STRUCTURAL EQUATION MODELING
2026, VOL. 33, NO. 1, 64–80
ISSN: 1070-5511 | E-ISSN: 1532-8007

OPEN ACCESS

Class Selection in Growth Mixture Models: Comparing Information Criteria to Nonparametric and Parametric Bayesian Approaches

Sarah Depaoli, Meng Qiu, Haiyan Liu, and Madelin Jauregui

University of California

ABSTRACT
Selecting the number of latent classes is a critical yet challenging aspect of latent growth mixture modeling (LGMM), with implications for model validity and substantive interpretation. Researchers commonly rely on information criteria to compare models with different numbers of classes, but these methods can be inconsistent, especially when class separation is poor or class sizes are unequal. This study evaluates two alternative Bayesian approaches: (1) the Dirichlet process mixture (DPM) model, a nonparametric method, and (2) the mixture of finite mixtures (MFM) model, a parametric method. Both impose a prior on the number of classes and estimate that number from the data. While the DPM model is theoretically appealing, previous research has found it tends to over-extract small classes. The MFM model, in contrast, offers a more reliable alternative by explicitly modeling the number of classes as a finite random variable. We compare these techniques to traditional information criteria (AIC, BIC, AICc, and aBIC) across varying conditions of sample size, class structure, separation, and indicator reliability. Simulation results highlight key performance differences, and we provide practical guidance for researchers selecting among class number determination methods. Illustrative R code is provided as online supplemental material.

KEYWORDS
Class enumeration; growth mixture modeling; information criteria; model fit; nonparameteric Bayesian

Determining the number of classes remains one of the most challenging and consequential decisions in latent class modeling. In latent growth mixture models (LGMMs), which allow for the modeling of unobserved heterogeneity in developmental trajectories, the selection of the number of latent classes directly affects the substantive conclusions drawn about heterogeneity in change over time.[1] Researchers typically rely on model comparison indices—such as the Akaike Information Criterion (AIC; Akaike, 1974), the Bayesian Information Criterion (BIC; Schwarz, 1978), and their sample-size-adjusted variants—to compare competing models that contain a different latent class structure (i.e., a different number or composition of latent classes). While widely used, these criteria depend on a discrete model comparison framework, and their performance can be sensitive to factors such as sample size, class separation, and within-class variability (Nylund et al., 2007).

Despite their prevalence, the use of conventional information criteria in class enumeration has notable limitations. Model comparison indices do not always converge on the true number of classes, especially in realistic conditions where classes are not well separated or where data include overlapping trajectories (e.g., so-called *fuzzy classification*; Tein et al., 2013; Tofighi & Enders, 2008). Moreover, the requirement to specify and compare a series of models with increasing class counts introduces a stepwise model-building process that may be inefficient and prone to overfitting or under-fitting the true latent structure (Bauer & Curran, 2003; Celeux et al., 2006; Nylund et al., 2007; Tofighi & Enders, 2008). These limitations motivate the need for alternative estimation strategies that allow greater flexibility and robustness in identifying the number of classes.

In this paper, we explore Bayesian methods for estimating the number of latent classes within LGMM, focusing specifically on the Dirichlet process mixture (DPM) model (Ferguson, 1973) and the mixture of finite mixtures (MFM) model (Miller & Harrison, 2018). These approaches offer conceptually distinct alternatives to the traditional model comparison paradigm by treating the number of classes as a random variable rather than a fixed model parameter. The DPM model, in particular, has garnered significant interest in the statistics and machine learning literature due to its ability to model data with an unknown and potentially infinite number of mixture classes (Ferguson, 1973; Rasmussen, 1999). However, previous work has shown that the DPM model can lead to inconsistent estimates of the number of latent classes under certain conditions, often producing small extraneous classes (Miller & Harrison, 2014).

© 2025 The Author(s).
This is an Open Access article distributed under the terms of the Creative Commons Attribution-NonCommercial-NoDerivatives License (http://creativecommons.org/licenses/by-nc-nd/4.0/), which permits non-commercial re-use, distribution, and reproduction in any medium, provided the original work is properly cited, and is not altered, transformed, or built upon in any way. The terms on which this article has been published allow the posting of the Accepted Manuscript in a repository by the author(s) or with their consent.

The MFM model was developed in part to address these limitations. Rather than assuming an infinite number of classes *a priori*, the MFM treats the number of mixture classes as an unknown but finite parameter drawn from a prior distribution. This formulation yields a posterior distribution over both the number of classes and the class-specific parameters, offering a more coherent and consistent framework for estimating the class structure. Although increasingly used in applied Bayesian statistics, the MFM has yet to be introduced to the LGMM literature.

Our goal in this study is twofold: (1) to introduce the DPM and MFM models as viable and theoretically grounded alternatives for model selection in LGMM, and (2) to systematically compare their performance to that of conventional information criteria across a range of simulation conditions relevant to the empirical application of LGMM. The simulation study includes a variety of conditions that examine the influence that class separation, sample size, class proportions, and indicator reliability have on the ability to properly detect the final latent class structure.

We aim to expand the methodological toolkit available to researchers using LGMM and to provide practical guidance on when (non)parametric Bayesian approaches may yield advantages over traditional model selection criteria. To our knowledge, this is the first study to examine the performance of the MFM in the context of LGMM, and one of the few to directly compare it with the DPM and information criteria-based approaches.

1. Organization of the Current Investigation

This paper is organized as follows. The next section details different approaches for the class enumeration issues within LGMM. We specifically highlight the use of iterative model comparison approaches (e.g., through the conventional use of information criteria) to determine the final class structure, as well as two (non)parametric approaches to directly estimate the number of classes—namely, the DPM and MFM models. The specification of the DPM and MFM models is presented next. We then present the specification of the LGMM model, which was used in the current investigation. Details of the simulation design and results follow, and the paper concludes with a discussion of findings and recommendations of use in applied research settings.

1.1. DPM and MFM as Alternatives for LGMM Class Enumeration

1.1.1. Information Criteria

In finite mixture modeling (including LGMM), the number of latent classes is typically unknown and must be specified before model estimation. Traditional model selection involves comparing models with varying numbers of classes using information criteria that balance model fit and model complexity (e.g., Burnham & Anderson, 2002; Jedidi et al., 1997). In the frequentist framework, widely used indices include the AIC (Akaike, 1974), the corrected Akaike Information Criterion (AICc; Hurvich & Tsai, 1989), the BIC (Schwarz, 1978), and the sample-size adjusted BIC (aBIC; Sclove, 1987).

The AIC, BIC, and aBIC are all computed as the sum of $-2\log$ likelihood and a penalty term that accounts for the model complexity. The penalty for the AIC is $2k$, where k is the number of estimated parameters. For the BIC, the penalty is $k\log(n)$, and for aBIC, it is $k\log\left(\frac{n+2}{24}\right)$, where n denotes the sample size.

Although the AIC is an asymptotically unbiased estimator of the expected out-of-sample deviance, it tends to be biased in small samples, particularly when the ratio n/k (sample size to number of parameters) is low (Burnham & Anderson, 2002; Hurvich & Tsai, 1989). To address this limitation, AICc includes an adjustment term, $\frac{2k(k+1)}{n-k-1}$, which increases the penalty for model complexity when the sample size is small.

The use of indices to determine the optimal number of classes typically involves fitting multiple finite mixture models with varying numbers of classes. This practice poses a potential concern, as it relies on the same dataset to extract the underlying structure and to evaluate the model, potentially leading to overfitting and optimistic model selection.

1.1.2. Alternative Approaches

As alternatives to traditional approaches, (non)parametric Bayesian methods, particularly the DPM and MFM models, offer more flexible frameworks to identify latent structures without requiring the number of classes to be specified in advance (Antoniak, 1974; Ferguson, 1973; Green & Richardson, 2001; Miller & Harrison, 2018).

The DPM model assumes a mixture model with a potentially unbounded number of classes. In the context of growth mixture modeling, individual growth parameters (such as intercepts and slopes) are assumed to follow a distribution drawn from a Dirichlet process (DP). In practice, the DP is typically implemented via the stick-breaking construction, which places probability mass on a countably infinite set of classes, or the Chinese restaurant process, which assigns probabilities to cluster assignments based on a sequential generative process. A detailed explanation of these DP constructions is beyond the scope of this paper; for further information, see Teh (2017); Qiu et al. (2025); Li et al. (2019). Although the DPM allows for theoretically infinite classes, only a finite number of classes are realized in a given dataset, and the posterior distribution adaptively infers the number of classes. When applied to longitudinal data, each mixture class naturally corresponds to a distinct growth trajectory.

In contrast, the MFM approach (Miller & Harrison, 2018) places a prior not only on the parameters of each class, but also explicitly on the number of classes (e.g., using a Poisson or uniform prior). Conditional on the number of classes, the model reduces to a standard finite mixture model with a fixed number of classes. A key advantage of the MFM is that it yields a posterior distribution over the number of classes, allowing researchers to quantify uncertainty in class enumeration directly, rather than fixed.

1.2. Specification of DPM

Let $\mathbf{y}_i$ denote the observed data for the individual $i = 1, \ldots, n$. The DPM model assumes that each observation is associated with a latent parameter θ_i, drawn from a random distribution G, which is itself drawn from a DP prior. The model is specified hierarchically as follows:

$$\begin{aligned} \mathbf{y}_i|\theta_i &\sim p\left(\mathbf{y}_i|\theta_i\right), \\ \theta_i|G &\sim G, \\ G &\sim \mathrm{DP}(\alpha, G_0), \end{aligned} \tag{1}$$

where $p(\mathbf{y}_i|\theta_i)$ is the likelihood function, θ_i is the latent parameter for observation i, G is a random probability measure, α is the concentration parameter of the Dirichlet Process, G_0 is the base distribution (prior over θ). The DP can be represented using the stick-breaking construction (Construction 1):

$$\begin{aligned} G &= \sum_{c=1}^{\infty} \pi_c \delta_{\theta_c^*}, \quad \theta_c^* \sim G_0, \\ \pi_c &= V_c \prod_{l=1}^{c-1} (1 - V_l), \quad V_c \sim \mathrm{Beta}(1, \alpha), \end{aligned} \tag{2}$$

where $\delta_{\theta_c^*}$ denotes a degenerate distribution with all its mass concentrated at θ_c^*, the atoms $(\theta_c^*)_{c=1}^{\infty}$ are generated independently from the base distribution G_0, $(V_c)_{c=1}^{\infty}$ are independent random variables drawn from a Beta distribution, π_c represents the probability mass at atom θ_c^*, and these probability masses are generated from a stick-breaking process that ensures that the class proportions add to 1.

The stick-breaking construction can be understood using a metaphor of breaking a stick of unit length into successive fractions π_c. Consider a stick of length one, we break a random piece of length V_1 where the length is sampled from $\mathrm{Beta}(1, \alpha)$. The remaining $(1 - V_1)$ of the stick is then recursively broken to obtain class proportions, $\pi_2 = (1 - V_1)V_2$, $\pi_3 = (1 - V_1)(1 - V_2)V_3$, and so on, with Vs from $\mathrm{Beta}(1, \alpha)$. By generating an infinite number of class proportions, this construction yields an infinite mixture model as follows:

$$\mathbf{y}_i \sim \sum_{c=1}^{\infty} \pi_c p(\mathbf{y}_i|\theta_c^*). \tag{3}$$

In practice, however, it is impossible to store a vector of infinite values on a computer. Therefore, Ishwaran and Zarepour (2000) proposed truncating this construction to a sufficiently large number H (e.g., $H = 10$), which represents an upper bound on the number of classes:

$$\pi_c = V_c \prod_{l=1}^{c-1} (1 - V_l), \quad c = 1, \ldots, H. \tag{4}$$

The DPM approach places a prior on an infinite number of classes, with a geometric-like decay in class probabilities. As a result, in finite samples, the posterior distribution often favors many small redundant classes—some of which may capture random noise rather than meaningful heterogeneity (Lu et al., 2018; Miller & Harrison, 2014). This tendency can hinder interpretability in applied settings, where clearly separated and substantively meaningful latent classes are typically desired.[2] To overcome the over-extraction of small classes, Miller and Harrison (2018) introduced the mixture of finite mixtures model.

1.3. Specification of MFM

Unlike the DPM model, where the number of classes can, in theory, be infinite–often resulting in an over-estimation of the number of classes, the MFM model treats the number of classes C as a random but finite quantity. A prior distribution can be specified for C, allowing the number of classes to be inferred directly from the data through its posterior distribution. The MFM model is specified as follows:

$$\begin{aligned} C &\sim p_C \quad \text{(prior on the number of classes)}, \\ \boldsymbol{\pi}|C &\sim \mathrm{Dirichlet}_C(\gamma, \ldots, \gamma), \\ z_i|\boldsymbol{\pi}, C &\sim \mathrm{Multinomial}(\pi_1, \ldots, \pi_C), \\ \boldsymbol{\theta}_c &\sim G_0, \quad c = 1, \ldots, C, \\ \mathbf{y}_i|z_i, \theta_1, \ldots, \theta_C &\sim p(\mathbf{y}_i|\boldsymbol{\theta}_{z_i}), \end{aligned} \tag{5}$$

where C denotes the (finite but unknown) number of classes, p_C is the prior distribution over the number of classes (e.g., Poisson or Uniform), $\boldsymbol{\pi}$ is the vector of class proportions, γ is the concentration parameter of the Dirichlet distribution, z_i is the latent class indicator for observation i, $\boldsymbol{\theta}_c$ is the vector of parameters associated with class c, G_0 is the base distribution for $\boldsymbol{\theta}_c$, and $p(\mathbf{y}_i|\boldsymbol{\theta}_{z_i})$ denotes the likelihood function.

The MFM can also be written using the stick-breaking construction (see Section 4.3 of Miller and Harrison (2018)). If we choose $(C - 1) \sim \mathrm{Poisson}(\lambda)$ and $\gamma = 1$, the class proportions $\pi_1, \ldots, \pi_C$ can be obtained through following steps (Construction 2)

1. Generate $\eta_1, \eta_2, \ldots \overset{\text{iid}}{\sim} \mathrm{Exponential}(\lambda)$,
2. $C = \min\{j : \sum_{c=1}^{j} \eta_c \geq 1\}$,
3. $\pi_c = \eta_c$, for $c = 1, \ldots, C - 1$,
4. $\pi_C = 1 - \sum_{c}^{C-1} \pi_c$.

Note that, in contrast to the infinite stick-breaking construction of the DP, the MFM's stick-breaking construction naturally truncates at Step 2, pruning small extraneous class proportions. This formulation allows the model to flexibly infer a finite but unknown number of latent classes, each characterized by its own parameters.

Despite its theoretical advantages, the performance of the MFM has not been systematically evaluated in the context of LGMM. To address this gap, we implement LGMM

[2] These methods are particularly relevant in applied settings where researchers seek to identify unobserved heterogeneity in developmental processes. For example, in psychology, LGMM is often used to distinguish subgroups of individuals with different trajectories of depressive symptoms over time. In education, researchers may apply mixture models to uncover distinct patterns of academic growth, such as students who improve steadily versus those who plateau. In fields such as marketing, similar models are used to classify consumers into groups with different purchasing or engagement trajectories.

under the MFM framework and compare its performance with the DPM approach and traditional information-criteria-based model selection methods. The following section outlines the technical specifications of the proposed models.

1.4. The DPM- and MFM-LGMM Models

The LGMM is a powerful tool for modeling trajectories of change over time, extending the simpler latent growth curve model by incorporating latent classes. This approach is based on the notation introduced in Depaoli (2021). In the LGMM framework, it is assumed that the observed data originate from a mixture distribution consisting of C latent classes indexed by $c = 1, 2, ..., C$, each with its own set of parameters. The class proportions are denoted by π_c. The model can be broken down into two primary classes: the measurement model and the structural model. The measurement model can be expressed as:

$$\mathbf{y}_{ic} = \Lambda \boldsymbol{\eta}_{ic} + \boldsymbol{\epsilon}_{ic}. \tag{6}$$

Here, $\mathbf{y}_{ic}$ represents the vector of observed repeated measurements for the individual i in class c, while Λ is a $T \times m$ matrix of factor loadings, where T is the number of time points and m is the number of latent factors. The first column of Λ is fixed at 1, and the remaining $m - 1$ columns contain information about the time structure and slope shape (e.g., 0, 1, 2, 3 for equally spaced time points with a linear slope). The vector $\boldsymbol{\eta}_{ic}$ holds the m latent growth parameters (such as the intercept and slope), and $\boldsymbol{\epsilon}_{ic}$ represents the vector of measurement errors, which are assumed to be normally distributed with a mean of zero.

The structural part of the model is defined as follows:

$$\boldsymbol{\eta}_{ic} = \boldsymbol{\beta}_c + \zeta_{ic}. \tag{7}$$

In this equation, $\boldsymbol{\eta}_{ic}$ represents the growth parameters, $\boldsymbol{\beta}_c$ is the vector of factor means for latent class c, and $\boldsymbol{\zeta}_{ic}$ captures the deviations of the growth parameters from the factor means, assumed to be normally distributed with a mean of zero. The simplified form of the model is then:

$$\mathbf{y}_{ic} = \Lambda(\boldsymbol{\beta}_c + \boldsymbol{\zeta}_{ic}) + \boldsymbol{\epsilon}_{ic}. \tag{8}$$

Since the expectation of $\boldsymbol{\eta}_c$ equals $\boldsymbol{\beta}_c$, the deviation term ζ_{ic} can be omitted. The mean and covariance of the model for class c can be expressed as follows:

$$\boldsymbol{\mu}_c = \Lambda \boldsymbol{\beta}_c, \tag{9}$$

$$\Sigma_c = \Lambda \Psi \Lambda' + \Phi_c. \tag{10}$$

In these equations, $\boldsymbol{\mu}_c$ represents the mean vector of the observed variables, while Σ_c denotes their covariance matrix. Additionally, Ψ signifies the covariance matrix of the latent factors, while Φc indicates the covariance matrix of the measurement errors. Note that Ψ does not include a subscript for c, implying homogeneity across latent classes, although this assumption can be relaxed by adding the c subscript. A diagram illustrating the basic structure of LGMM is shown in Figure 1. Therefore, the density function of $\mathbf{y}_i$ for an LGMM model can be expressed as:

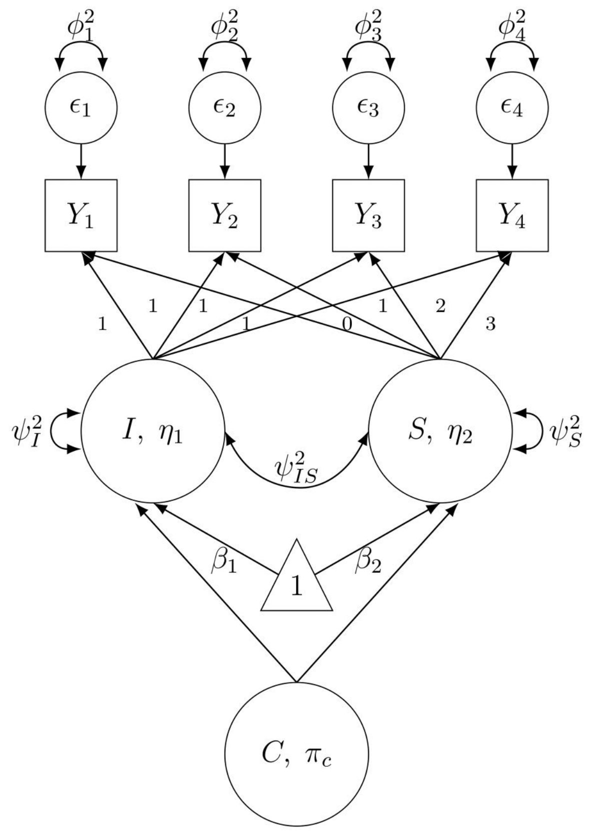

Figure 1. Latent growth mixture model.
Note. Path model of an LGMM with four measurements taken at equally-spaced time intervals. The intercept latent factor has loadings fixed to 1, whereas the slope factor has linearly increasing loadings (0, 1, 2, 3) to denote equally-spaced time intervals. ○; = Latent factor; □; = Observed variable; △; = Constant input (intercept); ϕ_t^2 = Measurement error variance; ϵ_t = Measurement error; Y_t = Measurements; η_1 = Intercept; η_2 = Slope; β_1 = Intercept mean; ψ_I^2 = Intercept variance; β_2 = Slope mean; ψ_S^2 = Slope variance; ψ_{IS}^2 = Growth factor covariance; C = Number of latent classes and π_c = Latent class proportions

$$p(\mathbf{y}_i) = \sum_{c=1}^{C} \pi_c \ \text{MultiNormal}\big(\mathbf{y}_i; \Lambda \boldsymbol{\beta}_c, \Lambda \Psi \Lambda' + \Phi_c\big). \tag{11}$$

To convert this conventional LGMM to its DPM and MFM alternatives, one needs to generate class proportions π_c using the corresponding stick-breaking construction specified in Construction 1 and Construction 2, respectively.

$$\begin{array}{llll} \boldsymbol{\pi} & \sim \text{Dirichlet}[d_1, ..., d_C], & \beta_{mc} & \sim \text{Normal}\big[\mu_{mc}, \sigma_{mc}^2\big], \\ \phi_{tc}^2 & \sim \text{InvGamma}[a_{tc}, b_{tc}], & \boldsymbol{\Psi} & \sim \text{InvWishart}[\boldsymbol{W}_0, \nu_0]. \end{array} \tag{12}$$

1.5. Determining the Number of Classes

1.5.1. Conventional LGMM

In the conventional LGMM, determining the optimal number of classes using an information criterion involves fitting a series of LGMM models with an increasing number of classes, starting

Table 1. Summary of simulation population parameters.

Simulation Factor	Levels
Sample Size	$n = 200, 400, 1000$
Class Proportion	Equal (34:33:33), Unequal (70:20:10)
Class Separation	MD $=$ 1 (small), 2 (medium), 3 (large), 5 (extra large)*
Average Growth Curve Reliability (AGCR)	**AGCR = 0.5** $\phi_1^2 = 1.5$, $\phi_2^2 = 2.2$, $\phi_3^2 = 3.0$, $\phi_4^2 = 3.8$ **AGCR = 0.9** $\phi_1^2 = 0.1$, $\phi_2^2 = 0.2$, $\phi_3^2 = 0.4$, $\phi_4^2 = 0.5$
Growth Factor Variance and Covariance	$\psi_I^2 = 1.0$, $\psi_S^2 = 0.5$, $\psi_{IS} = 0.1$

Note. Figure 1 illustrates parameters ϕ^2 and ψ^2 within the full LGMM. *See Table 2 for the population values associated to each degree of separation. MD = Mahalanobis distance.

from a single-class model up to a predefined maximum number of classes. For each model, the information criterion is calculated, and the model with the lowest value is selected as the optimal model, as it indicates the best trade-off between goodness-of-fit and parsimony. In practice, however, relying on a single information criterion may be insufficient; instead, employing a combination of criteria is recommended to enhance model selection robustness.

1.5.2. DPM- and MFM-LGMM

We follow the post-processing procedure utilized in Qiu et al. (2025), in which the authors employed a loss function called variation of information (VI) to identify the number of classes C. Grounded in information theory, the VI provides a point estimate of clustering that minimizes the posterior expected loss (Meilă, 2007; Wade & Ghahramani, 2018). It identifies a representative clustering based on all the posterior clusterings, and then utilizes the number of unique labels, denoted as $\hat{C}$, present in the representative clustering as the estimated value of C. It has been shown that VI is robust to the misspecification of α and can consistently recover the number of classes (Wade & Ghahramani, 2018). The VI has been implemented in the R package `mcclust.ext` (Wade & Wade, 2015). We demonstrate the use of the VI in the R code provided in the supplemental material.[3]

2. Simulation Design

This study sought to explore the performance of two Bayesian approaches, namely the DPM and MFM, to estimate the number of mixture classes in LGMM. We also aimed to compare the performance of the DPM and MFM with traditional approaches to model selection (e.g., AIC and BIC). Our design satisfied both goals by measuring the performance of the DPM, MFM, and the information criteria in estimating/selecting the correct number of classes in a variety of scenarios, which varied five design factors: number of classes (1 or 3), sample size (200, 400, 1000), class separation as defined by the Mahalanobis distance (e.g., 1, 2, 3, 5), latent class proportions (equal and unequal), and model reliability as measured by the average growth curve reliability (AGCR; Shryane, 2021). Figure 1 depicts the structure of LGMM; Table 1 includes a summary of the simulation factors and their corresponding levels; and Table 2 details the growth factor means for each number of classes and degree of class separation.

Table 2. Population intercept and slope parameters by class separation.

Class Enumeration		Population Intercept β_1 and Slope β_2			
1-class			(5.0, 0.8)		
3-class	*Class*	*MD = 1*	*MD = 2*	*MD = 3*	*MD = 5*
	$c = 1$	(6.0, 1.0)	(6.0,0.7)	(6.0,0.2)	(9.0,1.5)
	$c = 2$	(5.4,1.5)	(5.0,1.8)	(5.2,2.1)	(5.0,3.5)
	$c = 3$	(4.8,0.8)	(4.0,0.4)	(3.0,0.5)	(4.0,0.3) .

Note. Parameters are reported as (β_1, β_2). MD = Mahalanobis distance.

2.1. Population Model

At the population level, we generated data from an LGMM model with four time points and two growth factors (intercept and slope; Figure 1). Tables 1 and 2 summarize the population model's parameter values. The variance of the intercept and slope factors was fixed at 1.0 and 0.5, respectively, whereas the variance of error terms changed depending on the reliability (i.e., AGCR; Table 1). In conditions with more than one latent class, the class-specific growth factor means depended on the intended degree of class separation, as defined by the Mahalanobis distance and explained in the following section (see Table 2).

2.2. Design Factors

2.2.1. Number of Classes

A fundamental goal of this study was to measure the performance of alternative approaches in identifying the correct number of classes; thus, we varied the number of classes to include 1-class and 3-class conditions. By creating datasets with single and multiple latent classes, we evaluated the performance of all approaches in detecting mixture classes in multi-class scenarios (3-class) and accurately identifying the absence of mixtures in single-class scenarios (1-class).

Since 1-class conditions cannot vary class separation and class proportions, there were 3 (sample sizes) × 2 (reliability) = 6 conditions with a single latent class. In 3-class conditions, we manipulated all the factors, yielding 3 (sample size) × 4 (class separation) × 2 (class proportions) × 2 (reliability) = 48 conditions. In total, 6 (1-class) + 48 (3-class) = 54 unique simulation conditions, each with 200 replications, yielded a total of 10,800 simulated datasets.

2.2.2. Sample Size

Sample size is a factor known to impact the performance of model estimation in general and in LGMM specifically

[3] Supplemental material can be found at the following link: https://tinyurl.com/DPM-MFM.

(Enders & Tofighi, 2008; Lubke & Muthén, 2007; Nylund et al., 2007; Tein et al., 2013). Based on previous research, we included a small (200), medium (400), and large (1000) sample size (Diallo et al., 2017; Kim et al., 2022).

2.2.3. Class Separation

In 3-class conditions, the mean of the intercept and slope factors were determined based on the desired degree of class separation. Following prior research with LGMM, we used the Mahalanobis distance (MD) to measure class separation. Mathematically, MD for two different classes can be computed as:

$$MD = \sqrt{(\boldsymbol{\beta}_c - \boldsymbol{\beta}_{c'})^\top \boldsymbol{\Psi}^{-1} (\boldsymbol{\beta}_c - \boldsymbol{\beta}_{c'})} \tag{13}$$

where $\boldsymbol{\beta}_c$ and $\boldsymbol{\beta}_{c'}$ refer to the latent class parameter vector (i.e., intercept and slope) of two different classes, and $\boldsymbol{\Psi}$ the growth factor covariance matrix. We considered four levels of class separation: small (MD = 1), medium (MD = 2), large (MD = 3), and extra large (MD = 5). The work of Kim et al. (2022, 2021); Tong et al. (2022) informed the MD values for medium and large separation levels, and we added MD = 1 and 5 to include a clearly small and extra large degree of separation between classes. Noticeably, the degree of class separation was consistent across latent classes. For example, in a three-class model, the separation between classes 1 and 2, 2 and 3, and 1 and 3 was the same. Table 2 details the mean of growth factors under each degree of class separation.

2.2.4. Class Proportions

Prior research on mixture modeling shows that class proportions can impact the performance of traditional approaches to class enumeration (Depaoli, 2013; Nylund et al., 2007; Tein et al., 2013; Tueller & Lubke, 2010). Thus, we manipulated the 3-class conditions to reflect equal and unequal class proportions. Equal class proportions reflected a 34:33:33 ratio, whereas unequal conditions included a clear majority and a minority class in a 70:20:10 ratio.

2.2.5. Growth Curve Reliability

Reliability is defined as the ratio of true-score variance over the total variance of a measurement (Lord & Novick, 1968). For latent growth curve modeling, McArdle and Epstein (1987) proposed the *growth curve reliability* (GCR) at time point t as the ratio of the latent curve variance (true variance) over the total variance (the sum of true variance and error variance), representing the amount of variability explained by the latent growth factors. Given a linear growth curve model, the GCR at time point t can be computed as (see also McArdle & Epstein, 1987, Table 2B):

$$\begin{aligned} GCR_t &= Var(True)_t / \left[Var(True)_t + Var(Error)_t\right] \\ &= (\psi_I^2 + \lambda_t^2 \psi_S^2 + 2\lambda_t \psi_{IS}) / (\psi_I^2 + \lambda_t^2 \psi_S^2 + 2\lambda_t \psi_{IS} + \phi_t^2), \end{aligned} \tag{14}$$

where ψ_I^2 represents the intercept variance, ψ_S^2 the slope variance, ψ_{IS}^2 the covariance between the intercept and slope, and ϕ_t^2 the measurement error variance at measurement occasion t, λ_t the factor loading at measurement occasion t.

In the simulation study, we allowed the GCR to increase over time, which is a reasonable assumption in developmental research. To define a single metric, we averaged the GCR across all measurement occasions. Furthermore, we set this average GCR (AGCR) to 0.5 for low reliability and 0.9 for high reliability:

$$AGCR = \sum_{t=1}^{T} GCR_t / T. \tag{15}$$

2.3. Outcomes of Interest

As a measure of performance, we opted to use the proportion of replications that identified a model with C classes. In 1-class conditions, we considered the selection rate of a correctly specified 1-class solution and two over-specified solutions (2- and 3-class); and in 3-class conditions, we considered an under-specified 2-class solution, a correctly specified 3-class solution, and an over-specified 4-class solution. For each solution, we computed the selection rate as:

$$\widehat{pr}(\hat{C} = c) = \frac{1}{n_{reps}} \sum_{s=1}^{n_{reps}} 1(\widehat{C}_s = c), \tag{16}$$

where n_{reps} is the total number of replications (200 in this study), $s = 1, \ldots, n_{reps}$, $c = 1, 2, 3$ in 1-class conditions and $c = 2, 3, 4$ in 3-class conditions. This approach enabled us to not only observe the proportion of replications returning the correct solutions but also detect wether alternative (under- and over-specified) solutions were preferred when a small percentage of replications yielded the correct mixture model structure.

Because the proposed models estimate the number of classes, they do not require the specification of neighboring models. To facilitate comparison between the two frameworks, we made the following adjustments for the proposed LGMM models: 1) when the true C was 1, we merged the proportions for $C > 3$ into $C = 3$; and 2) when the true C was 3, we merged the proportions for $C \leq 2$ into $C = 2$ and those for $C \geq 4$ into $C = 4$.

2.4. Prior Distributions

In the current study, a noninformation normal prior $\text{Normal}(0, 10^3)$ was specified for the means of growth intercept and slope. An inverse gamma prior $\text{InvGamma}(1, 1)$ was selected for the variances of the measurement errors ϕ_t^2, which is considered relatively informative and has been recommended by Gelman (2006) to avoid improper posteriors. The covariance matrix of latent growth factors $\boldsymbol{\Psi}$ had a non-informative inverse-Wishart prior $\text{InvWishart}(\boldsymbol{W}_0, \nu_0)$ with an identify scale matrix $\boldsymbol{W}_0 = \mathbf{I}$ and degrees of freedom being $\nu_0 = 2$ (Zhang et al., 2007). For the concentration parameter of the DP, α, we set a weakly informative Gamma prior $\text{Gamma}(2, 2)$ to cover both small and large

Table 3. Prior distributions specified for model parameters.

Parameter	Meaning of Parameter	Prior Distribution
β_1	mean of intercept factor	$\mathcal{N}(0, 10^3)$
β_2	mean of linear slope factor	$\mathcal{N}(0, 10^3)$
Ψ	covariance of latent factors	$\mathcal{IW}(\boldsymbol{W}_0, 2)$
ϕ_t^2	variance of measurement error	$\mathcal{IG}(1, 1)$
α	concentration parameter of DP	$\Gamma(2, 2)$

Note. $\boldsymbol{W}_0$ is a 2×2 identity scale matrix.

values, as suggested by Ishwaran (2000). The specification of the priors are summarized in Table 3.

2.5. Software

In this study, posterior sampling of the model parameters was facilitated by NIMBLE (de Valpine et al., 2017), a flexible R-framework for hierarchical models and statistical programming. The `nimble` (version 0.9.1; de Valpine et al. (2020)) package in R (R Core Team et al., 2013) provides functionalities for fitting hierarchical models that involve DP priors either through a Chinese restaurant process or a truncated stick-breaking construction, which are currently not available in other commonly used statistical software, such as Stan and MATLAB. For example, NIMBLE provides ready-to-use functions `dCRP` and `stick_breaking` for easily constructing the Chinese restaurant process and stick-breaking process, respectively. Furthermore, NIMBLE offers a variety of MCMC samplers, such as the Gibbs sampler (Geman & Geman, 1984), the Metropolis-Hastings adaptive random-walk sampler (Roberts & Sahu, 1997), and the Hamiltonian Monte Carlo sampler (Duane et al., 1987). Another advantage of NIMBLE is that it allows users to write self-defined functions, similar to defining R functions (see https://r-nimble.org/manual/cha-lightning-intro.html#sec: creating-your-own for more information). Therefore, NIMBLE can handle a much broader class of models than those implemented in standard software packages.

3. Simulation Results

3.1. Convergence Diagnostics and Number of Iterations in a Chain

In a Bayesian analysis, it is essential to perform a convergence diagnosis to confirm that the chain has reached convergence—the retained samples are representative of the target distribution. Widely-used diagnostic tools include visual inspection of a trace plot and various diagnostic statistics, such as the Geweke statistic (Geweke, 1992), the Heidelberger-Welch test of stationarity (Heidelberger & Welch, 1983), and the Gelman-Rubin statistic (Gelman & Rubin, 1992). These diagnostic tools, however, were not easily implemented in this study for several reasons. First, the mixture model contained many parameters. Second, the diagnosis of a specific within-class parameter is subject to label switching, a unique issue in mixture modeling that can cause the position of a within-class parameter to swap across classes (see Jasra et al. (2005) for a review and solutions to label switching). Finally, the number of classes varied across the MCMC iterations. Therefore, in this case, a quick-and-dirty global diagnostic statistic is desirable. In the current study, we employed the normalized mutual information (NMI; McDaid et al. (2011); Vinh et al. (2009)), specifically the joint version, as a global diagnostic statistic. NMI measures the similarity between two clusterings (collections of individuals' class labels) and it ranges from 0 to 1, with 1 indicating perfect agreement between the two clusterings and 0 indicating no agreement beyond chance. In the analysis of a simulated dataset, one could compute NMI between the simulated and estimated clusterings for each iteration and then inspect the trace of NMI. An illustrative trace plot of NMI values obtained from a simulated example is presented in Figure 2, where the chain converged rapidly after 500 iterations. Another promising global diagnostic that has been employed in Bayesian nonparametric mixture modeling is the examination of marginal probabilities for all variables (Si & Reiter, 2013).

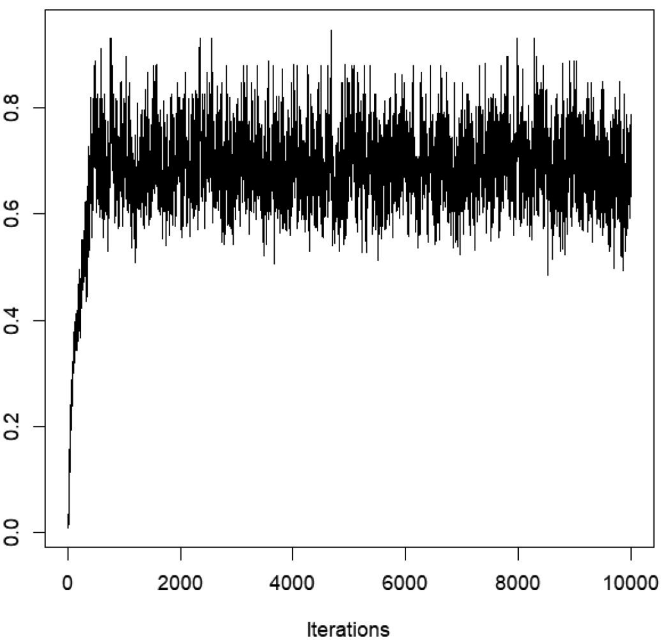

Figure 2. Trace plot of NMIs for a simulated example.

Through all simulation conditions, we observed that the Markov chain of NMI stabilized after 5,000. Given the complexity of mixture modeling, a chain length of 20,000 iterations with a burn-in of 10,000 was adopted for the simulation study.

3.2. Presentation of Results

The results in this section are partitioned based on the number of latent classes in the population model, and the tables contain a similar structure. The outcome listed in the table is the selection rate for each class solution. Table 4 presents the results of 1-class conditions, whereas Table 5 reports the results of 3-class conditions with low to moderate measurement reliability (AGCR = 0.5) and Table 6 the results from 3-class conditions with high reliability (AGCR = 0.9). The tables were structured to contain information on sample sizes (rows) and selection techniques (columns).

Table 4. $C = 1$: The case of a single class.

	AIC			AICc			BIC			aBIC			DPM			MFM		
n	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
AGCR = 0.5																		
200	**0.11**	0.31	0.59	**0.19**	0.37	0.45	**0.93**	0.07	0.00	**0.14**	0.33	0.54	**0.99**	0.01	0.00	**1.00**	0.00	0.00
400	**0.08**	0.29	0.64	**0.11**	0.32	0.58	**0.98**	0.03	0.00	**0.36**	0.32	0.33	**1.00**	0.00	0.00	**1.00**	0.01	0.00
1000	**0.11**	0.31	0.58	**0.12**	0.32	0.57	**1.00**	0.00	0.00	**0.77**	0.17	0.07	**1.00**	0.00	0.00	**1.00**	0.00	0.00
AGCR = 0.9																		
200	**0.11**	0.21	0.69	**0.21**	0.27	0.52	**0.98**	0.02	0.00	**0.14**	0.21	0.65	**1.00**	0.01	0.00	**1.00**	0.00	0.00
400	**0.10**	0.27	0.63	**0.15**	0.34	0.51	**0.99**	0.01	0.00	**0.49**	0.31	0.21	**1.00**	0.00	0.00	**1.00**	0.01	0.00
1000	**0.10**	0.34	0.57	**0.12**	0.35	0.54	**1.00**	0.00	0.00	**0.83**	0.15	0.03	**1.00**	0.00	0.00	**1.00**	0.00	0.00

Note. Columns in boldface indicate the true number of classes in the population. AGCR = averaged growth curve reliability; AIC = Akaike information criterion; AICc = corrected Akaike information criterion; BIC = Bayesian information criterion; aBIC = sample-size-adjusted Bayesian information criterion; DPM = Dirichlet process mixture model; MFM = Mixture of finite mixtures model.

In the case of multiple latent classes in the population, information about class separation and class proportions was also provided (main blocks of rows). The correct class solution is presented in bold font for ease of interpretation.

To facilitate comparison across models and simulation conditions, posterior class counts greater than three were combined into a single category (e.g., “C = 4”) in summary tables. This approach was necessary because the number of classes identified often ranged from two to as many as six, with the upper bound varying across replications and conditions. Retaining all distinct values would have made tabular presentation overly complex. Importantly, class counts exceeding three occurred infrequently, minimizing the risk of obscuring meaningful model behavior.

3.3. *C=1*: *The Case of a Single Class*

Results for $C = 1$, where a single class was specified in the population structure, are presented in Table 4. The columns for $C = 1$ are in bold font to track the selection rates for the correct class solution. The results for $C = 2$ and 3 highlight the tendency to over-extract the number of classes. Reliability values of 0.50 and 0.90 are presented in the main row blocks, with minimal noticeable differences between these reliability conditions regarding selection rate patterns.

Selection results for the DPM and MFM (two right columns) indicated proper selection in all sample size and reliability conditions examined. The DPM under the lower reliability (0.50) and lowest sample size ($n = 200$) was the only cell showing less than perfect selection rate. Otherwise, the DPM and MFM techniques were able to perfectly identify the class structure under these conditions.

The results for the information criteria were mixed, with some indices correctly selecting the population model structure and others mis-specifying the structure by selecting too many classes. Overall, the BIC performed the best among the information criteria examined here. The selection rates were high ($\geq .93$) in all sample sizes and reliability conditions. There was a small tendency (.00 to .07) to over-extract the number of classes to $C = 2$ when using the BIC for model selection.

The AIC and AICc performed similarly to each other in that they tended to over-extract the class solution in the majority of replications. $C = 1$ was selected with a rate ranging from 0.08 to 0.21 in all sample sizes and reliability conditions. These two indices tended to select $C = 2$ (.21 to .37) and 3 (.45 to .69) at a relatively high rate, with an overall preference for the largest class solution.

To some extent, the aBIC showed a pattern similar to the AIC and AICc, but its selection rates improved as sample size increased. With the largest sample size ($n = 1000$), the rates were 0.77 (reliability of 0.5) and .83 (reliability of 0.9). However, the smaller sample sizes produced patterns in which selection favored the highest number of classes ($C = 3$), and the selection rates for the correct solution $C = 1$ were $< .15$. Overall, the aBIC exhibited a clear sample size effect, with correct class selection improving with larger sample sizes. As sample sizes decreased, the aBIC was much less stable.

3.4. *C=3*: *Three Latent Classes with Low Observed Variable Reliability (0.50)*

The results for the $C = 3$ population model are presented in Table 5. In these conditions, the average growth curve reliability (AGCR), as defined in Equation (15), was set to 0.50 (see “Average Growth Curve Reliability” in Table 1). This table added complexity to the presentation of results compared to the $C = 1$ table of results because class proportions and class separation were also specified as conditions in this part of the simulation. The main row breaks represented class proportions, with equal class proportions (34:33:33) in the top half of the table and unequal class proportions (70:20:10) in the bottom half. The equal and unequal class proportions results were then further defined through class separation, with MD settings ranging from 1 (poorest separation) to 5 (largest separation). Sample sizes were varied from $n = 200$ (smallest) to $n = 1000$ (largest). In this table, the selection rate results for $C = 3$ are marked in bold font to highlight the correct class solution. Results for under-extraction ($C = 2$) and over-extraction ($C = 4$) of latent classes are presented alongside the correct class solution.

3.4.1. *Equal Class Proportions (34:33:33)*

Focusing on the equal class proportions first (top half of the table), there was a clear class separation effect and a more subtle sample size effect in the selection rates within each technique. Additionally, the techniques produced different selection rate patterns compared to one another.

Table 5. $C = 3$: Three latent classes with low observed variable reliability (0.50).

		AIC			AICc			BIC			aBIC			DPM			MFM		
MD	*n*	2	3	4	2	3	4	2	3	4	2	3	4	2	3	4	2	3	4
Equal (34:33:33)																			
1	200	0.12	**0.38**	0.51	0.24	**0.43**	0.34	0.90	**0.10**	0.01	0.16	**0.38**	0.47	1.00	**0.00**	0.00	1.00	**0.00**	0.00
	400	0.13	**0.35**	0.52	0.20	**0.39**	0.42	0.98	**0.03**	0.00	0.40	**0.32**	0.29	1.00	**0.00**	0.00	1.00	**0.00**	0.00
	1000	0.14	**0.35**	0.52	0.15	**0.38**	0.48	0.99	**0.02**	0.00	0.72	**0.19**	0.10	1.00	**0.00**	0.00	1.00	**0.00**	0.00
2	200	0.09	**0.34**	0.58	0.17	**0.41**	0.42	0.92	**0.08**	0.01	0.11	**0.37**	0.53	1.00	**0.00**	0.00	1.00	**0.00**	0.00
	400	0.08	**0.28**	0.65	0.13	**0.31**	0.57	0.98	**0.03**	0.00	0.36	**0.34**	0.31	1.00	**0.00**	0.00	0.99	**0.01**	0.00
	1000	0.04	**0.32**	0.65	0.04	**0.35**	0.62	1.00	**0.01**	0.00	0.68	**0.26**	0.06	1.00	**0.00**	0.00	1.00	**0.00**	0.00
3	200	0.04	**0.25**	0.71	0.12	**0.37**	0.52	0.90	**0.10**	0.00	0.06	**0.28**	0.66	0.97	**0.04**	0.00	0.91	**0.09**	0.00
	400	0.01	**0.22**	0.78	0.03	**0.31**	0.66	0.98	**0.03**	0.00	0.25	**0.32**	0.44	0.86	**0.15**	0.00	0.85	**0.15**	0.00
	1000	0.01	**0.14**	0.86	0.01	**0.16**	0.84	1.00	**0.00**	0.00	0.63	**0.25**	0.13	0.75	**0.24**	0.01	0.64	**0.37**	0.00
5	200	0.00	**0.09**	0.92	0.02	**0.25**	0.74	0.82	**0.14**	0.05	0.00	**0.11**	0.90	0.27	**0.72**	0.01	0.00	**0.97**	0.03
	400	0.00	**0.06**	0.94	0.00	**0.10**	0.90	0.43	**0.56**	0.02	0.00	**0.25**	0.75	0.10	**0.91**	0.00	0.00	**0.99**	0.01
	1000	0.00	**0.07**	0.94	0.00	**0.09**	0.91	0.01	**1.00**	0.00	0.00	**0.58**	0.43	0.14	**0.86**	0.00	0.00	**1.00**	0.00
Unequal (70:20:10)																			
1	200	0.12	**0.28**	0.60	0.27	**0.39**	0.35	0.94	**0.06**	0.01	0.15	**0.32**	0.54	1.00	**0.00**	0.00	1.00	**0.00**	0.00
	400	0.16	**0.30**	0.55	0.22	**0.32**	0.47	0.97	**0.03**	0.00	0.44	**0.32**	0.25	1.00	**0.00**	0.00	1.00	**0.00**	0.00
	1000	0.15	**0.32**	0.53	0.18	**0.34**	0.49	0.99	**0.01**	0.00	0.79	**0.18**	0.04	1.00	**0.00**	0.00	1.00	**0.00**	0.00
2	200	0.10	**0.39**	0.52	0.21	**0.45**	0.35	0.93	**0.08**	0.00	0.13	**0.40**	0.48	1.00	**0.01**	0.00	1.00	**0.01**	0.00
	400	0.11	**0.28**	0.62	0.15	**0.33**	0.53	0.98	**0.02**	0.00	0.31	**0.36**	0.33	1.00	**0.00**	0.00	1.00	**0.00**	0.00
	1000	0.06	**0.33**	0.62	0.07	**0.34**	0.59	1.00	**0.01**	0.00	0.65	**0.28**	0.08	1.00	**0.00**	0.00	1.00	**0.00**	0.00
3	200	0.05	**0.29**	0.67	0.15	**0.39**	0.47	0.91	**0.10**	0.00	0.06	**0.32**	0.63	0.98	**0.02**	0.00	0.98	**0.02**	0.00
	400	0.03	**0.26**	0.72	0.05	**0.32**	0.64	0.98	**0.02**	0.00	0.23	**0.37**	0.41	0.95	**0.05**	0.00	0.94	**0.07**	0.00
	1000	0.02	**0.13**	0.86	0.03	**0.15**	0.83	1.00	**0.01**	0.00	0.72	**0.14**	0.15	0.98	**0.03**	0.00	0.94	**0.06**	0.00
5	200	0.01	**0.14**	0.86	0.03	**0.27**	0.71	0.84	**0.15**	0.02	0.01	**0.16**	0.84	0.19	**0.81**	0.00	0.02	**0.98**	0.01
	400	0.00	**0.05**	0.95	0.00	**0.11**	0.90	0.69	**0.29**	0.03	0.01	**0.28**	0.71	0.01	**0.99**	0.01	0.00	**1.00**	0.01
	1000	0.00	**0.07**	0.93	0.00	**0.09**	0.92	0.01	**0.99**	0.01	0.00	**0.62**	0.38	0.03	**0.97**	0.00	0.00	**1.00**	0.00

Note. Columns in boldface indicate the true number of classes in the population. MP = mixing proportions; MD = Mahalanobis distance; AIC = Akaike information criterion; AICc = corrected Akaike information criterion; BIC = Bayesian information criterion; aBIC = sample-size-adjusted Bayesian information criterion; DPM = Dirichlet process mixture model; MFM = Mixture of finite mixtures model.

The AIC and AICc performed similarly to each other, with selection rates showing a tendency to over-extract the number of latent classes in most conditions. There was no clear benefit to larger sample sizes, with more accurate selection results yielded by conditions with smaller sample sizes. The pattern was more noticeable in the AICc, for example, with a selection rate of 0.25 for $n = 200$, 0.10 for $n = 400$, and 0.09 for $n = 1000$ when MD = 5. Specifically, there was a steeper drop-off in selection as sample sizes increased for the AICc compared to the AIC. Regarding class separation, another interesting observation emerged from the AIC and AICc results. The poorer separation conditions tended to exhibit higher selection rates for the correct $C = 3$ solution compared to the greater separation conditions. In other words, as class separation increased, the ability of the AIC and AICc to accurately detect the correct class structure decreased. In the case of both indices, there was an increased tendency to over-extract toward the 4-class solution as separation increased. Overall, these indices tended to over-extract, but the severity increased with greater separation and larger sample sizes.

The BIC showed a clear difference in sample size and class separation. Specifically, the ability for the BIC to properly detect the 3-class model improved drastically as separation increased to MD = 5, and the best detection was present under the largest sample size. Even with MD = 5, smaller sample sizes showed low-to-moderate selection rate results. Outside of conditions with larger sample sizes and greater class separation, the BIC displayed a strong tendency to under-extract and select the simpler model with $C = 2$. The aBIC showed a pattern similar to the BIC, with the best results under the largest separation and sample size condition. However, the remaining results for the aBIC were scattered, showing a mixed pattern of under- and over-extraction of the number of latent classes. Overall, the aBIC was highly variable in its selection pattern, appearing unstable under these conditions.

The DPM and MFM were unable to detect the correct class solution under the poorer separation conditions of MD = 1, 2, and (to a large extent) 3. Under MD = 3, a shift occurred in some replications showing correct selection, and that was present under the larger sample sizes. For instance, as the sample size increased from 200 to 1000, the selection rate for the correct class solution increased from 0.04 to 0.24 in the DPM and from 0.09 to 0.37 in the MFM. Albeit, MD = 3 still showed a large tendency of under-extraction of classes for these methods (.64 to .97). For MD = 5, the MFM outperformed the DPM across all sample sizes, and the MFM displayed perfect selection at $n = 1000$. Overall, the DPM appeared to be more sensitive to small sample sizes than the MFM.

3.4.2. Unequal Class Proportions (70:20:10)

A second condition of class proportions was examined for $C = 3$ with reliability set at 0.50. In previous research (Depaoli, 2013; Qiu et al., 2025; Tueller & Lubke, 2010) on mixture classes proportions, the relative size of the class was an important factor in properly determining the number of classes. Specifically, when class proportions were unequal, including a true minority and true majority class, estimation issues tended to arise. We aimed to explore this issue in terms of class enumeration to track the performance of the current techniques in situations with equal and unequal

Table 6. $C = 3$: Three latent classes with high observed variable reliability (0.90).

		AIC			AICc			BIC			aBIC			DPM			MFM		
MD	*n*	2	3	4	2	3	4	2	3	4	2	3	4	2	3	4	2	3	4
Equal (34:33:33)																			
1	200	0.13	**0.41**	0.47	0.28	**0.45**	0.28	0.97	**0.03**	0.00	0.17	**0.41**	0.42	1.00	**0.00**	0.00	1.00	**0.00**	0.00
	400	0.14	**0.30**	0.57	0.21	**0.34**	0.45	0.99	**0.02**	0.00	0.47	**0.31**	0.23	1.00	**0.00**	0.00	1.00	**0.00**	0.00
	1000	0.13	**0.38**	0.49	0.16	**0.41**	0.44	1.00	**0.01**	0.00	0.82	**0.16**	0.02	1.00	**0.00**	0.00	1.00	**0.00**	0.00
2	200	0.04	**0.32**	0.65	0.17	**0.42**	0.42	0.93	**0.08**	0.00	0.04	**0.37**	0.59	1.00	**0.00**	0.00	0.97	**0.04**	0.00
	400	0.01	**0.28**	0.71	0.04	**0.34**	0.62	1.00	**0.01**	0.00	0.30	**0.40**	0.31	0.98	**0.03**	0.00	0.94	**0.06**	0.00
	1000	0.04	**0.21**	0.76	0.05	**0.23**	0.73	1.00	**0.00**	0.00	0.73	**0.17**	0.10	0.98	**0.03**	0.00	0.93	**0.08**	0.00
3	200	0.01	**0.12**	0.88	0.05	**0.21**	0.74	0.93	**0.07**	0.01	0.02	**0.13**	0.85	0.56	**0.45**	0.00	0.20	**0.79**	0.02
	400	0.01	**0.07**	0.93	0.01	**0.14**	0.86	0.95	**0.06**	0.00	0.03	**0.35**	0.62	0.10	**0.90**	0.00	0.00	**0.99**	0.01
	1000	0.00	**0.09**	0.92	0.00	**0.11**	0.90	0.11	**0.89**	0.01	0.00	**0.76**	0.25	0.21	**0.79**	0.00	0.00	**1.00**	0.00
5	200	0.00	**0.06**	0.95	0.00	**0.23**	0.77	0.00	**0.98**	0.03	0.00	**0.08**	0.93	0.23	**0.77**	0.01	0.00	**0.99**	0.02
	400	0.00	**0.12**	0.89	0.00	**0.19**	0.82	0.00	**0.99**	0.02	0.00	**0.40**	0.60	0.07	**0.93**	0.00	0.00	**0.99**	0.01
	1000	0.00	**0.10**	0.91	0.00	**0.12**	0.89	0.00	**1.00**	0.00	0.00	**0.76**	0.24	0.16	**0.85**	0.00	0.00	**1.00**	0.00
Unequal (70:20:10)																			
1	200	0.17	**0.35**	0.48	0.35	**0.39**	0.27	0.96	**0.05**	0.00	0.22	**0.38**	0.41	1.00	**0.00**	0.00	1.00	**0.00**	0.00
	400	0.12	**0.32**	0.56	0.20	**0.36**	0.44	0.99	**0.01**	0.00	0.42	**0.37**	0.22	1.00	**0.00**	0.00	1.00	**0.00**	0.00
	1000	0.21	**0.34**	0.46	0.23	**0.34**	0.44	1.00	**0.00**	0.00	0.85	**0.13**	0.03	1.00	**0.00**	0.00	1.00	**0.01**	0.00
2	200	0.08	**0.30**	0.63	0.25	**0.37**	0.39	0.97	**0.04**	0.00	0.11	**0.31**	0.58	1.00	**0.00**	0.00	0.98	**0.02**	0.00
	400	0.09	**0.33**	0.59	0.14	**0.39**	0.48	1.00	**0.00**	0.00	0.37	**0.39**	0.25	1.00	**0.00**	0.00	0.99	**0.02**	0.00
	1000	0.06	**0.16**	0.78	0.07	**0.20**	0.74	1.00	**0.00**	0.00	0.77	**0.18**	0.06	1.00	**0.00**	0.00	1.00	**0.01**	0.00
3	200	0.03	**0.27**	0.71	0.10	**0.42**	0.48	0.97	**0.02**	0.01	0.05	**0.30**	0.66	0.63	**0.37**	0.00	0.42	**0.57**	0.02
	400	0.02	**0.13**	0.86	0.04	**0.17**	0.79	0.97	**0.03**	0.01	0.16	**0.29**	0.55	0.10	**0.90**	0.01	0.08	**0.92**	0.01
	1000	0.00	**0.07**	0.94	0.00	**0.09**	0.91	0.75	**0.26**	0.00	0.10	**0.62**	0.29	0.05	**0.95**	0.00	0.00	**1.00**	0.01
5	200	0.00	**0.07**	0.93	0.01	**0.28**	0.72	0.45	**0.52**	0.03	0.00	**0.11**	0.90	0.06	**0.93**	0.01	0.00	**0.99**	0.02
	400	0.00	**0.11**	0.89	0.00	**0.19**	0.81	0.04	**0.95**	0.02	0.00	**0.44**	0.57	0.01	**0.99**	0.01	0.00	**1.00**	0.00
	1000	0.00	**0.09**	0.91	0.00	**0.12**	0.88	0.00	**1.00**	0.01	0.00	**0.74**	0.26	0.03	**0.97**	0.00	0.00	**1.00**	0.00

Note. Columns in boldface indicate the true number of classes in the population. MP = mixing proportions; MD = Mahalanobis distance; AIC = Akaike information criterion; AICc = corrected Akaike information criterion; BIC = Bayesian information criterion; aBIC = sample-size-adjusted Bayesian information criterion; DPM = Dirichlet process mixture model; MFM = Mixture of finite mixtures model.

class size structures. The results of the current simulation did not pinpoint the class proportions as a notable factor that affected the performance in these indices or class selection techniques. The results for the 70:20:10 class proportions closely resembled those for the equal class proportions under the same reliability setting.

3.5. *C*=3: *Three Latent Classes with High Observed Variable Reliability (0.90)*

The results for the $C = 3$ population model under conditions of excellent reliability in the observed variables are presented in Table 6. Reliability for these observed variables was set to a high level of 0.90 to show the upper-bound performance of these indices in a set of conditions with high reliability. Results are presented in a similar manner within the table as above, with the top half displaying results for equal (34:33:33) class proportions and the bottom half showing results for unequal (70:20:10) class proportions.

3.5.1. *Equal Class Proportions (34:33:33)*

The AIC and AICc performed comparably to each other. They both tended to over-extract the number of latent classes, and this tendency increased as class separation increased. In other words, as classes became more distinct from one another, these indices tended to over-extract to a greater degree. There did not appear to be a noticeable *positive* sample size effect for these indices in that the pattern of over-extraction did not improve under larger sample sizes. In contrast, over-extraction became more common under large sample sizes and class separation values.

The BIC's results are quite clear: there is a strong tendency to under-extract and select $C = 2$ for all class separation conditions except for MD = 5; another exception was for $n = 1000$ under MD = 3, where the results produced a selection rate of 0.89. Otherwise, the BIC failed to pinpoint the correct class structure.

For poorer separation conditions (MD = 1 and 2), the aBIC showed a pattern of decreased selection rates for the correct model as sample sizes increased. Specifically, the largest sample size yielded the poorest selection rate in these conditions, as the aBIC tended to under-extract the number of classes. As separation increased, the pattern changed such that larger sample sizes and larger separation produced better selection rates for the correct $C = 3$ model, but there was still a tendency to over-extract under the larger separation conditions (MD = 3 and 5). For the largest separation conditions, this index performed just behind the BIC when $n = 1000$ but much worse than the BIC when sample size was small or medium. Overall, the BIC and aBIC performed much better than the AIC and AICc. However, the aBIC performed worse than the BIC when MD = 5, especially with a small sample size; the aBIC appeared more sensitive to sample size.

The results from the DPM and MFM demonstrated a clear sample size effect under larger class separation conditions. Specifically, at MD = 3 and 5, selection rates became more accurate as sample sizes increased. MD = 5 yielded the most accurate results for these techniques, with the MFM performing nearly perfectly across all sample sizes and the DPM showing improved results with larger sample sizes. Both techniques exhibited a strong tendency to under-extract (selecting $C = 2$) when class separation was relatively poor (MD = 1 and 2).

3.5.2. Unequal Class Proportions (70:20:10)

The AIC and AICc performed similarly for unequal class proportions, as observed for equal class proportions. Specifically, both tended to over-extract the number of latent classes, with this trend worsening for larger sample sizes and greater degrees of class separation. Overall, these indices did not accurately identify the number of latent classes and consistently leaned toward over-extraction.

Similar to the performance under equal class conditions, the BIC tended to under-extract in poorer class separation conditions. Selection rates did not exceed chance for the BIC until MD = 5, where rates were lower than those in the equal class size conditions reported above, especially when $n = 200$. These results suggest that the BIC was adversely impacted by the presence of the minority latent class due to unequal class sizes, and the under-extraction persisted under poorer separation settings.

The trend observed for the aBIC under unequal class proportions was consistent with the findings when class sizes were equal. Specifically, the aBIC tended to over-extract under the poorest separation and then shifted to under-extraction as separation increased.

The DPM and MFM showed a difference within MD = 3 conditions. With the presence of a true minority class, selection rates were lower for the correct class solution under this separation condition. Results for MD = 1 and MD = 2 were comparable to the above, where the techniques under-extracted in nearly all replications. For MD = 5, selection rates were high, indicating that the largest class separation condition could compensate for the presence of a minority class and correctly select $C = 3$ as the final model. As with equal-sized classes, the DPM slightly underperformed the MFM for the lowest sample size in this MD = 5 setting.

3.6. Computational Burden

All the models were estimated using a Linux cluster with two nodes having 12 cores and 128 GB RAM per node [Intel(R) Xeon(R) CPU E5-2680 v3 at 2.50 GHz]. Across simulation conditions, the estimation time ranged between 5 and 38 minutes for the two Bayesian models. It should be noted that the estimation time taken by the two Bayesian models is primarily influenced by sample size and the pre-specified upper limit of classes (set to $H = 10$ in the current simulation study). Larger sample sizes and higher values of H will lead to increased estimation time. For the DPM model, the choice of DP construction also affects the estimation time. Although not adopted in this study, according to Qiu et al. (2025), the Chinese restaurant process resulted in a shorter running time compared to the stick-breaking process.

4. Real Example: Analysis of ECLS-K Data

To demonstrate the application of MFM-GMM using NIMBLE, we analyzed a random subsample of mathematics achievement scores from the Early Childhood Longitudinal Study–Kindergarten Class of 1998–99 (ECLS-K). The ECLS-K is a longitudinal cohort study tracking child development and educational experiences, collecting multi-method, multi-source data from a nationally representative sample of approximately 22,000 students who entered kindergarten in 1998. Data were gathered in the fall and spring of kindergarten and first grade, and in the spring of third, fifth, and eighth grades, using direct assessments of student performance, parent interviews, teacher questionnaires, and administrator surveys. The original dataset comprises 51% male and 49% female students, with a racial distribution of approximately 57% White, 13% Black, 18% Hispanic, 7% Asian, and 5% other races. The subsample consists of complete data (no missing values) for the first four measurements ($T = 4$) from 400 students ($n = 400$), a sample size selected because it reflects a typical medium size in growth mixture modeling literature (e.g., Kohli et al. (2015)). The full sample public-use data can be found at http://nces.ed.gov/pubsearch/pubsinfo.asp?pubid=2009005.

One of the intriguing research questions in mathematics skill development is whether students form distinct subpopulations and whether these subpopulations exhibit differences in mathematics achievement trajectories over time. To answer this question, we fitted an MFM-GMM model to the subsample. The prior distributions used in the analysis were identical to those employed in the simulation study. The Gibbs sampler was run for 10,000 iterations with a burn-in period of 1,000 iterations. The subsample and complete R code for fitting both the DPM-GMM and MFM-GMM models are available in the online supplemental material.

The MFM-GMM model identified three classes. To visualize the quality of classification (the degree of class separation), a heatmap of the posterior similarity matrix, with data points reordered by hierarchical clustering, is presented in Figure 3. In this heatmap, darker shades of red indicate a higher probability of being in the same class, while darker shades of white indicate a lower probability of being in the same class. The heatmap reveals a clear configuration of three distinct red blocks, suggesting satisfactory class separation. The results for the three-class solution are presented in Figure 4 and Table 8. Figure 4 shows the estimated trajectories, while Table 8 shows the parameter estimates for the three classes. Class 1 (25% of the subsample), represented by the red trajectory, exhibited the highest initial score (around 25) followed by the steepest positive slope, culminating in a substantial increase to around 70 by the endpoint. Therefore, Class 1 can be labeled as "Rapid Accelerators." In contrast, Class 2 (26% of the subsample), depicted in green, commenced with the lowest initial score (around 15) and demonstrated minimal growth, rising modestly to about 32 by the endpoint, suggesting a stable low-performing subgroup. Thus, Class 2 can be labeled as "Gradual Stabilizers." Finally, Class 3 (49% of the subsample), shown in blue and comprising the largest

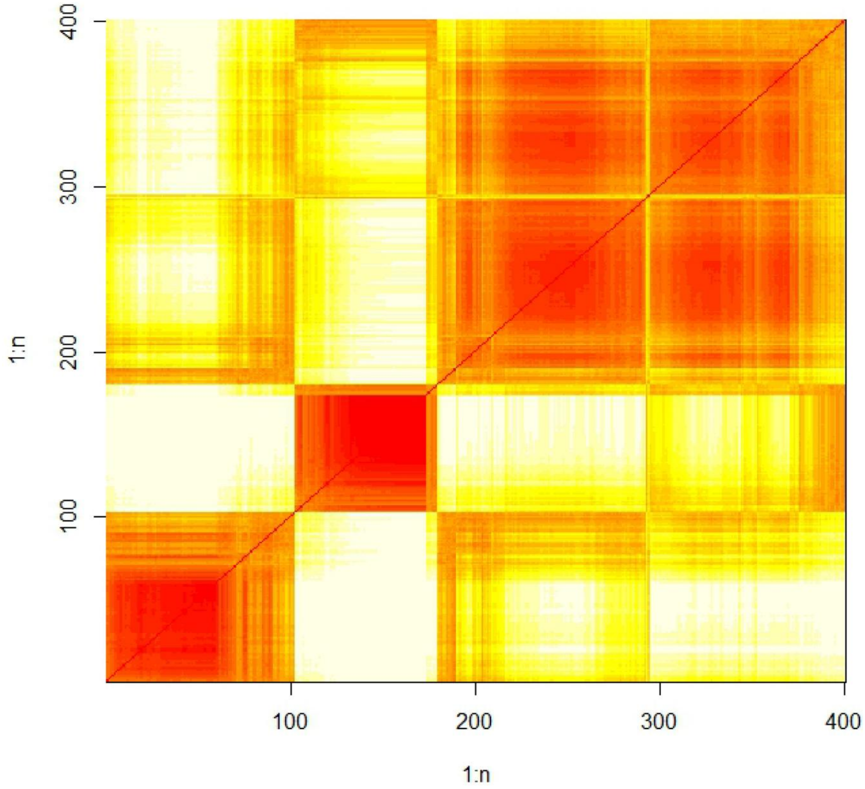

Figure 3. Heatmap for real example.

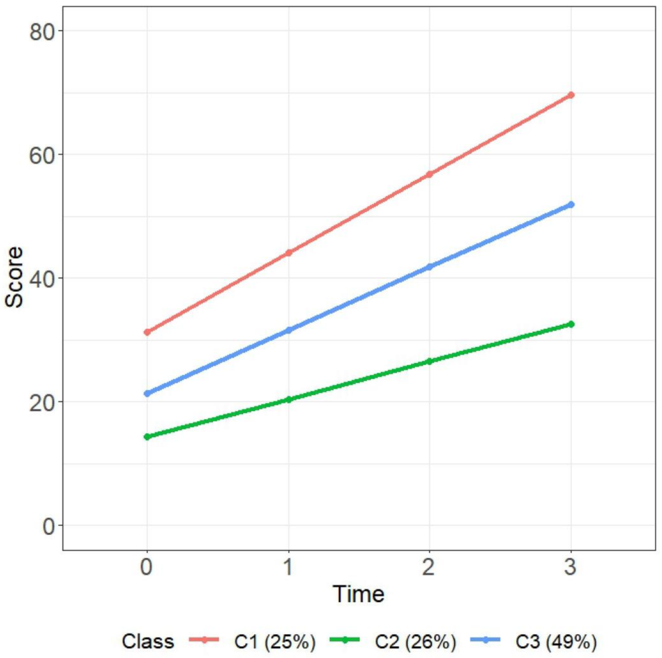

Figure 4. Estimated trajectories for the real example.

proportion, displayed a moderate initial score (around 20) with an intermediate upward trend, reaching approximately 50 by the endpoint, reflecting a normative pattern of gradual progress. Hence, Class 3 can be labeled as "Steady Progressors." The supplemental material includes fully executable R code for this example, accompanied by a detailed explanation in the appendix.

5. Discussion

When the true model did not contain latent classes (i.e., $C = 1$), the DPM, MFM, BIC, and, to a lesser extent, aBIC were all stable in correctly identifying the single-class structure. The AIC and AICc tended to over-extract latent classes. Notably, there was no significant effect of item reliability on detecting the correct latent class structure in this case.

As the latent class structure became more complex, especially when $C = 3$ (for both equal and unequal class proportions), poorer separation conditions led to variability in class selection. Under these conditions, the AIC, AICc, and aBIC were *relatively better* indices, although results were still highly variable, with common under- and over-extraction. In contrast, when class separation was large, the DPM and MFM were the most reliable methods, with the MFM demonstrating the most stable and accurate selection patterns across different conditions. These patterns held for both equal and unequal class proportions.

Overall, we believe the BIC's tendency to favor fewer classes (under-fitting), such as two instead of the true three, arises from its relatively strong penalty for model complexity. As the number of latent classes increases, the penalty term grows rapidly, often causing the BIC to under-fit by selecting a more parsimonious solution that fails to capture the true heterogeneity in the data.

The impact of item reliability on simulation conditions was evident, as higher reliability led to more accurate class selection. Specifically, when reliability was high, methods such as MFM and aBIC provided more stable results, ensuring that class extraction closely matched the true number of

Table 7. Summary Table for technique selection.

Condition	Best Method(s)	Notes
$C = 1$	DPM, MFM, BIC	DPM/MFM nearly perfect; BIC slightly over-extracts
$C = 3$, Low Rel., Low Sep.	None ideal	BIC/aBIC under-extract; DPM/MFM under-extract
$C = 3$, Low Rel., High Sep., Large N	MFM	Perfect selection possible
$C = 3$, High Rel., High Sep., Large N	MFM, aBIC	aBIC may still vary; MFM most stable
$C = 3$, High Rel., Low Sep.	None ideal	Over- or under-extraction common
Any, Tolerate Over-fitting	AIC, AICc	Prefer larger class solutions consistently

Table 8. Parameter estimates of MFM-GMM of random subsample from ECLS-K data.

	Estimate		
Parameter	Class 1	Class 2	Class 3
π	0.25	0.26	0.48
β_0	30.91	14.42	21.38
β_1	12.84	6.11	10.24
ψ_I^2	56.40	8.22	11.14
ψ_S^2	2.69	2.72	1.45
ψ_{IS}	7.95	4.05	2.86
ϕ_1^2	32.09	4.61	7.83
ϕ_2^2	31.92	3.85	23.74
ϕ_3^2	78.75	9.45	37.50
ϕ_4^2	157.24	75.53	71.94

classes. Conversely, low reliability often resulted in under- or over-extraction, with methods like the BIC struggling to accurately identify the optimal number of classes, particularly in the presence of lower separation among classes.

Although unequal class proportions did not substantially reduce performance across most indices, some noteworthy differences were observed. BIC tended to under-estimate the number of classes when groups were highly unbalanced, likely because the presence of smaller classes was more heavily penalized, particularly under conditions of lower observed variable reliability (Table 5). By contrast, the MFM approach demonstrated greater robustness overall but still exhibited variability when class sizes were highly unequal and class separation was limited (Tables 5 and 6). These findings suggest that the influence of unequal class proportions is method-dependent rather than uniform across criteria. This has implications for applied research in which small but substantively important classes may be present, and highlights the need for further investigation into how class imbalance interacts with different enumeration strategies.

5.1. Recommendations for Use in Applied Research

The results of our simulation study reveal that the performance of class enumeration techniques varies considerably depending on the number of latent classes in the population, the reliability of the observed indicators, class separation, class proportions, and sample size. To assist applied researchers in selecting an appropriate method for determining the number of latent classes, we present the following guidance in three formats: a written summary, a decision tree (Figure 5), and a summary table (Table 7).

5.2. Written Guidance for Applied Researchers

When deciding how many latent classes to extract from a dataset, there are several different aspects that must be considered by applied researchers. We will describe these considerations by putting them into context of our simulation design.

The first issue is about the suspected number of latent classes at the population level. If the researcher suspects (e.g., through expert evaluation or previous research) that there is only one class present in the population (homogeneity is present), then DPM and MFM are highly reliably across all conditions. The BIC is also recommended, particularly in larger samples, but we advise caution surrounding the issue of minor over-extraction that was present with the BIC.

The second aspect to consider is when there are multiple latent classes present ($C = 3$ in the current investigation). When reliability was low (as with the 0.50 condition in the simulation), certain patterns emerged worth noting. Specifically, when class separation was low to moderate ($MD = 1$ to 3), the DPM and MFM often under-extracted, especially at smaller sample sizes. The BIC and aBIC tended to under-extract, especially at low separation. When class separation was high ($MD = 5$) and sample size was large, MFM performed best. The BIC and aBIC also improved, though we note that results pointed toward the BIC being prone to under-extract at smaller samples. The AIC and AICc consistently over-extracted, especially as separation increases. As a result, they are not recommended unless over-fitting is preferred for substantive reasons. When reliability increased to 0.90, the patterns were largely the same but more extreme. Over-extraction by the AIC/AICc and under-extraction by the BIC/aBIC were more pronounced.

The third aspect to consider is class proportion. The effect of equal versus unequal class proportions (e.g., 70:20:10) was minimal for the techniques explored here. Most techniques showed similar performance regardless of differences in proportion. This stability is encouraging, as it indicates that when classes are present, the techniques will perform consistently regardless of the class proportions.

Finally, sample size issues are important to consider when selecting a technique for class enumeration purposes. Larger sample sizes generally improved performance across all methods, particularly for the BIC and aBIC. The DPM and MFM benefited most from high separation and large sample sizes.

5.2.1. General Recommendations

Some general recommendations are as follows: First, use the DPM or MFM when theoretical justification suggests a

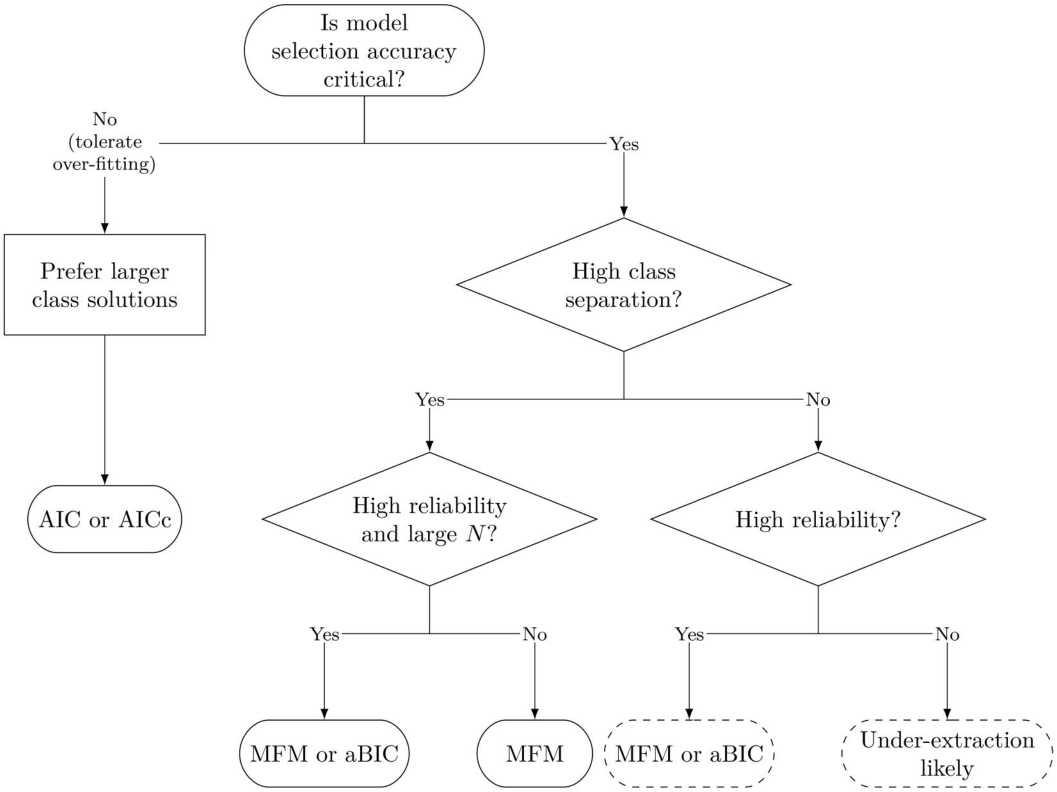
Is model selection accuracy critical?
No (tolerate over-fitting)
Yes
Prefer larger class solutions
AIC or AICc
High class separation?
Yes
No
High reliability and large N?
High reliability?
Yes
No
Yes
No
MFM or aBIC
MFM
MFM or aBIC
Under-extraction likely

Figure 5. Decision tree for selecting a latent class detection method based on study conditions.
Note. End nodes capture the best available latent class decision method based on the importance of model selection accuracy and population-level conditions (i.e., class separation and reliability). Dashed shapes denote non-ideal conditions under which any method should be implemented with caution. ◇ = Decision; ⬭ = Action/Recommendation; □; = Context; ⬭ = Cautionary recommendation

homogeneous population. Second, use the MFM in settings with high class separation and larger samples. Third, use the BIC if parsimony is important and you are comfortable with a slight risk of under-extraction. Finally, avoid using the AIC or AICc to determine the number of classes unless your goal is to maximize sensitivity at the risk of over-fitting.

We emphasize that the choice between traditional fit indices (e.g., BIC), DPM, or MFM should be viewed as relative rather than absolute. Importantly, there is no universal benchmark for what constitutes a "meaningful" difference in correct selection rates. For example, whether a difference between 0.85 and 0.90 is substantively important depends on the study context and research objectives. In some applied settings, even small improvements in selection accuracy may be highly valuable when the cost of misclassification is large, whereas in other contexts such differences may be negligible. Similar to the interpretation of statistical power, the evaluation of selection rates must consider factors such as sample size, class proportions, class separation, and estimation method. Our simulations therefore aim to provide a comprehensive perspective on how these factors shape selection rates, enabling researchers to judge their practical significance in relation to their own substantive goals.

5.3. Future Research Directions to Consider

In the current study, we used diffuse or weakly informative priors (e.g., $\mathrm{InvGamma}(1,1)$, $\mathrm{Normal}(0,10^3)$, $\mathrm{Gamma}(2,2)$) without conducting a formal sensitivity analysis. Prior specification can substantially influence Bayesian mixture models, particularly the Dirichlet process concentration parameter α, which governs the expected number of latent classes. Although our goal was to assess performance under commonly used priors, future research would benefit from systematic sensitivity analyses that evaluate how alternative prior choices affect both class enumeration and parameter recovery. This would strengthen conclusions about robustness and improve interpretability for applied researchers.

Although this study focused on LGMMs with linear trajectories and three latent classes, future work should investigate the performance of the DPM and MFM models in other modeling contexts. Extensions to factor mixture models, non-linear growth models (e.g., quadratic or piecewise), and settings with two or more than three population-level classes would offer insight into generalizability. Additionally, the current design included only four time points, which aligns with previous simulation studies but limits applicability to more complex longitudinal structures. Future studies should consider longer trajectories to examine how these methods scale in the presence of increased temporal complexity.

Another important consideration for future work is the handling of missing data. In longitudinal designs, missingness is common and can reduce class separation and effective sample size. While prior research has examined the performance of traditional information criteria under missing data conditions (Heo et al., 2024), there is a lack of corresponding investigations for the DPM and MFM models. Incorporating missing data mechanisms into simulation studies could help clarify how sensitive these models are to data loss and whether alternative estimation strategies are needed.

This study focused on evaluating class enumeration accuracy, but future work should also examine classification quality. Measures such as entropy, posterior classification probabilities, and external agreement indices (e.g., Adjusted Rand Index) can provide a more complete picture of model performance. Understanding the relationship between enumeration and classification accuracy may lead to more effective guidance for selecting and interpreting latent class models in applied settings.

A further area for development involves the refinement of information criteria tailored to latent growth mixture models. Current indices are not always well-suited to identifying under- or over-fitting of class structure, and improved criteria could provide more practical guidance for applied researchers. Such work would complement the advances in Bayesian (non)parametric modeling by strengthening the conventional toolkit available for class enumeration.

Finally, new algorithmic and machine learning-based methods for class enumeration represent a promising direction for future growth modeling research. Techniques such as decision tree-guided mixture modeling (Brandmaier et al., 2013), nonparametric Bayesian clustering (Bhattacharya & Dunson, 2011), and model-based recursive partitioning (Zeileis et al., 2008) offer alternatives to traditional fit indices by leveraging automated pattern detection. Future research should examine how these methods perform relative to conventional approaches, especially when applied to complex longitudinal data.

Taken together, these areas of future research may help refine best practices for class enumeration and improve the practical utility of both conventional and Bayesian (non)-parametric mixture models.

Disclosure statement

No potential conflict of interest was reported by the authors.

References

Akaike, H. (1974).). A new look at the statistical model identification [Conference Name: IEEE Transactions on Automatic Control]. *IEEE Transactions on Automatic Control*, *19*, 716–723. https://doi.org/10.1109/TAC.1974.1100705

Antoniak, C. E. (1974). Mixtures of Dirichlet Processes with applications to Bayesian nonparametric problems. *The Annals of Statistics*, *2*, 1152–1174. https://doi.org/10.1214/aos/1176342871

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

Bhattacharya, A., & Dunson, D. B. (2011). Sparse Bayesian infinite factor models. *Biometrika*, *98*, 291–306. https://doi.org/10.1093/biomet/asr013

Brandmaier, A. M., von Oertzen, T., McArdle, J. J., & Lindenberger, U. (2013). Structural equation model trees. *Psychological Methods*, *18*, 71–86. https://doi.org/10.1037/a0030001

Burnham, K. P., & Anderson, D. R. (2002). Advanced issues and deeper insights. In K. P. Burnham & D. R. Anderson (Eds.), *Model selection and multimodel inference: A practical information-theoretic approach.* (pp. 267–351). Springer. https://doi.org/10.1007/978-0-387-22456-5_6

Celeux, G., Forbes, F., Robert, C. P., & Titterington, D. M. (2006). Deviance information criteria for missing data models. *Bayesian Analysis*, *1*, 651–673. https://doi.org/10.1214/06-BA122

de Valpine, P., Paciorek, C., Turek, D., Michaud, N., Anderson-Bergman, C., Obermeyer, F., Cortes, C. W., Rodrıguez, A., Lang, D. T., Paganin, S., et al. (2020). NIMBLE: MCMC, particle filtering, and programmable hierarchical modeling (R package version 0.9.1).

de Valpine, P., Turek, D., Paciorek, C. J., Anderson-Bergman, C., Lang, D. T., & Bodik, R. (2017). Programming with models: Writing statistical algorithms for general model structures with NIMBLE. *Journal of Computational and Graphical Statistics*, *26*, 403–413. https://doi.org/10.1080/10618600.2016.1172487

Depaoli, S. (2021). *Bayesian structural equation modeling*. Guilford Press.

Depaoli, S. (2013). Mixture class recovery in GMM under varying degrees of class separation: Frequentist versus Bayesian estimation. *Psychological Methods*, *18*, 186–219. https://doi.org/10.1037/a0031609

Diallo, T. M. O., Morin, A. J. S., & Lu, H. (2017). Performance of growth mixture models in the presence of time-varying covariates. *Behavior Research Methods*, *49*, 1951–1965. https://doi.org/10.3758/s13428-016-0823-0

Duane, S., Kennedy, A. D., Pendleton, B. J., & Roweth, D. (1987). Hybrid Monte Carlo. *Physics Letters B*, *195*, 216–222. https://doi.org/10.1016/0370-2693(87)91197-X

Enders, C. K., & Tofighi, D. (2008). The impact of misspecifying class-specific residual variances in growth mixture models. *Structural Equation Modeling: A Multidisciplinary Journal*, *15*, 75–95. https://doi.org/10.1080/10705510701758281

Ferguson, T. S. (1973). A Bayesian analysis of some nonparametric problems. *The Annals of Statistics*, *1*, 209–230. Retrieved May 24, 2025, from https://www.jstor.org/stable/2958008 https://doi.org/10.1214/aos/1176342360

Gelman, A. (2006). Prior distributions for variance parameters in hierarchical models (comment on article by Browne and Draper). https://doi.org/10.1214/06-BA117A

Gelman, A., & Rubin, D. B. (1992). Inference from iterative simulation using multiple sequences. *Statistical Science*, *7*, 457–472. https://doi.org/10.1214/ss/1177011136

Geman, S., & Geman, D. (1984). Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *6*, 721–741. https://doi.org/10.1109/TPAMI.1984.4767596

Geweke, J. (1992). Evaluating the Accuracy of Sampling-Based Approaches to the Calculation of Posterior Moments. In *Bayesian Statistics 4: Proceedings of the Fourth Valencia International Meeting, Dedicated to the Memory of Morris H. DeGroot*, 1931–1989. Oxford University Press. https://doi.org/10.1093/oso/9780198522669.003.0010

Green, P. J., & Richardson, S. (2001). Modelling heterogeneity with and without the dirichlet process. *Scandinavian Journal of Statistics*, *28*, 355–375. https://doi.org/10.1111/1467-9469.00242

Heidelberger, P., & Welch, P. D. (1983). Simulation run length control in the presence of an initial transient. *Operations Research*, *31*, 1109–1144. https://doi.org/10.1287/opre.31.6.1109

Heo, I., Jia, F., & Depaoli, S. (2024). Performance of model fit and selection indices for Bayesian piecewise growth modeling with missing data. *Structural Equation Modeling: A Multidisciplinary Journal*, *31*, 455–476. https://doi.org/10.1080/10705511.2023.2264514

Hu, G., Yang, H.-C., Xue, Y., & Dey, D. K. (2023). Zero-inflated Poisson model with clustered regression coefficients: Application to heterogeneity learning of field goal attempts of professional basketball players. *Canadian Journal of Statistics*, *51*, 157–172. https://doi.org/10.1002/cjs.11684

Hurvich, C. M., & Tsai, C.-L. (1989). Regression and time series model selection in small samples. *Biometrika*, *76*, 297–307. https://doi.org/10.2307/2336663

Ishwaran, H. (2000). Inference for the random effects in Bayesian generalized linear mixed models. *ASA Proceedings of the Bayesian Statistical Science Section*, 1–10.

Ishwaran, H., & Zarepour, M. (2000). Markov chain Monte Carlo in approximate Dirichlet and beta two-parameter process hierarchical models. *Biometrika*, *87*, 371–390. https://doi.org/10.1093/biomet/87.2.371

Jasra, A., Holmes, C. C., & Stephens, D. A. (2005). Markov chain Monte Carlo methods and the label switching problem in Bayesian mixture modeling. *Statistical Science*, *20*, 50–67. https://doi.org/10.1214/088342305000000016

Jedidi, K., Jagpal, H. S., & DeSarbo, W. S. (1997). January 1). STEMM: A general finite mixture structural equation model. Retrieved May 24, 2025, from https://papers.ssrn.com/abstract=2791766

Kim, S., Tong, X., & Ke, Z. (2021). Exploring class enumeration in Bayesian Growth Mixture Modeling based on conditional medians. *Frontiers in Education*, *6*. https://doi.org/10.3389/feduc.2021.624149

Kim, S., Tong, X., Zhou, J., & Boichuk, J. P. (2022). Conditional median-based Bayesian Growth Mixture Modeling for nonnormal data. *Behavior Research Methods*, *54*, 1291–1305. https://doi.org/10.3758/s13428-021-01655-w

Kohli, N., Hughes, J., Wang, C., Zopluoglu, C., & Davison, M. L. (2015). Fitting a linear–linear piecewise growth mixture model with unknown knots: A comparison of two common approaches to inference. *Psychological Methods*, *20*, 259–275. https://doi.org/10.1037/met0000034

Li, Y., Schofield, E., & Gönen, M. (2019). A tutorial on dirichlet process mixture modeling. *Journal of Mathematical Psychology*, *91*, 128–144. https://doi.org/10.1016/j.jmp.2019.04.004

Lord, F. M., & Novick, M. R. (1968). *Statistical theories of mental test scores*. Addison-Wesley.

Lu, J., Li, M., & Dunson, D. (2018). Reducing over-clustering via the powered Chinese restaurant process. https://doi.org/10.48550/arXiv.1802.05392

Lubke, G., & Muthén, B. O. (2007). Performance of factor mixture models as a function of model size, covariate effects, and class-specific parameters. *Structural Equation Modeling: A Multidisciplinary Journal*, *14*, 26–47. https://doi.org/10.1080/10705510709336735

McArdle, J. J., & Epstein, D. (1987). Latent growth curves within developmental structural equation models. *Child Development*, *58*, 110–133. https://doi.org/10.2307/1130295

McDaid, A. F., Greene, D., & Hurley, N. (2011). Normalized mutual information to evaluate overlapping community finding algorithms. *arXiv preprint arXiv:*1110.2515. https://doi.org/10.48550/arXiv.1110.2515

Meilă, M. (2007). Comparing clusterings—an information based distance. *Journal of Multivariate Analysis*, *98*, 873–895. https://doi.org/10.1016/j.jmva.2006.11.013

Miller, J. W., & Harrison, M. T. (2014). Inconsistency of Pitman-Yor process mixtures for the number of components. *Journal of Machine Learning Research*, *15*, 3333–3370. http://jmlr.org/papers/v15/miller14a.html

Miller, J. W., & Harrison, M. T. (2018). Mixture models with a prior on the number of components. *Journal of the American Statistical Association*, *113*, 340–356. https://doi.org/10.1080/01621459.2016.1255636

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: A Multidisciplinary Journal*, *14*, 535–569. https://doi.org/10.1080/10705510701575396

Papastamoulis, P. (2016). Label.switching: An R package for dealing with the label switching problem in MCMC outputs. *Journal of Statistical Software*, *69*, 1–24. https://doi.org/10.18637/jss.v069.c01

Qiu, M., Paganin, S., Ohn, I., & Lin, L. (2025). Bayesian nonparametric latent class analysis with different item types. *Psychological Methods*. Advanced online publication, https://doi.org/10.1037/met0000728

R Core Team, R., et al. (2013). R: A language and environment for statistical computing.

Rasmussen, C. (1999). The infinite gaussian mixture model. *Advances in Neural Information Processing Systems, 12*. Retrieved May 24, 2025, from https://proceedings.neurips.cc/paper/1999/hash/97d98119037c5b8a9663cb21fb8ebf47-Abstract.html

Roberts, G. O., & Sahu, S. K. (1997). Updating schemes, correlation structure, blocking and parameterization for the gibbs sampler. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, *59*, 291–317. https://doi.org/10.1111/1467-9868.00070

Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics*, *6*, 461–464. https://doi.org/10.1214/aos/1176344136

Sclove, S. L. (1987). Application of model-selection criteria to some problems in multivariate analysis. *Psychometrika*, *52*, 333–343. https://doi.org/10.1007/BF02294360

Shryane, N. (2021). March 18). Reliability in latent growth curve models. In A. Cernat & J. W. Sakshaug (Eds.), *Measurement error in longitudinal data.* (p. 0). Oxford University Press. https://doi.org/10.1093/oso/9780198859987.003.0009

Si, Y., & Reiter, J. P. (2013). Nonparametric bayesian multiple imputation for incomplete categorical variables in large-scale assessment surveys. *Journal of Educational and Behavioral Statistics*, *38*, 499–521.

Teh, Y. W. (2017). Dirichlet process. In *Encyclopedia of machine learning and data mining.* (pp. 361–370). Springer.

Tein, J.-Y., Coxe, S., & Cham, H. (2013). Statistical power to detect the correct number of classes in latent profile analysis. *Structural Equation Modeling: a Multidisciplinary Journal*, *20*, 640–657. https://doi.org/10.1080/10705511.2013.824781

Tofighi, D., & Enders, C. K. (2008). Identifying the correct number of classes in growth mixture models. In G. R. Hancock & K. M. Samuelsen (Eds.), *Advances in latent variable mixture models.* (pp. 317–341). Information Age Publishing.

Tong, X., Kim, S., & Ke, Z. (2022). Impact of likelihoods on class enumeration in Bayesian Growth Mixture Modeling. In M. Wiberg, D. Molenaar, J. González, J.-S. Kim, & H. Hwang (Eds.), *Quantitative psychology.* (pp. 111–120). Springer International Publishing. https://doi.org/10.1007/978-3-031-04572-1_9

Tueller, S., & Lubke, G. (2010). Evaluation of structural equation mixture models parameter estimates and correct class assignment. *Structural Equation Modeling: A Multidisciplinary Journal*, *17*, 165–192. https://doi.org/10.1080/10705511003659318

Vinh, N., Epps, J., & Bailey, J. (2009). Information theoretic measures for clusterings comparison: Is a correction for chance necessary?. *Proceedings of the 26th Annual International Conference on Machine Learning*, 1073–1080. https://doi.org/10.1145/1553374.1553511

Wade, S., & Ghahramani, Z. (2018). Bayesian cluster analysis: Point estimation and credible balls (with discussion). *Bayesian Analysis*, *13*, 559–626. https://doi.org/10.1214/17-BA1073

Wade, S., & Wade, M. S. (2015). Package ‘mcclust. ext. *Journal of Computational and Graphical Statistics*, *16*, 526–558.

Zeileis, A., Hothorn, T., & Hornik, K. (2008). Model-based recursive partitioning. *Journal of Computational and Graphical Statistics*, *17*, 492–514. https://doi.org/10.1198/106186008X319331

Zhang, Z., Hamagami, F., Lijuan Wang, L., Nesselroade, J. R., & Grimm, K. J. (2007). Bayesian analysis of longitudinal data using growth curve models. *International Journal of Behavioral Development*, *31*, 374–383. https://doi.org/10.1177/0165025407077764

Appendix:

A1. Explanation of the Supplemental R Code

The supplemental material provides fully executable R code for analyzing the ECLS-K subsample, implementing the Dirichlet Process Mixture (DPM) and Mixture of Finite Mixtures (MFM) models. The two NIMBLE models are named gmm_dpm and gmm_mfm, respectively. The subsequent explanation focuses on four key components: (1) the stick-breaking process specific to the DPM and MFM models; (2) running the Markov Chain Monte Carlo (MCMC) sampler in NIMBLE; (3) creating a heatmap and determining the number of classes using the variation of information (VI) loss function; and (4) addressing the label-switching issue. Comments on nuanced code lines can be found in the R document. For interested readers, additional information about NIMBLE is available in its online user manual at https://r-nimble.org/manual/cha-welcome-nimble.html.

A1.1. *Stick-Breaking Process Construction*

After loading the required packages and importing the data, the MFM example begins with a user-defined NIMBLE function, sbpMFM(), which mirrors the stick-breaking process of the MFM (see Construction 2; Hu et al. (2023)). In the code line below, pi presents the class weight variable and the index M is the upper bound of the number of classes. In particular, for those familiar with the BUGS or JAGS code, one point worth mentioning is that NIMBLE does not allow variables with blank indices, such as in pi[]. The dimensions of the variable must be specified, so that pi[1:M].

```
pi [1: M ] < - sbpMFM (r=r _ beta [1: M ])
```

The NIMBLE code for the stick-breaking process construction for the DPM model is in the gmm_dpm model, where we provide two approaches to build the construction in NIMBLE: 1) an automatic construction using the NIMBLE function stick_breaking(); and 2) a manual construction exactly follow the hierarchical structure of the truncated stick-breaking process. In the code line below, pi presents the class weight variable and the index H is the upper bound of the number of classes. The function stick_breaking() builds class weights from the stick-breaking proportions saved in the vector v. The manual construction is commented out, as it is equivalent to the stick_breaking() function.

```
pi [1: H ] < - stick _ breaking (v [1:(H −1)])
```

A1.2. *MCMC Sampling Procedure*

After building our model, we then run the MCMC sampler. We first prepare the associated constants, initial values, and data list for the model (data must be saved as a list in NIMBLE). Constants are values that never change in the model (e.g., for-loop indices) and are stored in the list const. Initial values are contained in the list inits. Although NIMBLE does not require the model to be fully initialized, providing initial values for all parameters can prevent unexpected errors. To inspect model initialization, one can use the function model$initializeInfo().

The next steps involve creating and running the MCMC. The nimbleModel() function builds the model specified in the nimbleCode() function. The compileNimble() function performs the following four tasks: 1) creates C++ code files for all the model components; 2) calls the system's C++ compiler; 3) loads the compiled object(s) back into R; and 4) generates R objects for using the compiled model. The configureMCMC() function creates an MCMC configuration, which contains information needed to build an MCMC for the considered model, including variables that need to be sampled and their corresponding samplers. The monitors argument specifies which variables' MCMC samples will be recorded. Then, we create the MCMC algorithm by first using the function buildMCMC(), compiling the MCMC object via compileNimble(), and finally running it with one or more chains via runMCMC(). To specify values of the burn-in and thinning periods after sampling, one can set values for the nburnin and thin arguments.

```
model < - nimbleModel (gmm _ mfm, data=dat.
gmm, inits=inits,
constants=consts)
cmodel < - compileNimble (model)
conf < - configureMCMC (model, monitors=c
(‘ z ‘,’pi’,’ muLS ‘,
‘cov_b’,’ sig2 ‘), print=TRUE)
mcmc < - buildMCMC(conf)
cmcmc < - compileNimble(mcmc, project
=model)
samples < - runMCMC(cmcmc, niter=nmcmc,
nburnin=nburn, thin=nthin,
setSeed=TRUE)’
```

A1.3. *Determining the Number of Classes*

After extracting MCMC samples for the monitored objects, one can compute the posterior similarity matrix based on individuals' class membership stored in z.nogap using the comp.psm() function. The posterior similarity matrix can then be visualized through a heat map using plotpsm function. The function minVI() identifies the partition minimizing the posterior expected variation of information, which we use as a point estimate of the clustering (Wade & Ghahramani, 2018). On the basis of this representative clustering (out.vi$cl[1,]), the object nc.vi serves as a point estimate of the number of classes.

```
psm < - comp. psm (z. nogap)
plotpsm (psm)
out. vi < - minVI (psm, z. nogap, method =‘
all ‘, include.greedy=TRUE)
nc.vi < - length(unique(out.vi$cl[1,]))’
```

A1.4. *Addressing Label Switching*

Before summarizing posterior samples, one must address label switching in class labels for each retained MCMC sample. In the current study, we use an iterative version of the Equivalence Classes Representatives (ECR) algorithm through the ecr.iterative.1() function implemented in the label.switching R package (Papastamoulis, 2016). The object, neworder, contains a vector of indexes that can be used to reorder the posterior parameters using the permute.mcmc() function. These reordered posterior samples can then be used to obtain point or interval summaries.

```
run < - ecr. iterative .1(z=z. filter,
K=nc. vi)
neworder < - run $ permutations
pi. reorder < - matrix (0, nrow=ndraw,
ncol=nc. vi)
for (i in 1: ndraw) { pi. reorder [ i,] < -
pi. matrix [ i,][ neworder [ i,]]}
mu. reorder < - permute. mcmc (mu. array,
neworder)
cov. reorder < - permute. mcmc (cov. array,
neworder)
sig2. reorder < - permute. mcmc (sig2.
array, neworder)
```