degree of certainty remains elusive. This uncertainty is compounded further when there is no a priori hypothesized or known number of classes in the data. For instance, in the crab data analyzed by Pearson, he hypothesized the existence of two possible classes—subspecies. However, he acknowledges that he could have fit more than two normal distributions to the data, but as the number of mixtures increases the number of calculations would have increased exponentially (Pearson, 1894). Herein lays a major problem with the mixture modeling method. The number of distributions—mixtures—fitted to the data is discretionary based on how many classes the researcher hypothesizes, or suspects, may be in the data. The resulting models, with differing numbers of mixtures, are evaluated by how well they fit the data. When competing models fit the data equally well, it becomes extremely difficult to empirically determine the true nature of the distribution; particularly in the absence of strong theoretical justifications.

Finite Mixture Models

Multivariate Normal Mixture Model. A finite mixture model assumes a population density such that

$$f(\mathbf{x}; p, \theta) = \sum_{k=1}^{K} p_k g_k(\mathbf{x}; \boldsymbol{\theta}_k) \qquad 2.1$$

where p_k represents the proportion of the overall distribution accounted for by the g_k component distributions with $\boldsymbol{\theta}_k$ parameters. The proportions p_k must be greater than zero for all *k* and must sum to one. (Note the symbols p and p_k represent population parameters that are normally represented with the Greek letter π. However, equation 2.2 includes the numeric value of π so the symbols p and p_k were used to avoid confusion.) While the component distributions can be of any distributional form, it is most common to assume a multivariate normal distribution (McLachlan & Basford, 1988; McLachlan & Peel, 2000). Under the assumption of multivariate normality the population density becomes

$$f(\mathbf{x}; p, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = \sum_{k=1}^{K} p_k \frac{exp\left\{-\frac{1}{2}(\mathbf{x}_i - \boldsymbol{\mu}_k)'\Sigma_k^{-1}(\mathbf{x}_i - \boldsymbol{\mu}_k)\right\}}{2\pi^{m/2}|\Sigma_k|^{1/2}} \qquad 2.2$$

where $\boldsymbol{\mu}_k$ is the mean vector with *m* dimensions and Σ_k is the *m* x *m* covariance matrix for the *k*th class. Adhering to the assumption of multivariate normal component distributions may increase the likelihood of identifying spurious classes because the normal component distributions must be forced in to account for the thicker elongated tail of the skewed overall distribution (Asparouhov & Muthen, 2014). Therefore, relaxing the assumption of multivariate normal component distributions should allow the component distributions to more accurately represent the overall observed skewed distribution. One recently developed method for relaxing the multivariate normal component assumption is the fitting of multivariate skew t distributions.

Multivariate Skew t Mixture Model. Lee and McLachlan (2014) show the derivation of an unrestricted multivariate skew t component distribution model and a restricted multivariate skew t component distribution model. They note that the restricted multivariate skew t model is not nested within the unrestricted model; however, when the skewing function is univariate, the models are the same. In other words, the restricted model estimates a univariate latent skew variable for fitting the component distributions, but the unrestricted model estimates a multivariate latent skew variable for fitting the component distributions. Consequently, the estimation of the unrestricted model takes considerably longer than the estimation of the restricted model. Additionally, maximum likelihood model estimation is not a straight forward process when estimating the unrestricted model. To calculate the intractable conditional expectations, in the E-step of the EM algorithm additional estimation procedures must be included such as Monte Carlo integration (Lin, 2010) or a one-step late approach (Lee & McLachlan, 2011). Both of these estimation methods increase computational time which can become burdensome particularly with high dimensional data. Since the restricted multivariate skew t model estimates a univariate latent skew variable, maximum likelihood estimation is accomplished via the EM algorithm without additional computational steps.

Asparouhov and Muthen (2014) argue that the assumption of a univariate latent variable responsible for skewness in the data is reasonable for data commonly used with structural equation and mixture models. Therefore, the estimation of a multivariate latent skew variable may be unnecessary. Given this argument and that the restricted multivariate skew t model is a more parsimonious model with less computationally intensive model estimation, in most cases, it should be the preferred model. Following the notation of Lee and McLachlan (2014)—also used in Asparouhov and Muthen (2014)—the assumed population density of g multivariate restricted skew t components is given by,

$$f(\boldsymbol{Y};\boldsymbol{\Psi}) = \sum_{\mathrm{h=1}}^{\mathrm{g}} \pi_h\, f(y;\ \boldsymbol{\mu}_h, \boldsymbol{\Sigma}_h, \boldsymbol{\delta}_h, \upsilon_h \qquad 2.3$$

where $\boldsymbol{\mu}_h$ is a vector of means with m dimensions, $\boldsymbol{\Sigma}_h$ is a m x m covariance matrix, $\boldsymbol{\delta}_h$ is a vector of skew parameters with m dimensions, and υ_h is the degrees of freedom. The mixing proportions π_h must be greater than or equal to zero and the sum of the mixing proportions must equal one. With this population density, the multivariate t distribution density function is,

$$t_{m,\upsilon}(y, \mu, \Omega) = \frac{\Gamma\left(\frac{\upsilon + m}{2}\right) |\boldsymbol{\Omega}|^{-1}}{(\pi\upsilon)^{m/2}\Gamma(\frac{\nu}{2})[1 + d(y)/\nu]^{(\nu+m)/2}} \qquad 2.4$$

where

$$\boldsymbol{\Omega} = \boldsymbol{\Sigma} + \boldsymbol{\delta}\boldsymbol{\delta}^T \qquad 2.5$$

and

$$d(y) = (y - \boldsymbol{\mu})^T \boldsymbol{\Omega}^{-1}(y - \boldsymbol{\mu}) \qquad 2.6$$

It follows from the equations that the multivariate skew t becomes a multivariate t normal distribution when $\delta = 0$, it becomes a multivariate skew normal distribution as $\nu \rightarrow \infty$, and when under both conditions, $\delta = 0$ and $\nu \rightarrow \infty$, it becomes a multivariate normal distribution. The fitting of multivariate skew t distributions provides researchers with a tool that will enable them to more accurately model their observed data. In fact, the anticipated improvement in the correct identification of subpopulations within an overall population distribution brought about by the fitting of multivariate skew t distributions led Asparouhov and Muthen (2014) to assert that "Spurious class formation due to non-normality and skewness will be eliminated" (p.6).

Up to this point I have focused on the conception of FMM in the context of distributions of observed data. However, the method—unlike *K*-means clustering—can be extended to the identification of classes based on latent variable differences by using a factor finite mixture model (FaFMM) or an item response theory finite mixture model (IRTFMM).

Factor Finite Mixture Models. FaFMM is based on the common factor model (Jöreskog, 1971) with the addition of a latent variable for class membership. The FaFMM should not be confused with the mixtures of factor analyzers model (McLachlan & Peel, 2000). This model, also referred to as a mixture of factors model, is based on the exploratory factor model where variables are reduced to class specific factors. In FaFMM a single factor structure is assumed to hold for all classes. The FaFMM can be expressed with the regression equations:

$$y_{ik} = \boldsymbol{\nu}_k + \boldsymbol{\Lambda}_{yk}\boldsymbol{\eta}_{ik} + \boldsymbol{\varepsilon}_{ik} \qquad 2.7$$

$$\boldsymbol{\eta}_{ik} = \mathbf{A}\mathbf{c}_i + \boldsymbol{\zeta}_{ik} \qquad 2.8$$

In the first equation, y_{ik} is the score on variable Y for individual *i* in the *k*th class, $\boldsymbol{\nu}_k$ are the regression intercepts, $\boldsymbol{\Lambda}_{yk}$ are the factor loadings, $\boldsymbol{\eta}_{ik}$ is an individual's score on the latent factor, and $\boldsymbol{\varepsilon}_{ik}$ are the residuals that capture measurement error and variance attributable to factors not in the model. In the second equation $\mathbf{A}$ is the intercept(s) of the factor(s), $\mathbf{c}_i$ is the individual's standing on the multinomial latent class variable *C*, and $\boldsymbol{\zeta}_{ik}$ is the residual of the factor scores. Muthén and Shedden (1999) and Lubke and Muthén (2005) use a similar model specification, but also include the modeling of a continuous covariate and categorical class predictor variable.

As with any comparison between groups using a common factor model, the model must meet the conditions of measurement invariance (MI) (Mellenbergh, 1989; Meredith, 1993). MI means the differences between classes are attributable only to differences on the modeled latent factor(s) and not to class specific differences on variables not in the model. For MI to hold, three conditions must be met in progressively more restrictive order: equal factor loadings between classes, equal observed variable intercepts between classes, and equal residual variance between classes. If factor loadings are not equal between classes, then the observed variables have a differential influence on the class factor score that is due to some class specific characteristic rather than on the latent factor(s) of interest. If the intercepts of the observed variables are not equal between classes, then the mean responses of the classes on the observed variables is, in part, due to an unmodeled latent variable rather than the factor(s) of interest and any comparison based on factor scores is unwarranted because of the influence of the unmodeled latent variable. Finally, in the most restrictive level of MI, if the residual variances are not equal between classes, then either the measurement errors and/or the class specific errors are not the same. Some researchers have suggested that the model is sufficiently invariant if the first two conditions of MI are met (Little, 1997; Widaman & Reise, 1997). However, Lubke and Muthén (2005) warn that making this assumption has pitfalls that can adversely affect the inferences made based on the model results and suggest a partial measurement invariance (PMI) evaluation. Bryne, Shavelson, and Muthén (1989) show how to evaluate PMI and show that class comparison based on factor score are still valid as long as invariance holds for at least two observed variables per factor and the varying parameters are modeled for their factor and class specific influence. For instance, if invariance of factor loadings holds and invariance of intercepts does not, the observed items that have varying class intercepts can be regressed on the latent class variable *C* and the latent factor variable. This model specification will account for the class specific variation in the intercept as well as the shared factor variation. While PMI is a feasible method for addressing some invariance issues, it complicates interpretation considerably and, if there are a sufficient number of observed variables, it may be simpler to remove the ones that are not invariant. Of course, as with any decision to remove a variable from the model, this action must be consistent with the theoretical underpinnings of the model. In sum, since FaFMM seeks to uncover latent classes based on their standing on a latent factor, it is imperative that model invariance holds. The invariance assumption is also critical to IRTFMM. However, within the framework of IRT the assumption of parameter invariance must be violated if more than one class is found through the data.

Mixture Item Response Models. One of the central assumptions of IRT is that item and person parameters are invariant in the population. Differential item function (DIF) analysis is a statistical evaluation testing whether examinees from different groups exhibit differing probabilities of responding correctly to an item after matching the examinees from both groups on a variable that measures the same construct that the item is intended to measure. In other words, do members of one group—e.g. females—have a different probability of correctly responding to an item than members of another group—e.g. males—when the members of both groups are matched on their ability level. If DIF is found, then the item parameter estimates in the sample are not invariant with respect to those groups. IRTFMM can be conceptualized as a DIF analysis when unknown groups, classes, may exist in the sample. If examinees from the sample can be classified in such a way that item parameter estimates differ after matching the latent classes on ability, then the lack of item parameter invariance is an indicator of the existence of different classes in the sample.

The different invariance requirements for FaFMM and IRTFMM stem directly from the theories underlying each approach. Classical Test Theory (CTT) focuses its analysis on the whole test—all of the items at once—and then works back to examine items. IRT focuses on the individual items and works forward to examine the whole test. This fundamental difference between the two theories influences how data are handled in application. CTT factor analysis uses a reduced form of the observed variables correlation matrix where the item communalities replace the diagonal elements of the matrix. The level of measurement of the item data determines which correlation matrix should be calculated. If the items are continuous, then a Pearson's **R** matrix is appropriate.

However, when items are scored dichotomously—categorical right / wrong—or polytomously—ordinal partial credit based on which choice is made of a limited number of options—then the item scores will violate one of the assumptions for calculating Pearson's *r*. While the **R** matrix can be calculated from the raw scores the relationship among the items will be underestimated. Therefore, with dichotomously scored item data a tetrachoric correlation matrix is appropriate and with polytomously scored data a polychoric correlation matrix is appropriate. Because of its focus on the item rather than on the whole test combination of items IRT does not concern itself with the calculation of correlation matrices because it uses raw item scores to calculate the probability of correct responses. With dichotomously scored data it models the probability of a right answer and with polytomously scored data it calculates step thresholds—the probability where selecting one score level and the next higher score level is fifty percent. IRT is not appropriate for items that are scored at the continuous level of measurement.

A number of different IRTFMMs have been proposed and implemented including a mixture partial credit mixture model (Rost, 1991), a nominal response mixture model (Bolt, Cohen, & Wollack, 2001), a multilevel mixture model (Cho and Cohen, 2010), a multidimensional mixture model (De Jong & Steenkamp, 2010), and a multilevel multidimensional mixture model (Finch & Finch, 2013). However, the most basic form of IRTFMM is the mixture Rasch model (MRM; Rost, 1990). MRMs are used with dichotomously scored data where the probability of a response (x_i) is expressed as:

$$p(\boldsymbol{x}|\theta) = \sum_g \pi_g p(\boldsymbol{x}|\,\theta, g) \qquad 2.9$$

where g is indicates class membership and the mixing proportions π_g must be greater than or equal to zero and the sum of the mixing proportions must equal one and $p(\boldsymbol{x}|\theta, g)$ is the conditional probability of the response which is the product of response probabilities across all items,

$$p(\boldsymbol{x}|\theta, g) = \prod_{i=1}^{I} \frac{\exp\,(x_i(\theta - \beta_{ig}))}{1 + \exp\,(x_i(\theta - \beta_{ig}))} \qquad 2.10$$

In the above equation, β_{ig} is the item difficulty for a class of respondents and θ is the person ability parameter. When $\theta = \beta_{ig}$ the probability of a correct response is .5 and classes are determined by differences in their item difficulty parameters.

In the context of cluster analysis, the purpose of finite mixture modeling is to estimate the model parameters and compute the probability that each observation belongs to each of the classes. The model is fit several times using a different number of classes and fit statistics allow the researcher to identify the model with the correct number of classes. Fit statistics can be broadly categorized into three types: information criteria based, classification (entropy and entropy penalty) based, and likelihood ratio test based.

Evaluating Competing Models

Information criteria are a function of the log-likelihood and some other penalty imposed for model complexity. These fit statistics differ on the way the penalty is computed, but the penalty is necessary given that statistical models with more parameters fit data better than similar models with fewer parameters. In the mixture modeling context, this means that models with more component distributions will fit better than those with fewer components. The penalty helps prevent over identifying the number of classes. The fit statistics employed in this study are reviewed below.

Akaike's Information Criterion (AIC; Akaike, 1973) is one of the earliest information criteria and is defined as

$$\text{AIC} = -2\,LL + 2p \qquad 2.11$$

where LL is the maximized log likelihood and p is the total number of free parameters in the model. Unfortunately, the AIC is not theoretically consistent; selection of the correct model does not increase consistently as sample size nears infinity (Woodruffe, 1982). The Bayesian Information Criterion (BIC; Schwartz, 1978) is defined as

$$\text{BIC} = -2\,LL + p \log N. \qquad 2.12$$

Multiplying the total number of free parameters by the log of the sample size imbues BIC with a consistent quality (Haughton, 1988) and makes the criteria slightly more conservative than the AIC. Following the BIC, a consistent version of AIC (CAIC; Bozdogan, 1987) is defined as

$$\text{CAIC} = -2\,LL + p\,(\log N + 1). \qquad 2.13$$

The CAIC applies a more severe penalty on the number of parameters than does either BIC or AIC, and, therefore, tends to favor models with fewer parameters. Contemporaneously, Sclove (1987) proposed a sample size adjustment for information criteria given by the equation

$$N^* = (N + 2)/24 \qquad 2.14$$

and N^* replaces sample size (N) in the sample adjusted information criteria equations:

$$\text{SABIC} = -2\,LL + p\,\log\,(N + 2)/24 \qquad 2.15$$

$$\text{SACAIC} = -2\,LL + p\,(\log\,((N + 2)/24) + 1). \qquad 2.16$$

This sample size adjustment to the AIC and BIC makes these consistent information criteria slightly more liberal than their non-adjusted forms.

Classification (entropy and entropy penalty) based. Classification fit statistics use entropy, a measure of classification accuracy, as penalty when determining the fit of competing models. Essentially, the posterior probabilities of class membership are estimated and clearer assignment of cases to latent classes results in a better fit. When the separation among multivariate means of the component distributions for the classes is large, entropy, and the entropy penalty, should be smaller than when the multivariate means of the component distributions are closer together and assignment of cases to classes becomes less clear. This also implies that as the number of component distributions fitted to the data increase, the less clear the assignment of cases to classes becomes and, therefore, the larger entropy and entropy penalty becomes. Thus, the entropy and entropy penalty based fit statistics act to encourage the selection of a more parsimonious model. The most commonly used entropy and entropy penalty based fit statistics are the normalized entropy criterion (NEC; Celeux & Soromenho, 1996), the classification likelihood criterion (CLC; Biernacki & Govaert, 1997), and the integrated classification likelihood criterion-BIC (ICL-BIC; Biernacki, Celeux, & Govaert, 1998).

Entropy and entropy penalty fit statistics are based on the observation that the estimated log-likelihood of a model may be broken down into two component parts:

$$LL = LL_c + EN(\hat{\tau}) \qquad 2.17$$

where LL_c is the log-likelihood obtained if posterior probabilities of class membership are constrained to zero and one—one representing perfect classification—and $EN(\hat{\tau})$ is entropy, which captures the error associated with the classification. Entropy is given as

$$EN(\hat{\tau}) = -\sum_{i-1}^{N}\sum_{k-1}^{K}\hat{\tau}_{ik}\log\hat{\tau}_{ik} \qquad 2.18$$

where $\hat{\tau}_{ik}$ is the estimated posterior probability that individual *i* is a member of group *k*. Therefore, as entropy approaches zero, *LL* approaches LL_c or perfect classification. Thus, entropy is a measure of classification quality with smaller values indicating higher quality.

The NEC as proposed by Celeux and Soromenho (1996) is defined as

$$\text{NEC} = \frac{EN(\hat{\tau})}{LL_k - LL_{k=1}} \qquad 2.19$$

where LL_k is the log likelihood of a model with *K* component distributions. As the equations used for the calculation of the NEC show, the NEC is undefined when testing a $K = 1$ model. To address this issue Celeux and Soromenho (1996) and Biernacki, Celeux, and Govaert (1999) suggest a logical test where a model with $K > 1$ is preferred if the calculated NEC ≤ 1; if no *K* component model meets this criteria, then the $K = 1$ model is preferred.

The CLC (Biernacki & Govaert, 1997) is given as

$$\text{CLC} = -2LL + 2EN(\hat{\tau}) \qquad 2.20$$

The entropy component of the CLC is undefined in the K = 1 model. However, the CLC can still be employed without the logical test necessary with the NEC. In the K = 1 model, since there is only one distribution, the correct classification equals one and the entropy component of the CLC calculation becomes zero. Therefore, in the K = 1, the CLC reduces to –2LL which is used in comparison to the CLC statistics from the K > 1 models where the penalty $2EN(\hat{\tau})$ is greater than zero.

The ICL–BIC uses entropy as an adjustment to the information criterion BIC and is defined as

$$\text{ICL} - \text{BIC} = -2LL + p\log N + 2EN(\hat{\tau}). \qquad 2.21$$

As in the CLC, in the K =1 model the $2EN(\hat{\tau})$ penalty component of the formula is zero. Therefore, the ICL – BIC reduces to the BIC statistic in the one component model and subsequent K > 1 models will have a penalty ($2EN(\hat{\tau})$) greater than zero. Since the ICL – BIC incorporates the penalties employed in the BIC and the CLC, it tends to be more conservative in model selection.

Likelihood ratio test based. The likelihood ratio test (LRT) is a nested model test defined as

$$\text{LRT} = -2(LL_{k-1} - LL_k) \qquad 2.22$$

The LRT statistic is a widely used method for testing nested models that is chi-square distributed with degrees of freedom equal to the difference in the number of estimated parameters for the competing models. However, in the mixture modeling context, the LRT cannot be used to test models where the number of classes is nested. The reason for this is that the parameters of the *k* class model must be set to zero to specify the *k* – 1 model. That is to say the probability of being in the *k*th class must be set to zero. Since probabilities range from zero to one, the parameter is being set at the border of admissible space. Consequently, the LRT statistic is no longer asymptotically chi-square distributed (McLachlan & Peel, 2000). To overcome this distributional dilemma, researchers have proposed alternative methods for employing the LRT in evaluating the fit of models with nested component structures: the bootstrap likelihood ratio test (BLRT; McLachlan, 1987) and the Lo, Mendell, and Rubin likelihood ratio test (LMR-LRT; 2001).

The BLRT is based on the bootstrap method (Efron, 1979) where a statistic is estimated on many sample distributions whose cases are drawn, with replacement, from the original distribution. This creates a sampling distribution of the statistic that is based on data from the original observed distribution and allows for the estimation of bootstrapped standard errors and confidence intervals for the statistic. In the context of mixture models and the LRT, the process is simply: 1) calculate the LRT for the original sample, 2) draw a bootstrap sample using the maximum likelihood estimated parameters from the null model (*k* – 1) and calculate the LRT for the sample, 3) repeat step 2 many times to create the true sampling distribution of the LRT statistic under the null hypothesis, and 4) compare the originally obtained LRT in step 1 with the distribution obtained after step 3 to get a *p* value that indicates whether the *k* – 1 (null) model should be rejected in favor of the *k* (alternative) model (McLachlan, 1987; McLachlan & Peel, 2000).

Lo, Mendell, and Rubin (2001) extended the work of White (1982) and Vuong (1989) to mixture model applications. Vuong showed that the LRT statistic can be viewed as a weighted sum of independent random variables that are chi-square distributed with one degree of freedom. From this Lo, Mendell, and Rubin defined the LMR – LRT as

$$\text{LMR} - \text{LRT} = 2\sum_{j=1}^{n} log\frac{f(X_j;\hat{\theta})}{g(X_j;\hat{\gamma})} \qquad 2.23$$

where $\hat{\gamma}$ is the set of unknown parameters estimated under the null (*k* – 1) component model and $\hat{\theta}$ is the set of unknown parameters estimated under the alternative (*k*) component model. In the same article Lo, Mendell, and Rubin proposed an adjustment to the LMR – LRT to improve its accuracy that they gave as

$$\text{aLMR} - \text{LRT} = \frac{\text{LMR} - \text{LRT}}{1 + \{(p-q)\log n\}^{-1}} \qquad 2.24$$

where *p* and *q* are the number of independent chi-square distributed random variables for the alternative (*k*) and null (*k* – 1) models respectively. They note that as $n \rightarrow \infty$ the adjustment $\{(p-q)\log n\}^{-1} \rightarrow 0$. The aLMR – LRT, and the LMR – LRT, result in a *p* value that indicates whether the *k* – 1 (null) model should be rejected in favor of the *k* (alternative) model. Jefferies (2003) contends that one of the assumptions in the mathematical proof for the LMR – LRT is not fully met for normal outcomes. However, Nylund, Asparouhov, and Muthén (2007) assert that the findings from the simulations conducted by Lo, et al. (2001) provide sufficient evidence of the usefulness of the test in determining the number of classes in a distribution.

The purpose of the fit statistics is to aid the researcher in determining the number of component distributions in an overall distribution. Their different formulations and methods of penalizing models that estimate a larger number of parameters can often lead to markedly different decisions based upon which fit statistic is used. To date there is no consensus on the “best” fit statistic to use in determining the number of component distributions in an overall distribution. Comparing the ability of the various fit statistics in identifying the correct number of component distribution provides important information for researchers seeking to uncover the true characteristics of the membership in their sample.

Efficacy of Fit Statistics

Identifying the number of clusters or components in the data depends on the researcher’s choice of fit statistic. Two standards are of interest when evaluating the efficacy of a fit statistic. One standard is like statistical power. It concerns how often the correct number of clusters is identified when there really are clusters in the data. The second standard is like type I error. It concerns how often multiple clusters are identified in the data when no clusters actually exist. A large portion of research on the efficacy of fit statistics in mixture model has focused on the first criteria while only a few have considered the second.

Table 1 shows the fit statistics compared in seven efficacy studies with an indication of the statistics that performed well relative to the others in the study. Many of the statistics found in the other studies are not included in this study. The information criteria statistics not included in this study are D – BIC (Draper, 1995), deviance information criterion (DIC; Speigelhalter, Best, & Carlin, 1988), Efron information criteria (EIC; Ishiguro, Sakamoto, & Kitagawa, 1997), HQ (Hannan & Quinn, 1979), HT – AIC (Hurvich & Tsai, 1989), The classification (entropy penalty) statistic not included in this study is integrated classification likelihood criterion-(ICL; Biernacki, Celeux, & Govaert, 1998) without the BIC adjustment and the only likelihood ratio statistic not included is the stand alone LRT, which is not suitable for testing against a true one class condition. Other fit statistics such as the matrix based Laplace-empirical criterion (LEC; Roberts, Husmeier, Rezek, & Penny, 1998), two goodness of fit tests: the multivariate skewness test (MST) and multivariate kurtosis test (MKT) (Muthén, 2003), and the Bayesian pseudo-Bayes factor (PBF; Geisser & Eddy, 1979) and posterior predictive model checks (PPMC; Gelman, Carlin, Stern, & Rubin, 1996) are also not included in this study.

Tofighi and Enders (2007) in a growth mixture modeling (GMM) format found that, across conditions, the SABIC and the LMR-LRT outperformed all other fit statistics they examined. Henson, Reise, and Kim (2007) in a structural equation mixture modeling format that, across conditions, the ICL-BIC and the CLC outperformed all other fit statistics in selecting the true two class model. However, Nylund, Asparouhov, and Muthén (2007) found that the BLRT outperformed the other likelihood tests and the BIC outperformed the information criteria tests. McLachlan and Ng (2000) reported the results of three simulation studies. They found that the ICL-BIC, the CLC, and the LEC identified the true model and, of these three, the ICL-BIC is the easiest to implement.

Table 1.
Fit statistics by efficacy study.

	Bauer & Curran (2003a)	Henson, Reise, & Kim (2007)	Li, Cohen, Kim, & Cho (2009)	McLachlan & Ng (2000)	Nylund, Asparouhov, & Muthén (2007)	Peugh & Fan (2013)	Tofighi & Enders (2007)
Information Criteria							
AIC	x	x	x	x	x	x	x
BIC	x	x	x*	x	x*	x*	x
CAIC	x	x			x	x*	x
D-BIC						x*	
DIC			x				
EIC				x			
HQ						x	
HT-AIC						x	
SABIC		x			x	x	x*
SACAIC						x*	x
SAD-BIC						x	
SAHT-AIC						x	
SAHQ						x	
Classification							
CLC	x	x*		x*		x*	
Entropy		x					
ICL				x			
ICL-BIC	x*	x*		x*		x*	
NEC	x	x				x	
SAICL-BIC						x*	
Likelihood Ratio							
aLMR-LRT		x				x	
BLRT					x*	x	
LMR-LRT		x			x		x*
LRT					x		
Other							
LEC				x*			
MST		x					x
MSK		x					x
PBF			x				
PPMC			x				

Note: SA proceeds sample size adjusted fit statistics and * indicates the statistic performed better than the others in the study.

While these studies examine the effectiveness of log likelihood based, classification (entropy and entropy penalty) based, and likelihood ratio test based fit statistics in their role of determining the correct number of classes in a mixture distribution, they do not examine the most basic condition of fitting more than one normal distribution to a true single class nonnormal distribution (i.e. the second standard of performance described above). Recently, Peugh and Fan (2013) compared the performance of the AIC, CAIC, BIC, D – BIC, HQ, HT – AIC, CLC, NEC, ICL – BIC, aLMR – LRT, and the BLRT using true k = 1 and true k = 3 models. (Also included were sample size adjusted versions of the information criteria fit statistics.) In their simulation, they varied sample size (300, 600, 900, 1200, 1500, and 3000), Mahalanobis distance—separation of the multivariate means of the generated distributions creating the overall distribution—(.50, .80, 1.20, and 2.00), and dimensionality of the data (4 and 8 variables) with 200 replications for each of the 48 conditions. In the true k = 1 condition they fit models with 1, 2, and 3 components with homogeneous variances and then with heterogeneous variances. With the homogeneous variances the fit statistics that use entropy as a penalty—CLC, ICL – BIC, and the sample size adjusted ICL – BIC—correctly identified the true k -1 model 100% of the time except for the N = 3000 condition where they were 0% correct. All of the information criteria fit statistics failed (0% correct) to identify the correct model. When the variances of the component distributions were heterogeneous the fit statistics that use entropy as a penalty correctly identified the true k -1 model 100% of the time across all conditions. Additionally, the CAIC, sample size adjusted CAIC, BIC, and D – BIC also correctly identified the correct model 100% of the time across all conditions. Oddly, Peugh and Fan (2013) did not use the NEC, aLMR – LRT, and the BLRT for the *k* = 1 true model citing the undefined nature of the NEC and the boundary limitations associated with the aLMR – LRT and the BLRT. However, each of these statistics has been shown to be useful in identifying a correct *k* = 1 model (Biernacki, Celeux, & Govaert, 1999; Lo, Mendell, & Rubin, 2001; McLachlan, 1987). Additionally, Peugh and Fan (2013) did not examine what effect different levels of skew and kurtosis of the overall distribution would have on the accuracy of the fit statistics.

Li, Cohen, Kim, and Cho (2009) investigated the AIC, BIC, PBF, DIC, and PPMC using IRT models with Markov chain Monte Carlo (MCMC) estimation. (The versions of the AIC and BIC used in this study are the equivalent forms created by Congdon (2003) for use with MCMC model estimation.) They used 1PL, 2PL, and 3PL mixture models to examine true one class, true two classes, true three classes, and true four classes conditions with tests of 15 and 30 items and sample sizes of 600 and 1200, Each of the 48 models were fit with mixtures from one to five classes. They found that the BIC correctly identified the true number of classes on all replications for the true one, two, and three class conditions across all other conditions and was nearly perfect in the four class condition except for the small sample size (600) 3PL model where it performed poorly. Regardless, the BIC outperformed all of the other fit statistics by a wide margin and the authors recommend its use with the one, two, and three parameter logistic mixture models. While their findings support the use of BIC for determining the correct number of classes, it should be noted that the results of their simulation study are based on only 30 replications.

Bauer and Curran (2003a) examined the BIC, AIC, CAIC, ICL-BIC, NEC, and the CLC using a true $k = 1$ compared with a $k = 2$ alternative mixture model. Three conditions of non-normality were created by manipulating skewness and kurtosis (denoted skewness : kurtosis). Their choices were (0:0, 1:1, 1.5:6). Their finding showed that the ICL-BIC outperformed the other fit statistics. However, in the mixture model condition where skew = 1 and kurtosis = 1 for $n = 200$ within class variance allowed to vary the ICL-BIC selected the incorrect $k = 2$ solution 69.60% of the time and 92.51% when skew = 1.5 and kurtosis = 6. The problem became worse for $n = 600$ where the ICL-BIC, while still the best criterion, selected the incorrect $k = 2$ solution 91.08% of the time and 99.18% respectively (Bauer & Curran, 2003a).

Muthén (2003) acknowledges the limitations of the BIC, and the other criteria that use the same basic information (i.e. the log likelihood), to distinguish between the two possibilities that a distribution contains more than one class or is one class nonnormal and suggests the use of additional covariates. Bauer and Curran (2003a) also suggest that additional covariates in the model could increase between class separations and improve the ability of the fit indices to identify the correct number of classes. Lubke and Muthén (2007) showed that the inclusion of covariates improved correct class assignment in a FaFMM. Of course, this assumes that the researcher has access to data regarding potentially salient covariates that were left out from the original model specification. Further, Tofighi and Enders (2007) found that the inclusion of covariates reduced the ability of the fit indices to correctly identify the number of classes, particularly in sample sizes less than 1000. Therefore, adding covariates may not help distinguish the correct model because model complexity is another factor that must be considered.

In practice, the importance of substantive theory in selecting the correct number of classes is not disputed (Bauer & Curran, 2003a; Muthén, 2003). However, as Bauer and Curran noted, when competing models fit the data equally well, it becomes extremely difficult to determine empirically the true nature of the distribution; particularly in the absence of strong theoretical justifications.

Given the very recent implementation of fitting multivariate skew t distributions, there are no studies examining the efficacy of fit statistics in identifying the correct number of classes when skew t component distributions are fit to an overall distribution. Additionally, of the three studies that focus on fitting skew t distributions, two (Asparouhov & Muthen, 2014; Muthen & Asparouhov, 2014) focus on growth mixture models rather than finite mixture models and limit the fits statistics to the AIC and BIC. Lee and McLachlan (2014) also limited the fit statistics to the AIC and BIC, but within a finite mixture context. They fit models to real data with known $k = 2$ components and did not address the issue of a true $k = 1$ distribution. Researchers who employ FaFMMs also tend to limit the selection of fit statistic to one of the information criteria statistics—AIC, BIC, etc. and the one study that specifically examined the identification of spurious classes within IRT—Alexeev, Templin, and Cohen (2011)—used only the BIC to determine model selection.

Therefore, the purpose of this study is to explore the extent to which finite mixture models identify spurious classes. This simulation study examines conditions when the observed data forms a single-component nonnormal distribution. Mixture models with one to four components will be fitted and log likelihood based, classification (entropy and entropy penalty) based, and likelihood ratio based fit statistics will be employed to identify the best fitting model.

Chapter 3 Method

Data were simulated with using R and models were run in Mplus 7.3 using the R package Mplus Automation. The study involves six conditions: (a) type of fit statistic, (b) type of component distribution, (c) type of component covariance structure, (d) sample size, (e) degree of dimensionality, and (f) degree of skewness and kurtosis. These conditions can be described generally as analysis conditions or data conditions.

Analysis Conditions

The study uses eleven different fit statistics from three general categories, log likelihood based, classification based, and likelihood ratio based. The log likelihood based fit statistics are, AIC, BIC, CAIC, SABIC, and SACAIC. The classification (entropy and entropy penalty) based fit statistics are, NEC, CLC, and ICL-BIC. Mplus reports entropy as relative entropy in its output. Relative entropy can be expressed as,

$$REN(\hat{\tau}) = 1 - \frac{EN(\hat{\tau})}{N \log k} \qquad 3.1$$

where $EN(\hat{\tau})$ is entropy, N is the sample size, and k is the number of classes in the model. Since the entropy estimate is required for the calculation of the classification based fit statistics, entropy was calculated from relative entropy using the equation,

$$EN(\hat{\tau}) = \left(1 - REN(\hat{\tau})\right) N \log k \qquad 3.2$$

during the parameter extraction and fit statistic calculation processes. The final three fit statistics are the likelihood ratio based BLRT, LMR-LRT, and aLMR-LRT. Since McLachlan (1987) asserts that the BLRT is only appropriate for models that fit normally distributed component distributions, the fit statistic was not calculated for the nonnormal component distribution conditions. Three types of component distributions: multivariate normal, multivariate skew normal, and multivariate restricted skew t were used in the analysis. The final analysis condition employed two types of covariance structure. The first type allows the covariance matrix to differ among the individual class distributions. The second type freely estimated the covariance matrices, but constrained the matrices to be the same across classes. Table 2 summarizes the analysis conditions.

Table 2.
*Summary of analysis conditions.**

	Component Distribution	Component Covariance
A1	Multivariate Normal	Unconstrained
A2	Multivariate Normal	Constrained
A3	Multivariate Skew Normal	Unconstrained
A4	Multivariate Skew Normal	Constrained
A5	Multivariate Skew t	Unconstrained
A6	Multivariate Skew t	Constrained

*Analysis conditions A1 and A2 are evaluated by all eleven fit statistics. However, analysis conditions A3 to A6 are evaluated with only ten fit statistics – no BLRT.

Data Conditions

Data are simulated with *N* = 500 and *N* = 2,000 cases. There are four and eight dimensions simulated to have a correlation of .52 among the variables. Population skewness and kurtosis are 0.00:0.00, 0.75:0.25, and 1.00:1.00. These values were selected to be consistent with values encountered in applied settings and for comparability to other research on mixture modeling. Table 3 summarizes the data conditions.

Table 3.

Summary of data conditions.

	N	Dimensions	skew:kurtosis
C1	500	4 variables	0.00:0.00
C2	500	4 variables	0.75:0.25
C3	500	4 variables	1.00:1.00
C4	2000	4 variables	0.00:0.00
C5	2000	4 variables	0.75:0.25
C6	2000	4 variables	1.00:1.00
C7	500	8 variables	0.00:0.00
C8	500	8 variables	0.75:0.25
C9	500	8 variables	1.00:1.00
C10	2000	8 variables	0.00:0.00
C11	2000	8 variables	0.75:0.25
C12	2000	8 variables	1.00:1.00

Two frequently used methods for simulating nonnormal data are the one proposed by Vale and Maurelli (1983) and the method proposed by Headrick (2002). There is some controversy over which method performs best and their relative performance may differ based on the degree of nonnormality being simulated (Olvera Astivia & Zumbo, 2014). Therefore, prior to simulating the data, both methods were tested using the 1.00:1.00 level of the skew and kurtosis. (Simulation code for the Vale and Maurelli (1983) method was adapted from code written by Cengiz Zopluoglu (2011) and the simulation code for the Headrick (2002) method was obtained through a link provided in Olvera Astivia and Zumbo (2014)). The results of the test indicated that the Vale and Maurelli method performed better than the Headrick method. Therefore, the Vale and Maurelli was used to simulate the data – 250 data sets for each data condition. A check of the simulated data showed that for most of the conditions on average target values of skew and kurtosis for the variables were achieved when using a criterion of 0.05 deviation from the median and mean centrality statistics. The exceptions to meeting this criterion occurred with the kurtosis parameter when the sample size was 500. In the four variable condition one variable had a – 0.06 deviation from the target median. With a kurtosis of 0.25 two of the variables had deviations of – 0.07 and – 0.06 from the target median value. When target kurtosis was 1.00 with four variables, deviations that exceeded the criterion were, - 0.15, - 0.22, - 0.12, and – 0.14 from the median and two variables had deviations of – 0.8 and – 0.10 from the target mean. In the eight variable condition, with a target kurtosis of 0.25, five variables exceeded the criterion deviation from the median. The average deviation from the median for the five variables is – 0.07 with the largest being – 0.08 and the smallest – 0.051. Additionally, only one of the eight variables in the 0.25 kurtosis condition had a deviation from the mean greater than the criterion, - 0.053. When the target kurtosis was 1.00 all eight variables exceeded the 0.05 criterion from the median and five variables exceeded the criterion for the mean. The average deviation from the target median was – 0.14 with the largest deviation of – 0.23 and smallest of – 0.08. For the five variables that exceeded the criterion for the target mean the average deviation was – 0.08 with a largest deviation of – 0.09 and smallest deviation of – 0.06. While mathematical adjustments to the simulation syntax code could have been applied to reduce these deviations, it was determined that for the sake of comparability and reproducibility that no adjustments should be made. Tables for the full simulation check and comparison between the two simulation methods as well as the R syntax code for the simulation methods can be found in Appendix A.

Summary of Conditions

Fully crossing all analysis and data conditions results in 72 different combinations for analysis that are evaluated with the fit statistics. For instance, analysis condition one (A1) crossed with data condition one (C1) produces the condition A1C1 that fits multivariate normal component distributions with unconstrained class covariance matrices to data with 500 cases and four variables whose distribution was simulated to have a skew of 0.00 and kurtosis of 0.00. Similarly, analysis condition six (A6) crossed with data condition twelve (C12) produces the condition A6C12 that fits multivariate skew t component distributions with class covariance matrices constrained to be equal to data with 2000 cases and eight variables whose distribution was simulated to have a skew of 1.00 and kurtosis of 1.00.

Each data condition was replicated 250 times and one to four components were fitted to each condition's distribution. Initially the number of component distributions that were to be fit was much higher. However, because of the additional computation required for the models and a substantial nonconvergence with these models, it was decided that four component models were sufficient because this number meets or exceeds the number of component distributions used in previous efficacy studies. Additionally, because of the amount of time required to calculate estimates from models fitting multivariate skew t component distributions, these conditions were limited to 125 replications. Log-likelihood based, classification (entropy and entropy penalty) based, and likelihood ratio based fit statistics were employed to identify the best fitting model. Additionally, for all models, each analysis is conducted using 500 random starts of 20 iterations to help avoid the problem of the EM algorithm resolving on local maxima as well as improving the probability of obtaining model convergence, particularly when fitting the four component models. The percentage of times each fit statistic identified a model with *k* number of components is computed, where *k* = 1, 2, 3, and 4. The outcome of interest is the percentage of times that the correct one class model fit the data best. For each of the 72 conditions the percent number of replications that each fit statistic identified each of the one through four class models as the correct model was calculated. These were then combined into a single data set for analysis. Therefore, each fit statistic in the data set has a distribution of percentages for the one through four class solutions. Discussion of the results focuses on the distribution of the identification of the correct one class model. The mean of the distributions for each fit statistic is the average percent correct across the 72 conditions. This can also be interpreted as the mean percent correct across 15,000 replications. [(24 normal + 24 skew normal component distribution conditions x 250 replications = 12,000 replications) + (24 skew t component distribution conditions x 125 = 3,000 replications) = 15,000 total replications] As noted above, R was used to generate the data. The R package Mplus Automation is used to run the models in Mplus 7.3. Samples of Mplus Automation createModels syntax files, the runModels syntax, and the extractModelSummaries syntax are found in Appendix B.

Chapter 4: Results

There are 72 fully crossed conditions fit with four separate class models evaluated by 11 fit statistics in the normal distribution conditions and ten fit statistics in both of the nonnormal distribution conditions. Across all data sets this results in 624,000 fit statistic calculations that took over 25,000 computation hours to make. The first step in reducing the complexity of the results was to calculate the percent number of times each fit statistic preferred each of the one through four class models in each of the 72 analysis by data conditions. The second step in reducing the complexity of the results was to run an ANOVA model using the percent correct identification of the true one class model as the dependent variable and each of the conditions (fit statistic, sample size, skew and kurtosis, number of variables, component distribution type, and component covariance type) as fixed factors. A main effects only model was chosen because of the large number of possible interaction effects – 57. The results of the main effects ANOVA and R^2 effect sizes are presented in table 4. The results show that the skew and kurtosis condition is statistically significant and accounts for approximately 33% of the variance in percent correct true one class model. Additionally, type of fit statistic is also statistically significant and accounts for approximately 27% of the variance in percent correct true one class model. The type of component distribution, while statistically significant, only explains an additional 1.4% of the variance. Sample size, number of variables, and component covariance type were not statistically significant and the three effects combined accounted for less than 0.5% of the variance in percent correct true one class model. In sum, this analysis suggests that evaluation of the results should focus on the skew and kurtosis condition, the fit statistic condition, and the type of component distribution condition.

Table 4.

Main effects and effect sizes from fixed effects ANOVA using percent correct true one class model as the dependent variable.

	df	*SS*	*MS*	*F*	*p*	R^2
Fit Statistic	10	403359	40336	53.39	< .001	.267
Sample Size	1	2766	2766	3.66	.056	.002
Skew and Kurtosis	2	493884	246942	326.84	< .001	.327
Variables	1	2748	2748	3.64	.057	.002
Component Distribution	2	20860	10430	13.80	< .001	.014
Component Covariance	1	316	316	0.42	.518	< .001
Residuals	774	584801	756			

Skew and Kurtosis

The skew and kurtosis condition has three levels: skew = 0.00 and kurtosis = 0.00, skew = 0.75 and kurtosis = 0.25, and skew = 1.00 and kurtosis = 1.00. Table 5 shows the distribution of percent correct identification of the true one class model by skew and kurtosis condition. The table displays the mean, standard deviation, and five number summaries of the distribution of the percent correct true one class model for the three levels of the skew and kurtosis condition. The statistics in table 5 show that as the degree of skew and kurtosis increases, the identification of the true one class model decreases. For instance, the mean for the skew and kurtosis level of 0.00:0.00 is 86.33% correct but the mean for the skew and kurtosis level of 0.75:0.25 is 36.09% correct. Additionally,

Table 5.

Distribution of percent correct identification of the true one class model by skew and kurtosis condition. (N = 248)

	M	*SD*	*Min*	*Q1*	*MDN*	*Q3*	*Max*
0.00:0.00	86.33	26.49	0.00	87.30	99.20	100.00	100.00
0.75:0.25	36.09	39.30	0.00	0.00	17.40	73.30	100.00
1.00:1.00	25.34	35.92	0.00	0.00	5.60	44.20	100.00

at the median the correct one class model was identified 99.20% of the time at the skew and kurtosis level of 0.00:0.00; however, at the 0.75:0.25 level the median was only 17.40% correct identification of the one class model. Closer examination of the frequencies of correct identification of the one class model revealed that in the 0.00:0.00 level 100% correct identification occurred in 46.37% (*n* = 115) of the 248 data conditions where this level of skew and kurtosis was simulated. Additionally, over half 67.34% (*n* = 167) of the data conditions with the 0.00:0.00 level of skew and kurtosis had a correct identification of 95% or greater. At the other end of the distribution 0% correct identification occurred in 1.21% (*n* = 3) and 4.84% (*n* = 12) had a correct identification rate of 5% or less. At the 0.75:0.25 level of skew and kurtosis100% correct identification occurred in 8.47% (*n* = 21) of the 248 data conditions where this level of skew and kurtosis was simulated. Additionally, 16.53% (*n* = 41) of the data conditions with the 0.75:0.25 level of skew and kurtosis had a correct identification of 95% or greater. At the other end of the distribution for this level of skew and kurtosis 0% correct identification occurred in 34.68% (*n* = 86) 43.95% (*n* = 109) had a correct identification rate of 5% or less. For the 1.00:1.00 level of skew and kurtosis100% correct identification occurred in

4.84% (*n* = 12) of the 248 data conditions where this level of skew and kurtosis was simulated. Additionally, 11.69% (*n* = 29) of the data conditions with the 1.00:1.00 level of skew and kurtosis had a correct identification of 95% or greater. At the other end of the distribution for this level of skew and kurtosis 0% correct identification occurred in 40.32% (*n* = 100) and almost half 49.60% (*n* = 123) had a correct identification rate of 5% or less. Overall, these results show that higher skew and kurtosis values inhibit the ability to identify a true one class model. Of course, in this study model selection is based solely upon the fit statistics, which is the next largest source of variation accounting for the percent correct one class model identification.

Fit Statistics

The fit statistic condition has eleven categories: Akaike's information criterion (AIC), Bayesian information criterion (BIC), consistent AIC (CAIC), sample size adjusted BIC (SABIC), sample size adjusted CAIC (SACAIC), normalized entropy criterion (NEC), classification likelihood criterion (CLC), integrated classification likelihood BIC (ICL-BIC), bootstrap likelihood ratio test (BLRT), Lo, Mendell, and Rubin likelihood ratio test (LMR-LRT), and adjusted LMR (aLMR-LRT). Table 6 shows the distribution of percent correct identification of the true one class model by fit statistic. The table groups the fit statistics into their respective categories of information criteria based, classification based, and likelihood ratio based test statistics. It displays the mean, standard deviation, and five number summaries of the distribution of the percent correct true one class model for the eleven different fit statistics. Each of the fit statistics was used in each of the 72 analysis by data conditions with the exception of the BLRT, which is only calculated when fitting models with normally distributed components. Table 6 shows that across all analysis and data conditions the classification based fit statistics

Table 6.

Distribution of percent correct identification of the true one class model by fit statistic. (N = 72)

	M	*SD*	*Min*	*Q1*	*MDN*	*Q3*	*Max*
Information Criteria							
AIC	13.68	24.03	0.00	0.00	0.00	13.90	79.20
BIC	43.18	46.39	0.00	0.00	8.80	100.00	100.00
CAIC	46.16	46.75	0.00	0.00	22.60	100.00	100.00
SABIC	32.72	43.12	0.00	0.00	0.00	85.00	100.00
SACAIC	37.14	46.15	0.00	0.00	0.80	99.90	100.00
Classification							
NEC	70.42	35.73	0.00	45.20	88.00	98.40	100.00
CLC	71.06	35.98	0.00	89.30	100.00	100.00	100.00
ICL-BIC	88.22	23.75	6.40	89.30	100.00	100.00	100.00
Likelihood Ratio							
BLRT*	30.67	44.42	0.00	0.00	0.00	89.80	99.20
LMR-LRT	47.91	36.54	0.00	11.40	46.00	83.70	100.00
aLMR-LRT	48.25	36.64	0.00	11.40	46.00	85.20	100.00

*Statistics for the BLRT are from the conditions fitting normal distributions (*n* = 24).

outperformed the log likelihood and likelihood ratio test based fit statistics. The ICL-BIC outperformed all other fit statistics (*M* = 88.22%). It was the only fit statistic that was able to correctly identify the true one class model in all 72 analysis by data conditions – *Min* = 6.40% correct identification. At the first quartile of the percent correct identification distribution it had an 89.30% that rose to 100% correct identification at the median of the distribution. A closer examination of the distribution showed that in 55.56% (*n* = 40) of the analysis by data conditions the ICL-BIC identified the correct one class model 100% of the time. Additionally, in 72.22% (*n* = 52) of the analysis by data conditions the ICL-BIC had a 98.4% or greater accuracy in identifying the correct one class model. The next most accurate fit statistic was the CLC with a mean of 71.06 on the percent correct identification distribution. While the quartile values for the CLC are identical to the quartile values of the ICL-BIC, a closer examination of the CLC distribution of percent correct identification revealed that in 18.06% (*n* = 13) of the analysis by data conditions the CLC identified the correct one class model 100% of the time. Additionally, in 33.33% (*n* = 24) of the analysis by data conditions the CLC's accuracy was 98.4% and in 47.22% (*n* = 34) of the conditions the CLC had a greater than 95% accuracy in identifying the correct one class model. At the other end of the distribution for the CLC 0% correct identification occurred in 6.94% (*n* = 5) and 11.11% (*n* = 8) had a correct identification rate of less than 5%. These results show that the second best fit statistic – the CLC – had three times fewer instances of 100% and two times fewer instances of 98.4% or greater correct identification than the most accurate fit statistic – the ICL-BIC. The final classification fit statistic – the NEC – performed worse than the ICL-BIC and had slightly worse but very similar performance compared to the CLC. The NEC had a mean of 70.42 on the percent correct identification distribution. At the first quartile the NEC had a much smaller 45.20% correct than the 89.30% correct identification level in the first quartile of the other two entropy statistics. Additionally, whereas the ICL-BIC and the CLC reached the level of 100% correct at the median of their distributions, this level of accuracy was only attained at the maximum of the NEC percent correct distribution. Further examination of the NEC distribution revealed that in 15.28% (*n* = 11) of the analysis by data conditions the NEC identified the correct one class model 100% of the time. Additionally, in 29.17% (*n* = 21) of the analysis by data conditions the NEC's accuracy was 98.4% and in 45.83% (*n* = 33) of the conditions the NEC had a greater than 95% accuracy in identifying the correct one class model. At the other end of the distribution for the NEC 0% correct identification occurred in 6.94% (*n* = 5) and 9.72% (*n* = 7) had a correct identification rate of less than 5%. Although the NEC had, by a slight margin, the poorest performance of the classification fit statistics, it still outperformed the information criteria and likelihood ratio statistics.

Likelihood ratio fit statistics. The aLMR-LRT had a slightly higher mean (*M* = 48.25%) than the LMR-LRT (*M* = 47.91%) on the percent correct identification distribution. Their respective five number summaries were identical up to quartile three where the aLMR-LRT (85.20) was slightly higher than the LMR-LRT (83.70). However, closer examination of the distributions of percent correct identification revealed that the upper and lower ends of the respective distributions were the same for both statistics. In 5.56% (*n* = 4) of the analysis by data conditions they identified the correct one class model 100% of the time. Additionally, in 8.33% (*n* = 6) of the analysis by data conditions their accuracy was 98.4% and in 16.67% (*n* = 12) of the conditions had a greater than 95% accuracy in identifying the correct one class model. At the other end of the distributions, 0% correct identification occurred in 2.78% (*n* = 2) and 13.89% (*n* = 10) had a correct identification rate of less than 5%. Overall, there was very little difference between the LMR-LRT and the aLMR-LRT with the aLMR-LRT having a slightly better performance in identifying the correct one class model. The final likelihood ratio fit statistic is the BLRT. The BLRT is only calculated for models fitting normal component distributions. Therefore, the percent correct identification distribution is a third the size (*n* = 24) of the distributions for the other fit statistics. Nonetheless, the BLRT had a 0% at the median of it correct identification distribution and did not achieve 100% accuracy on any of the analysis by data conditions (*Max* = 99.20%). Further inspection of the distribution revealed that 16.67% (*n* = 4) had greater than 95% accuracy in identifying the correct one class model. At the other end of the distribution a 0% correct identification occurred in 58.33% (*n* = 14) of the conditions and 66.67% (*n* = 16) had less than 5% correct. This pattern of inaccuracy is only exceeded by the information criterion AIC.

Information criteria fit statistics. When looking at the overall distribution of percent correct identification the best performing information criteria fit statistic is the CAIC. It had a mean of 46.16% and an accuracy of 22.60% correct at the median of its distribution that rose to 100% at the third quartile. A closer inspection of the distribution revealed that in 33.33% (*n* = 24) of the analysis by data conditions the CAIC was 100% correct in identifying the true one class model and 36.11% (*n* = 26) had greater than 95% accuracy. At the other end of the distribution 30.56% (*n* = 22) had 0% correct and 43.06% (*n* = 31) had a correct identification accuracy of less than 5%. The BIC performed worse than the CAIC. The BIC had a mean of 43.18% and an accuracy of just 8.80% correct at the median of its distribution that rose to 100% at the third quartile. A closer inspection of the distribution revealed that in 33.33% (*n* = 24) of the analysis by data conditions the BIC was 100% correct in identifying the true one class model and the next highest percent was a single condition with a 94.40% accuracy. At the other end of the distribution 38.89% (*n* = 28) had 0% correct and 44.44% (*n* = 32) had a correct identification accuracy of less than 5%. The SACAIC and the SABIC performed worse than, but with a similar pattern to, their nonsample size adjusted counter parts. The SACAIC had a mean of 37.14% and an accuracy of just 0.80% correct at the median of its distribution that rose to 99.90% at the third quartile. A closer inspection of the distribution revealed that in 25.00% (*n* = 18) of the analysis by data conditions the SACAIC was 100% correct in identifying the true one class model and 31.94% (*n* = 23) had greater than 95% accuracy. At the other end of the distribution 47.22% (*n* = 34) had 0% correct and over half 54.17% (*n* = 39) had a correct identification accuracy of less than 5%. The SABIC performed worse than the SACAIC. The SABIC had a mean of 32.72% and an accuracy of 0% correct at the median of its distribution that rose to 85.00% at the third quartile. Closer inspection of the distribution revealed that in 13.89% (*n* = 10) of the analysis by data conditions the SABIC was 100% correct in identifying the true one class model and 23.61% (*n* = 17) had greater than 95% accuracy. At the other end of the distribution over half 51.39% (*n* = 37) of the conditions had 0% correct and over half 54.16% (*n* = 39) had a correct identification accuracy of less than 5%. By far the worst performing fit statistic was the AIC. It had a mean of 13.68% and an accuracy of only 13.90% at the third quartile and a maximum 79.20% (*n* = 1). In fact, a closer inspection of the distribution revealed that in only 13.89% (*n* = 10) of the analysis by data conditions did the AIC exceed 50% correct in identifying the true one class model. At the other end of the distribution over half 55.56% (*n* = 40) had 0% correct and over two thirds 68.06% (*n* = 49) had a correct identification accuracy of less than 5%. These results taken as a whole allow us to construct a ranking of fit statistics based on their ability to identify the correct one class model.

Summary ranking of fit statistics. When we examine the distributions of percent correct from table 6, we can see a ranking from best to worst: ICL-BIC, CLC, NEC, aLMR-LRT, LMR-LRT, CAIC, BIC, SACAIC, SABIC, BLRT, and AIC. However, if we look at 100% accuracy only, a different ranking emerges: ICL-BIC (55.56%, n = 40), CAIC (33.33%, n = 24), BIC (33.33%, n = 24), SACAIC (25.00%, n = 18), CLC (18.06%, n = 13), NEC (15.28%, n = 11), SABIC (13.89%, n = 10), aLMR-LRT (5.56%, n = 4), LMR-LRT (5.56%, n = 4). In this view of the results the CAIC and the BIC are tied in rank as are the LMR-LRT and its adjusted form aLMR-LRT while the BLRT and AIC are not ranked because they failed to meet the 100% criterion. While the CLC, NEC, aLMR-LRT, and LMR-LRT, except for the tie between the aLMR-LRT and LMR-LRT, maintained the relative rank amongst themselves, they have been moved down in rank by the CAIC, BIC, SACAIC and the SABIC has been ranked between them. This indicates that changing the analysis by data conditions has a much more drastic effect on the ability of the information criteria fit statistics to accurately identify the correct one class model. In essence, they are more consistent in succeeding or failing to identify the correct model across all the replications in a condition. One data condition that can influence whether the correct model is identified is nonnormality. The analysis condition that could account for nonnormality is component distribution type, which is the third largest source of variation explaining the percent correct true one class model.

Component Distributions

The component distribution condition has three levels: multivariate normal, multivariate skew normal, and multivariate skew t. Table 7 shows the distribution of percent correct identification of the true one class model by component distribution condition. The table displays the mean, standard deviation, and five number summaries of the distribution of the percent correct true one class model for the three levels of the component distribution condition. The statistics in the table have been adjusted for no estimation of the BLRT in the nonnormal component distribution conditions. The statistics in table 7 show that the more the component distributions are allowed to vary their shape the identification of the true one class model increases. For instance, the mean for the normal distribution is 41.78% correct but the mean for the skew normal

Table 7.

*Distribution of percent correct identification of the true one class model by distribution condition.**

	M	*SD*	*Min*	*Q1*	*MDN*	*Q3*	*Max*
Normal	41.78	44.15	0.00	0.00	20.00	95.20	100.00
Skew	47.94	41.16	0.00	1.60	44.00	97.60	100.00
Skew T	59.80	43.13	0.00	5.60	80.00	100.00	100.00

*Statistics adjusted for no BLRT estimation in the nonnormal distribution conditions: normal (*N* = 264), skew and skew t (*N* = 240).

distribution is 47.94% correct. Additionally, at the first quartile the correct one class model was identified 0% of the time in the normal component distribution condition; however, when the skew normal component distribution was used the correct identification at the first quartile was 1.60%. Closer examination of the frequencies of correct identification of the one class model revealed that in the normal component distributions 100% correct identification occurred in 15.91% (*n* = 42) of the 264 analysis by data conditions that fit this type of distribution. Additionally, 25.38% (*n* = 67) of the analysis conditions that fit normal component distributions had a correct identification of 95% or greater. At the other end of the distribution 0% correct identification occurred in 36.74% (*n* = 97) and 45.08% (*n* = 119) had a correct identification rate of 5% or less. When skew normal component distributions were fit 100% correct identification occurred in 16.67% (*n* = 40) of the 240 analysis by data conditions. Additionally, 27.50% (*n* = 66) of the analysis conditions that used the skew normal component distributions had a correct identification of 95% or greater. At the other end of the distribution for this type of component distribution 0% correct identification occurred in 18.75% (*n* = 45) and 27.50% (*n* = 66) had a correct identification rate of 5% or less. For the conditions where the skew t component distribution were fit, 100% correct identification occurred in 27.50% (*n* = 66) of the 240 analysis by data conditions. Additionally, 42.50% (*n* = 102) of the conditions that fit skew t component distributions had a correct identification of 95% or greater. At the other end of the distribution for this type of component distribution 0% correct identification occurred in 19.58% (*n* = 47) and 24.58% (*n* = 59) had a correct identification rate of 5% or less. Overall, these results show that estimating additional parameters that allowed the shape of the component distributions to change increased the ability to identify a true one class model. However, as the initial ANOVA analysis showed, component distribution type accounted for only 1.4% of the variance in percent correct identification of the true one class model. Of course, when evaluating the ability to identify the true one class model in this study, and in any study, that requires fitting multiple models on multiple data sets, it is important to examine model convergence.

Model Convergence

It is expected that less complex models will converge at a higher rate than more complex models. In other words, models fitted with fewer class component distributions are more likely to converge than models fitting more classes. Therefore, in this study it was expected that for each data replication the one class model would have the highest convergence rate and that the subsequent two through four class models would have decreasing rates of convergence. This expectation implies that if the one class model fails to converge on any given data replication, all of the subsequent two through four class models fitted to that data replication will also fail to converge. In practice the failure to converge of a more complex model is often taken as evidence in support of preference being given to a less complex model. To a certain extent this study follows this method of preference because, for each data replication, the best fitting model is selected based on the models that converged for that replication. However, in some of the analysis by data conditions in this study, the expected pattern of convergence was not achieved.

The majority of the analysis by data conditions followed the expected pattern of convergence where the less complex one class models converged at a higher rate than the more complex two through four class models. However, in 27.78% (n = 20) of the conditions the true one class model failed to converge in at least one replication. Table 8 shows the condition identifier, the percent that the one class model converged, and the condition characteristics for the conditions where the one class model failed to converge on at least one replication. Tables C1 and C2 in Appendix C provide convergence rates

Table 8.
Percent convergence of one class model when 100% convergence was not achieved by condition and condition characteristics.

	%	Distribution	Covariance	*N*	Variables	Sk:Ku
A3C6	99.60	Skew	Unconstrained	2000	4	1.00:1.00
A4C6	99.60	Skew	Constrained	2000	4	1.00:1.00
A3C5	98.40	Skew	Unconstrained	2000	4	0.75:0.25
A4C5	98.40	Skew	Constrained	2000	4	0.75:0.25
A5C2	98.40	Skew t	Unconstrained	500	4	0.75:0.25
A5C3	92.00	Skew t	Unconstrained	500	4	1.00:1.00
A3C2	81.20	Skew	Unconstrained	500	4	0.75:0.25
A4C2	81.20	Skew	Constrained	500	4	0.75:0.25
A3C8	73.60	Skew	Unconstrained	500	8	0.75:0.25
A4C8	73.60	Skew	Constrained	500	8	0.75:0.25
A3C11	68.40	Skew	Unconstrained	2000	8	0.75:0.25
A4C11	68.40	Skew	Constrained	2000	8	0.75:0.25
A3C3	55.20	Skew	Unconstrained	500	4	1.00:1.00
A4C3	54.80	Skew	Constrained	500	4	1.00:1.00
A5C8	48.00	Skew t	Unconstrained	500	8	0.75:0.25
A3C12	46.80	Skew	Unconstrained	2000	8	1.00:1.00
A4C12	46.80	Skew	Constrained	2000	8	1.00:1.00
A3C9	14.80	Skew	Unconstrained	500	8	1.00:1.00
A4C9	14.80	Skew	Constrained	500	8	1.00:1.00
A5C9	6.40	Skew t	Unconstrained	500	8	1.00:1.00

for all conditions and models in the study. The percent column in table eight indicates that the failure of the one class model to converge ranged from one failure in conditions A3C6 and A4C6 to 213 failures in A3C9 and A4C9. (Condition A5C9 has the lowest percent of one class convergence, but 117 failures because the conditions that fit the multivariate skew t component distribution were limited to 125 replications.) Examination of the distribution column indicates that 80% (*n* = 16) of the conditions where the one class model failed to converge at least once had fit multivariate skew normal component distributions and the other 20% (*n* = 4) of the conditions had fit multivariate skew t component distributions. The type of covariance matrices was more evenly split, 60% (*n* = 12) unconstrained and 40% (*n* = 8) constrained; however, the distinction between the types of covariance structure is irrelevant because in the one class model the covariance matrix is freely estimated in both types of covariance structures. The sample size condition was similarly split, 60% (*n* = 12) of the conditions had 500 cases and 40% (*n* = 8) of the conditions had 2000 cases. The variables condition was evenly split (50%; *n* = 10) between the four variable level and the eight variable level; although the eight variable level is found more frequently at higher rates of failure. The skew and kurtosis condition was evenly split (50%; *n* = 10) between the 0.75:0.25 level and the 1.00:1.00 level; although the 1.00:1.00 level tends to be found more frequently at higher rates of failure. Notably absent from table eight are any conditions that had fit multivariate normal component distribution or any conditions where the data was simulated to have zero skew and zero kurtosis. Additionally, although few patterns emerge when the interaction of the analysis and data conditions present in table eight are considered, it should be noted that when the skew t component distribution appears, it is always paired with the smaller sample size of 500 cases. These findings for the failure of convergence of the one class model have important practical implications.

In the context of practice, if the one class model fails to converge, a researcher could increase the number of random starts in an attempt to get the model to converge.

This is impractical and, more poignantly, unwarranted in the context of a simulation study. If the number of random starts were increased to encourage convergence, this would have to be done for all of the analyses in the simulation study. Changing the random starts analysis condition that was held constant would influence the results for the manipulated conditions in the study. Nonetheless, in practice, fitting a one class model successfully is an important step even if it is only intended as a comparison for more complex multi class models. The results have focused on the percent correct identification of the true one class model. This means that the replications where the one class model failed to converge reduced the percent correct identification by virtue of there being no one class model to select. This should have little effect on the on the relative ranking of the effectiveness of the fit statistics in identifying the correct one class model. However, in practice it may be more important to know the accuracy of the fit statistics when the one class model converged. Additionally, looking only at replications where the one class model converged should decrease the differences associated with levels of skew and kurtosis and increase the differences associated with the type of component distribution. Therefore, the results were reexamined using only the replications where the one class model converged.

One Class Model Converged Replications

Replications where the one class model failed to converge were removed and percent correct identification was recalculated for all fit statistics in all 72 analysis by data conditions. The recalculated percent correct identification then served as the dependent variable in a second ANOVA model. The results of the main effects ANOVA and R^2 effect sizes are presented in table 9. The results show that there were statistically significant effects for all of the conditions except for the number of variables condition. As in the first ANOVA model that used the percent correct identification dependent variable calculated using all replications, skew and kurtosis, fit statistic, and type of component distribution were the main contributors in explaining variation in the dependent variable. However, when using only the replications where the one class model converged, the fit statistic condition accounted for the most variance – 30%. The skew and kurtosis condition accounts for approximately 24% of the variance and the type of component distribution explains an additional 2.5% of the variance. The remaining three conditions combined accounted for less than 2% of the variance.

Table 9.
Main effects and effect sizes from fixed effects ANOVA using percent correct model calculated from replications where the one class model converged.

	df	*SS*	*MS*	*F*	*p*	R^2
Fit Statistic	10	475124	47512	55.88	< .001	.302
Sample Size	1	20847	20847	24.52	< .001	.013
Skew and Kurtosis	2	371940	185970	218.74	< .001	.236
Variables	1	1765	1765	2.08	.150	.001
Component Distribution	2	39544	19772	23.26	< .001	.025
Component Covariance	1	5444	5444	6.40	.012	.003
Residuals	774	658059	850			

Fit Statistics – Replications with One Class Model Convergence

Table 10 shows the distribution of percent correct identification of the true one class model by fit statistic when using only the replications where the one class model converged. The table groups the fit statistics into their respective categories of information criteria, classification based, and likelihood ratio based test statistics. It displays the mean, standard deviation, and five number summaries of the distribution of the percent correct true one class model for the eleven different fit statistics when using only the replications where the one class model converged. Since the BLRT is only calculated when fitting models with normally distributed components and the one class model converged for all replications in the conditions that fit normally distributed components, the BLRT distribution statistics are the same as in table 6 where the

Table 10.
Distribution of percent correct identification of the true one class model by fit statistic when using only replications where the one class model converged. (N = 72)

	M	*SD*	*Min*	*Q1*	*MDN*	*Q3*	*Max*
Information Criteria							
AIC	16.01	26.55	0.00	0.00	0.00	26.53	81.70
BIC	47.42	47.27	0.00	0.00	37.62	100.00	100.00
CAIC	51.13	47.86	0.00	0.00	59.23	100.00	100.00
SABIC	35.40	44.00	0.00	0.00	0.00	90.62	100.00
SACAIC	40.20	47.13	0.00	0.00	0.80	100.00	100.00
Classification							
NEC	75.68	33.83	0.00	60.86	95.43	99.20	100.00
CLC	76.33	34.04	0.00	60.86	100.00	100.00	100.00
ICL-BIC	96.47	11.94	29.20	99.53	100.00	100.00	100.00
Likelihood Ratio							
BLRT*	30.67	44.42	0.00	0.00	0.00	89.80	99.20
LMR-LRT	54.14	37.23	0.00	19.30	51.84	92.56	100.00
aLMR-LRT	54.51	37.27	0.00	19.31	52.80	92.56	100.00

*Statistics for the BLRT are from the conditions fitting normal distributions (*n* = 24).

percent correct one class identification distribution statistics are based on all replications. Table 10 shows that, as expected, all of the fit statistics, except the BLRT, performed better when using only the replications where the one class model converged. Notably, the mean of the ICL-BIC percent correct one class distribution was 96.47%. The distribution had a minimum of 29.20% and by the first quartile it was 99.53% correct. Additionally, as expected, the performance of the fit statistics relative to one another remained the same when examining only the replications where the one class model converged. For instance, on average the aLMR-LRT (*M* = 54.54%) and LMR-LRT (*M* = 54.14%) outperformed the CAIC (*M* = 51.13%). However, at the median and third quartiles of their respective distributions the percent correct one class model identification was higher for the CAIC (*Mdn* = 59.23%, *Q3* = 100.00%) than it was for either the aLMR-LRT (*Mdn* = 52.80%, *Q3* = 92.56%) or the LMR-LRT (*Mdn* = 51.84%, *Q3* = 92.56%). These results suggest that the CAIC is more consistent in its identification, or failure of identification, of the correct one class model on all of the replications within each condition. Examining the upper and lower ends of the percent correct true one class distributions for each of the fit statistics sheds greater light on how consistent the each fit statistic was in identifying, or failing to identify, the correct one class model on the replications within the conditions.

Table 11 shows the percent and number of conditions at the levels of, 100% correct, 95% or greater correct, 5% or less correct, and 0% correct for each fit statistic when using only the replications where the one class model converged. As table 10 shows, the ICL-BIC identified the correct one class model 100% of the time on all replications for 70.83% (*n* = 51) of the 72 analysis by data conditions. Additionally, it had an accuracy 95% or greater for 88.89% (*n* = 64) of the analysis by data conditions. Of the remaining eight conditions, half were above 85% correct and the remaining four conditions were between 76.00% and 29.29% correct identification of the one class model. The other two classification based fit statistics – CLC and NEC - outperform

Table 11.

Percent and number of conditions in the upper and lower ends of the percent correct one class model distribution for each fit statistic when using only thee replications where the one class model converged.(N = 72)

	100%		95% or greater		5% or less		0%	
	%	*n*	*%*	*n*	*%*	*n*	*%*	*n*
Information Criteria								
AIC	0	0	0	0	66.67	48	55.56	40
BIC	36.10	26	40.28	29	44.44	32	38.89	28
CAIC	40.28	29	43.06	31	43.06	31	30.56	22
SABIC	13.89	10	25.00	18	54.17	39	51.39	37
SACAIC	26.39	19	34.72	25	54.17	39	47.22	34
Classification								
NEC	20.83	15	54.17	39	11.11	8	6.94	5
CLC	26.39	19	55.56	40	11.11	8	6.94	5
ICL-BIC	70.83	51	88.89	64	0	0	0	0
Likelihood Ratio								
BLRT*	0	0	16.67	4	66.67	16	58.33	14
LMR-LRT	8.33	6	20.83	15	15.28	11	2.78	2
aLMR-LRT	8.33	6	20.83	15	15.28	11	2.78	2

*Statistics for the BLRT are from the conditions fitting normal distributions (*n* = 24).

the information criteria and likelihood ratio test based fit statistics when looking at 95% or greater accuracy in identifying the correct one class model within the conditions.

However, with the exception of the SABIC and the AIC, the information criteria based fit statistics have an equal to or higher percent and number of conditions where they were 100% accurate in identifying the correct one class model than either the CLC or the NEC. Additionally, the information criteria based fit statistics, with the exception of the AIC, had higher a percent and number of conditions in both the 100% and 95% or greater levels than did any of the likelihood ratio based statistics. However, at the other end of their respective distributions, with the exception of the BLRT, the information criteria based fit statistics had higher a percent and number of conditions in both the 5% or less and 0% levels than did any of the classification based or likelihood ratio based fit statistics. These results show that the information criteria fit statistics are more consistent in identifying, or failing to identify, the correct one class model across replications within a condition. Of course the other conditions influence the ability of the fit statistics to identify the correct one class model. When analyzing only the replications where the one class model converged, the skew and kurtosis condition accounted for the most variance (R^2 = .236) in percent correct one class model identification after the fit statistic condition (R^2 = .302).

Skew and Kurtosis – Replications with One Class Model Convergence

Table 12 shows the distribution of percent correct identification of the true one class model by the degree of data nonnormality when using only the replications where the one class model converged. It displays the mean, standard deviation, and five number summaries of the distribution of the percent correct true one class model for the three different levels of skew and kurtosis when using only the replications where the one class model converged. The statistics in table 12 show that as the degree of skew and kurtosis increases, the identification of the true one class model decreases. As would be expected, since all of the one class models converged in the conditions with the skew and kurtosis level of 0.00:0.00, the distribution of the percent correct one class model is essentially identical whether using all replications or just replications where the one class model converged. However, using just the replications where the one class model converged did have an effect on the distribution

Table 12.

Distribution of percent correct identification of the true one class model by skew and kurtosis condition when using only the replications where the one class model converged. (N = 248)

	M	*SD*	*Min*	*Q1*	*MDN*	*Q3*	*Max*
0.00:0.00	86.42	26.50	0.00	87.60	99.20	100.00	100.00
0.75:0.25	41.22	43.43	0.00	0.00	24.00	95.10	100.00
1.00:1.00	34.71	40.72	0.00	0.00	9.00	80.00	100.00

of the percent correct one class model identification in both the 0.75:0.25 level and the 1.00:1.00 level of skew and kurtosis. At the 0.75:0.25 level the mean correct one class model distribution was 5.12% higher when using only the replications where the one class model converged. Additionally, although the minimum and the first quartile were the same, the median was 6.60% higher and the third quartile was 21.80% higher when using only the replications where the one class model converged. Similarly, at the 1.00:1.00 level the mean correct one class model distribution was 9.37% higher when using only the replications where the one class model converged. Additionally, although the minimum and the first quartile were the same, the median was 3.40% higher and the third quartile was 38.50% higher when using only the replications where the one class model converged. Overall, these results show that, although the effect is reduced, higher skew and kurtosis values inhibit the ability to identify a true one class model even under the condition that the one class model converged for the replications. Thus, nonnormality in the data decreases the probability of the fit statistics identifying the true one class model and increases the probability of the fit statistics identifying spurious classes.

Examination of the fit statistics performance at the levels of skew and kurtosis revealed that for the conditions that examined simulated data with skew and kurtosis of 0.00:0.00, only the ICL-BIC, the CAIC, and the BIC were 100% correct on all replications in the conditions. The next best performance was the SACAIC (98.78%), which is followed in order by the BLRT (91.90%), aLMR-LRT (91.20%), LMR-LRT (90.70%), SABIC (88.35%), CLC (80.05%), NEC (78.62%), and AIC (33.77%). In the conditions that examined data simulated at the 0.75:0.25 level of skew and kurtosis, none of the fit statistics were 100% accurate in identifying the correct one class model. However, the ICL-BIC was 100% correct on all replications in 18 of the 24 conditions. Its lowest percent correct was 98.25% on the A3C11 condition and it had an average of 99.77% correct across all of the 24 conditions. The next best performance was the CLC with an average of 83.22% correct across all 24 conditions. The CLC was 100% correct on seven of the conditions and had a 95% or greater performance on an additional seven conditions. Examination of the remaining conditions with less than 95% accuracy, showed that all but one (A1C2) had eight variables and more than half had 500 cases. In the conditions that examined data simulated at the 1.00:1.00 level of skew and kurtosis, none of the fit statistics were 100% accurate in identifying the correct one class model. However, the ICL-BIC was 100% correct on all replications in nine of the 24 conditions and had accuracy 95% or greater in an additional seven conditions. Its lowest percent correct was 29.20% on the A1C12 condition and it had an average of 89.63% correct across all of the 24 conditions. The next best performance was the CLC with an average of 65.47% correct across all 24 conditions. The CLC was 100% correct on seven of the conditions and had a 95% or greater performance on an additional four conditions at this level of data nonnormality. On over half of the conditions, the CLC had less than 95% accuracy. Of these 13 conditions, nine had eight variables and of the 13 condition, nine had 500 cases. Examining the eight conditions where the ICL-BIC was less than 95% accurate revealed that six had sample sizes of 500 cases and none of the conditions fit the skew t component distribution. This suggest that the analysis condition that could account for nonnormality is component distribution type, which is the third largest source of variation explaining the percent correct one class model when using just the replications where the one class model converged.

Component Distributions – Replications with One Class Model Convergence

Table 13 shows the distribution of percent correct identification of the true one class model by component distribution condition when using only the replications where the one class model converged. The table displays the mean, standard deviation, and five number summaries of the distribution of the percent correct true one class model for the three levels of the component distribution condition. The statistics in the table have been adjusted for no estimation of the BLRT in the nonnormal component distribution conditions. The statistics in table 13 show that as the more the component distributions are allowed to vary their shape the identification of the true one class model increases. Since the one class model converged for all replications in the conditions that fit normally distributed components, the distribution statistics for the multivariate normal component distribution are the same as in table 7 where the percent correct one class identification distribution statistics are based on all replications. Table 13 shows that, as expected, when using only the replications where the correct one class model converged, the ability to identify the correct one class model improved for conditions that fit the multivariate skew normal and multivariate skew t component distributions. For instance, the mean for the multivariate skew normal distribution was 47.94% correct when using all replications, but the mean for the multivariate skew normal was 56.33% when using only the replications where the one class model converged. Similarly, the mean for the

Table 13.

*Distribution of percent correct identification of the true one class model by distribution condition when using only replications where the one class model converged.**

	M	*SD*	*Min*	*Q1*	*MDN*	*Q3*	*Max*
Normal	41.78	44.15	0.00	0.00	20.00	95.20	100.00
Skew	56.33	41.92	0.00	1.60	44.00	97.60	100.00
Skew T	64.97	42.87	0.00	5.60	80.00	100.00	100.00

*Statistics adjusted for no BLRT estimation in the nonnormal distribution conditions: normal (N = 264), skew and skew t (N = 240).

multivariate skew t component distribution went from 59.80% correct to 64.97% correct when using only the replications where the one class model converged. Additionally, for the multivariate skew normal components although the minimum was the same, the first quartile was 0.75%, the median was 26.80% higher and the third quartile was 1.55% higher when using only the replications where the one class model converged. Similarly, for the multivariate skew t component distributions although the minimum and the first quartile were the same, the median was 16.00% higher when using only the replications where the one class model converged. In both distributions – all replications and only replications where the one class model converged – the third quartile was 100.00%. Overall, these results show that allowing the component distributions to vary increases the ability of the fit statistics to identify a true one class model and that is enhanced under the condition that the one class model converged for the replications.

Examination of the fit statistics performance when fitting different types of component distributions revealed that for the conditions that fit skew t component distributions, none were 100% correct on all replications in the conditions. However, the ICL-BIC was 100% correct on all replications in 21 of the 24 conditions and had accuracy 98% or greater in the other three conditions. Its lowest percent correct was 98.40% on the A6C9 and A6C11 conditions and it had an average of 99.83% correct across all of the 24 conditions. The next best performance was the CLC with an average of 96.32% correct across all 24 conditions. The CLC was 100% correct on ten of the conditions and had a 95% or greater performance on an additional ten conditions when fitting multivariate skew t component distributions. In the conditions that fit multivariate skew normal components, none of the fit statistics were 100% accurate in identifying the correct one class model. However, the ICL-BIC was 100% correct on all replications in 15 of the 24 conditions and had accuracy 95% or greater in an additional six conditions.

On the remaining three conditions it had accuracies of 91.24% (A4C3), 55.20% (A3C3), and 46.15% (A3C12) and it had an overall average of 95.00% correct across all of the 24 conditions. The next best performance was the CLC with an average of 75.24% correct across all 24 conditions. The CLC was 100% correct on five of the conditions and had a 95% or greater performance on an additional seven conditions. Notably, although the CAIC's average accuracy (54.64%) was lower than the CLC, NEC (74.83%), aLMR-LRT (61.80%), and the LMR-LRT (61.50%), it had 100% correct accuracy on more conditions – nine – than any of those fit statistics and with a total of 10 conditions it trailed only the CLC and NEC, both with a total of 12 conditions, at the level of 95% or greater. In the conditions that fit multivariate normal component distributions, none of the fit statistics were 100% accurate in identifying the correct one class model. However, the ICL-BIC was 100% correct on all replications in 15 of the 24 conditions and had accuracy 95% or greater in an additional four conditions. Its lowest percent correct was 29.20% on the A1C12 condition and on the remaining four conditions it had accuracies of 92.80% (A1C3), 88.40% (A2C3), 85.20% (A1C9), and 76.00% (A1C12) and it had an overall average of 94.57% correct across all of the 24 conditions. The next best performances were the identically performing CLC and NEC with averages of 57.43% correct across all 24 conditions. They were both 100% correct on four of the conditions and had a 95% or greater performance on an additional four conditions. Notably, although the CAIC (37.48%) and the BIC (35.60%) had average accuracies lower than the CLC, NEC, aLMR-LRT (43.77%), and the LMR-LRT (43.17%), they had 100% correct accuracy on more conditions – eight – than any of those fit statistics and they were tied with the CLC and NEC, eight total conditions, at the level of 95% or greater.

Overall, these results indicate that nonnormality, the type of component distribution, and choice of fit statistic all influence the identification of the correct model. The results show that even mild nonnormality in the data decreases the identification of the correct one class model and encourages the identification of spurious classes. Allowing the shape of the component distributions to vary increases the identification of the correct one class model and decreases the identification of spurious classes. However, the best method to increase the identification of the correct one class model and decrease the identification of spurious classes is the choice of fit statistic. The ICL-BIC outperformed all of the other fit statistics. On average, the other classification based fit statistics – CLC and NEC – outperformed the likelihood ratio based and information criteria based fit statistics. Also, on average, the likelihood ratio based aLMR-LRT and the LMR-LRT outperformed the information criteria based fit statistics. However, when comparing the number of conditions where the fit statistics had 100% accuracy, the information criteria based fit statistics, with the exception of the AIC, outperformed the likelihood ratio based fit statistics and, except for the AIC and the SABIC, they outperformed the classification based CLC and NEC. When comparing the number of conditions where the fit statistics had 95% or greater accuracy, the information criteria based fit statistics, with the exception of the AIC, outperformed the likelihood ratio based fit statistics, but they were outperformed the classification based CLC and NEC. This indicates that the information based statistics tend to me more consistent, whether correct or incorrect, across the replications within the data by analysis condition.

Chapter 5: Discussion

Although, the study contained another analysis condition – type of component covariance matrices – and two other data conditions – sample size and number of variables – these conditions combined accounted for a relatively small amount of the variance in percent correct one class identification. Therefore, the results and the subsequent discussion focus on the three conditions that accounted for the majority of the variance in percent correct one class identification.

The results indicate that the best method to increase the identification of the correct one class model and decrease the identification of spurious classes is the choice of fit statistic. However, nonnormality of the data and the type of component distribution also influence the identification of the correct model. The results showed that across all conditions the ICL-BIC outperformed all of the other fit statistics in identifying the correct one class model. Additionally, the results showed that even levels of mild nonnormality simulated in the data for this study decreases the identification of the correct one class model and encourages the identification of spurious classes. Further, allowing the shape of the component distributions to vary increases the identification of the correct one class model and decreases the identification of spurious classes. Since, in practice, a researcher would have the least control over the distribution of the data, the data condition of skew and kurtosis are discussed before proceeding to the analysis condition of type of component distribution. Further, since the fit statistics are the evaluative tool for determining the model with the correct number of classes, much of the discussion regarding the fit statistics is interspersed within the discussion of the skew and kurtosis and type of component distribution conditions.

Skew and Kurtosis

The levels of skew and kurtosis (skew:kurtosis) explored in this study were 0.00:0.00, 0.75:0.25, and 1.00:1.00. As the results showed, across all other conditions, the greater the nonnormality in the data, the larger the failure rate was for identifying the correct model and, consequently, preference given for models with spurious classes. In practice, when the data is normally distributed (0.00:0.00), there would be no distributional reason for a researcher to suspect that more than one class gave rise to the data (Bauer & Curran, 2003a). However, this level of skew and kurtosis was included as a baseline for the two levels of nonnormality. Theoretically it would be expected that, when given normal data, the fit statistics should prefer the one class model across all conditions on all replications. However, only the ICL-BIC, CAIC, and BIC met this 100% correct across all replications within all conditions criterion. While the SACAIC (98.78%) was very close to meeting the criterion and the BLRT (91.90%), aLMR-LRT (91.20%), and LMR-LRT (90.70%) were all within 10% of the criterion, the SABIC (88.35%), CLC (80.05%), NEC (78.62%), and especially the AIC (33.77%) performed somewhat poorly at the level of skew and kurtosis where the one class model should be preferred as a matter of course. These results are similar to those that Bauer and Curran (2003a) found with a growth mixture model where the ICL-BIC, CAIC, and BIC all had perfect one class model identification and the AIC had similar poor performance.

However, the CLC and NEC had markedly better performance in Bauer and Curran (2003a) than in this study. It is unclear why this would occur, but it is likely that class assignment for individual cases becomes more definitive, and the error (entropy) becomes smaller, over five repeated measurements than when a single measurement is used as in this study. Additionally, Bauer and Curran used a method where an initial parameter estimate was used as a set of priors and then they used five random starts to try to avoid the problem of the likelihood function resolving on a local maximum. Using so few random starts may have been insufficient to avoid local maxima. This study used 500 initial random starts with 20 final stage optimizations. If the likelihood function were to resolve on a local maximum, this would likely increase the separation between the modeled classes and, consequently, reduce the error in assigning cases to classes. The only other study that used normally distributed data, fit with normal component distributions, and a true one class condition was Peugh and Fan (2013). This study more closely matched the findings in the current study; however, the SACAIC was 100% correct versus 98.78% in this study and the CLC was 100% correct versus 80.05% in this study. Although it is unclear why this would occur, as in Bauer and Curran, Peugh and Fan used fewer random starts than in this study. Peugh and Fan used the default of 20 initial random starts with 4 final stage optimizations for the one class model and doubled these values for the multiple class models.

When the first level of mild nonnormality as used to simulate the data, the average percent correct one class identification across all conditions was reduced by more than half – M = 86.42 at the 0.00:0.00 level and M = 41.22 at the 0.75:0.25 level. However, the effect was not consistent across the fit statistics. Across all replications in

18 out of 24 conditions the ICL-BIC was 100% correct and the next most accurate, the CLC, was 100% correct on seven conditions with 95% or greater accuracy on an additional seven conditions. The average percent correct identification for the ICL-BIC was 99.77% and its lowest percent correct was 98.25% on the A3C11 condition. The CLC had an average of 83.22% correct across all 24 conditions. Examination of the conditions where the CLC had less than 95% accuracy suggests that it may be better suited for data with fewer dimensions.

When the data was simulated to have skew = 1.00 and kurtosis = 1.00, the average percent correct one class identification went down – $M = 34.71$. Once again the ICL-BIC outperformed all of the other statistics with an average accuracy of 89.63%. It was 100% correct on nine of the conditions and 95% or greater on an additional seven condition. The next best performance was the CLC with an average of 65.47% correct across all 24 conditions. The CLC was 100% correct on seven of the conditions and had a 95% or greater performance on an additional four conditions. These results are considerably different than those found in Bauer and Curran (2003a) who also used a condition with skew = 1.00 and kurtosis = 1.00. In their study, the ICL-BIC, although still the best performer of the fit statistics they examined, failed to identify the correct one class model 91% of the time. The next best performing fit statistics – CLC and NEC – were almost 99% incorrect in their model selection. Of course, Bauer and Curran only fit normally distributed components. However, in this study, when examining only the conditions that fit normally distributed components, the ICL-BIC had an average percent correct one class model identification of 83.85% even though the condition where the ICL-BIC performed the worst (A1C12; 29.20% correct) fit normally distributed components. Additionally, when examining only the conditions that fit normal components in this study the CLC had an average correct one class identification of 46.05% and this includes three conditions where the CLC was less than one percent correct: A2C9, A1C9, and A1C12. Of these, the first two have 500 cases and the last one has 2000, the first has constrained covariance matrices and the second two unconstrained, and all three conditions have eight variables. It is unclear as to why the ICL-BIC, CLC, and NEC, which had an identical performance to the CLC, performed much better in this study. This could be due to the number of random starts used. Regardless of whether random starts and resolving on local maxima are responsible for the difference in the studies, the results in the current study agree that the amount of nonnormality in the data increases the identification of spurious classes; however, it does not appear that the nonnormality problem is as severe as previously reported. Nonetheless, recent improvements in statistical software have allowed for varying the shape of the component distributions that are fit in the model, which should mitigate the effect of data nonnormality on the identification of spurious classes.

Component Distribution

The recent implementation of methods allowing for varying the shape of component distributions that are fitted to mixture models led Asparouhov and Muthen (2014) to assert that, "Spurious class formation due to non-normality and skewness will be eliminated" (p.6). The results of this study encourage partial credence to this rather strongly worded assertion. When multivariate skew normal and multivariate skew t component distributions where fit to the data, the average percent correct one class model identification across all other conditions was higher than when multivariate normal component distributions were fit to the data. However, the identification of spurious classes was by no means eliminated. As with the effects of nonnormality in identifying spurious classes, it is important to consider the evaluative fit statistics in preventing the identification of spurious classes when nonnormal component distributions are fit to the data.

As with the nonnormality condition, the fit statistics that performed best were the classification based ICL-BIC and CLC. When multivariate skew normal components were fit to the data, the ICL-BIC was, on average, 95.00% correct in identifying the correct one class model and this average percent correct increased to 99.83% correct when multivariate skew t component distributions were fit to the data. The CLC had a similar, if somewhat less effective, performance with an average of 75.24% when multivariate skew normal component distributions were fit to the data and an average of 96.32% when multivariate skew t components were fit to the data. These results, obviously, do not meet the strict criterion of elimination evoked by Asparouhov and Muthen (2014, p.6). However, these authors probably intended a less strict criterion of the elimination of the identification of spurious classes due to nonnormality within a reasonable statistical probability. Using this criterion, on average across all conditions, the ICL-BIC has a five percent error rate when multivariate skew normal components are fit to the data and less than one percent error when the multivariate skew t component distributions are fit to the data. Although the CLC had an unacceptably high error rate of almost 25% when multivariate skew normal components were fit to the data, its error rate was less than five percent when multivariate skew t component distributions were fit to the data. In practice, given the improvement in correct model identification when multivariate skew normal and, more acutely, when multivariate skew t component distributions are fit to the data, one or the other, if not both, of the nonnormal types of component distribution should be fit to the data.

Fitting nonnormal component distributions requires the assumption that the classes within an overall distribution should have a nonnormal distribution. In the social sciences, and in particular education, there are variables such as test scores and response times where nonnormality is a common occurrence (Keats & Lord, 1962). If a variable has, or is expected to have, a population nonnormal distribution, then it is not unwarranted to assume that subpopulations – classes within the population – will share the nonnormal quality. Of course, another consideration is the additional computation time required to estimate models fitting nonnormal components. The computation time associated with fitting nonnormal component distributions increases dramatically depending on the type of component distribution and the complexity of the model. For instance, frequently in this study, fitting a single four class model with multivariate skew t component distributions required in excess of eight hours of computation time. In practice, while the additional computation time is inconvenient, it is a small sacrifice for accurate research results. After all, if Pearson (1894) had the patience to calculate a ninth order polynomial by hand, we can certainly leave a computer running overnight to obtain accurate fit statistics.

Fit Statistics

Across all analysis and data conditions the ICL-BIC was the most accurate fit statistic in identifying the correct one class mode. This finding is consistent with all four of the fit statistic efficacy studies that included the ICL-BIC (McLachlan & Ng, 2000; Bauer & Curran, 2003a, Henson, Reise, & Kim, 2007; Peugh & Fan, 2013). Additionally, in three of these studies, the CLC performed equal to the ICL-BIC and in Bauer and Curran (2003a), except for one condition where the two fit statistics performed equally, the CLC was the second best performing fit statistic, as it was in this study. Of the four studies, only Bauer and Curran (2003a) and Peugh and Fan (2013) examined a true one class condition and fit only normally distributed components and Peugh and Fan did not examine data nonnormality in the true one class condition. Nonetheless, as noted earlier the CLC performed better in the Peugh and Fan study on the one analysis by data condition it shared with this study. The remaining classification fit statistic – NEC – was not examined for the true one class condition in Peugh and Fan. In the Bauer and Curran study, the NEC had identical performance as the CLC and this equal performance was mirrored in this study in the conditions the two studies shared. Of course, as noted earlier, the performance of the CLC and NEC were not as good in this study. Regardless, this study is consistent with previous research that supports the use of classification fit statistics in identifying the correct model.

The results of this study also show that, across all data by analysis conditions, on average the classification fit statistics performed better than the likelihood ratio and information criteria fit statistics. This finding concurs with findings in Henson, Reise, and Kim (2007) and Peugh and Fan (2013), which are the only other studies that also included fit statistics from all three general types of fit statistics. However, Henson, Reise, and Kim investigated a true two class model in a SEM context while Peugh and Fan only employed the likelihood ratio based fit statistics in evaluating a true three class model. Additionally, Henson, Reise, and Kim was the only fit statistic efficacy study that compared the LMR-LRT to its adjusted form the aLMR-LRT. They found that the two fit statistics performed identically. However, in this study, although the two were similar in performance, the aLMR-LRT slightly outperformed its unadjusted form in line with the original expectations when they were first proposed (Lo, Mendell, & Rubin, 2001). The last and poorest performing likelihood ratio statistic was the BLRT. In fact, the only conditions where the BLRT performed reasonably well in identifying the true one class model were the ones that had normally distributed data (skew = 0 and kurtosis = 0); conditions where, in practice, there would be no reason to suspect a mixture of classes. The poor performance of the BLRT echoes the findings in Peugh and Fan (2013) when they examined a true three class model; they did not use the statistic in the true one class condition. The findings regarding the BLRT were considerably different in Nylund, Asparouhov, & Muthén (2007) where the fit statistic outperformed all of the other fit statistics in many of the conditions. However, these authors constrained the fitted class sizes and no such constraints were used in this study. On average, with the exception of the BLRT, the likelihood ratio based statistics outperformed the information criteria based fit statistics. However, when considering the number of conditions where the two types fit statistics had 95% or greater accuracy in identifying the correct one class model, all of the information criteria fit statistics, with the exception of the AIC, outperformed the likelihood ratio fit statistics. However, at the other end of their respective distributions, with the exception of the BLRT, the information criteria based fit statistics had higher a percent and number of conditions in both the 5% or less and 0% levels than did any of the likelihood ratio based fit statistics. These results show that the information criteria fit statistics are more consistent in identifying, or failing to identify, the correct one class model across replications within analysis by data conditions. When looking of the relative performance of the information criteria fit statistics compared to one another, the best performing fit statistic was the CAIC followed by the BIC, SACAIC, SABIC, and the AIC. However, it should be noted that the best performing information criteria fit statistic – CAIC – had an average 51.13% correct one class model identification across all analysis by data conditions. This is somewhat mitigated by the fact that in 43.06% or 31 of the 72 analysis by data conditions the CAIC had a 95% or greater accuracy in identifying the correct one class model.

Conclusions

Not surprisingly, in a study with a true one class model, the most conservative fit statistics within each of the general fit statistic types, ICL-BIC, aLMT-LRT, and CAIC were the best performers in their categories. However, when the number of classes was greater than one, the most conservative fit statistic in the entire study – the ICL-BIC – has been found to underestimate the true number of classes (Puegh & Fan, 2013), while others have shown it to be the most effective in identifying the correct model (McLachlan & Ng, 2000; Henson, Reise, & Kim, 2007). Regardless, in practice, when faced with a nonnormal distribution where one might suspect that the distribution is actually a mixture of latent classes, the null hypothesis should be that the distribution is really a one class distribution. When the one class model is the null hypothesis, the best statistical defense against a type one error is the most conservative fit statistic. Of course statistical protection against a type one error can lead to making a type two error; failing to reject the one class model when the distribution is really made up of two or more classes. Therefore, a researcher would want to consider a more liberal fit statistic to help avoid a type two error. Additionally, consistency is a desirable quality for reproducibility. Therefore, a recommendation for applied use of the results of this study is the adoption of a practice commonly used in other latent variable modeling – the use of multiple fit statistics in evaluating competing models. The process would use the ICL-BIC, aLMR-LRT, and the BIC. The reason for selecting the BIC instead of the slightly more conservative CAIC is that, in this study, their performances were similar, the BIC has the consistent quality, and it is already a part of the Mplus output. While the aLMR-LRT is easily obtainable by requesting the TECH 11 output, Mplus does not calculate the ICL-BIC. Therefore, the process would be to run the competing models; for instance a one class versus a two class model. If the BIC prefers the one class model to the two class model and the *p* value of the aLMR-LRT is greater than .05, then the one class model is statistically the best model. However, if the BIC prefers the two class model and the *p* value of the aLMR-LRT is close to .05, then the ICL-BIC can be calculated by converting the relative entropy reported in the Mplus output to entropy using equation 3.2 and adding this value to the BIC of the two class model. The ICL-BIC from the two class model is then compared to the BIC from the one class model. There is no calculation for the one class model because the error associated with class assignment is zero when there is only one class. Therefore, the entropy penalty drops away and in a one class model the ICL-BIC equals the BIC. Essentially, this tests whether the improvement in model fit to the data when adding an additional class is sufficient to overcome the error associated with assigning cases to classes. Of course, as this study shows, allowing the component distributions to vary their shape increases the ability of the fit statistics to identify the correct one class model when the components are fit to a nonnormal distribution. Therefore, in practice, multiple sets of models with different types of component distributions should be fit.

Models fitted with nonnormal component distributions will require fewer classes to model the data, but they also require considerably more computation time than models fit with normal component distributions. So, the process of fitting models with different types of component distributions begins with finding the best fitting model with normally distributed components. Then fit the nonnormal components model beginning with one fewer class than the result from the normal components modeling. The ICL-BIC and BIC can be used for the comparison between the models using different types of components; however, the aLMR-LRT is only appropriate for testing models within component types. If the fit statistics indicate a better fit for the nonnormal component model with one less class than the normal component model, then nonnormal component models are subsequently fit while reducing the number of classes one at a time until the best fitting model is found.

Of course, while it was not practical to evaluate them in this study, in practice, statistics such as the separation of class means and the proportion of cases in each class are important considerations in determining whether the preferred solution makes theoretical sense. While improvements such as nonnormal component modeling have brought us closer, to Pearson's (1895) belief that a definitive statistical solution will be found for identifying the true number of classes in a distribution, this has yet to be achieved. Therefore, the overarching consideration in finite mixture modeling, as in other statistical modeling, remains and must be that the solution makes theoretical sense.

Limitations and Future Research

This study focused on the identification of a one class model versus models with two through four classes. It did not examine the degree of inaccuracy in identifying the correct model. For this, one would need to examine which model was preferred by the fit statistics rather than whether the fit statistics identified the correct model. A study of this nature would require fitting models with even more classes (e.g. six) than the number used in this study. Given the failure of convergence in some of the conditions in this study, a study fitting even more classes would require using even more random starts to encourage convergence, which would considerably increase the computation time required. Additionally, this study did not examine a condition where the correct model had more than one class. The only research examining conditions where the correct model had more than one class and fit models using nonnormal components, appear to be the studies introducing the implementation of the nonnormal component strategy (Asparouhov & Muthén, 2014; Lee & McLachlan, 2014; Muthén & Asparouhov, 2015). Therefore, future studies are needed to assess the effectiveness of nonnormal component distributions when the correct model is two or more classes. Further, as noted above, this study made no attempt to evaluate other statistical considerations such as the separation of class means and the proportion or number of cases in the classes. Future studies could constrain acceptable solutions to those where the smallest class size contains an acceptable proportion or number of cases from the overall distribution for additional descriptive and simple statistical evaluations such as effect size evaluations of class mean differences. One final limitation is that throughout the analysis, as would be expected, the best log likelihood estimated by the EM algorithm was used on the models that converged. However, the individual models were not checked to ensure that the best log likelihood had been replicated through the random starts process. Therefore, there is no guarantee that every analysis converged to a global maximum.

References

Akaike, H. (1987). Factor analysis and AIC. *Psychometrika, 52*, 317–332.

Alexeev, N., Templin, J., & Cohen, A. S. (2011). Spurious latent classes in the mixture Rasch model. *Journal of Educational Measurement*, *48*(3), 313-332.

Almansa, J., Vermunt, J. K., Forero, C. G., Alonso, J., & European Study of the Epidemiology of Mental Disorders– Mental Health Disability: a European Assessment in the Year 2000 Investigators (2014), A factor mixture model for multivariate survival data: An application to the analysis of lifetime mental disorders. *Journal of the Royal Statistical Society: Series C (Applied Statistics), 63*. 85–102. doi:10.1111/rssc.12026

Asparouhov T. & Muthén, B. (2014). Structural equation models and mixture models with continuous non-normal skewed distributions. *Mplus Web Notes: No. 19 Version 2*, (July 3, 2014). Retrieved from: http://www.statmodel.com/download/Skew.pdf

Bauer, D. J., & Curran, P. J. (2003a). Distributional assumptions of growth mixture models: Implications for overextraction of latent trajectory classes. *Psychological Methods*, *8*, 338-363. doi:10.1037/1082-989X.8.3.338

Bauer, D. J., & Curran, P. J. (2003b). Overextraction of latent trajectory classes: Much ado about nothing? Reply to Rindskopf (2003), Muthen (2003), and Cudeck and Henly (2003). *Psychological Methods, 8*, 384-393. doi:10.1037/1082-989X.8.3.384

Biernacki, C., Celeux, G., & Govaert, G. (1998). Assessing a mixture model for clustering with the integrated classification likelihood. *Technical Report No. 3521*, Rhône-Alpes: INRIA.

Biernacki, C., Celeux, G., & Govaert, G. (1999). An improvement of the NEC criterion for assessing the number of clusters in a mixture model. *Pattern Recognition Letters, 20,* 267–272.

Biernacki, C., & Govaert, G. (1997). Using the classification likelihood to choose the number of clusters. *Computing Science and Statistics, 29*, 451-457.

Bolt, D. M., Cohen, A. S., & Wollack, J. A. (2001). A mixture item response model for multiple-choice data. *Journal of Educational and Behavioral Statistics, 26*, 381–409.

Bolt, D. M., Cohen, A. S., & Wollack, J. A. (2002). Item parameter estimation under conditions of test speededness: Application of a mixture Rasch model with ordinal constraints. *Journal of Educational Measurement, 39*, 331-348.

Bozdogan, H. (1987). Model selection and Akaike's information criterion (AlC): the general theory and its analytic extensions. *Psychometrika, 52*, 345-370.

Bozdogan, H. (1994). Mixture-model cluster analysis using model selection criteria and a new informational measure of complexity. *Proceedings of the First US/Japan*

Conference on the Frontiers of Statistical Modeling: An Informational Approach, 2, 69-113.

Brown, R. S. (2007). Using Latent Class Analysis to Set Academic Performance Standards. *Educational Assessment,* 12(3/4), 283-301.

Büsch, D., Hagemann, N., & Bender, N. (2010). The dimensionality of the Edinburgh Handedness Inventory: An analysis with models of the item response theory. *Laterality, 15*(6), 610-628. doi: 10.1080/13576500903081806

Byrne, B. M., Shavelson, R. J., & Muthe´n, B. O. (1989). Testing for the equivalence of factor covariance and mean structures: The issue of partial measurement invariance. *Psychological Bulletin, 105*(3), 456–466.

Celeux, G., & Soromenho, G. (1996). An entropy criterion for assessing the number of clusters in a mixture model. *Journal of Classification, 13,* 195–212.

Chan, W.-H., Leu, Y.-C., & Chen, C.-M. (2007). Exploring group-wise conceptual deficiencies of fractions for fifth and sixth graders in Taiwan. *Journal of Experimental Education, 76*(1), 26-57.

Cho, H.-J., Lee, J., & Kingston, N. (2012). Examining the effectiveness of test accommodation using DIF and a mixture IRT model. *Applied Measurement in Education, 25,* 281-304. doi:10.1080/08957347.2012.714682

Cho, S.-J. & Cohen, A. S. (2010). A multilevel mixture IRT model with an application to DIF. *Journal of Educational and Behavioral Statistics, 35*(3) 336-370. doi:10.3102/1076998609353111

Cho, S.-J., Cohen, A. S., & Bottge (2013). Detecting intervention effects using a multilevel latent transition analysis with a mixture IRT model. *Psychometrika, 78*(3) 576-600. doi:10.1007/s11334-012-9314-0

Choen, A. S. & Bolt, D. M. (2005) A mixture model analysis of differential item functioning. *Journal of Educational Measurement, 42*(2), 133-148.

Congdon, P. (2003). *Applied Bayesian modelling.* New York: John Wiley.

Cox, D. R. (1957). Note on grouping. *Journal of the American Statistical Association, 52*(280), 543-547. doi:10.1080/01621459.1957.10501411

De Jong, M. G. & Steenkamp, J.-B. E. M. (2010). Finite mixture multilevel multidimensional ordinal IRT models for large scale cross-cultural research. *Psychometrika, 75*(1), 3–32. doi:10.1007/s11336-009-9134-z

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological), 39*(1), 1-38. Retrieved from http://www.jstor.org.proxy.its.virginia.edu/stable/2984875

Ding, C. S. (2006). Multidimensional scaling modelling approach to latent profile analysis in psychological research. *International Journal of Psychology, 41*, 226–238.

Draper, D. (1995). Assessment and propagation of model uncertainty. *Journal of the Royal Statistical Society: Series B, 57*, 45–97.

Efron, B. (1979). Bootstrap methods: Another look at the jackknife. *The Annals of Statistics, 7*(1), 1-26. Retrieved from https://projecteuclid.org/download/pdf_1/euclid.aos/1176344552

Finch, W. H. & Finch, M. E. H. (2013). Investigation of specific learning disability and testing accommodations based differential item functioning using a multilevel multidimensional mixture item response theory model. *Educational and Psychological Measurement, 76*(6), 973-993. doi:10.1177/0013164413494776

Fisher, W. D. (1958). On grouping for maximum homogeneity. *Journal of the American Statistical Association, 53*(284), 789-798. Retrieved from http://www.jstor.org.proxy.its.virginia.edu/stable/2281952

Forgy, E. W. (1965). Cluster analysis of multivariate data: Efficiency versus interpretability of classifications. *Biometrics, 21*, 768-769.

Frederickx, S., Tuerlinckx, F., De Boeck, P., & Magis, D. (2010), RIM: A random item mixture model to detect differential item functioning. *Journal of Educational Measurement, 47*(4), 432-457.

Freund, P. A., Tietjens, M., & Strauss, B. (2013), Using rating scales for the assessment of physical self-concept: Why the number of response categories matters. *Measurement in Physical Education and Exercise Science*, *17*, 249–263. doi:10.1080/1091367X.2013.807265

Gebhardt, C., Rose, N. & Mitte, K. (2013). Fact or artefact: An item response theory analysis of median split based repressor classification. *British Journal of Health Psychology, 19,* 36-51. DOI:10.1111/bjhp.12029

Geisser, S., & Eddy, W. F. (1979). A predictive approach to model selection. *Journal of the American Statistical Association, 74*, 153-160.

Gelman, A., Carlin, J. B., Stern, H. S., & Rubin, D. B. (1996). *Bayesian data analysis.* London: Chapman & Hall.

Giang, M. T., & Graham, S. (2008). Using latent class analysis to identify aggressors and victims of peer harassment. *Aggressive Behavior*, 34(2), 203-213.

Hannan, E. J., & Quinn, B. G. (1979). The determination of the order of an autoregression. *Journal of the Royal Statistical Society: Series B, 41*, 190–195.

Haughton, D. (1988). On the choice of a model to fit data from an exponential family. *The Annals of Statistics, 16*, 342-355.

Headrick, T. C. (2002). Fast fifth-order polynomial transforms for generating univariate and multivariate nonnormal distributions. Computational Statistics & Data Analysis, 40, 685-711.

Henson, J. M., Reise, S. P., & Kim, K. H. (2007). Detecting mixtures from structural model differences using latent variable mixture modeling: A comparison of relative model fit statistics, *Structural Equation Modeling, 14*(2), 202-226, doi:10.1080/10705510709336744

Hesser, H. & Andersson, G. (2014). Dimensional or categorical approach to tinnitus severity: An item response mixture modeling analysis of tinnitus handicap. *International Journal of Behavioral Medicine, 21*(6), 982-988. doi:10.1007/s12529-013-9375-1

Hoijtink, H., & Notenboom, A. (2004). Model based clustering of large data sets: Tracing the development of spelling ability. *Psychometrika, 69*, 481–498.

Howe, A. J. & Bozdogan, H. (2013). Robust mixture model cluster analysis using adaptive kernels. *Journal of Applied Statistics, 40*(2), 320-336. doi:10.1080/02664763.2012.740630

Hurvich, C. M., & Tsai, C. (1989). Regression and time series model selection in small samples. *Biometrika, 76*, 297–307.

Ishiguro, M., Sakamoto, Y., & Kitagawa, G. (1997). Bootstrapping log-likelihood and EIC, an extension of AIC. *Annals of the Institute of Statistical Mathematics, 49*, 411-434.

Jeffries, N. (2003). A note on "Testing the number of components in a normal mixture." *Biometrika, 90*, 991–994.

Jin, K.-Y. & Wang, W.-C. (2014) Item response theory models for performance decline during testing, *Journal of Educational Measurement*, *51*(2), 178-200.

Jöreskog, K. G. (1971). Simultaneous factor analysis in several populations. *Psychometrika, 36*(4), 409-426. doi:10.1007/BF02291366

Keats, J. A., & Lord, F. M. (1962) A theoretical distribution for mental test scores. *Psychometrika, 27*, 59-72.

Kelava, A. & Brandt, H. (2014). A general nonlinear multilevel structural equation mixture model. *Frontiers in Psychology* 5:748. doi:10.3389/fpsyg.2014.00748

Kim, J., & Mahmassani, H. S. (2014). A finite mixture model of vehicle-to-vehicle and day-to-day variability of traffic network travel times. *Transportation Research Part C: Emerging Technologies, 46*(0), 83-97. doi:http://dx.doi.org/10.1016/j.trc.2014.05.011

Kogan, J. (2007). *Introduction to clustering large and high-dimensional data.* New York, NY: Cambridge University Press.

Larsson, G., Kempe, C., & Starrin, B. (1988). Appraisal and coping processes in acute time-limited stressful situations: A study of police officers. *European Journal of Personality, 2,* 259–276.

Lee, H. Y. & Beretvas, S. N. (2014). Evaluation of two types of differential item functioning in factor mixture models with binary outcomes. *Educational and Psychological Measurement, 74*(5), 831-858. doi:10.1177/0013164414526881

Lee, S. & McLachlan, G. J. (2011). On fitting of mixtures of multivariate skew *t*-distributions via the EM algorithm. arXiv:1109.4706v2 [stat.ME]. Retrieved from: http://arxiv.org/pdf/1109.4706v2.pdf

Lee, S. & McLachlan, G. J. (2014). Finite mixtures of multivariate skew *t*-distributions: some recent and new results. *Statistics and Computing, 24*(2), 181-202. doi:10.1007/s11222-012-9362-4

Leong, S. H., Mahdi, R., & Ling, S. E. (2013). Mixture Rasch model for guessing group identification. *Proceedings of the 20th National Symposium on Mathematical Sciences,* AIP Conf. Proc. 1522, 1248-1253. doi:10.1063/1.4801273

Li, F., Cohen, A. S., Kim, S.-H., & Cho, S.-J. (2009). Model selection methods for mixture dichotomous IRT models. *Applied Psychological Measurement, 33*(5), 353-373. doi:10.1177/0146621608326422

Lin, T.-I. (2010). Robust mixture modeling using multivariate skew *t* distribution. *Statistics and Computing 20*(3), 343-356. doi:10.1007/s11222-009-9128-9

Little, T. D. (1997). Mean and covariance structures (MACS) analyses of cross-cultural data: Practical and theoretical issues. *Multivariate Behavioral Research, 32,* 53–76.

Lloyd, S. (1982). Least squares quantization in PCM. *Information Theory, IEEE Transactions on, 28*(2), 129-137. doi:10.1109/TIT.1982.1056489

Lo, Y, Mendell, N. R, & Rubin, D. B. (2001). Testing the number of components in a normal mixture. *Biometrika, 88*, 767-778.

Lubke, G. H. & Muthén, B. (2005). Investigating population heterogeneity with factor mixture models. *Psychological Methods, 10*(1), 21-29. doi:10.1037/1082-989X.10.1.21

Lubke, G. H. & Muthén, B. (2007). Performance of factor mixture models as a function of model size, covariate effects, and class-specific parameters. *Structural Equation Modeling, 14*(1), 26-47.

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. Paper presented at the *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, 1*(14), 281-297.

Magidson, J., & Vermunt J. K. (2002a, March). Latent class modeling as a probabilistic extension of *K*-means clustering. *Quirk's Marketing Research Review, 20*(3), 77–80.

Magidson, J., & Vermunt, J. K. (2002b). Latent class models for clustering: A comparison with *K*-means. *Canadian Journal of Marketing Research, 20*(1), 36–43.

Maij-de Meij, A. M., Kelderman, H., & van der Flier, H. (2008). Fitting a mixture item response theory model to personality questionnaire data: Characterizing latent classes and investigating possibilities for improving prediction. *Applied Psychological Measurement, 32*(8), 611–631. doi:10.1177/0146621607312613

McCrae, J. S., Chapman, M. V., & Christ, S. L. (2006). Profile of children investigated for sexual abuse: Association with psychopathology symptoms and services. *American Journal of Orthopsychiatry, 76*, 468–481. doi:10.1037/0002-9432.76.4.468

McCrea, R. S., Morgan, B. J. T., & Cole, D. J. (2013). Age-dependent mixture models for recovery data on animals marked at unknown age. *Journal of the Royal Statistical Society: Series C (Applied Statistics), 62*(1), 101-113. doi:10.1111/j.1467-9876.2012.01043.x

McLachlan, G. J. (1987). On bootstrapping the likelihood ratio test statistic for the number of components in a normal mixture. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, *36*(3), 318-324. Retrieved from http://www.jstor.org/stable/2347790

McLachlan, G. J. (2011). Commentary on Steinley and Brusco (2011): Recommendations and cautions. *Psychological Methods, 16*(1), 80–81. doi:10.1037/a0021141

McLachlan, G. J., & Basford, K. E. (1988). *Mixture models: Inference and applications to clustering.* New York, NY: Marcel Dekker.

McLachlan, G. J., & Ng, S. K. (2000). A comparison of some information criteria for the number of components in a mixture model. In McLachlan, G. J., & Peel, D. *Finite mixture models.*(217-220) New York, NY: Wiley.

McLachlan, G. J., & Peel, D. (2000). *Finite mixture models.* New York, NY: Wiley.

Mellenbergh, G. J. (1989). Item bias and item response theory. *International Journal of Educational Research, 13*(2), 127–143. doi:10.1016/0883-0355(89)90002-5

Meredith, W. (1993). Measurement invariance, factor analysis, and factorial invariance. *Psychometrika, 58*, 525–543. doi:10.1007/BF02294825

Meyer, J. P. (2010). A mixture Rasch model with item response time components. *Applied Psychological Measurement*, *34*(7), 521-538. doi:10.1177/0146621609355451

Muthén, B. (2003). Statistical and substantive checking in growth mixture modeling: Comment on Bauer and Curran (2003). *Psychological Methods*, *8*(3), 369-377. doi:10.1037/1082-989X.8.3.369

Muthén, B. & Asparouhov T. (2015). Growth mixture modeling with non-normal distributions. *Statistics in Medicine*, *34*(6), 1041–1058. doi: 10.1002/sim6388

Muthén, B. O., & Shedden, K. (1999). Finite mixture modeling with mixture outcomes using the EM algorithm. *Biometrics*, *55*(2), 463–469. doi:10.1111/j.0006-341X.1999.00463.x

Olvera Astivia, O. L. & Zumbo, B. D. (2014). A cautionary note on the use of the Vale and Maurelli method to generate multivariate, nonnormal data for simulation purposes. *Educational and Psychological Measurement*, *xx*(x), 1-27. doi:10.1177/0013164414548894

Nylund, K. L., Asparouhov, T., & Muthén, B. O. (2007). Deciding on the number of classes in latent class analysis and growth mixture modeling: a Monte Carlo simulation study. *Structural Equation Modeling*, *14*(4), 535–569.

Park, B., Lord, D., & Lee, C. (2014). Finite mixture modeling for vehicle crash data with application to hotspot identification. *Accident Analysis & Prevention*, *71*(0), 319-326. doi:http://dx.doi.org/10.1016/j.aap.2014.05.030

Pastor, D. A., Barron, K. E., Miller, B. J., & Davis, S. L. (2007). A latent profile analysis of college students' achievement goal orientation profiles. *Contemporary Educational Psychology, 32*, 8-47.

Pearson, K. (1894). Contributions to the mathematical theory of evolution. *Philosophical Transactions of the Royal Society of London. A, 185*, 71-110. Retrieved from http://www.jstor.org/stable/90667

Pearson, K. (1895). Contributions to the mathematical theory of evolution. II. skew variation in homogeneous material. *Philosophical Transactions of the Royal Society of London. A, 186*, 343-414. Retrieved from http://www.jstor.org/stable/90649

Punzo, A. (2014). Flexible mixture modeling with the polynomial Gaussian cluster-weighted model. *Statistical Modeling*, *14*(3), 257-291. doi:10.1177/1471082X13503455

Rindskopf, D. (2003). Mixture or homogenous? Comment on Bauer and Curran (2003). *Psychological Methods*, *8*, 364-368.

Roberts, S. J., Husmeier, D., Rezek, I., & Penny, W. (1998). Bayesian approaches to Gaussian modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 20*, 1133-1142.

Rost, J. (1990). Rasch models in latent classes: An integration of two approaches to item analysis. *Applied Psychological Measurement*, *14*, 271–282.

Rost, J. (1991). A logistic mixture distribution model for polychotomous item responses. *British Journal of Mathematical and Statistical Psychology, 44*, 75-92.

Schwartz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics*, 6, 461-464.

Sclove, L. S. (1987). Application of model-selection criteria to some problems in multivariate analysis. *Psychometrika, 52*, 333-343.

Spiegelhalter, D. J., Best, N. G., & Carlin, B. P. (1998). *Bayesian deviance, the effective number of parameters, and the comparison of arbitrarily complex models.* Technical report, MRC Biostatistics Unit, Cambridge. Retrieved from http://www.sph.umn.edu/faculty1/wp-content/uploads/2012/11/rr98-009.pdf

Steinley, D. (2006). *K*-means clustering: A half-century synthesis. *British Journal of Mathematical and Statistical Psychology, 59,* 1–34. doi:10.1348/000711005X48266

Steinley, D., & Brusco, M. J. (2011). Evaluating Mixture Modeling for Clustering: Recommendations and cautions. *Psychological Methods, 16*(1), 63–79. doi:10.1037/a0022673

Sawatzky, R., Ratner, P. A., Kopec, J. A., & Zumbo, B. D. (2012). Latent variable mixture models: a promising approach for the validation of patient reported outcomes. *Quality of Life Research, 21*(4), 637-650. doi:10.1007/s11136-011-9976-6

Thorndike, R. (1953). Who belongs in the family? *Psychometrika, 18*(4), 267-276. doi:10.1007/BF02289263

Tofighi, D., & Enders, C. K. (2007). Identifying the correct number of classes in a growth mixture model. In G. R. Hancock (Ed.), Mixture models in latent variable research (pp. 317–341). Greenwich, CT: Information Age.

Tuma, M. & Decker, R. (2013). Finite mixture models in market segmentation: A review and suggestions for best practices. *The Electronic Journal of Business Research Methods, 11*(1), 2-15. Available online at www.ejbrm.com

Vale, C. & Maurelli, V. (1983). Simulating multivariate nonnormal distributions. *Psychometrika, 48*(3), 465-471.

Vermunt, J. K. (2011). *K*-means may perform as well as mixture model clustering but may also be much worse: Comment of Steinley and Brusco (2011). *Psychological Methods, 16*(1), 82–88. doi:10.1037/a0020144

Vermunt, J.K., and Magidson, J. Latent class cluster analysis. (2002). J.A. Hagenaars and A. McCutcheon (Eds.). *Applied latent class analysis*, 89-106. Cambridge: Cambridge University Press.

Voung, Q. H. (1989). Likelihood ratio tests for model selection and non-nested hypotheses. *Econometrica*, *57*, 307–333.

Wade, T. D., Crosby, R. D., & Martin, N. G. (2006). Use of latent profile analysis to identify eating disorder phenotypes in an adult Australian twin cohort. *Archives of General Psychiatry, 63*, 1377–1384. Retrieved from www.archgenpsychiatry.com

White, H. (1982). Maximum likelihood estimation of misspecified models. *Econometrica, 50*, 1-25.

Widaman, K. F., & Reise, S. P. (1997). Exploring the measurement invariance of psychological instruments: Applications in the substance use domain. In K. J. Bryant, M. Windle, & S. G. West (Eds.), *The science of prevention: Methodological advances from alcohol and substance abuse research* (281–324).

Washington, DC: American Psychological Association. Retrieved from http://psycnet.apa.org/books/10222/009.pdf

Woodruffe, M. (1982). On model selection and the arcsine laws. *The Annals of Statistics, 10,* 1182-1194.

Wu, P.-C. & Huang, T.-W. (2010). Person heterogeneity of the BDI-II-C and its effects on dimensionality and construct validity: Using mixture item response models. *Measurement and Evaluation in Counseling and Development*, *43*(3), 155–167. doi:10.1177/0748175610384808

Yampolskaya, S., Greenbaum, P., & Berson, I. (2009). Profiles of Child Maltreatment Perpetrators and Risk for Fatal Assault: A Latent Class Analysis. *Journal of Family Violence*, *24*(5), 337-348. doi:10.1007/s10896-009-9233-8

Yu, J. (2012). Gaussian mixture models-based control chart pattern recognition. International *Journal of Production Research*, *50*(23), 6746–6762. doi:10.1080/00207543.2011.623724

Appendix A

Simulation Check Tables

Table A1.
Simulation check: Skewness in the four variable conditions.

		Mdn	*M*	*S*	*Min*	*Max*
N = 500	V1	0.00644	-0.00137	0.10657	-0.29189	0.26107
Sk = 0.0	V2	0.00060	0.00253	0.11347	-0.38010	0.36509
	V3	0.00264	0.00344	0.10304	-0.23825	0.36516
	V4	0.00953	0.00195	0.10795	-0.30493	0.34742
N = 500	V1	0.73953	0.74060	0.10639	0.44158	1.13850
Sk = 0.75	V2	0.74261	0.75437	0.09408	0.49671	1.01562
	V3	0.73908	0.74119	0.09237	0.52514	0.96335
	V4	0.74332	0.74692	0.10233	0.48105	1.00232
N = 500	V1	0.98100	0.98367	0.12876	0.66029	1.40420
Sk = 1.0	V2	0.96638	0.97467	0.13216	0.64863	1.47419
	V3	0.99071	0.99196	0.14340	0.66095	1.40980
	V4	0.98430	0.98865	0.13159	0.69428	1.42614
N = 2000	V1	-0.00722	-0.00672	0.05406	-0.18082	0.14364
Sk = 0.0	V2	-0.00199	0.00091	0.05629	-0.16138	0.16827
	V3	-0.00113	-0.00488	0.05486	-0.16965	0.13413
	V4	-0.00930	-0.00620	0.05508	-0.15932	0.18076
N = 2000	V1	0.74983	0.75014	0.05126	0.60617	0.87871
Sk = 0.75	V2	0.75119	0.75061	0.05037	0.61638	0.90514
	V3	0.75033	0.75032	0.04958	0.62396	0.87645
	V4	0.74522	0.74807	0.05332	0.61225	0.93346
N = 2000	V1	1.00185	0.99943	0.06630	0.81436	1.18227
Sk = 1.0	V2	0.99841	1.00031	0.06700	0.81480	1.19638
	V3	0.99640	0.99295	0.06817	0.83381	1.21450
	V4	0.99236	0.99776	0.07090	0.80722	1.22962

Table A2.

Simulation check: Kurtosis in the four variable conditions.

		Mdn	*M*	*S*	*Min*	*Max*
N = 500	V1	-0.03089	-0.02333	0.21157	-0.44070	0.84217
Ku = 0.0	V2	**-0.06099**	-0.02820	0.20698	-0.43140	0.57475
	V3	-0.03614	-0.01380	0.21716	-0.48677	0.85028
	V4	-0.02956	-0.01812	0.21098	-0.44395	0.67821
N = 500	V1	0.21549	0.24205	0.36890	-0.56489	1.92669
Ku = 0.25	V2	0.21566	0.24731	0.32318	-0.52291	1.19469
	V3	**0.17656**	0.22441	0.33178	-0.37962	1.38604
	V4	**0.19477**	0.22670	0.32919	-0.56665	1.21531
N = 500	V1	**0.84701**	**0.91864**	0.60345	-0.24797	3.78106
Ku = 1.0	V2	**0.77981**	**0.90251**	0.57907	-0.18802	3.46289
	V3	**0.88123**	0.97542	0.63828	-0.28707	3.40701
	V4	**0.86219**	0.96084	0.61411	-0.24886	3.06473
N = 2000	V1	-0.01297	-0.00859	0.09588	-0.24852	0.30156
Ku = 0.0	V2	-0.02093	-0.01643	0.10160	-0.27496	0.30661
	V3	-0.00064	-0.00277	0.10527	-0.28499	0.37057
	V4	-0.01007	-0.00799	0.11166	-0.30247	0.38176
N = 2000	V1	0.25147	0.25769	0.17396	-0.10263	0.85750
Ku = 0.25	V2	0.23244	0.25340	0.17820	-0.19586	0.78112
	V3	0.24449	0.24836	0.15525	-0.20309	0.63270
	V4	0.22951	0.24903	0.17320	-0.12256	0.82839
N = 2000	V1	0.98457	1.00692	0.31730	0.25870	2.27317
Ku = 1.0	V2	0.97874	1.00174	0.30906	0.29825	2.00747
	V3	0.95042	0.96975	0.31621	0.26938	1.85086
	V4	0.94300	0.99070	0.34439	0.25778	2.22627

Note: Bold type indicates statistic > 0.05 off target value.

Table A3.

Simulation check: Skewness in the eight variable conditions with N = 500.

		Mdn	*M*	*S*	*Min*	*Max*
N = 500	V1	0.00548	0.00335	0.09489	-0.24453	0.27916
Sk = 0.0	V2	-0.00971	-0.00579	0.10549	-0.33679	0.30212
	V3	0.00898	0.01018	0.10259	-0.23419	0.31412
	V4	0.00775	-0.00036	0.09776	-0.25336	0.26703
	V5	-0.00126	0.00031	0.10282	-0.33431	0.27399
	V6	0.00282	-0.00436	0.11376	-0.29371	0.28410
	V7	-0.00285	0.00010	0.11644	-0.30165	0.28414
	V8	0.00105	-0.00383	0.11343	-0.41642	0.26385
N = 500	V1	0.73244	0.73618	0.10022	0.48151	1.01718
Sk = 0.75	V2	0.73651	0.74032	0.10150	0.49554	1.03844
	V3	0.74046	0.74761	0.10328	0.46291	1.12954
	V4	0.74302	0.74406	0.09837	0.46913	1.03550
	V5	0.72665	0.73413	0.10525	0.48845	1.02412
	V6	0.74350	0.74911	0.10698	0.51163	1.06781
	V7	0.74918	0.74864	0.10634	0.41320	1.08194
	V8	0.74914	0.74369	0.09853	0.48491	1.07711
N = 500	V1	0.98519	0.99363	0.13605	0.64779	1.45310
Sk = 1.0	V2	0.96941	0.98005	0.12914	0.65074	1.34027
	V3	0.96399	0.99196	0.14653	0.70919	1.45468
	V4	0.96934	0.98402	0.13154	0.61147	1.43791
	V5	0.97025	0.98582	0.13756	0.67751	1.47500
	V6	0.99039	1.00235	0.14033	0.59925	1.54492
	V7	0.97984	0.98059	0.13708	0.66531	1.46756
	V8	0.98299	0.98589	0.12559	0.64247	1.41921

Table A4.

Simulation check: Skewness in the eight variable conditions with N = 2000.

		Mdn	*M*	*S*	*Min*	*Max*
N = 2000	V1	0.00651	0.00287	0.05559	-0.13933	0.14273
Sk = 0.0	V2	0.00105	0.00064	0.05563	-0.16832	0.13769
	V3	0.00479	0.00419	0.05462	-0.14363	0.16518
	V4	-0.00148	-0.00063	0.05462	-0.13359	0.13690
	V5	-0.00107	-0.00144	0.05498	-0.16254	0.13922
	V6	0.00486	0.00477	0.05556	-0.13159	0.16482
	V7	0.00339	0.00022	0.05378	-0.11930	0.17532
	V8	0.00860	0.00340	0.05441	-0.13595	0.15083
N = 2000	V1	0.74580	0.74950	0.05141	0.62824	0.94845
Sk = 0.75	V2	0.75448	0.75074	0.05214	0.57566	0.89710
	V3	0.74800	0.74923	0.05391	0.62960	0.91133
	V4	0.75390	0.75254	0.05152	0.62262	0.89305
	V5	0.75292	0.75195	0.04984	0.59810	0.89295
	V6	0.74992	0.75142	0.05381	0.60296	0.92688
	V7	0.74951	0.74875	0.05122	0.60383	0.93427
	V8	0.74690	0.75422	0.05040	0.63447	0.89189
N = 2000	V1	0.99038	0.99649	0.06439	0.83977	1.17789
Sk = 1.0	V2	1.00692	1.00499	0.06593	0.82900	1.18326
	V3	1.00540	0.99971	0.06894	0.82974	1.18321
	V4	0.99247	1.00226	0.06883	0.83443	1.18687
	V5	0.99673	0.99806	0.06970	0.80678	1.18390
	V6	0.99708	0.99562	0.06608	0.81898	1.16767
	V7	0.99381	0.99618	0.06885	0.77883	1.26226
	V8	0.99568	0.99932	0.07243	0.81685	1.19381

Table A5.

Simulation check: Kurtosis in the eight variable conditions with N = 500.

		Mdn	*M*	*S*	*Min*	*Max*
N = 500	V1	-0.04248	-0.02059	0.20719	-0.51923	0.68186
Ku = 0.0	V2	-0.03424	-0.00649	0.22912	-0.44216	0.73864
	V3	-0.04834	-0.01141	0.22609	-0.41747	0.89718
	V4	-0.01830	-0.00078	0.21281	-0.44111	0.57241
	V5	-0.06954	-0.04587	0.20831	-0.47273	0.86098
	V6	-0.02345	-0.00278	0.20760	-0.46639	0.64372
	V7	-0.02129	-0.02095	0.20905	-0.53902	0.73668
	V8	-0.01256	0.00555	0.22722	-0.49556	1.40011
N = 500	V1	**0.17566**	0.20923	0.35297	-0.56870	1.35814
Ku = 0.25	V2	0.20421	0.22088	0.31514	-0.40291	1.28213
	V3	0.23026	0.25157	0.35782	-0.61434	1.71316
	V4	**0.18206**	0.21606	0.31582	-0.57369	1.33482
	V5	**0.16688**	**0.19748**	0.34172	-0.46256	1.73070
	V6	**0.18778**	0.25912	0.38415	-0.49270	1.57075
	V7	0.22823	0.23651	0.35479	-0.61093	1.62997
	V8	**0.19894**	0.21899	0.32578	-0.48039	1.23943
N = 500	V1	**0.86363**	0.96638	0.65438	-0.23057	3.8159
Ku = 1.0	V2	**0.87953**	**0.92066**	0.57419	-0.24452	3.37294
	V3	**0.87283**	0.96506	0.69388	-0.28591	3.87444
	V4	**0.84127**	**0.91316**	0.60465	-0.08148	3.28883
	V5	**0.76797**	**0.90548**	0.60655	-0.16215	2.81700
	V6	**0.92373**	1.00698	0.63354	-0.17598	4.50836
	V7	**0.83015**	**0.91622**	0.62208	-0.26429	3.40547
	V8	**0.87489**	**0.94223**	0.57927	-0.21917	3.38764

Note: Bold type indicates statistic > 0.05 off target value.

Table A6.

Simulation check: Kurtosis in the eight variable conditions with N = 2000.

		Mdn	*M*	*S*	*Min*	*Max*
N = 2000	V1	-0.00903	-0.00585	0.09905	-0.24831	0.40077
Ku = 0.0	V2	-0.00879	-0.00656	0.11506	-0.32764	0.36475
	V3	0.00062	0.00305	0.11189	-0.26778	0.33758
	V4	-0.01940	-0.01180	0.10966	-0.23555	0.33580
	V5	-0.02061	-0.01027	0.11084	-0.26865	0.28535
	V6	-0.02538	-0.00463	0.10824	-0.21832	0.61508
	V7	-0.01626	-0.01343	0.10393	-0.29327	0.36824
	V8	-0.01426	-0.01121	0.10388	-0.27410	0.26942
N = 2000	V1	0.24433	0.25345	0.17424	-0.15167	0.94410
Ku = 0.25	V2	0.23291	0.24347	0.17086	-0.24758	0.77270
	V3	0.23572	0.24920	0.17630	-0.13273	0.95334
	V4	0.25340	0.26416	0.18071	-0.20812	0.82448
	V5	0.26169	0.26290	0.16867	-0.30199	0.73563
	V6	0.22248	0.24538	0.18851	-0.12717	0.85507
	V7	0.24164	0.24802	0.17777	-0.16254	0.99996
	V8	0.25726	0.26532	0.18111	-0.14083	0.81751
N = 2000	V1	0.95380	0.99232	0.30156	0.33886	1.92922
Ku = 1.0	V2	1.02156	1.03083	0.31224	0.18418	2.20004
	V3	0.97500	0.99814	0.32548	0.27436	1.85370
	V4	0.96475	1.00803	0.32506	0.29719	2.16480
	V5	0.97006	1.00167	0.34250	0.29317	2.46093
	V6	0.97167	0.98398	0.29466	0.22910	1.94187
	V7	0.95194	0.99403	0.32783	0.18680	2.74306
	V8	0.98858	1.02766	0.35681	0.25557	2.48759

Comparison of Simulation Methods

Table A7.

Comparison of simulation methods for skewness target value 1.00.

	Headrick (2002)				Vale & Maurelli (1983)		
	Mdn	*M*	*S*		*Mdn*	*M*	*S*
Reps = 1000, *N* = 10000							
V1	0.99814	0.99990	0.03722	←	**0.99815**	**0.99998**	**0.03002**
V2	0.99865	**0.99995**	0.03733	→	**1.00008**	0.99986	**0.03011**
V3	0.99790	**0.99931**	0.04080	→	**0.99939**	0.99919	**0.03061**
V4	**0.99858**	**1.00053**	0.03757	→	0.99828	0.99907	**0.03004**
Reps = 1000, *N* = 2000							
V1	0.99120	**0.99769**	0.07653	→	**0.99168**	0.99300	**0.06493**
V2	0.99144	**0.99805**	0.08425	→	**0.99597**	0.99596	**0.06905**
V3	**0.99596**	**1.00161**	0.07747	→	0.99337	0.99679	**0.06795**
V4	**1.00111**	1.00408	0.08164	→	0.99437	**0.99710**	**0.06754**
Reps = 1000, *N* = 500							
V1	0.97329	0.98760	0.14134	→	**0.97870**	**0.98763**	**0.13291**
V2	0.97711	0.98988	0.14381	→	**0.98623**	**0.99521**	**0.13267**
V3	0.96312	0.97821	0.13937	→	**0.98519**	**0.99168**	**0.13879**
V4	0.96634	0.98685	0.14913	→	**0.98237**	**0.98989**	**0.13660**
Reps = 250, *N* = 2000							
V1	**1.00014**	1.00867	0.08494	→	0.99225	**0.99199**	**0.06969**
V2	0.98852	0.99073	0.07269	←	**0.98900**	**0.99487**	**0.06804**
V3	1.00353	1.00709	0.08031	←	**1.00163**	**0.99590**	**0.06783**
V4	**0.99816**	**0.99864**	0.08458	←	0.99160	0.99749	**0.06952**
Reps = 250, *N* = 500							
V1	0.96702	**0.98621**	0.14403	→	**0.97623**	0.98309	**0.13490**
V2	0.95059	0.98032	0.15687	→	**0.99965**	**0.99735**	**0.12848**
V3	0.96172	0.96781	0.14040	←	**0.99140**	**0.99793**	**0.13019**
V4	0.94287	0.96921	0.15181	→	**0.98553**	**0.99292**	**0.13648**

Note: bold type indicates statistic across methods closest to target and arrow indicates which method the mean is closest to median.

Table A8.

Comparison of simulation methods for kurtosis target value 1.00.

		Headrick (2002)				Vale & Maurelli (1983)		
		Mdn	*M*	*S*		*Mdn*	*M*	*S*
Reps = 1000, *N* = 10000								
	V1	0.96883	1.00296	0.26149	→	**0.98776**	**1.00135**	**0.14265**
	V2	0.95932	0.99104	0.22860	→	**0.99259**	**1.00362**	**0.14122**
	V3	0.97049	1.00656	0.32501	→	**0.98873**	**0.99671**	**0.14367**
	V4	0.96699	0.99820	0.25211	→	**0.98685**	**0.99538**	**0.13841**
Reps = 1000, *N* = 2000								
	V1	0.89887	**0.98216**	0.47072	→	**0.93310**	0.96717	**0.31247**
	V2	0.90301	**0.98871**	0.56669	→	**0.95423**	0.98104	**0.32221**
	V3	0.90776	**0.99609**	0.46704	→	**0.94956**	0.98890	**0.32947**
	V4	0.93723	**1.01701**	0.50987	→	**0.95290**	0.97842	**0.31750**
Reps = 1000, *N* = 500								
	V1	0.76645	0.92531	0.73834	→	**0.84957**	**0.94422**	**0.60829**
	V2	0.80317	0.91963	0.72645	→	**0.88533**	**0.97293**	**0.61459**
	V3	0.75101	0.86159	0.69739	→	**0.87205**	**0.97052**	**0.62977**
	V4	0.74186	0.92510	0.87262	→	**0.83992**	**0.95750**	**0.64371**
Reps = 250, *N* = 2000								
	V1	0.95145	1.04945	0.54738	→	**0.95384**	**0.96828**	**0.31665**
	V2	0.89255	0.94492	0.40937	←	**0.91029**	**0.98352**	**0.31898**
	V3	0.94216	**1.02231**	0.47682	→	**0.95882**	0.97415	**0.31814**
	V4	0.90128	**1.00822**	0.63977	→	**0.94488**	0.98815	**0.34080**
Reps = 250, *N* = 500								
	V1	0.73625	**0.94205**	0.77557	→	**0.83105**	0.93639	**0.67230**
	V2	0.69213	0.91904	0.93673	→	**0.93256**	**0.97855**	**0.56160**
	V3	0.76285	0.85326	0.67843	→	**0.94409**	**0.99338**	**0.60115**
	V4	0.62883	0.85229	0.84522	→	**0.85431**	**0.99365**	**0.64878**

Note: bold type indicates statistic across methods closest to target and arrow indicates which method the mean is closest to median.

Simulation Code: R Syntax for Headrick (2002)

```
#Code adapted from link in Astivia and Zumbo (2014) A cautionary note on the use of the Vale and Maurelli
        #method to generate multivariate, nonnormal data for simulation purposes. Educational and Psychological
        #Measurement, 1-17. DOI: 10.1177/0013164414548894

        #https://psychometroscar.wordpress.com/headricks-5thorder-polynomial-method/

        #https://raw.githubusercontent.com/OscarOlvera/R-code-for-publications/master/Headrick02.md
######################################################################################################################

f_skew <- function(x){

  sd.x <- sd(x)
  mu3.x <- mean((x-mean(x))^3)
  mu3.x/sd.x^3

}

f_kurt <- function(x){

  sd.x <- sd(x)
  mu4 <- mean((x-mean(x))^4)
  mu4/sd.x^4 - 3

}


f_gamma3 <- function(x){
  sd.x <- sd(x)
  mu3 <- mean((x-mean(x))^3)
  mu5 <- mean((x-mean(x))^5)
  mu5/sd.x^5-10*mu3/sd.x^3
}

f_gamma4 <- function(x){
  sd.x <- sd(x)
  mu3 <- mean((x-mean(x))^3)
  mu4 <- mean((x-mean(x))^4)
  mu6 <- mean((x-mean(x))^6)
  mu6/sd.x^6-15*mu4/sd.x^4 + 45-10*mu3^2/sd.x^3-15
}

headrick02.poly.coeff <- function(skewness, kurtosis, gam3, gam4, control = list(trace = T, max.ntry = 10, obj.tol = 1e-10, n.valid.sol
= 2)){

  gam1 <- skewness
  gam2 <- kurtosis

  gam <- c(gam1, gam2, gam3, gam4)

  obj.fun <- function(x, gam){

    if(length(x) != 6){
      stop("coefficients of fifth-order polynomial should be length-six")
    }

    c0 <- x[1]
    c1 <- x[2]
    c2 <- x[3]
    c3 <- x[4]
    c4 <- x[5]
    c5 <- x[6]

    gam1 <- gam[1]
    gam2 <- gam[2]
```

```
gam3 <- gam[3]
gam4 <- gam[4]

eq.18 <- 0 + c0 + c2 + 3 * c4
eq.22 <- -1 + c1^2 + 2 * c2^2 + 24 * c2 * c4 +
  6 * c1 * (c3 + 5 * c5) +
  3 * (5 * c3^2 + 32 * c4^2 + 70 * c3 * c5 + 315 * c5^2)

eq.B1 <- -gam1 + 2 * (
  4 * c2^3 + 108 * c2^2 * c4 + 3 * c1^2 * (c2 + 6 * c4) +
    18 * c1 * (2 * c2 * c3 + 16 * c3 * c4 + 15 * c2 * c5 + 150 * c4 * c5) +
    9 * c2 * (15 * c3^2 + 128 * c4^2 + 280 * c3 * c5 + 1575 * c5^2) +
    54 * c4 * (25 * c3^2 + 88 * c4^2 + 560 * c3 * c5 + 3675 * c5^2)
  )
eq.B2 <- -gam2 + 24 * (
  2 * c2^4 + 96 * c2^3 * c4 + c1^3 * (c3 + 10 * c5) +
    30 * c2^2 * (6 * c3^2 + 64 * c4^2 + 140 * c3 * c5 + 945 * c5^2) +
    c1^2 * (2 * c2^2 + 18 * c3^2 + 36 * c2 * c4 + 192 * c4^2 + 375 * c3 * c5 + 2250 * c5^2) +
    36 * c2 * c4 * (125 * c3^2 + 528 * c4^2 + 3360 * c3 * c5 + 25725 * c5^2) +
    3 * c1 * (45 * c3^3 + 1584 * c3 * c4^2 + 1590 * c3^2 * c5 + 21360 * c4^2 * c5 + 21525 * c3 * c5^2 +
            110250 * c5^3 + 12 * c2^2 * (c3 + 10 * c5) + 8 * c2 * c4 * (32 * c3 + 375 * c5)) +
    9 * (45 * c3^4 + 8704 * c4^4 + 2415 * c3^3 * c5 + 932400 * c4^2 * c5^2 + 3018750 * c5^4 +
         20 * c3^2 * (178 * c4^2 + 2765 * c5^2) + 35 * c3 * (3104 * c4^2 * c5 + 18075 * c5^3))
  )
eq.B3 <- -gam3 + 24 * (
  16 * c2^5 + 5 * c1^4 * c4 + 1200 * c2^4 * c4 + 10 * c1^3 * (3 * c2 * c3 + 42 * c3 * c4 + 40 * c2 * c5 + 570 * c4 * c5) +
    300 * c2^3 * (10 * c3^2 + 128 * c4^2 + 280 * c3 * c5 + 2205 * c5^2) +
    1080 * c2^2 * c4 * (125 * c3^2 + 3920 * c3 * c5 + 28 * (22 * c4^2 + 1225 * c5^2)) +
    10 * c1^2 * (2 * c2^3 + 72 * c2^2 * c4 + 3 * c2 * (24 * c3^2 + 320 * c4^2 + 625 * c3 * c5 + 4500 * c5^2) +
              9 * c4 * (109 * c3^2 + 528 * c4^2 + 3130 * c3 * c5 + 24975 * c5^2)) +
    30 * c1 * (8 * c2^3 * (2 * c3 + 25 * c5) + 40 * c2^2 * c4 * (16 * c3 + 225 * c5) +
             3 * c2 * (75 * c3^3 + 3168 * c3 * c4^2 + 3180 * c3^2 * c5 + 49840 * c4^2 * c5 + 50225 * c3 * c5^2 + 294000 * c5^3) +
             6 * c4 * (555 * c3^3 + 8704 * c3 * c4^2 + 26225 * c3^2 * c5 + 152160 * c4^2 * c5 + 459375 * c3 * c5^2 + 2963625 *
c5^3)) +
    90 * c2 * (270 * c3^4 + 16905 * c3^3 * c5 + 280 * c3^2 * (89 * c4^2 + 1580 * c5^2) +
             35 * c3 * (24832 * c4^2 * c5 + 162675 * c5^3) +
             4 * (17408 * c4^4 + 2097900 * c4^2 * c5^2 + 7546875 * c5^4)) +
    27 * c4 * (14775 * c3^4 + 1028300 * c3^3 * c5 + 50 * c3^2 * (10144 * c4^2 + 594055 * c5^2) +
             700 * c3 * (27904 * c4^2 * c5 + 598575 * c5^3) +
             3 * (316928 * c4^4 + 68908000 * c4^2 * c5^2 + 806378125 * c5^4))
  )
eq.B4 <- -gam4 + 120 * (
  32 * c2^6 + 3456 * c2^5 * c4 + 6 * c1^5 * c5 +
    3 * c1^4 * (9 * c3^2 + 16 * c2 * c4 + 168 * c4^2 + 330 * c3 * c5 + 2850 * c5^2) +
    720 * c2^4 * (15 * c3^2 + 224 * c4^2 + 490 * c3 * c5 + 4410 * c5^2) +
    6048 * c2^3 * c4 * (125 * c3^2 + 704 * c4^2 + 4480 * c3 * c5 + 44100 * c5^2) +
    12 * c1^3 * (4 * c2^2 * (3 * c3 + 50 * c5) + 60 * c2 * c4 * (7 * c3 + 114 * c5) +
              3 * (24 * c3^3 + 1192 * c3 * c4^2 + 1170 * c3^2 * c5 + 20440 * c4^2 * c5 +
                   20150 * c3 * c5^2 + 124875 * c5^3)) +
    216 * c2^2 * (945 * c3^4 + 67620 * c3^3 * c5 +
               560 * c3^2 * (178 * c4^2 + 3555 * c5^2) +
               315 * c3 * (12416 * c4^2 * c5 + 90375 * c5^3) +
               6 * (52224 * c4^4 + 6993000 * c4^2 * c5^2 + 27671875 * c5^4)) +
    6 * c1^2 * (8 * c2^4 + 480 * c2^3 * c4 +
             180 * c2^2 * (4 * c3^2 + 64 * c4^2 + 125 * c3 * c5 + 1050 * c5^2) +
             72 * c2 * c4 * (327 * c3^2 + 1848 * c4^2 + 10955 * c3 * c5 + 99900 * c5^2) +
             9 * (225 * c3^4 + 22824 * c3^2 * c4^2 + 69632 * c4^4 + 15090 * c3^3 * c5 +
                  830240 * c3 * c4^2 * c5 + 412925 * c3^2 * c5^2 +
                  8239800 * c4^2 * c5^2 + 5475750 * c3 * c5^3 + 29636250 * c5^4)) +
    1296 * c2 * c4 * (5910 * c3^4 + 462735 * c3^3 * c5 +
                   c3^2 * (228240 * c4^2 + 14851375 * c5^2) +
                   175 * c3 * (55808 * c4^2 * c5 + 1316865 * c5^3) +
                   3 * (158464 * c4^4 + 37899400 * c4^2 * c5^2 + 483826875 * c5^4)) +
    27 * (9945 * c3^6 + 92930048 * c4^6 + 1166130 * c3^5 * c5 +
          35724729600 * c4^4 * c5^2 + 977816385000 * c4^2 * c5^4 +
          1907724656250 * c5^6 + 180 * c3^4 * (16082 * c4^2 + 345905 * c5^2) +
          140 * c3^3 * (1765608 * c4^2 * c5 + 13775375 * c5^3) +
          15 * c3^2 * (4076032 * c4^4 + 574146160 * c4^2 * c5^2 +
                    2424667875 * c5^4) +
```

```
          210 * c3 * (13526272 * c4^4 * c5 + 687499200 * c4^2 * c5^3 +
                      1876468125 * c5^5)) +
      18 * c1 * (80 * c2^4 * (c3 + 15 * c5) + 160 * c2^3 * c4 * (32 * c3 + 525 * c5) +
               12 * c2^2 * (225 * c3^3 + 11088 * c3 * c4^2 + 11130 * c3^2 * c5 +
                          199360 * c4^2 * c5 + 200900 * c3 * c5^2 + 1323000 * c5^3) +
               24 * c2 * c4 * (3885 * c3^3 + 69632 * c3 * c4^2 + 209800 * c3^2 * c5 +
                            1369440 * c4^2 * c5 + 4134375 * c3 * c5^2 + 29636250 * c5^3) +
               9 * (540 * c3^5 + 48585 * c3^4 * c5 +
                    20 * c3^3 * (4856 * c4^2 + 95655 * c5^2) +
                    80 * c3^2 * (71597 * c4^2 * c5 + 513625 * c5^3) +
                    4 * c3 * (237696 * c4^4 + 30726500 * c4^2 * c5^2 +
                           119844375 * c5^4) +
                    5 * c5 * (4076032 * c4^4 + 191074800 * c4^2 * c5^2 +
                           483826875 * c5^4)))
  )

  eqs <- c(eq.18, eq.22, eq.B1, eq.B2, eq.B3, eq.B4)
  obj <- sum(eqs^2)

  obj

}

OPT <- list()
ntry <- 0
cnt <- 0
while(ntry+1 < control[["max.ntry"]]){

  ntry <- ntry + 1
  start <- rnorm(6, sd = .5)
  opt <- nlminb(start = start, objective = obj.fun, scale = 10,
                lower = -2, upper = 2,
                control = list(trace = F, abs.tol = 1e-20, rel.tol = 1e-15, eval.max = 1e6, iter.max = 1e6), gam = gam)
  #print(opt$objective)
  if(opt$convergence == 0 && opt$objective <= control[["obj.tol"]]){
    cnt <- cnt + 1
    OPT[[cnt]] <- opt
    if(control[["trace"]]){
      #cat(cnt, "/", ntry, "\n", sep="")
    }
  }


  if(length(OPT) >= control[["n.valid.sol"]] || (opt$objective <= min(1e-15, control[["obj.tol"]]) && opt$convergence == 0)){
    break
  }
}

if(length(OPT) == 0){
  return(NULL)
  stop(paste0("cannot find the coefficients of polynomial after ", control[["max.ntry"]], " attempts"))
}

min.obj <- 1e20
idx <- -1
for(i in 1:length(OPT)){
  #print(OPT[[i]]$objective)
  if(OPT[[i]]$objective < min.obj){
    min.obj <- OPT[[i]]$objective
    idx <- i
  }
}
if(control[["trace"]]){
  #cat("minimum objective: ", min.obj, "\n", sep="")
}

coeff <- OPT[[idx]]$par
list(coeff = coeff, min.obj = min.obj)
```

```
}

headrick02.corr.match <- function(poly.coeff, corr){

 obj.fun2 <- function(x, c0, c1, c2, c3, c4, c5, i, j, rho.Y){

  eq <- -rho.Y +
   3*c0[j]*c4[i] + 3*c2[j]*c4[i] + 9*c4[i]*c4[j] + c0[i]*(c0[j] + c2[j] + 3*c4[j]) +
   c1[i]*c1[j]*x + 3*c1[j]*c3[i]*x + 3*c1[i]*c3[j]*x + 9*c3[i]*c3[j]*x +
   15*c1[j]*c5[i]*x + 45*c3[j]*c5[i]*x + 15*c1[i]*c5[j]*x +
   45*c3[i]*c5[j]*x + 225*c5[i]*c5[j]*x + 12*c2[j]*c4[i]*x^2 +
   72*c4[i]*c4[j]*x^2 + 6*c3[i]*c3[j]*x^3 + 60*c3[j]*c5[i]*x^3 +
   60*c3[i]*c5[j]*x^3 + 600*c5[i]*c5[j]*x^3 + 24*c4[i]*c4[j]*x^4 +
   120*c5[i]*c5[j]*x^5 +
   c2[i]*(c0[j] + c2[j] + 3*c4[j] + 2*c2[j]*x^2 + 12*c4[j]*x^2)
  obj <- eq^2
  obj

 }

 k <- ncol(poly.coeff)

 c0 <- as.vector(poly.coeff[1,], mode = "numeric")
 c1 <- as.vector(poly.coeff[2,], mode = "numeric")
 c2 <- as.vector(poly.coeff[3,], mode = "numeric")
 c3 <- as.vector(poly.coeff[4,], mode = "numeric")
 c4 <- as.vector(poly.coeff[5,], mode = "numeric")
 c5 <- as.vector(poly.coeff[6,], mode = "numeric")


 l <- 0
 inter.corr <- diag(1, k)
 obj <- matrix(NA, k , k)
 for(i in 1:(k-1)){
  for(j in (i+1):k){
   l <- l + 1
   rho.Y <- corr[i, j]
   opt <- nlminb(start = rho.Y, objective = obj.fun2, scale = 10, lower = -1, upper = 1,
          control = list(trace = F, abs.tol = 1e-20, eval.max = 1e5, iter.max = 1e3),
          c0 = c0, c1 = c1, c2 = c2, c3 = c3, c4 = c4, c5 = c5, i = i, j = j, rho.Y = rho.Y)
   if(opt$convergence == 0){
    inter.corr[i, j] <- opt$par
    inter.corr[j, i] <- opt$par
    obj[i, j] <- opt$objective
   }else{
    stop("error in solving intermediate correlation")
   }

  }
 }

 list(inter.corr = inter.corr, obj = obj)

}

headrick02 <- function(n, mean, sd, corr, skewness, kurtosis, gam3=NaN, gam4=NaN, replication = 1, control = list(seed = NULL,
trace = T, max.ntry = 5, obj.tol = 1e-10, n.valid.sol = 1)){

 ##setting up

 if (! file.exists("compiled.txt")){
        print("Error: compiled.txt not found. Please change the working directory and try again.")
        return
 }


 if(!is.null(control[["seed"]])){
  set.seed(control[["seed"]])
```

```
}

if(is.null(control[["trace"]])){
  control[["trace"]] <- T
}

if(is.null(control[["max.ntry"]])){
  control[["max.ntry"]] <- 5
}

if(is.null(control[["obj.tol"]])){
  control[["obj.tol"]] <- 1e-10
}

if(is.null(control[["n.valid.sol"]])){
  control[["n.valid.sol"]] <- 1
}

start = Sys.time()

k <- nrow(corr)


if (is.nan(gam3[1]) && !is.nan(gam4[1])){
        print("Error: Please provide both gam3 and gam4, or neither.")
        return
}

if (is.nan(gam4[1]) && !is.nan(gam3[1])){
        print("Error: Please provide both gam3 and gam4, or neither.")
        return
}

default_gam3 = F
default_gam4 = F

if(is.nan(gam3[1])){
  gam3 = pmax(skewness, kurtosis)
        default_gam3 = T
}

if(is.nan(gam4[1])){
  gam4 = pmax(skewness,kurtosis)^2
        default_gam4 = T
}

len <- c(length(mean), length(sd), length(skewness), length(kurtosis), length(gam3), length(gam4))
if(var(len) != 0){
  stop("Lengths of mean, std, skewness, kurtosis, gam3 and gam4 must be equal")
}

if(len[1] != 1 && len[1] != k){
  stop("Inconsistent length/dim of moments and correlation")
}

if(len[1] == 1){
  mean <- rep(mean, k)
  sd <- rep(sd, k)
  skewness <- rep(skewness, k)
  kurtosis <- rep(kurtosis, k)
  gam3 <- rep(gam3, k)
  gam4 <- rep(gam4, k)
}

  for (i in 1:k){
        if (kurtosis[i] <= skewness[i]^2 - 2){
                cat("Error: the ", i," th component of kurtosis is not bigger than skewness squared minus 2.\n")
                return
```

```
        }
 }


 ##Solve for coefficients c0-c5 using equation 18, 22, B1-B4

 coeff <- NULL
 obj.poly.coeff <- NULL
 poly.coeff <- NULL
 gam4_fit = rep(0,k)
 gam3_fit = rep(0,k)

 for(i in 1:k){
		if (control[["trace"]]){
			cat("Time elapsed ", as.numeric(Sys.time()-start, units="secs") ," seconds. Start fitting c0 - c5 for
distribution ", i, ".\n", sep="")
		}

		compiled = read.table("compiled.txt", header = T)

		if (default_gam3 && default_gam4){
			matched = compiled[compiled["g1"]==skewness[i] & compiled["g2"]==kurtosis[i] &
compiled["tol"]<=control[["obj.tol"]],]
			if (nrow(matched)>0){
				if (control[["trace"]]){
					cat("Configuration found in compiled list. Compiled coefficients will be used.
\n")
				}
				matched = matched[order(-matched$tol),]
				curr.coeff = c(as.vector(matched[1,c("c0", "c1", "c2", "c3", "c4", "c5")]))
				curr.obj = matched[1, "tol"]
				gam3_fit[i] = matched[1, "g3"]
				gam4_fit[i] = matched[1, "g4"]
			}
			else{
				j <- 1
				j3 <- 1
				poly.coeff <- NULL
				upper = 4
				step_size = 4
				iterations = 0
				while(j<=15 && j3<=15 && is.null(poly.coeff)){
					iterations = iterations + 1
					tic = Sys.time()
					gam3_fit[i] = gam3[i]/2*2^j3
					gam4_fit[i] = gam4[i]/2*2^j
					poly.coeff <- headrick02.poly.coeff(skewness[i], kurtosis[i], gam3_fit[i],
gam4_fit[i], control = control)
					if(is.null(poly.coeff)){
						if (control[["trace"]]){
							cat("Trial ",iterations," unsuccessful. Time spent: ",
as.numeric(Sys.time()-tic, units="secs") , " seconds.\n", sep = "")
						}
						j3<-j3+1
						if(j3==upper+1 && j<upper){
							j3 <- 1
							j <- j+1
						}
						else if(j3==upper+1 && j==upper){
							input = "y"
							cat("No solutions found after ", iterations," iterations.
Do you want to continue searching? (y/n)\n", sep = "")

							input = readline()
							while(input!="y" && input !="n" && input != "Y" &&
input != "N"){

								cat("Invalid input. Please try again.\n")
								input = readline()
							}
							if(input=="y" || input == "Y"){
```

```
                                        j3 = 1
                                        j = j + 1
                                        upper = upper + step_size

                                    }
                                    if(input == "n" || input == "N"){
                                        break
                                    }
                                }
                            }
                        }
                        if(!is.null(poly.coeff)){
                            if (control[["trace"]]){
                                cat("Trial ",iterations," successful. Time spent: ",
as.numeric(Sys.time()-tic, units="secs") , " seconds.\n", sep = "")
                            }
                            curr.coeff = poly.coeff$coeff
                            curr.obj = poly.coeff$min.obj
                            to_append = matrix(c(skewness[i],kurtosis[i],curr.obj,gam3_fit[i],
gam4_fit[i], curr.coeff), nrow=1)
                            write.table(to_append, "compiled.txt",append = T, row.names = F, col.names
= F)
                        }

                        if (is.null(poly.coeff)){
                            cat("Error: no solution found for the combination of skewness: ", skewness[i],
"; kurtosis: ",

                                kurtosis[i], ".\n", sep = "")
                            return(NULL)
                        }
                    }
                }

                else{
                    matched = compiled[compiled["g1"]==skewness[i] & compiled["g2"]==kurtosis[i]
                        & compiled["g3"] == gam3[i] & compiled["g4"] == gam4[i]
                        & compiled["tol"]<=control[["obj.tol"]],]
                    if (nrow(matched)>0){
                        if (control[["trace"]]){
                            cat("Configuration found in compiled list. Compiled coefficients will be used.
\n")
                        }
                        matched = matched[order(-matched$tol)]
                        curr.coeff = c(as.vector(matched[1,c("c0", "c1", "c2", "c3", "c4", "c5")]))
                        curr.obj = matched[1, "tol"]
                        gam3_fit[i] = matched[1, "g3"]
                        gam4_fit[i] = matched[1, "g4"]
                    }
                    else {
                        gam3_fit[i]  = gam3[i]
                        gam4_fit[i] = gam4[i]
                        poly.coeff <- headrick02.poly.coeff(skewness[i], kurtosis[i], gam3_fit[i], gam4_fit[i],
control = control)
                        if(is.null(poly.coeff)){
                            cat("Error: no solution found for the combination of skewness: ", skewness[i],
"; kurtosis: ",

                                kurtosis[i], " gam3: ", gam3_fit[i], "; gam4: ", gam4_fit[i], ".\n",
sep = "")

                                return
                        }
                        else{
                            curr.coeff = poly.coeff$coeff
                            curr.obj = poly.coeff$min.obj
                            to_append = matrix(c(skewness[i],kurtosis[i],curr.obj,gam3_fit[i],
gam4_fit[i], curr.coeff), nrow=1)
                            write.table(to_append, "compiled.txt",append = T, row.names = F, col.names
= F)
                        }
                    }
```

```
}

coeff <- c(coeff, curr.coeff)
obj.poly.coeff <- c(obj.poly.coeff, curr.obj)
}

coeff = matrix(coeff, nrow = 6)

desired.moments <-data.frame(mean = mean, sd=sd, skewness = skewness, kurtosis = kurtosis, gam3 = gam3_fit,
gam4=gam4_fit)
rownames(desired.moments) <- paste0("Y", 1:nrow(desired.moments))

if (control[["trace"]]){
cat("Finished fitting c0 - c5. Time elapsed ", as.numeric(Sys.time()-start, units="secs") ," seconds.
\n", sep="")
}

summary.poly.coeff <- rbind(obj.poly.coeff, coeff)
colnames(summary.poly.coeff) <- paste0("Distribution ", 1:ncol(summary.poly.coeff))
rownames(summary.poly.coeff) <- c("obj value @ convergence", paste0("c", 0:5))
colnames(coeff) <- paste0("Distribution ", 1:ncol(summary.poly.coeff))
rownames(coeff) <- paste0("c", 0:5)

##Solve for intermediate correlation using equation 26
if(k>1){
if (control[["trace"]]){
cat("\nStart solving for intermediate correlation matrix...\n")
}

corr.match <- headrick02.corr.match(coeff, corr)
inter.corr <- corr.match$inter.corr
obj.corr.match <- corr.match$obj
colnames(inter.corr) <- paste0("Z", 1:ncol(inter.corr))
rownames(inter.corr) <- paste0("Z", 1:nrow(inter.corr))

colnames(obj.corr.match) <- paste0("Z", 1:ncol(obj.corr.match))
rownames(obj.corr.match) <- paste0("Z", 1:nrow(obj.corr.match))

if (control[["trace"]]){
cat("Finished solving for intermediate correlation matrix. Time elapsed ", as.numeric(Sys.time()-
start, units="secs") ," seconds.\n", sep="")
}
}
else{
inter.corr <- corr
colnames(inter.corr) <- paste0("Z", 1:ncol(inter.corr))
rownames(inter.corr) <- paste0("Z", 1:nrow(inter.corr))

}

c0 <- as.vector(coeff[1,], mode = "numeric")
c1 <- as.vector(coeff[2,], mode = "numeric")
c2 <- as.vector(coeff[3,], mode = "numeric")
c3 <- as.vector(coeff[4,], mode = "numeric")
c4 <- as.vector(coeff[5,], mode = "numeric")
c5 <- as.vector(coeff[6,], mode = "numeric")
library("MASS")

obs.mean = NULL
obs.sd = NULL
obs.skew = NULL
obs.kurt = NULL
obs.gam3 = NULL
obs.gam4 = NULL

for (replica in 1:replication){
## Generate intermediate normal distribution with desired intermediate correlation

Z <- mvrnorm(n, mu = rep(0, k), Sigma = inter.corr)
```

```
                    Z2 <- Z^2
                    Z3 <- Z^3
                    Z4 <- Z^4
                    Z5 <- Z^5

                    ## Generate multivariate distribution with desired property

                    Y <- matrix(0, nrow = n, ncol = k)
                    for(i in 1:k){
                            Y[, i] <- mean[i] + sd[i]*(c0[i] + c1[i] * Z[, i] + c2[i] * Z2[, i] + c3[i] * Z3[, i] + c4[i] *
Z4[, i] + c5[i] * Z5[, i])
                        }

                    obs.mean = rbind(obs.mean, apply(Y, 2, mean))
                    obs.sd = rbind(obs.sd, apply(Y, 2, sd))
                    obs.skew = rbind(obs.skew, apply(Y, 2, f_skew))
                    obs.kurt = rbind(obs.kurt, apply(Y, 2, f_kurt))
                    obs.gam3 = rbind(obs.gam3, apply(Y, 2, f_gamma3))
                    obs.gam4 = rbind(obs.gam4, apply(Y, 2, f_gamma4))
                   }


                   obs.moments <- data.frame(mean = apply(obs.mean, 2, mean), sd = apply(obs.sd, 2, mean), skewness =
apply(obs.skew, 2, mean),
                                   kurtosis = apply(obs.kurt, 2, mean), gam3 =
apply(obs.gam3,2,mean), gam4 = apply(obs.gam4,2,mean))
                   obs.moments.sd <- data.frame(mean = apply(obs.mean, 2, sd), sd = apply(obs.sd, 2, sd), skewness =
apply(obs.skew, 2, sd),
                                   kurtosis = apply(obs.kurt, 2, sd), gam3 = apply(obs.gam3,2,sd),
gam4 = apply(obs.gam4,2,sd))
                   rownames(obs.moments) <- paste0("Y", 1:nrow(obs.moments))
                   rownames(obs.moments.sd) <- paste0("Y", 1:nrow(obs.moments.sd))

                   obs.corr <- cor(Y)
                   rownames(obs.corr) <- paste0("Y", 1:nrow(obs.corr))
                   colnames(obs.corr) <- paste0("Y ", 1:ncol(obs.corr))

                   if (replication>1){
                        obs.corr = NULL
                   }

          if (control[["trace"]]){
                        cat("\nDesired moments:\n")
                        print(desired.moments)

                        cat("\nSampling moments:\n")
                        print(obs.moments)

                        if (replication >1){
                            cat("\nSampling moment standard deviations:\n")
                            print(obs.moments.sd)
                        }

                        if(replication == 1){
                            cat("\nDesired correlation matrix:\n")
                            print(corr)

                            cat("\nSampling correlation matrix:\n")
                            print(obs.corr)
                        }
                        cat("\nTotal time elapsed ", as.numeric(Sys.time()-start, units="secs"), " seconds.\n", sep="")
                   }

          list(obs = Y, obs.corr = obs.corr, obs.moments = obs.moments, obs.moments.sd = obs.moments.sd, desired.corr = corr,
desired.moments = desired.moments,
                    summary.poly.coeff = summary.poly.coeff, inter.corr = inter.corr)

}
```

```
##################################################################################################
######################

#Set-up

library(moments)

Sigm1<-cor <- matrix(c(1,.52,.52,.52,.52,1,.52,.52,.52,.52,1,.52,.52,.52,.52,1),4,4) #correlation matrix

setwd("C:/Users/New User/Documents/Dissertation Code/") #set the working directory to the one where headrick02.R is located

pathToData<-"C:/Users/New User/Documents/Dissertation Code/testHn500r250/"

N <- 500
mean <- c(rep(0,4))
sd <- c(rep(1,4))
Sigm1
skewness <- c(rep(1.0,4))
kurtosis <- c(rep(1.0,4))

reps <- 250

#simulate multiple data sets
for(i in 1:reps){
	y<-as.data.frame(headrick02(N, mean, sd, Sigm1, skewness, kurtosis)$obs)
	write.table(x=y, file=paste(pathToData, "x", i, ".txt", sep=""), sep=",",
		row.names=FALSE, quote=FALSE, col.names=FALSE )
}

#check skewnwss and kurtosis values
simcheck<-matrix(NA, nrow=reps, ncol=8)
for(i in 1:nrow(simcheck)){
	y<-read.table(file=paste(pathToData, "x", i, ".txt", sep=""), sep=",",
		header=FALSE )
	simcheck[i,1]<-skewness(y[,1])
	simcheck[i,2]<-skewness(y[,2])
	simcheck[i,3]<-skewness(y[,3])
	simcheck[i,4]<-skewness(y[,4])
	simcheck[i,5]<-kurtosis(y[,1])-3
	simcheck[i,6]<-kurtosis(y[,2])-3
	simcheck[i,7]<-kurtosis(y[,3])-3
	simcheck[i,8]<-kurtosis(y[,4])-3
}

mystats<-matrix(c(median(simcheck[,1]), mean(simcheck[,1]), sd(simcheck[,1]), median(simcheck[,5]), mean(simcheck[,5]),
sd(simcheck[,5]),
		median(simcheck[,2]), mean(simcheck[,2]), sd(simcheck[,2]), median(simcheck[,6]), mean(simcheck[,6]),
sd(simcheck[,6]),
		median(simcheck[,3]), mean(simcheck[,3]), sd(simcheck[,3]), median(simcheck[,7]), mean(simcheck[,7]),
sd(simcheck[,7]),
		median(simcheck[,4]), mean(simcheck[,4]), sd(simcheck[,4]), median(simcheck[,8]), mean(simcheck[,8]),
sd(simcheck[,8])),
		nrow=4, ncol=6, byrow=TRUE)

write.table(x=mystats, file=paste(pathToData, "simcheck.txt", sep=""), sep=",",
		row.names=FALSE, quote=FALSE, col.names=FALSE)
```

Simulation Code: R Syntax for Vale and Maurelli (1983)

```
# 2011

#University of Minnesota

#Cengiz Zopluoglu

#  Applications in R:

#Generating Multivariate Non-normal Variables


#######################################################################
#                                                                     #
#                                                                     #
# R Script to Generate Non-normal Distributions                  #
#                                                                               #
#                                                                               #
# Method described in:                                                #
# Vale, C. & Maurelli, V. (1983). Simulating multivariate        #
# nonnormal distributions. Psychometrika, 48(3):465-471.         #
#                                                                               #
# Programmed by: Cengiz Zopluoglu                                  #
# April 20,2011                                                       #
#######################################################################
#######################################################################

#Inputs:
#n , sample size
#k , number of variables
#cor , desired correlation matrix between bivariate non-normal variables, k x k
#skew , a vector of k elements, skewness for the variables
#kurt , a vector of k elements, kurtosis for the variables
#k , number of variables
#############################################################################

library(moments)

gen.nonnormal <- function(n,cor,skew,kurt,k) { #Start Main Function

#Internal Function to compute the a,b,c,d for a variable given the skewness and
#kurtosis. Use Newton-Raphson Iteration with a Jacobian matrix to solve the system of
#non-linear equations. Equation 2,3, and 4 in Vale & Maurelli(1983)

tol=.00001

constant <- function(sk,ku,start){ #Start Internal Function 1

#sk , desired skewness
#ku , desired kurtosis
#start, starting values for the iteratin, based on Fleishman(1978) using c(1,0,0)
#is reasonable

start=c(1,0,0)
max.iter <- 500
F <- function(x){
                    F <- matrix(0,nrow=3)
                    b=x[1]
                    c=x[2]
                    d=x[3]
                    F[1]= b^2+6*b*d+2*c^2+15*d^2-1
                    F[2]= 2*c*(b^2+24*b*d+105*d^2+2)-sk
                    F[3]=24*(b*d+c^2*(1+b^2+28*b*d)+d^2*(12+48*b*d+141*c^2+225*d^2))-ku
                    F
        }
```

```
J <- function(x){
                b=x[1]
                c=x[2]
                d=x[3]
                j=matrix(0,ncol=3,nrow=3)
                j[1,1]= 2*b+6*d
                j[1,2]= 4*c
                j[1,3]= 6*b+30*d

                j[2,1]= 4*b*c+48*c*d
                j[2,2]= 2*b^2+48*b*d+210*d^2+4
                j[2,3]= 48*b*c+420*c*d
                j[3,1]=24*d+48*c^2*b
                j[3,2]=48*c+48*c*b^2+1344*c*b*d+6768*c*d^2
                j[3,3]=24*b+672*c^2*b+576*d+3456*b*d^2+6768*d*c^2+21600*d^3
                j
        }

x0 <- start
fx <- F(x0)
jx <- J(x0)
d <- solve(J(x0))%*%F(x0)
iter <- 0

d=det(J(x0))
if (identical(all.equal(d,0),TRUE))
{cat("Jacobian has no inverse. Try a different initial point.","\n")
break}

while((abs(d)> tol) && (iter < max.iter)) {
                x0 <- x0-solve(J(x0))%*%F(x0)
                d <- solve(J(x0))%*%F(x0)
                fx <- F(x0)
                jx <- J(x0)
                iter <- iter+1
        }
x0
} #End internal function 1

#Compute the constants a,b,c, and d for each variable with a desired skewness and
#kurtosis

constants <- matrix(nrow=k,ncol=4)
        for(i in 1:k) {
                constants[i,2:4]=t(constant(skew[i],kurt[i],start=c(1,0,0)))
                constants[i,1]=-(constants[i,3])
        }

#Internal Function to solve the polynomial function to find the intermediate
#correlation between two normal variables for a given desired correlation between two
#non-normal variables and constants a,b,c,d Use Newton-Raphson iteration to
#approximate the root

solve.p12 <- function(r12,a1,a2,b1,b2,c1,c2,d1,d2) { #Start Internal Function 2

        max.iter=500
        start=.5

        ftn <- function(p12) {
                a <-((b1*b2+3*b1*d2+3*d1*b2+9*d1*d2)*p12)+((2*c1*c2)*p12^2)+((6*d1*d2)*p12^3)-r12
                b <-(b1*b2+3*b1*d2+3*d1*b2+9*d1*d2)+((4*c1*c2)*p12)+((12*d1*d2)*p12^2)
                c(a,b)
        }

p12 <- start
fx <- ftn(p12)
iter <- 0
while((abs(fx[1]) > tol) && (iter < max.iter)) {
                p12 <- p12-fx[1]/fx[2]
```

```
            fx <- ftn(p12)
            iter <- iter+1
        }
p12
} #End Internal Function 2

#Compute the intermediate intercorrelation matrix required for normal variables
#These normal variables are used to construct non-normal variables

inter <- matrix(0,k,k)
        for(i in 1:k) {
            for(j in 1:k) {
                inter[i,j]=solve.p12(cor[i,j],constants[i,1],constants[j,1],constants[i,2],constants[j,2],constants[i,3],
                constants[j,3],constants[i,4],constants[j,4])
            }
        }
diag(inter) <- 1

#Compute the multivariate normal variables based on the intermediate intercorrelation #matrix
#Eigen decomposition of correlation matrix

U <- eigen(inter)$vectors
L <- eigen(inter)$values
b <- U%*%diag(sqrt(L))

#Creating independent multivariate normal variables

normal <- matrix(nrow=n,ncol=k)
        for(i in 1:k) { normal[,i]=rnorm(n,0,1) }

#Creating correlated multivariate normal variables

d <- as.data.frame(normal%*%t(b))

#Creating correlated non-normal multivariate variables from correlated multivariate #normal variables using constants a,b,c, and d

nonnormal <- as.data.frame(matrix(nrow=n,ncol=k))
        for(i in 1:k) {
            nonnormal[,i]=constants[i,1]+ constants[i,2]*d[,i]+constants[i,3]*d[,i]^2+constants[i,4]*d[,i]^3
        }
nonnormal
} #End Main function




#Create two functions to compute the skewness and kurtosis of a variable:

kurt <- function(x) {
            m4 <- mean((x-mean(x))^4)
            kurt <- m4/(sd(x)^4)-3
            kurt
        }
skew <- function(x) {
            m3 <- mean((x-mean(x))^3)
            skew <- m3/(sd(x)^3)
            skew
        }

#Check if the function gen.nonnormal() does what you expect:

#1) Set the parameters

n <- 2000 # sample size
```

```
r12=.52 # correlation between v1 and v2
r13=.52 # correlation between v1 and v3
r14=.52 # correlation between v1 and v4
r15=.52 # correlation between v1 and v5
r16=.52 # correlation between v1 and v6
r17=.52 # correlation between v1 and v7
r18=.52 # correlation between v1 and v8
r23=.52 # correlation between v2 and v3
r24=.52 # correlation between v2 and v4
r25=.52 # correlation between v2 and v5
r26=.52 # correlation between v2 and v6
r27=.52 # correlation between v2 and v7
r28=.52 # correlation between v2 and v8
r34=.52 # correlation between v3 and v4
r35=.52 # correlation between v3 and v5
r36=.52 # correlation between v3 and v6
r37=.52 # correlation between v3 and v7
r38=.52 # correlation between v3 and v8
r45=.52 # correlation between v4 and v5
r46=.52 # correlation between v4 and v6
r47=.52 # correlation between v4 and v7
r48=.52 # correlation between v4 and v8
r56=.52 # correlation between v5 and v6
r57=.52 # correlation between v5 and v7
r58=.52 # correlation between v5 and v8
r67=.52 # correlation between v6 and v7
r68=.52 # correlation between v6 and v8
r78=.52 # correlation between v7 and v8

cor <- matrix(c(1,r12,r13,r14,r15,r16,r17,r18,
                r12,1,r23,r24,r25,r26,r27,r28,
                r13,r23,1,r34,r35,r36,r37,r38,
                r14,r24,r34,1,r45,r46,r47,r48,
                r15,r25,r35,r45,1,r56,r57,r58,
                r16,r26,r36,r46,r56,1,r67,r68,
                r17,r27,r37,r47,r57,r67,1,r78,
                r18,r28,r38,r48,r58,r68,r78,1),8,8) #correlation matrix

av1 <- 1.00
av2 <- 1.00
vec1 <- c(av1,av1,av1,av1,av1,av1,av1,av1)
vec2 <- c(av2,av2,av2,av2,av2,av2,av2,av2)

sk <- vec1
ku <- vec2
#sk <- c(1.0,1.0,1.0,1.0,1.0,1.0,1.0,1.0) #desired level of skewness for each variable
#ku <- c(1.0,1.0,1.0,1.0,1.0,1.0,1.0,1.0) # desired level of kurtosis for each variables
k <-8 # number of variables

#2) Generate multivariate non-normal data with desired level of skewness and kurtosis

nonnormal <- gen.nonnormal(n,cor,sk,ku,k)

#3) See if it’s enough for you (It’s not always perfect, and does not work for all combinations of skewness and kurtosis)

nonnormal <- gen.nonnormal(n,cor,sk,ku,k)
mean(nonnormal[,1]);mean(nonnormal[,2]);mean(nonnormal[,3]);mean(nonnormal[,4]);
      mean(nonnormal[,5]);mean(nonnormal[,6]);mean(nonnormal[,7]);mean(nonnormal[,8])
sd(nonnormal[,1]);sd(nonnormal[,2]);sd(nonnormal[,3]);sd(nonnormal[,4]);
      sd(nonnormal[,5]);sd(nonnormal[,6]);sd(nonnormal[,7]);sd(nonnormal[,8])
skew(nonnormal[,1]);skew(nonnormal[,2]);skew(nonnormal[,3]);skew(nonnormal[,4]);
      skew(nonnormal[,5]);skew(nonnormal[,6]);skew(nonnormal[,7]);skew(nonnormal[,8])
kurt(nonnormal[,1]);kurt(nonnormal[,2]);kurt(nonnormal[,3]);kurt(nonnormal[,4]);
      kurt(nonnormal[,5]);kurt(nonnormal[,6]);kurt(nonnormal[,7]);kurt(nonnormal[,8])
cor(nonnormal)
#hist(nonnormal[,1])
#hist(nonnormal[,2])
#hist(nonnormal[,3])
#require(MASS)
```

```
#bivn.kde1 <- kde2d(nonnormal[,1],nonnormal[,2], n = 50)
#bivn.kde2 <- kde2d(nonnormal[,1],nonnormal[,3], n = 50)
#bivn.kde3 <- kde2d(nonnormal[,2],nonnormal[,3], n = 50)
#persp(bivn.kde1, phi = 30, theta = 30,xlab="X")
#persp(bivn.kde2, phi = 30, theta = 30,xlab="X")
#persp(bivn.kde3, phi = 30, theta = 30,xlab="X")


#simulate multiple data sets

pathToData<-"C:/Dissertation/SIMDATA8/c6/"

reps <- 250

#simulate multiple data sets
for(i in 1:reps){
        y<-as.data.frame(gen.nonnormal(n,cor,sk,ku,k))
        write.table(x=y, file=paste(pathToData, "x", i, ".txt", sep=""), sep=",",
                row.names=FALSE, quote=FALSE, col.names=FALSE )
}

#check skewnwss and kurtosis values

simcheck<-matrix(NA, nrow=reps, ncol=16)

for(i in 1:nrow(simcheck)){
        y<-read.table(file=paste(pathToData, "x", i, ".txt", sep=""), sep=",",
                header=FALSE )
        simcheck[i,1]<-skewness(y[,1])
        simcheck[i,2]<-skewness(y[,2])
        simcheck[i,3]<-skewness(y[,3])
        simcheck[i,4]<-skewness(y[,4])
        simcheck[i,5]<-kurtosis(y[,1])-3
        simcheck[i,6]<-kurtosis(y[,2])-3
        simcheck[i,7]<-kurtosis(y[,3])-3
        simcheck[i,8]<-kurtosis(y[,4])-3
        simcheck[i,9]<-skewness(y[,5])
        simcheck[i,10]<-skewness(y[,6])
        simcheck[i,11]<-skewness(y[,7])
        simcheck[i,12]<-skewness(y[,8])
        simcheck[i,13]<-kurtosis(y[,5])-3
        simcheck[i,14]<-kurtosis(y[,6])-3
        simcheck[i,15]<-kurtosis(y[,7])-3
        simcheck[i,16]<-kurtosis(y[,8])-3

}

mystats<-matrix(c(median(simcheck[,1]), mean(simcheck[,1]), sd(simcheck[,1]), min(simcheck[,1]), max(simcheck[,1]),
                median(simcheck[,5]), mean(simcheck[,5]), sd(simcheck[,5]), min(simcheck[,5]), max(simcheck[,5]),
                median(simcheck[,2]), mean(simcheck[,2]), sd(simcheck[,2]), min(simcheck[,2]), max(simcheck[,2]),
                median(simcheck[,6]), mean(simcheck[,6]), sd(simcheck[,6]), min(simcheck[,6]), max(simcheck[,6]),
                median(simcheck[,3]), mean(simcheck[,3]), sd(simcheck[,3]), min(simcheck[,3]), max(simcheck[,3]),
                median(simcheck[,7]), mean(simcheck[,7]), sd(simcheck[,7]), min(simcheck[,7]), max(simcheck[,7]),
                median(simcheck[,4]), mean(simcheck[,4]), sd(simcheck[,4]), min(simcheck[,4]), max(simcheck[,4]),
                median(simcheck[,8]), mean(simcheck[,8]), sd(simcheck[,8]), min(simcheck[,8]), max(simcheck[,8]),
                median(simcheck[,9]), mean(simcheck[,9]), sd(simcheck[,9]), min(simcheck[,9]), max(simcheck[,9]),
                median(simcheck[,13]), mean(simcheck[,13]), sd(simcheck[,13]), min(simcheck[,13]), max(simcheck[,13]),
                median(simcheck[,10]), mean(simcheck[,10]), sd(simcheck[,10]), min(simcheck[,10]), max(simcheck[,10]),
                median(simcheck[,14]), mean(simcheck[,14]), sd(simcheck[,14]), min(simcheck[,14]), max(simcheck[,14]),
                median(simcheck[,11]), mean(simcheck[,11]), sd(simcheck[,11]), min(simcheck[,11]), max(simcheck[,11]),
                median(simcheck[,15]), mean(simcheck[,15]), sd(simcheck[,15]), min(simcheck[,15]), max(simcheck[,15]),
                median(simcheck[,12]), mean(simcheck[,12]), sd(simcheck[,12]), min(simcheck[,12]), max(simcheck[,12]),
                median(simcheck[,16]), mean(simcheck[,16]), sd(simcheck[,16]), min(simcheck[,16]), max(simcheck[,16])),
                nrow=8, ncol=10, byrow=TRUE)

write.table(x=mystats, file=paste(pathToData, "simcheck.txt", sep=""), sep=",",
                row.names=FALSE, quote=FALSE, col.names=FALSE)

mystats
```

Appendix B

R Syntax for Creating Mplus Input Files

```
library(MplusAutomation)


#Change the last statement to change Mplus model (i.e. a1 to a6)
#Change file name var4 to var8 for data conditions 7 through 12.

createModels("C:/Dissertation/MplusFiles/var4/a1.txt")

#This calls the file that contains the instructions for creating the Mplus input files:

[[init]]
iterators = condition datanum classes;
condition = 1:6;
datanum = 1:250;
classes = 1:4;
filename = "x[[datanum]]-class-[[classes]]-a1.inp";
outputDirectory = "C:/Dissertation/MplusFiles/var4/c[[condition]]/a1/o[[datanum]]";
[[/init]]

TITLE: a1

DATA: FILE IS C:\Dissertation\SIMDATA4\c[[condition]]\x[[datanum]].txt;
VARIABLE: NAMES ARE x1-x4;
CLASSES = c ([[classes]]);
ANALYSIS: TYPE = MIXTURE;
        STARTS = 200 50;
        STITERATIONS = 20;
        ALGORITHM = EM;
           PROCESSORS = 8;
           DISTRIBUTION = NORMAL;
         [[classes > 1]]
                K-1STARTS = 200 50;
                LRTSTARTS = 200 50 200 50;

         [[/classes > 1]]



MODEL:
                %OVERALL%
                x1 WITH x2 x3 x4;
                x2 WITH x3 x4;
                x3 WITH x4;
```

```
[[classes > 1]]
[[classes = 2]]
        %c#1%
        x1 WITH x2 x3 x4;
        x2 WITH x3 x4;
        x3 WITH x4;

        %c#1%
        x1 WITH x2 x3 x4;
        x2 WITH x3 x4;
        x3 WITH x4;
[[/classes = 2]]

[[classes = 3]]
        %c#1%
        x1 WITH x2 x3 x4;
        x2 WITH x3 x4;
        x3 WITH x4;

        %c#2%
        x1 WITH x2 x3 x4;
        x2 WITH x3 x4;
        x3 WITH x4;

        %c#3%
        x1 WITH x2 x3 x4;
        x2 WITH x3 x4;
        x3 WITH x4;
[[/classes = 3]]


[[classes = 4]]
        %c#1%
        x1 WITH x2 x3 x4;
        x2 WITH x3 x4;
        x3 WITH x4;

        %c#2%
        x1 WITH x2 x3 x4;
        x2 WITH x3 x4;
        x3 WITH x4;

        %c#3%
        x1 WITH x2 x3 x4;
        x2 WITH x3 x4;
        x3 WITH x4;
```

```
            %c#4%
            x1 WITH x2 x3 x4;
            x2 WITH x3 x4;
            x3 WITH x4;
        [[/classes = 4]]

        [[/classes > 1]]


OUTPUT: TECH1 TECH8 [[classes > 1]] TECH11 TECH14 [[/classes > 1]];
```

R Syntax for Running the Created Mplus Input Files

```
library(MplusAutomation)

#This will open the created Mplus input files, run each in turn, and return output files.


#Be sure that the path object refers to the correct model folder which will
#match the name of the createModels file.

numConditions = 6
datanum = 250
basePath<-"C:/Dissertation/MplusFiles/var4/"
for(i in 1:numConditions){
	path<-paste(basePath, "c", i, "/a1/", sep="")

	for(j in 1:datanum){
		path2<-paste(path, "o", j, sep="")
		runModels(path2, recursive=TRUE, replaceOutfile="always")
	}
}
```

R Syntax for Extracting Mplus Output and Calculating Fit Statistics

```
library(MplusAutomation)
library(stringi)


#Be sure that the fitpath object below refers to the correct model folder.

#Also change the condition folder (c1, c2, c3,...) in both the fitpath
#object and the write.table file statement at the end of the loop.

datanum = 250
fitpath<-"C:/Dissertation/MplusFiles/var4/c1/a1/"
for(j in 1:datanum){
        path3<-paste(fitpath, "o", j, sep="")
        x=extractModelSummaries(path3,recursive=TRUE)

        for(i in 1:datanum){

        File=x$Filename
        n<-x$Observations

                nParx<-c(x$Parameters)
        nPar<-if(is.null(nParx)) (c(NA,NA,NA,NA)) else nParx

                LLx<-c(x$LL)
        LL<-if(is.null(LLx)) (c(NA,NA,NA,NA)) else LLx
        ll2<--2*LL

        class <- as.integer(stri_sub(File,-9,-9))

                ENx<-c(x$Entropy)
                ENy<-if(is.null(ENx)) (c(NA,NA,NA,NA)) else ENx
                ENa<-replace(ENy,1,1)
        EN<- ((1-ENa)*(n*log(class)))

                BLRTKM1LLx<-c(x$BLRT_KM1LL)
        BLRTKM1LL<-if(is.null(BLRTKM1LLx)) (c(NA,NA,NA,NA)) else BLRTKM1LLx

                BLRTpx<-c(x$BLRT_PValue)
        BLRTp<-if(is.null(BLRTpx)) (c(NA,NA,NA,NA)) else BLRTpx

                LMRKM1LLx<-c(x$T11_KM1LL)
        LMRKM1LL<-if(is.null(LMRKM1LLx)) (c(NA,NA,NA,NA)) else LMRKM1LLx

                LMR2xLLDiffx<-c(x$T11_VLMR_2xLLDiff)
        LMR2xLLDiff<-if(is.null(LMR2xLLDiffx)) (c(NA,NA,NA,NA)) else LMR2xLLDiffx

                LMRPDiffx<-c(x$T11_VLMR_ParamDiff)
        LMRPDiff<-if(is.null(LMRPDiffx)) (c(NA,NA,NA,NA)) else LMRPDiffx

                LMRMeanx<-c(x$T11_VLMR_Mean)
        LMRMean<-if(is.null(LMRMeanx)) (c(NA,NA,NA,NA)) else LMRMeanx

                LMRSDx<-c(x$T11_VLMR_SD)
        LMRSD<-if(is.null(LMRSDx)) (c(NA,NA,NA,NA)) else LMRSDx

                LMRpx<-c(x$T11_VLMR_PValue)
        LMRp<-if(is.null(LMRpx)) (c(NA,NA,NA,NA)) else LMRpx

                aLMRx<-c(x$T11_LMR_Value)
```

```
aLMR<-if(is.null(aLMRx)) (c(NA,NA,NA,NA)) else aLMRx

        aLMRpx<-c(x$T11_LMR_PValue)
aLMRp<-if(is.null(aLMRpx)) (c(NA,NA,NA,NA)) else aLMRpx

aic<-ll2+(2*nPar)
bic<-ll2+(nPar*log(n))
caic<-ll2+(nPar*(log(n)+1))
sabic<-ll2+(nPar*(log((n+2)/24)))
sacaic<-ll2+(nPar*(log((n+2)/24)+1))
clc<-ll2+(2*EN)
iclbic<-ll2+(2*EN)+(nPar*log(n))


aicR=rank(aic, na.last = "keep", ties.method = "first")
bicR=rank(bic, na.last = "keep", ties.method = "first")
caicR=rank(caic, na.last = "keep", ties.method = "first")
sabicR=rank(sabic, na.last = "keep", ties.method = "first")
sacaicR=rank(sacaic, na.last = "keep", ties.method = "first")
clcR=rank(clc, na.last = "keep", ties.method = "first")
iclbicR=rank(iclbic, na.last = "keep", ties.method = "first")

fit1=data.frame(File=File,class=class,LL=LL,ll2=ll2,EN=EN,aic=aic,bic=bic,caic=caic,
                sabic=sabic,sacaic=sacaic,clc=clc,iclbic=iclbic,aicR=aicR,bicR=bicR,caicR=caicR,

sabicR=sabicR,sacaicR=sacaicR,clcR=clcR,iclbicR=iclbicR,BLRTKM1LL=BLRTKM1LL,

BLRTp=BLRTp,LMRKM1LL=LMRKM1LL,LMR2xLLDiff=LMR2xLLDiff,LMRPDiff=LMRPDiff,

LMRMean=LMRMean,LMRSD=LMRSD,LMRp=LMRp,aLMR=aLMR,aLMRp=aLMRp)

conv=ifelse(sum(fit1$aicR)==10, 1, NA)

fit2=data.frame(File=File,class=class,LL=LL,ll2=ll2,EN=EN,aic=aic,bic=bic,caic=caic,
                sabic=sabic,sacaic=sacaic,clc=clc,iclbic=iclbic,aicR=aicR,bicR=bicR,caicR=caicR,
                sabicR=sabicR,sacaicR=sacaicR,clcR=clcR,iclbicR=iclbicR,conv=conv,

BLRTKM1LL=BLRTKM1LL,BLRTp=BLRTp,LMRKM1LL=LMRKM1LL,LMR2xLLDiff=LMR2xLLDif
f,

LMRPDiff=LMRPDiff,LMRMean=LMRMean,LMRSD=LMRSD,LMRp=LMRp,aLMR=aLMR,
                aLMRp=aLMRp)

}
write.table(fit2, file = "H:/Results/A1C1Results",
                append = TRUE, sep = " ", na = "NA", col.names = FALSE)
}
```

Appendix C

Table C1.

Convergence rates for the one through four class models for analysis conditions one through three and data conditions one through twelve.

Condition	*1N*	1P	*2N*	2P	*3N*	3P	*4N*	4P
A1C01	250	100	198	79.2	172	68.8	139	55.6
A1C02	250	100	250	100	243	97.2	218	87.2
A1C03	250	100	250	100	248	99.2	233	93.2
A1C04	250	100	186	74.4	180	72	144	57.6
A1C05	250	100	250	100	249	99.6	250	100
A1C06	250	100	250	100	248	99.2	249	99.6
A1C07	250	100	232	92.8	212	84.8	206	82.4
A1C08	250	100	250	100	237	94.8	226	90.4
A1C09	250	100	249	99.6	244	97.6	238	95.2
A1C10	250	100	236	94.4	223	89.2	215	86
A1C11	250	100	250	100	250	100	249	99.6
A1C12	250	100	250	100	250	100	250	100
A2C01	250	100	250	100	250	100	249	99.6
A2C02	250	100	250	100	250	100	250	100
A2C03	250	100	250	100	250	100	250	100
A2C04	250	100	246	98.4	230	92	168	67.2
A2C05	250	100	250	100	250	100	250	100
A2C06	250	100	250	100	250	100	250	100
A2C07	250	100	249	99.6	250	100	249	99.6
A2C08	250	100	250	100	250	100	250	100
A2C09	250	100	250	100	250	100	250	100
A2C10	250	100	250	100	247	98.8	235	94
A2C11	250	100	250	100	249	99.6	250	100
A2C12	250	100	250	100	250	100	250	100
A3C01	250	100	196	78.4	201	80.4	158	63.2
A3C02	**203**	**81.2**	**179**	**71.6**	**137**	**54.8**	**96**	**38.4**
A3C03	**138**	**55.2**	**162**	**64.8**	**117**	**46.8**	**58**	**23.2**
A3C04	250	100	229	91.6	226	90.4	210	84
A3C05	**246**	**98.4**	**235**	**94**	**212**	**84.8**	**185**	**74**
A3C06	**249**	**99.6**	**242**	**96.8**	**213**	**85.2**	**175**	**70**
A3C07	250	100	191	76.4	105	42	28	11.2
A3C08	**184**	**73.6**	**96**	**38.4**	**36**	**14.4**	**12**	**4.8**
A3C09	**37**	**14.8**	**115**	**46**	**28**	**11.2**	**8**	**3.2**
A3C10	250	100	222	88.8	194	77.6	126	50.4
A3C11	**171**	**68.4**	**245**	**98**	**170**	**68**	**141**	**56.4**
A3C12	**117**	**46.8**	**244**	**97.6**	**181**	**72.4**	**153**	**61.2**

Note: conditions when one class model did not converge 100% in bold type.

Table C2.
Convergence rates for the one through four class models for analysis conditions four through six and data conditions one through twelve.

Condition	*1N*	1P	*2N*	2P	*3N*	3P	*4N*	4P
A4C01	250	100	244	97.6	243	97.2	243	97.2
A4C02	203	81.2	242	96.8	240	96	243	97.2
A4C03	**137**	**54.8**	**243**	**97.2**	**248**	**99.2**	**243**	**97.2**
A4C04	250	100	249	99.6	247	98.8	236	94.4
A4C05	**246**	**98.4**	**250**	**100**	**248**	**99.2**	**247**	**98.8**
A4C06	**249**	**99.6**	**250**	**100**	**249**	**99.6**	**248**	**99.2**
A4C07	250	100	247	98.8	248	99.2	242	96.8
A4C08	**184**	**73.6**	**225**	**90**	**237**	**94.8**	**241**	**96.4**
A4C09	**37**	**14.8**	**198**	**79.2**	**242**	**96.8**	**235**	**94**
A4C10	250	100	248	99.2	249	99.6	248	99.2
A4C11	**171**	**68.4**	**250**	**100**	**247**	**98.8**	**244**	**97.6**
A4C12	**117**	**46.8**	**248**	**99.2**	**247**	**98.8**	**248**	**99.2**
A5C01	125	100	99	79.2	93	74.4	56	44.8
A5C02	**123**	**98.4**	**82**	**65.6**	**51**	**40.8**	**33**	**26.4**
A5C03	**115**	**92**	**69**	**55.2**	**35**	**28**	**20**	**16**
A5C04	125	100	111	88.8	115	92	95	76
A5C05	125	100	123	98.4	119	95.2	107	85.6
A5C06	125	100	124	99.2	124	99.2	110	88
A5C07	125	100	96	76.8	64	51.2	55	44
A5C08	**60**	**48**	**28**	**22.4**	**13**	**10.4**	**10**	**8**
A5C09	**8**	**6.4**	**22**	**17.6**	**9**	**7.2**	**4**	**3.2**
A5C10	125	100	103	82.4	94	75.2	38	30.4
A5C11	125	100	122	97.6	83	66.4	47	37.6
A5C12	125	100	118	94.4	48	38.4	23	18.4
A6C01	125	100	125	100	125	100	125	100
A6C02	125	100	122	97.6	122	97.6	119	95.2
A6C03	125	100	123	98.4	123	98.4	117	93.6
A6C04	125	100	123	98.4	121	96.8	118	94.4
A6C05	125	100	123	98.4	125	100	124	99.2
A6C07	125	100	125	100	124	99.2	125	100
A6C08	125	100	125	100	124	99.2	125	100
A6C09	125	100	124	99.2	125	100	123	98.4
A6C10	125	100	122	97.6	122	97.6	121	96.8
A6C11	125	100	125	100	125	100	125	100
A6C12	125	100	125	100	125	100	125	100

Note: conditions when one class model did not converge 100% in bold type.