Bayesian mixture modeling using a mixture of finite mixtures with normalized inverse Gaussian weights

Fumiya Iwashige and Shintaro Hashimoto Department of Mathematics, Hiroshima University

February 3, 2025

Abstract

In Bayesian inference for mixture models with an unknown number of components, a finite mixture model is usually employed that assumes prior distributions for mixing weights and the number of components. This model is called a mixture of finite mixtures (MFM). As a prior distribution for the weights, a (symmetric) Dirichlet distribution is widely used for conjugacy and computational simplicity, while the selection of the concentration parameter influences the estimate of the number of components. In this paper, we focus on estimating the number of components. As a robust alternative Dirichlet weights, we present a method based on a mixture of finite mixtures with normalized inverse Gaussian weights. The motivation is similar to the use of normalized inverse Gaussian processes instead of Dirichlet processes for infinite mixture modeling. Introducing latent variables, the posterior computation is carried out using block Gibbs sampling without using the reversible jump algorithm. The performance of the proposed method is illustrated through some numerical experiments and real data examples, including clustering, density estimation, and community detection.

Keywords: Bayesian nonparametrics; Clustering; Dirichlet distribution; Inverse Gaussian distribution; Markov chain Monte Carlo; Mixture models

1 Introduction

We consider a finite mixture model with an unknown number of components:

$$f(y|\theta_1, \dots, \theta_M, \pi) = \sum_{j=1}^M \pi_j f(y|\theta_j), \qquad (1.1)$$

where $M \in \{1, 2, \dots\}$ is the number of components, $y \in \mathbb{R}^d$ is a d-dimensional observation vector, and $\pi = (\pi_1, \dots, \pi_M)^{\top}$ is the mixing weight vector, satisfying $\sum_{j=1}^{M} \pi_j = 1$ and $\pi_j > 0$. For $j = 1, \ldots, M$, $f(\cdot|\theta_j)$ is a probability density or probability mass function parameterized by the component-specific parameter θ_i . The function $f(\cdot|\theta_i)$ is also called the kernel of the mixture model. Each observation is assumed to arise from one of the Mcomponents, and each component is weighted based on its frequency of occurrence. Mixture models are important statistical models used in model-based clustering and density estimation because they can model complex data-generating distributions in random phenomena by using multiple components. Finite mixture models have been applied in many fields, for instance, sociology (Handcock et al., 2007), economics (Frühwirth-Schnatter and Kaufmann, 2008) and genetics (McLachlan et al., 2002) and so on. In application, the determination of M is a very important issue. Correctly estimating M improves model interpretability, the accuracy of kernel parameter estimates and model predictions, and reduces computation time. Although various methods have been proposed for the selection of M, such as model selection criteria and hypothesis testing, using these criterion makes it difficult to quantify the uncertainty of M, and also introduces the bias that model selection is carried out. The comprehensive review of (finite) mixture models is provided by McLachlan (2000); Frühwirth-Schnatter (2006); Frühwirth-Schnatter et al. (2019); McLachlan et al. (2019). In this paper, we focus on the case where M is finite and make a clear distinction between the number of components M and the number of clusters k. In general, the number of components is written as $M = k + M_{na}$, where k is the number of components for which data are actually assigned, and M_{na} is the number of empty components (see also Argiento and De Iorio, 2022).

In Bayesian analysis, a finite mixture model that assumes prior distributions for the

mixing weights and the number of components is usually employed (see, e.g. Nobile, 1994; Miller and Harrison, 2018). Such a model is also called a mixture of finite mixtures (MFM), and the model is often used, as well as infinite mixture models represented by Dirichlet process mixture models. Although the reversible jump algorithm (Green, 1995; Richardson and Green, 1997) has been used to obtain samples from the posterior distribution based on MFM, it faces significant computational and implementation challenges. Methods based on marginal likelihoods have also been proposed (see, e.g. Nobile and Fearnside, 2007), while, as with reversible jump, the computational aspect is an issue. A method called sparse MFM (Rousseau and Mengersen, 2011; Malsiner-Walli et al., 2016) has also been proposed, which focuses on estimating the number of clusters k by using the overfitting model with a large fixed M, and choosing the prior distribution of the component rate well. Under same conditions, they showed that k has consistency for true M. Since it allows for many empty components, we cannot estimate the number of components M and it is not easy to quantify the uncertainty. Recently, the use of nonparametric Bayesian methods for MFM to estimate M has attracted much attention. In Miller and Harrison (2018), the exchangeable partition distribution is derived by marginalizing out M, and the restaurant process is constructed to overcome computational difficulties. In Frühwirth-Schnatter et al. (2021), a generalized MFM in which the parameters of the Dirichlet distribution depend on M is proposed. Miller and Harrison (2018) derived the theoretical result that the marginal posterior probabilities of k and M are asymptotically equivalent when the sample size is large, while only a sampling of k is obtained as with general nonparametric Bayesian methods. Thus, the number of components M cannot be sampled directly from the posterior distribution. Although Frühwirth-Schnatter et al. (2021)'s method can sample M directly, as a sparse MFM, it has the disadvantage of producing a large number of empty components and overestimating M. In addition, most MFM studies use a symmetric Dirichlet distribution as the mixing weight vector, and the choice of a hyperparameter is very important. Although it is possible to set a prior distribution and learn from the data, the Metropolis-Hastings algorithm is required.

The main purposes of this paper are 1) sampling M directly and efficiently compared to

MFM based on the Dirichlet distribution; 2) estimating M reasonably by suppressing M_{na} ;

3) proposing a MFM that is robust to the choice of hyperparameters of the distribution

on a simplex. To this end, we employ a normalized inverse Gaussian distribution as

the mixing weight vector of the finite mixture model. To the best of our knowledge,

there are few studies using the normalized inverse Gaussian distributions as the mixing

weights in the field of finite mixture models. In Bayesian nonparametric inference with

infinite mixtures, Lijoi et al. (2005) proposed a normalized inverse Gaussian process and

showed some analytical results of the posterior distribution under the process. Similarly,

since Miller and Harrison (2018) is a finite version of the Dirichlet process, the model

presented in this paper corresponds to a kind of finite alternative of Lijoi et al. (2005).

Furthermore, leveraging data augmentation with a latent gamma random variable and

the result of Argiento and De Iorio (2022), we construct an efficient posterior sampling

algorithm.

This paper is organized as follows. In Section 2, we introduce the MFM and its

equivalent representation using discrete probability measures. We also discuss the data

augmentation and the conditional posterior distribution of M_{na} , which are crucial for

constructing the computational algorithm. In Section 3, we present the proposal method

as well as the posterior computation algorithm. In Section 4, we illustrate some numerical

studies to compare the proposed method with existing methods. R code implementing

the proposed methods is available at Github repository:

https://github.com/Fumiya-Iwashige/MFM-Inv-Ga

2 Mixture of finite mixtures

We introduce a mixture of finite mixtures and its equivalent representation of the dis-

crete probability measure. The proposed model and the posterior computation algorithm

presented in the next section are largely based on the basic model presented in this section.

4

2.1 Formulation

Let each observations Y_1, \ldots, Y_n be univariate or multivariate in an Euclidean space. A MFM model is a statistical model defined by the following hierarchical representation.

$$M-1 \sim q_M, \quad q_M \text{ is a probability mass function (p.m.f.) on } \{0,1,2,\dots\},$$

$$\pi|M \sim P_{\pi}(\pi|M),$$

$$c_i|M,\pi \sim \operatorname{Categorical}_M(\pi),$$

$$\xi_m|M \overset{\text{i.i.d}}{\sim} p_0(\xi), \quad m=1,\dots,M,$$

$$\theta_i|c_i,\xi \overset{\text{ind}}{\sim} \delta_{\xi_{c_i}}(\theta_i), \quad i=1,\dots,n,$$

$$Y_i|\theta_i \overset{\text{ind}}{\sim} f(y_i|\theta_i), \quad i=1,\dots,n,$$

$$(2.1)$$

where $f(y|\theta_i)$ is a parametric model with parameter θ_i , $\delta_x(\cdot)$ is a point mass at x and ξ is a element of a vector $\xi = (\xi_1, \dots, \xi_M)^{\top}$. $p_0(\xi)$ is a prior density function of the parameter in mixture components. The parameter space is denoted by $\Theta \subset \mathbb{R}^d$. Each c_i is a latent allocation indicates to which component each observation is allocated. Given the number of components M, $P_{\pi}(\pi|M)$ is a probability distribution on the M-dimensional unit simplex, in that, this is the prior distribution for the mixing weights. We focus the case where the mixing weights can be further hierarchically expressed as follows:

$$S_m|M \stackrel{\text{ind}}{\sim} h$$
, h is a probability density function (p.d.f.) on $(0, \infty)$,
$$\pi = \left(\frac{S_1}{T}, \dots, \frac{S_M}{T}\right) \sim P_{\pi}(\pi|M), \quad T = \sum_{m=1}^M S_m,$$
(2.2)

where S_1, \ldots, S_M is unnormalized weights and π is (normalized) mixing weight vector. This representation is the most basic way to generate a probability distribution on the d-dimensional unit simplex $\mathbb{S}_d := \{w = (w_1, \ldots, w_d); w_i \geq 0, \sum_{i=1}^d w_i = 1\}$. One of the most famous examples of a distribution on \mathbb{S}_d is the Dirichlet distribution. If $S_m | M \sim \operatorname{Gamma}(\gamma_i, 1)$ in (2.2), $\pi \sim \operatorname{Dirichlet}(\gamma_1, \ldots, \gamma_d)$, where $\operatorname{Gamma}(a, b)$ is the gamma distribution with shape parameter a > 0 and scale parameter $\beta > 0$, and $\operatorname{Dirichlet}(\gamma_1, \ldots, \gamma_d)$ is the Dirichlet distribution with parameter $\gamma = (\gamma_1, \ldots, \gamma_d) > 0$.

If $\gamma_1 = \cdots = \gamma_d =: \gamma$, the distribution obtained by normalization is Dirichlet (γ, \ldots, γ) . The symmetric Dirichlet distribution is often used in the framework of MFM, because of conjugacy with categorical distributions and simplicity of computation. The symmetric structure of Dirichlet (γ, \ldots, γ) is also essential for marginalizing out M and deriving the exchangeable partition distribution. However, it is known that the estimation result is sensitive to the choice of the shape parameter γ . For example, Miller and Harrison (2018) recommended to use $\gamma = 1$ as a default choice and the value works well in many cases. The use of a small γ was recommended in terms of sparse or generalized MFMs (Rousseau and Mengersen, 2011; Malsiner-Walli et al., 2016; Frühwirth-Schnatter et al., 2021). A small value of γ indicates that many empty components can be created. Although most previous studies allow for the appearance of the empty components, there are several problems. First, M tends to be overestimated, making the interpretation of the model difficult. Second, the existing of the empty components decreases the predictive performance of the model, as we will see in a later section through density estimations. Finally, they lead to an increase in computation time. When we focus on the estimation of the number of components M, it is desirable to have few empty components. In other words, the discrepancy between M and k should be small. To achieve this, we use the normalized inverse Gaussian distribution as mixing weights, instead of the Dirichlet distribution.

2.2 Equivalent representations using discrete probability measures

We give an equivalent representation of the MFM. We can construct a discrete measure $P(\cdot) = \sum_{m=1}^{M} \pi_m \delta_{\xi_m}(\cdot)$ in the parameter space Θ , almost surely, where M, ξ and π are realizations from the distributions q_M , p_0 and P_{π} , respectively. Let $\theta_1, \ldots, \theta_n$ be random variables according to P. Then, the model (2.1) is equivalent to the following hierarchical

representation using a random measure P.

$$Y_i | \theta_i \stackrel{\text{ind}}{\sim} f(y_i | \theta_i), \quad i = 1, \dots, n,$$

$$\theta_1, \dots, \theta_n | P \stackrel{\text{i.i.d}}{\sim} P, \qquad (2.3)$$

$$P \sim \mathcal{P}(q_M, h, p_0),$$

where \mathcal{P} is a probability distribution of P with parameters q_M , h and p_0 . The representation (2.3) is the MFM described as a nonparametric Bayesian framework with infinite mixtures. If we replace \mathcal{P} with the Dirichlet process, (2.3) represents the well-known Dirichlet process mixture models (e.g., Escobar and West, 1995). If we replace \mathcal{P} with the normalized inverse Gaussian process, (2.3) represents the normalized inverse Gaussian process mixture models (Lijoi et al., 2005). For MFM, Argiento and De Iorio (2022) proposed a normalized independent finite point process (Norm-IFPP), which is a class of flexible prior distributions for P. We employ the representation (2.3) with the Norm-IFPP. The advantages are as follows. We can directly estimate M and k. This is a major difference from Miller and Harrison (2018), which estimates M indirectly through the number of clusters k. Furthermore, an efficient Gibbs sampler can be constructed by incorporating the data augmentation with a latent Gamma random variable. This data augmentation enables us to overcome the lack of conjugacy in the categorical distribution. Instead of using the probability density function of P_{π} , we can use the density function of h. This is the key to building an efficient MCMC algorithm for the proposed model.

Our study is in the spirit of nonparametric Bayes in that it accounts for uncertainty in the prior distribution of the parameters by means of a random measure based on the Norm-IFPP. The details of Norm-IFPP and and the independent finite point process (IFPP) are given in Argiento and De Iorio (2022).

2.3 Data augmentation and conditional distribution of M_{na}

We illustrate the data augmentation and conditional posterior distribution of M_{na} . We introduce the data augmentation in models (2.1) and (2.2). This technique is employed

in James et al. (2009) and Argiento and De Iorio (2022). Let \mathcal{M}_a and \mathcal{M}_{na} be the allocated and unallocated index sets, respectively. The conditional joint distribution of $\xi = (\xi_1, \dots, \xi_M)^{\top}$ and $S = (S_1, \dots, S_M)^{\top}$ given M and a label vector c is

$$p(S, \tau | M, c) \propto \left(\prod_{i=1}^{n} \frac{S_{c_i}}{T} \right) \left(\prod_{m=1}^{M} h(s_m) p(\tau_m) \right)$$
$$= \left(\prod_{m \in \mathcal{M}^a} \left(\frac{S_m}{T} \right)^{n_m} h(s_m) p(\tau_m) \right) \left(\prod_{m \in \mathcal{M}^{n_a}} h(s_m) p(\tau_m) \right),$$

where $n_m = \#\{i : c_i = m\}$. Since this equation consists of categorical distributions, conjugacy with such a distribution is required as P_{π} for Gibbs sampling. This restriction is relaxed by using latent a random variable $U_n|T \sim \text{Gamma}(n,T)$, where $T = \sum_{m=1}^{M} S_m$. In fact,

$$p(S, \tau, U_n | M, c) \propto u^{n-1} \left(\prod_{m \in \mathcal{M}} e^{-S_m u} S_m^{n_m} h(s_m) p(\tau_m) \right) \left(\prod_{m \in \mathcal{M}^{n_a}} e^{-S_m u} h(s_m) p(\tau_m) \right).$$

Thus, the conditional distribution of each S_m is

$$S_m|U_n, M, c \stackrel{\text{ind}}{\sim} e^{-S_m u} S_m^{n_m} h(s_m), \quad m \in \mathcal{M}_a,$$
 (2.4)

$$S_m|U_n, M, c \stackrel{\text{i.i.d}}{\sim} e^{-S_m u} h(s_m), \quad m \in \mathcal{M}_{na}.$$
 (2.5)

If it is easy to generate random variables from the distributions in (2.4) and (2.5), an efficient Gibbs sampling algorithm can be constructed. For example, when the prior distribution of the unnormalized weight is an inverse Gaussian distribution, (2.4) and (2.5) are generalized inverse Gaussian distributions. Introducing U_n , the update of the variable π is replaced by the update of the variable S_m . Thus, the selection of h is essential in MCMC updates.

Under this data augmentation, the conditional distribution of M_{na} is established in Theorem 5.1 in Argiento and De Iorio (2022). This theorem states that if $\mathcal{P}(q_M, h, p_0)$ follows a Norm-IFPP, then the posterior distribution of the random measure P, given U_n , is a superposition of a finite point process with fixed points and an IFPP. The IFPP characterizes the process of unallocated jumps, where the discrete probability distribution that serves as its parameter corresponds to the distribution of M_{na} . The conditional distribution of M_{na} is given by

$$\mathbb{P}(M_{na} = m | \theta, U_n = u) \propto \frac{(m+k)!}{m!} \psi(u)^m q_M(m+k), \quad m = 1, 2, \dots,$$
 (2.6)

where $\theta = (\theta_1, \dots, \theta_n)^{\top}$, k is the number of cluster (unique values of θ) and ψ is a Laplace transform of h. We sample M_{na} from (2.6) in the MCMC algorithm. Then, we straightforwardly obtain M by adding k to M_{na} .

Remark 2.1. Although we use the data augmentation $U_n|T \sim \text{Gamma}(n,T)$, the distributions (2.4) and (2.5) are generally complex. In addition, the result of Argiento and De Iorio (2022) applies only to h that have a Laplace transform. Thus, the inverse Gaussian distribution is one of the few examples that satisfy both computational and theoretical constraints.

3 Methodology

In this section, we propose a mixture of finite mixtures with normalized inverse Gaussian weights. Moreover, we develop an efficient posterior sampling algorithm based on Argiento and De Iorio (2022).

3.1 Mixture of finite mixtures with normalized inverse Gaussian weights

We propose a mixture of finite mixtures with normalized inverse Gaussian weights (denoted by MFM-Inv-Ga), where the notation explicitly specifies h because it is essential for computation. Our proposal model only requires (2.2) to be

$$S_m | M \stackrel{\text{ind}}{\sim} \text{Inv-Ga}(\alpha, 1), \quad \alpha > 0,$$

$$\pi = \left(\frac{S_1}{T}, \dots, \frac{S_M}{T}\right) \sim \text{Norm-Inv-Ga}(\alpha, \dots, \alpha),$$

where Inv-Ga(α, β) is the inverse Gaussian distribution with shape parameter $\alpha > 0$ and scale parameter $\beta > 0$, and Norm-Inv-Ga($\alpha_1, \ldots, \alpha_d$) is the normalized inverse Gaussian distribution with parameters $\alpha_i > 0$ for $i = 1, \ldots, d$. The probability density function of the inverse Gaussian distribution is given by

$$h(s) = \frac{\alpha}{\sqrt{2\pi}} s^{-3/2} \exp\left(-\frac{1}{2} \left(\frac{\alpha^2}{s} + \beta^2 s\right) + \beta \alpha\right), \tag{3.1}$$

where $\alpha > 0$ is the shape parameter and $\beta > 0$ is the scale parameter. The mean and variance of S are given by

$$\mathbb{E}[S] = \frac{\alpha}{\beta}, \quad \text{Var}[S] = \frac{\alpha}{\beta^3}.$$
 (3.2)

From Lijoi et al. (2005), the probability density function of the normalized inverse Gaussian distribution with parameters $\alpha_i > 0$, i = 1, ..., d is given by

$$f(\pi) = \frac{e^{\sum_{i=1}^{d} \alpha_i} \prod_{i=1}^{d} \alpha_i}{2^{d/2 - 1} \pi^{d/2}} \times K_{-d/2} \left(\sqrt{\mathcal{A}_d(\pi_1, \dots, \pi_{d-1})} \right) \times \left(\pi_1^{3/2} \cdots \pi_{d-1}^{3/2} \left(1 - \sum_{i=1}^{d-1} \pi_i \right)^{3/2} \times \mathcal{A}_d(\pi_1, \dots, \pi_{d-1})^{d/4} \right)^{-1},$$
(3.3)

where $\mathcal{A}_d(\pi_1, \ldots, \pi_{d-1}) = \sum_{i=1}^{d-1} \alpha_i^2/\pi_i + \alpha_d^2 \left(1 - \sum_{i=1}^{d-1} \pi_i\right)^{-1}$ and K_m is the modified Bessel functions of the third kind of order m (see, also Ghosal and van der Vaart, 2017). Due to the breakdown of conjugacy with the categorical distribution, constructing an efficient Gibbs sampler is challenging. Because of the complexity of (3.3), the calculation in Miller and Harrison (2018) is intractable: constructing an MCMC algorithm based on the restaurant process by marginalizing out M is extremely challenging. To overcome this difficulty, we use the data augmentation in Subsection 2.3 and the method proposed in Argiento and De Iorio (2022). As a result, it is sufficient to use not (3.3) but (3.1) and (3.8) to construct our MCMC algorithm.

Algorithm 1 Block Gibbs sampler

Step 0. Set initial values.

Step 1. Sample U_n from Gamma(n,T), where $T = \sum_{m=1}^{M} S_m$.

Step 2. Sample c_i , for i = 1, ..., n, the following discrete probability distribution defined by each probabilities

$$\mathbb{P}(c_i = j | \text{rest}) \propto S_i f(y_i | \tau_i), \quad j = 1, \dots, M.$$

Step 3. Sample the hyper parameter η of the q_M from

$$p(\eta|\text{rest}) \propto \Psi(u,k)p(\eta), \quad \Psi(u,k) := \sum_{m=0}^{\infty} \frac{(m+k)!}{m!} \psi(u)^m q_M(m+k).$$

Step 4. Sample M_{na} from the following discrete distribution

$$\mathbb{P}(M_{na} = m | \text{rest}) \propto \frac{(m+k)!}{m!} \psi(u)^m q_M(m+k), \quad m = 0, 1, \dots$$

Step 5. Sample S_m , for m = 1, ..., k from $GIG(2u + 1, \alpha^2, n_m - 1/2)$.

Step 6. Sample τ_m , for $m = 1, \ldots, k$ from

$$p(\tau_m|\text{rest}) \propto \left\{ \prod_{i;\theta_i = \tau_m} f(y_i|\tau_m) \right\} p(\tau).$$

Step 7. Sample $S_{m_{na}}$, for $m_{na} = k + 1, ..., M_{na} + k$ from $GIG(2u + 1, \alpha^2, -1/2)$.

Step 8. Sample $\tau_{m_{na}}$, for $m_{na} = k + 1, \dots, M_{na} + k$ from the prior $p(\tau)$.

3.2 Posterior computation

We present a fast and efficient posterior computation algorithm for the proposed method. To this end, we adopt the blocked Gibbs sampling scheme in Argiento and De Iorio (2022). Let η be a hyper-parameter of the prior distribution q_M , and $p(\cdot)$ be a joint prior density function except for h and q_M . $n_m = \#\{i; c_i = m\}$ is the size of the mth cluster. $\psi(u)$ is the Laplace transform of h, which is defined by $\psi(u) := \mathbb{E}[e^{-uS_m}]$ for $u \geq 0$.

We summarize the algorithm in Algorithm 1. The point of this algorithm is the data augmentation through the latent variable such as $U_n|T \sim \text{Gamma}(n,T)$, where n is the sample size and $T = \sum_{m=1}^{M} S_m$. It is important to sample the number of empty components M_{na} from (2.6) in Step 4. This step allows for direct sampling of M by adding k and label variables to be updated as in the finite mixture model with given M

in step 2. The update is the same as the telescoping sampling proposed by Frühwirth-Schnatter et al. (2021), and the method is more efficient than the classical restaurant process. However, in implementation, it is important to determine q_M so that the series $\Psi(u,k)$ in Steps 3 and 4 can be written analytically and random variables can be easily generated from the full conditional distributions of η and M_{na} . In Steps 5 and 7, if h is the Inv-Ga(α , 1), each generalized inverse Gaussian distribution (denoted by GIG) is immediately derived from (2.4) and (2.5). It is easy to generate random numbers from the GIG distribution. In Steps 6 and 7, the assigned unnormalized weights and kernel parameters are updated. When M_{na} sampled in Step 4 is greater than or equal to 1, Steps 8 and 9 are executed. Thus, when we get many empty components, it takes longer to compute.

3.3 Prior distributions for the number of mixture components

Assume that M-1 follows a discrete probability distribution with the support $\{0, 1, 2, ...\}$. In this paper, we consider Poisson(Λ) ($\Lambda > 0$) for M-1, because the constraints in Steps 3 and 4 are satisfied. In fact, from Argiento and De Iorio (2022), we have

$$\Psi(u,k) = \Lambda^{k-1}(\Lambda\psi(u) + k) \exp(\Lambda(\psi(u) - 1)). \tag{3.4}$$

Assuming $\Lambda \sim \text{Gamma}(a_{\Lambda}, b_{\Lambda})$ for $a_{\Lambda}, b_{\Lambda} > 0$, the full conditional distributions of Λ and M_{na} are given by

$$\begin{split} p(\Lambda|\text{rest}) &\propto \psi(u)(k+a_{\Lambda}-1)\text{Ga}(k+a_{\Lambda}+1,1-\psi(u)+b_{\Lambda}) \\ &\qquad \qquad + k(b_{\Lambda}+1-\psi(u))\text{Ga}(k+a_{\Lambda},1-\psi(u)+b_{\Lambda}), \end{split}$$

$$\mathbb{P}(M_{na}=m|\text{rest}) &\propto \Lambda \psi(u)\text{Shifted-Poisson}(\Lambda \psi(u),1) + k\text{Poisson}(\Lambda \psi(u)), \end{split}$$

respectively, where Shifted-Poisson(Λ , t) is parallel shifted of the distribution Poisson(Λ) by t. If we consider the negative binomial distribution as a prior distribution for M-1, the full conditional distributions are also obtained but learning hyper-parameters become

slightly troublesome.

3.4 Specification of kernels

The choice of kernel is important, and the appropriate kernel must be selected for the purpose. In this paper, although we do not discuss the details of the selection of kernels, we present some famous kernels for the sake of completeness.

3.4.1 Cluster analysis and density estimation

One of the most famous and useful kernels is the (multivariate) normal kernel $N(\cdot|\mu,\sigma^2)$. In the univariate case, we often use the normal inverse gamma model $N(\mu|m_0,\sigma^2/\eta) \times IG(\sigma^2|c_0,C_0)$ as a prior distribution of $\xi=(\mu,\sigma^2)$, where $m_0\in\mathbb{R}$, η , c_0 , $C_0>0$. The parameter η is called a smoothing parameter and plays an important role in density estimation. It is possible to include a hierarchical prior for η . In the multivariate case, we often employ the normal inverse Wishart model $N_r(\mu|m_0,\Sigma/B_0)\times W^{-1}(\Sigma|c_0,C_0)$ as a prior distribution of (μ,Σ) , where $m_0\in\mathbb{R}^r$, $B_0>0$, $c_0>r-1$ and C_0 is a positive definite matrix. We will use normal kernels in later numerical experiments in Sections 4.1 and 4.2 for clustering and density estimation. In the context of finite mixture models, various kernels have been proposed. If we have prior information that the data have skewness, the skew normal or skew-t kernel is also useful (see e.g., Frühwirth-Schnatter and Pyne, 2010). The skew-normal and skew-t distributions is easy to handle because these kernels have the scale mixtures of normal representations, except for the degree of freedom of skew-t distribution.

3.4.2 Network analysis

As an application of the proposed method, we perform community detection on network data. Community detection is the task of identifying dense subclasses in network data and corresponds to clustering and estimating the number of components, called the number of communities in network analysis. Note that the number of components of finite mixture models is equivalent to the number of communities in the network, and both are denoted

M. Estimating the number of communities is an important problem and various methods have been proposed (Shi and Malik, 2000; Newman, 2004; White and Smyth, 2005). From a model-based perspective, it is essentially the same as estimating the number of components in a finite-mixture model, and we can apply MFM. The stochastic block model is a famous statistical model of network data (Henze, 1986; Nowicki and Snijders, 2001; Geng et al., 2019), which assumes a stochastic block structure behind and specifies the community structure by estimating the probability of edges being drawn between each group. We estimate the number of communities with the MFM in the framework of this stochastic block model. Geng et al. (2019) proposed a stochastic block model based on MFM with Dirichlet weights, and also constructed a similar algorithm to Miller and Harrison (2018).

MFM can be easily applied to community detection by modifying the kernel. Data y are replaced by the adjacency matrix $A \in \mathbb{R}^{n \times n}$, where $A = (A_{ij}) \in \{0, 1\}^{n \times n}$ and n is the size of the node. When $A_{ij} = 1$, this indicates that an edge is drawn from the i th node to the j th node, and when $A_{ij} = 0$, this does not. For simplicity, we assume that the adjacency matrix is not direct and does not have a self-loop, in that $A_{i,j} = A_{ji}$ and $A_{ii} = 0$, where $1 \le i < j \le n$. The stochastic block model is formulated as follows,

$$A_{ij}|Q, M \sim \text{Bernoulli}(\theta_{ij}), \quad \theta_{ij} = Q_{c_i c_j}, \quad 1 \le i < j \le n,$$

$$Q_{rs} \sim \text{Beta}(a_Q, b_Q), \quad 1 \le r \le s \le M,$$

$$(3.5)$$

where a_Q , $b_Q > 0$ and M is the number of communities. $Q = (Q_{rs}) \in [0,1]^{M \times M}$ is a symmetric matrix and defines the stochastic block structure of a network. Each element Q_{rs} represents the probability that an edge is drawn between any node belonging to the community label r and any node belonging to the community label s. To perform community detection based on the proposed model, we just set the Bernoulli likelihood as the kernel.

3.5 Evaluation of the number of empty components

From the point of view of the interpretability of the model, the generalization performance, and the computational efficiency, it is reasonable that the number M_{na} should be small. For the full conditional distribution of M_{na} , the following inequality holds, where $M - 1|\Lambda \sim \text{Poisson}(\Lambda)$ and $\Lambda \sim \text{Gamma}(a_{\Lambda}, b_{\Lambda})$ for $a_{\Lambda}, b_{\Lambda} > 0$.

Proposition 3.1. For the full conditional distribution of M_{na} , the inequalities

$$\mathbb{P}(M_{na} \ge 1 | U_n = u, \ k, \ M = m, \ \Lambda) \le \Lambda \psi(u) \left(1 + \frac{1}{\Lambda \psi(u) + k} \right), \tag{3.6}$$

$$\mathbb{P}(M_{na} \ge 1 | U_n = u, \ k, \ M = m) \le \psi(u) \frac{a_{\Lambda}}{b_{\Lambda}} \left(1 + \frac{1}{k} \right)$$
(3.7)

holds, where ψ is the Laplace transform of the probability distribution for h.

Proof. When $m \neq 0$,

$$\mathbb{P}(M_{na} = m | U_n = u, k, M = m, \Lambda) = \frac{k}{\Lambda \psi(u) + k} \mathcal{P}_0(m; \Lambda \psi(u)) + \frac{\Lambda \psi(u)}{\Lambda \psi(u) + k} \mathcal{P}_1(m; \Lambda \psi(u)).$$

Thus, the conditional expectation of M_{na} is

$$\mathbb{E}[M_{na}|U_n = u, \ k, \ M = m, \ \Lambda] = \frac{k}{\Lambda\psi(u) + k} \times \Lambda\psi(u) + \frac{\Lambda\psi(u)}{\Lambda\psi(u) + k} \times (\Lambda\psi(u) + 1)$$
$$= \Lambda\psi(u) \left(1 + \frac{1}{\Lambda\psi(u) + k}\right).$$

Furthermore,

$$\mathbb{E}[M_{na}|U_n = u, \ k, \ M = m] = \mathbb{E}_{\Lambda}[\mathbb{E}[M_{na}|U_n = u, \ k, \ M = m, \ \Lambda]]$$

$$= \int \left(\Lambda \psi(u) \left(1 + \frac{1}{\Lambda \psi(u) + k}\right)\right) \frac{b_{\Lambda}^{a_{\Lambda}}}{\Gamma(a_{\Lambda})} \Lambda^{a_{\Lambda} - 1} \exp(-b_{\Lambda}\Lambda) d\Lambda$$

$$\leq \psi(u) \times \frac{a_{\Lambda}}{b_{\Lambda}} + \frac{\psi(u)}{k} \times \frac{a_{\Lambda}}{b_{\Lambda}}$$

$$= \psi(u) \times \frac{a_{\Lambda}}{b_{\Lambda}} \left(1 + \frac{1}{k}\right).$$

Finally, the result follows from Markov's inequality.

Proposition 3.1 shows that $\psi(u)$ plays an important role in the generation of empty components. The critical difference between the inverse Gaussian and gamma distributions in estimating M in the MFM is the Laplace transform. Let $\psi_{\text{Inv-Ga}}(u)$ and $\psi_{\text{Ga}}(u)$ be the Laplace transforms of Inv-Ga (α, β) and Gamma (γ, η) , respectively. The we have

$$\psi_{\text{Inv-Ga}}(u) = \exp\left(\alpha \left(1 - \sqrt{1 + 2u}\right)\right), \quad u \ge 0,$$
(3.8)

$$\psi_{\text{Ga}}(u) = \left(\frac{\eta}{\eta + u}\right)^{\gamma}, \quad u \ge 0.$$
(3.9)

Laplace transforms $\psi_{\text{Inv-Ga}}$ and ψ_{Ga} are decreasing functions with respect to u. The former has exponential decay, while the latter is polynomial. Figure 1 shows the graphs of the Laplace transforms of inverse Gaussian and gamma distributions when the shape parameters are $\alpha, \gamma = 1.0, 0.2, 10^{-1}, 10^{-2}, 10^{-3}$ and the scale parameters are 1. It can be seen that $\psi_{\text{Inv-Ga}}$ decreases much faster than ψ_{Ga} . The speed of this decrease has a significant impact on the appearance of the empty component in estimating the number of components.

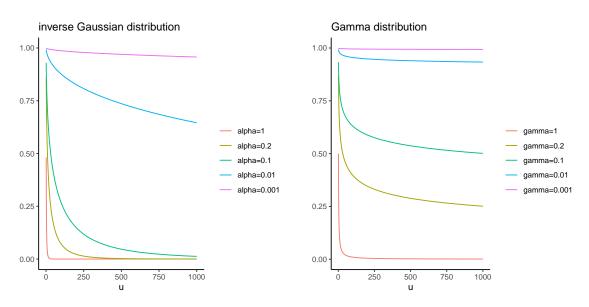


Figure 1: When shape parameters are $\alpha = \gamma = 1, 0.2, 10^{-1}, 10^{-2}, 10^{-3}$, left graphs are Laplace transforms of Inv-Ga(α , 1), right graphs are Laplace transforms of Gamma(γ , 1).

Inequalities (3.6) and (3.7) are not for marginal posterior distributions, but for full conditional distributions of M_{na} . This is due to the fact that the marginal posterior distributions of U_n are analytically intractable.

4 Empirical demonstrations

We evaluate the performance of the MFM-Inv-Ga and MFM-Ga methods through some numerical experiments. Recall that γ is the shape parameter of the prior distribution of unnormalized weights in the proposed method (MFM-Inv-Ga), while α is that of MFM-Ga.

4.1 Inference for the number of mixture components and clustering

In this subsection, we illustrate the performance of clustering and inference for the number of components using artificial and real data.

4.1.1 Artificial data

In this simulation, we assume that $M_{\text{true}} = 3$. We generate data from the following multivariate normal distribution:

$$f(y|\mu_1, \mu_2, \mu_3) = 0.8 \cdot N_2(y|\mu_1, I_2) + 0.1 \cdot N_2(y|\mu_2, I_2) + 0.1 \cdot N_2(y|\mu_3, I_2),$$

where $\mu_1 = (0,0)^{\top}$, $\mu_2 = (0,10)^{\top}$, $\mu_3 = (7.5,10)^{\top}$ and I_2 is the 2×2 identity matrix. We set n=300 and generate 50 dataset. Each MCMC iteration is 2000 and the first half of 1000 samples is not used as a burn-in period. We assume that $M-1|\Lambda \sim \text{Poisson}(\Lambda)$ and $\Lambda \sim \text{Gamma}(1,1)$. We employ the multivariate normal kernel $f(y|\mu,\Sigma) \sim N_2(y|\mu,\Sigma)$ and the normal inverse Wishart model $N_2(\mu|m_0,\Sigma/B_0) \times \text{W}^{-1}(\Sigma|c_0,C_0)$ as the prior of the parameters in the kernel, where m_0 is the sample mean vector, $B_0=1$, $c_0=2+1.5$ and C_0 is the sample covariance matrix. The shape parameters are $\alpha, \gamma=1.0, 0.2, 10^{-1}, 10^{-2}, 10^{-3}$. This choice of α and γ induces the mean and variance of Inv-Ga $(\alpha,1)$ and Gamma $(\gamma,1)$ to be equivalent. Hence, the first and second moments of the inverse Gaussian and gamma distribution are matched for each shape parameter (see, also Lijoi et al., 2005). To measure performance, we consider the three criteria.

- The posterior mean of the number of components M.
- The posterior mean of the rand index. The rand index is a measure of the clustering fitting, and the value takes [0, 1]. When it is close to 1, the assignment estimate is reasonable.
- The posterior probability that the number of empty components M_{na} is equal to 0.

The respective averages over 50 repetitions are denoted by \widehat{RI} and $\widehat{\mathbb{P}}(M_{na}=0)$.

We report the posterior mean of M in Figure 2. It is observed that the results of the MFM-Inv-Ga method remain almost the same even if the shape parameter is varied. However, the MFM-Ga method tends to overestimate the number of mixture components as the shape parameter decreases. Table 1 shows the results of the clustering performance

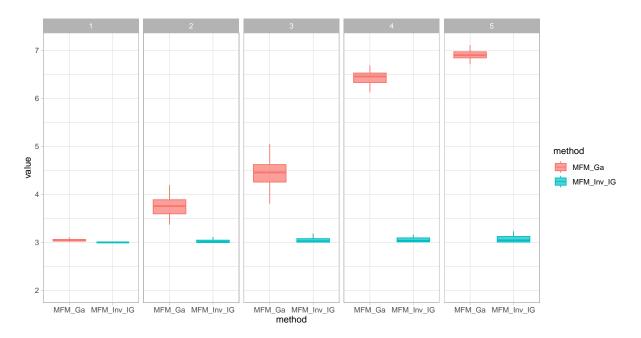


Figure 2: Box plots of the posterior mean of the number of components M for each shape parameter. From left to right, shape parameters are set as γ , $\alpha = 1, 0.2, 10^{-1}, 10^{-2}, 10^{-3}$.

and the posterior probability that M_{na} is equal to 0. For clustering, the MFM-Inv-Ga and MFM-Ga methods are comparable. Both methods have reasonable clustering accuracy, only slightly better for MFM-Ga than for MFM-Inv-Ga. As the shape parameters α and γ decrease, \widehat{RI}_i increases slightly and its standard deviation decreases. This is natural given that for data that have components with relatively very large cluster sizes, a suitable prior

on a simplex is one with large mass at the edges or vertices. It is important to note that Figure 2 and Table 1 show that the MFM-Inv-Ga method produces few empty components for all scenarios and provides reasonable estimates of M, but not MFM-Ga. This can be

Table 1: Results of the rand index $\widehat{\mathrm{RI}}_i$ and posterior probability that the number of empty components is equal to zero $\widehat{\mathbb{P}}(M_{na}=0)$ (their standard deviations are shown in parentheses) averaged over 50 Monte Carlo replications for each shape parameters.

		$\widehat{ ext{RI}}_i$			
α and γ	1	0.2	10^{-1}	10^{-2}	10^{-3}
MFM-Inv-Ga	0.993	0.994	0.995	0.995	0.995
	(0.013)	(0.006)	(0.006)	(0.004)	(0.005)
MFM-Ga	0.995	0.995	0.995	0.996	0.996
	(0.006)	(0.004)	(0.003)	(0.004)	(0.003)
		$\widehat{\mathbb{D}}/M$	0)		
		$\widehat{\mathbb{P}}(M_{na} =$			
α and γ	1	0.2	10^{-1}	10^{-2}	10^{-3}
MFM-Inv-Ga	1.000	0.995	0.982	0.947	0.952
	(0.001)	(0.008)	(0.032)	(0.135)	(0.087)

0.555

(0.080)

0.328

(0.066)

0.088

(0.012)

0.066

(0.013)

seen in Table 2. It can also be seen that MFM-Inv-Ga assigns a very high posterior probability to $M_{\rm true}$ than MFM-Ga, and the behavior of the posterior distributions for k and M is similar, together with Table 1. Table 3 shows the CPU times of MFM-Inv-Ga and MFM-Ga for 50 repetitions of MCMC with 2000 iterations, where $\alpha, \gamma = 1.0$ and 10^{-3} . When $\alpha = \gamma = 1.0$, MFM-Inv-Ga is faster than MFM-Ga on average, but MFM-Inv-Ga has more variability than MFM-Ga. On the other hand, MFM-Ga is very time-consuming in $\gamma = 10^{-3}$, because many empty components are created. As a result, MFM-Inv-Ga has the same clustering accuracy as MFM-Ga and outperforms MFM-Ga in terms of M estimation and CPU time.

4.1.2 Thyroid Data

MFM-Ga

0.979

(0.008)

We apply the proposed method to famous thyroid data. The data is available from the R package mclust and is well known as benchmark data for clustering. The sample size is 215 and the dimension of the data is 6. The thyroid disease of each patient is included

Table 2: Posterior probabilities and their standard deviations (shown in parentheses) of the number of components M averaged over 50 Monte Carlo replications. The highest value is bolded.

		MF	FM-Inv-G	a		
	M = 1	M=2	M = 3	M = 4	M = 5	$M \ge 6$
$\alpha = 1$	0.000	0.000	0.973	0.027	0.000	0.000
	(0.000)	(0.000)	(0.093)	(0.093)	(0.000)	(0.000)
$\alpha = 0.2$	0.000	0.080	0.886	0.033	0.001	0.000
	(0.000)	(0.274)	(0.268)	(0.048)	(0.003)	(0.000)
$\alpha = 10^{-1}$	0.000	0.000	0.942	0.051	0.006	0.001
	(0.000)	(0.000)	(0.084)	(0.069)	(0.013)	(0.005)
$\alpha = 10^{-2}$	0.000	0.020	0.890	0.063	0.015	0.012
	(0.000)	(0.141)	(0.199)	(0.078)	(0.039)	(0.048)
$\alpha = 10^{-3}$	0.000	0.050	0.860	0.068	0.015	0.007
	(0.000)	(0.209)	(0.225)	(0.084)	(0.029)	(0.020)
		N	AFM-Ga			
$\gamma = 1$	0.000	0.000	0.948	0.049	0.002	0.000
	(0.000)	(0.000)	(0.055)	(0.049)	(0.006)	(0.001)
$\gamma = 0.2$	0.000	0.010	0.508	0.301	0.121	0.061
	(0.000)	(0.070)	(0.093)	(0.037)	(0.037)	(0.035)
$\gamma = 10^{-1}$	0.000	0.000	0.307	0.306	0.196	0.191
	(0.000)	(0.007)	(0.093)	(0.037)	(0.037)	(0.035)
$\gamma = 10^{-2}$	0.000	0.003	0.088	0.155	0.176	0.578
	(0.000)	(0.002)	(0.021)	(0.016)	(0.013)	(0.071)
$\gamma = 10^{-3}$	0.000	0.003	0.066	0.132	0.163	0.636
	(0.000)	(0.020)	(0.019)	(0.013)	(0.011)	(0.072)

Table 3: Comparison of CPU times (in seconds) and their standard deviations (shown in parentheses) for artificial data averaged over 50 replications, α , $\gamma=1.0,10^{-3}$

	MFM-Inv-Ga	MFM-Ga
$\alpha = \gamma = 1$	88.545	90.898
	(5.686)	(2.774)
$\alpha = \gamma = 10^{-3}$	87.061	192.234
	(12.286)	(8.932)

and classified into three categories: Normal, hypo and hyper. The number of diseases is 150, 30 and 35, respectively. Using these labels as true labels, the main interest is the accuracy of the clustering and whether the number of components is estimated to be 3.

We use the same model and shape parameters as in Section 4.1.1. We independently run the 5 MCMC chain with different initial values. The number of iterations for each chain is 20000 and finally we get the 10000 samples. We compare MFM-Inv-Ga and MFM-Ga through the posterior mean of M, the posterior probability $\mathbb{P}(M_{na} = 0|y_1, \dots, y_n)$ and the posterior mean of the rand index (RI). Table 4 shows that the result of the real data analysis is similar to that of Section 4.1.1. MFM-Inv-Ga provides a reasonable estimate of M for all shape parameters by not producing empty components. However, MFM-Ga overestimates M. Furthermore, MFM-Inv-Ga is slightly more accurate than MFM-Ga.

Table 4: Posterior means of M (M), posterior probabilities of $M_{na}=0$ ($\mathbb{P}(M_{na}=0|y_1,\ldots,y_n)$) and posterior means of rand index (RI) for thyroid data.

MFM-Inv-Ga						
	\hat{M}	$\mathbb{P}(M_{na}=0 y_1,\ldots,y_n)$	RI			
$\alpha = 1$	3.200	1.000	0.895			
$\alpha = 0.2$	3.213	0.994	0.895			
$\alpha = 10^{-1}$	3.225	0.987	0.894			
$\alpha = 10^{-2}$	3.222	0.987	0.895			
$\alpha = 10^{-3}$	3.489	0.960	0.893			

		MFM- Ga	
	\hat{M}	$\mathbb{P}(M_{na}=0 y_1,\ldots,y_n)$	RI
$\gamma = 1$	3.670	0.962	0.893
$\gamma = 0.2$	4.944	0.434	0.890
$\gamma = 10^{-1}$	4.945	0.394	0.893
$\gamma = 10^{-2}$	7.425	0.067	0.893
$\gamma = 10^{-3}$	7.626	0.046	0.895

4.2 Density estimation

As seen in the previous subsection, the main difference between the MFM-Inv-Ga and MFM-Ga methods is the frequency of occurrence of the empty components. The MFM-Ga method can achieve very high clustering accuracy by setting a small γ and allowing

empty components. However, small γ not only makes the model difficult to interpret but also degrades the predictive accuracy of the model. In this subsection, we examine this phenomenon through density estimation using predictive distributions.

We used the famous galaxy dataset, which is a small data set consisting of 82 velocities (km/sec) of different galaxies. The data is widely used in nonparametric Bayesian statistics as a benchmark for density estimation and cluster analysis. The details of the data are found in Roeder (1990). We set $M-1|\Lambda \sim \text{Poisson}(\Lambda)$ and $\Lambda \sim \text{Ga}(1,1/5)$. We also employ the univariate normal kernel $N(y|\mu,\sigma^2)$, and the normal-inverse gamma conjugate prior $N(\mu|m_0, \sigma^2/\tau) \times \text{IG}(\sigma^2|c_0, C_0)$ as the prior of the parameters in the kernel. Moreover, we assume that the prior of C_0 is $Gamma(d_0, D_0)$, and we set $m_0 =$ $(\max(\text{data}) + \min(\text{data}))/2$, $d_0 = 0.2$, $D_0 = 10/(\max(\text{data}) + \min(\text{data}))^2$ as Richardson and Green (1997). The parameter τ controls the smoothness of the estimated density function. We assume that the prior of the smoothing parameter τ is Gamma(w, W), where w = 0.5 and W = 50 as Escobar and West (1995). Parameters τ and C_0 should be carefully learned from the data, since density estimation is sensitive to their choice. The MCMC iterations are 100000 and the first half of the 90000 samples are not used as a burn-in period. The shape parameters are the same as in Section 4.1. We evaluated the result of density estimation and posterior probabilities of k, M, and M_{na} for each shape parameter. We show the estimated density functions using the posterior means in Figure 3. From the figure, it is observed that the shapes of the estimated densities using the MFM-Inv-Ga do not depend on the choice of the shape parameter α . However, results using the MFM-Ga method seem to be strongly influenced by the choice of the shape parameter γ . For $\gamma = 10^{-2}$ and 10^{-3} , MFM-Ga cannot capture two large peaks in the middle. The number of clusters in the galaxy data has been reported as 5 or 6 in existing studies. From table 5, in MFM-Inv-Ga, the posterior distributions of k have large probabilities at k=5 and 6 for all α and each posterior distribution induced by MFM-Inv-Ga is more similar than by MFM-Ga. For M, both MFM-Inv-Ga and MFM-Ga overestimate when α and γ are small. This can be seen in Table 6. The reason is why the data size n = 82 is small and M_{na} is easily produced. Tables 5 and 6 show that the

Table 5: Posterior probabilities of the number of components M and clusters k for Galaxy data.

			MI	FM-Inv-G	la			
	$M \leq 3$	M = 4	M = 5	M = 6	M = 7	M = 8	M = 9	$M \ge 10$
$\alpha = 1.0$	0.000	0.089	0.224	0.31	0.170	0.097	0.054	0.056
$\alpha = 0.2$	0.136	0.194	0.171	0.152	0.114	0.086	0.054	0.091
$\alpha = 10^{-1}$	0.071	0.096	0.154	0.144	0.137	0.102	0.079	0.216
$\alpha = 10^{-2}$	0.039	0.078	0.119	0.151	0.123	0.102	0.081	0.308
$\alpha = 10^{-3}$	0.057	0.111	0.170	0.153	0.122	0.101	0.071	0.217
	$k \le 3$	k = 4	k = 5	k = 6	k = 7	k = 8	k = 9	$k \ge 10$
$\alpha = 1.0$	0.000	0.092	0.241	0.331	0.164	0.092	0.046	0.034
$\alpha = 0.2$	0.156	0.215	0.206	0.182	0.118	0.063	0.033	0.026
$\alpha = 10^{-1}$	0.089	0.127	0.214	0.224	0.169	0.094	0.052	0.031
$\alpha = 10^{-2}$	0.088	0.123	0.201	0.232	0.160	0.107	0.054	0.035
$\alpha = 10^{-3}$	0.082	0.152	0.256	0.221	0.143	0.081	0.039	0.025
			1	MFM-Ga				
	$M \leq 3$	M=4	M = 5	$\frac{\text{MFM-Ga}}{M=6}$	M = 7	M = 8	M = 9	$M \ge 10$
$\frac{}{\gamma = 1.0}$	$\frac{M \le 3}{0.038}$	M = 4 0.104			M = 7 0.176	M = 8 0.122	M = 9 0.072	$\frac{M \ge 10}{0.091}$
$\frac{\gamma = 1.0}{\gamma = 0.2}$			M = 5	M = 6				
$\gamma = 1.0$ $\gamma = 0.2$ $\gamma = 10^{-1}$	0.038 0.018 0.019	0.104	M = 5 0.191	M = 6 0.206	0.176	0.122	0.072	0.091
$\gamma = 1.0$ $\gamma = 0.2$ $\gamma = 10^{-1}$ $\gamma = 10^{-2}$	0.038 0.018 0.019 0.009	0.104 0.042	M = 5 0.191 0.076	M = 6 0.206 0.098	0.176 0.114	0.122 0.117	0.072 0.103	0.091 0.432
$\gamma = 1.0$ $\gamma = 0.2$ $\gamma = 10^{-1}$ $\gamma = 10^{-2}$ $\gamma = 10^{-3}$	0.038 0.018 0.019	0.104 0.042 0.040	M = 5 0.191 0.076 0.057	M = 6 0.206 0.098 0.071	0.176 0.114 0.075	0.122 0.117 0.083	0.072 0.103 0.079	0.091 0.432 0.577
$\gamma = 1.0$ $\gamma = 0.2$ $\gamma = 10^{-1}$ $\gamma = 10^{-2}$ $\gamma = 10^{-3}$	0.038 0.018 0.019 0.009	0.104 0.042 0.040 0.023 0.004	M = 5 0.191 0.076 0.057 0.042 0.013	M = 6 0.206 0.098 0.071 0.059 0.019	0.176 0.114 0.075 0.071	0.122 0.117 0.083 0.076 0.033	0.072 0.103 0.079 0.075 0.038	0.091 0.432 0.577 0.645 0.865
$\gamma = 0.2$ $\gamma = 10^{-1}$ $\gamma = 10^{-2}$ $\gamma = 10^{-3}$	$0.038 \\ 0.018 \\ 0.019 \\ 0.009 \\ 0.003$ $k \le 3$	0.104 0.042 0.040 0.023	M = 5 0.191 0.076 0.057 0.042	M = 6 0.206 0.098 0.071 0.059	0.176 0.114 0.075 0.071	0.122 0.117 0.083 0.076	0.072 0.103 0.079 0.075	0.091 0.432 0.577 0.645
$\gamma = 0.2$ $\gamma = 10^{-1}$ $\gamma = 10^{-2}$ $\gamma = 10^{-3}$ $\gamma = 1.0$	$\begin{array}{c} 0.038 \\ 0.018 \\ 0.019 \\ 0.009 \\ 0.003 \\ \\ k \leq 3 \\ 0.044 \\ \end{array}$	$0.104 \\ 0.042 \\ 0.040 \\ 0.023 \\ 0.004$ $k = 4 \\ 0.120$	M = 5 0.191 0.076 0.057 0.042 0.013 $k = 5$ 0.232	M = 6 0.206 0.098 0.071 0.059 0.019 $k = 6$ 0.236	0.176 0.114 0.075 0.071 0.025	0.122 0.117 0.083 0.076 0.033	0.072 0.103 0.079 0.075 0.038	0.091 0.432 0.577 0.645 0.865
$\gamma = 0.2$ $\gamma = 10^{-1}$ $\gamma = 10^{-2}$ $\gamma = 10^{-3}$ $\gamma = 1.0$ $\gamma = 0.2$	$\begin{array}{c} 0.038 \\ 0.018 \\ 0.019 \\ 0.009 \\ 0.003 \\ \hline \\ k \leq 3 \\ 0.044 \\ 0.145 \\ \end{array}$	$0.104 \\ 0.042 \\ 0.040 \\ 0.023 \\ 0.004$ $k = 4 \\ 0.120 \\ 0.198$	M = 5 0.191 0.076 0.057 0.042 0.013 $k = 5$	M = 6 0.206 0.098 0.071 0.059 0.019	$0.176 \\ 0.114 \\ 0.075 \\ 0.071 \\ 0.025$ $k = 7$	0.122 0.117 0.083 0.076 0.033 $k = 8$	0.072 0.103 0.079 0.075 0.038	$0.091 \\ 0.432 \\ 0.577 \\ 0.645 \\ 0.865$ $k \ge 10$
$\gamma = 0.2$ $\gamma = 10^{-1}$ $\gamma = 10^{-2}$ $\gamma = 10^{-3}$ $\gamma = 1.0$ $\gamma = 0.2$ $\gamma = 10^{-1}$	$\begin{array}{c} 0.038 \\ 0.018 \\ 0.019 \\ 0.009 \\ 0.003 \\ \\ k \leq 3 \\ 0.044 \\ 0.145 \\ 0.145 \\ \end{array}$	$0.104 \\ 0.042 \\ 0.040 \\ 0.023 \\ 0.004$ $k = 4 \\ 0.120$	M = 5 0.191 0.076 0.057 0.042 0.013 $k = 5$ 0.232	M = 6 0.206 0.098 0.071 0.059 0.019 $k = 6$ 0.236	$0.176 \\ 0.114 \\ 0.075 \\ 0.071 \\ 0.025$ $k = 7 \\ 0.188$	0.122 0.117 0.083 0.076 0.033 $k = 8$ 0.104	$0.072 \\ 0.103 \\ 0.079 \\ 0.075 \\ 0.038$ $k = 9 \\ 0.044$	$0.091 \\ 0.432 \\ 0.577 \\ 0.645 \\ 0.865$ $k \ge 10 \\ 0.033$
$\gamma = 0.2$ $\gamma = 10^{-1}$ $\gamma = 10^{-2}$ $\gamma = 10^{-3}$ $\gamma = 1.0$ $\gamma = 0.2$	$\begin{array}{c} 0.038 \\ 0.018 \\ 0.019 \\ 0.009 \\ 0.003 \\ \hline \\ k \leq 3 \\ 0.044 \\ 0.145 \\ \end{array}$	$0.104 \\ 0.042 \\ 0.040 \\ 0.023 \\ 0.004$ $k = 4 \\ 0.120 \\ 0.198$	M = 5 0.191 0.076 0.057 0.042 0.013 $k = 5$ 0.232 0.304	M = 6 0.206 0.098 0.071 0.059 0.019 $k = 6$ 0.236 0.186	$0.176 \\ 0.114 \\ 0.075 \\ 0.071 \\ 0.025$ $k = 7 \\ 0.188 \\ 0.098$	$0.122 \\ 0.117 \\ 0.083 \\ 0.076 \\ 0.033$ $k = 8 \\ 0.104 \\ 0.040$	$0.072 \\ 0.103 \\ 0.079 \\ 0.075 \\ 0.038$ $k = 9 \\ 0.044 \\ 0.019$	$0.091 \\ 0.432 \\ 0.577 \\ 0.645 \\ 0.865$ $k \ge 10 \\ 0.033 \\ 0.009$

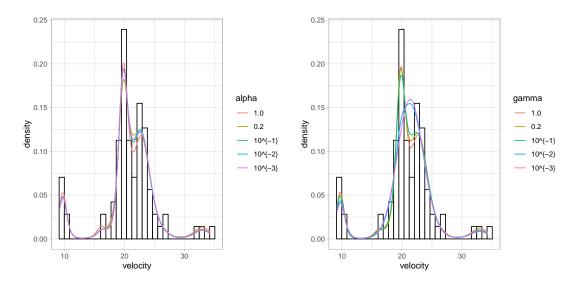


Figure 3: Results of density estimation for galaxy data using MFM-Inv-Ga (left) and MFM-Ga (right).

degree of overestimation is much smaller for MFM-Inv-Ga than for MFM-Ga.

Table 6: Posterior probabilities of the number of empty components is equal to zero for Galaxy data

$\mathbb{P}(M_{na}=0 y_1,\ldots,y_n)$						
α and γ	1	0.2	10^{-1}	10^{-2}	10^{-3}	
MFM-Inv-Ga	0.854	0.654	0.480	0.347	0.443	
MFM-Ga	0.648	0.081	0.040	0.010	0.002	

Focusing on the number of clusters k, it is interesting that MFM-Inv-Ga is more robust for shape parameter estimates k than MFM-Ga. This suggests that the prior distribution of k based on MFM-Inv-Ga is less informative than MFM-Ga, i.e., the same relationship holds for MFM-Inv-Ga and MFM-Ga as for normalized inverse Gaussian process and Dirichlet process. In Lijoi et al. (2005), the prior of k induced by a normalized inverse Gaussian process is not more informative for the precision parameter than the Dirichlet process. However, the prior k for the normalized inverse Gaussian process can be written in closed form, while that for MFM-Inv-Ga is given in complex integral form.

The shape parameter of MFM-Ga should be chosen carefully because it has a significant impact on clustering, density estimation, and the appearance of empty components. However, MFM-Inv-Ga is much more robust than MFM-Ga with respect to the choice of the shape parameter. Hence, MFM-Inv-Ga is superior to MFM-Ga in that it is much

easier to use than MFM-Ga.

4.3 Community detection

We apply the proposed method to the community detection for network data. Similar comparisons as in Section 4.1 are made for both artificial and real data. Since the number of components of the finite mixture models is equivalent to the number of communities of the network, we use the same notation M to denote the number of communities.

4.3.1 Artificial data

First, we illustrate the performance of the proposed method using simulation data. We assume that the true number of communities is 3, denoted by $M_{\text{true}} = 3$ and the number of nodes in the network is set as n = 150. We here consider the balanced network, in that the true allocation consists of M_{true} communities with 50 nodes. For the true probability matrix Q, we assume that each component is expressed by $q_{rs} = q + (p - q)I(r = s)$, where q = 0.1, p = 0.8 and $1 \le r \le s \le M_{\text{true}}$. The assumption indicates that the edges are more easily drawn within the same community and less easily drawn between different communities. In the setup, we generate the 50 data set.

We set $M - 1 | \Lambda \sim \text{Poisson}(\Lambda)$ and $\Lambda \sim \text{Ga}(1, 1)$, and employ (3.5) as a kernel and a prior, where $a_Q = 1$ and $b_Q = 1$. The values of the shape parameters, the evaluations and the MCMC setting are the same as those of Section 4.1.1.

We report the posterior mean of the number of communities in Figure 4. It is observed that the results are almost the same as those of Figure 2. Table 7 also shows that the results of the posterior probability of M_{na} are the same as Table 1. However, in terms of clustering accuracy for network data, MFM-Inv-Ga is higher and more accurate than MFM-Ga. From table 8, MFM-Inv-Ga has a much higher posterior probability at M_{true} for all shape parameters than MFM-Ga. As a result, in the context of community detection, MFM-Inv-Ga also achieves better clustering and community estimation than MFM-Ga.

We also compared the computation time with Geng's method (Geng et al., 2019), denoted MFM-Geng, when the shape parameters are 1.0 and 10^{-3} . The result in Table 9

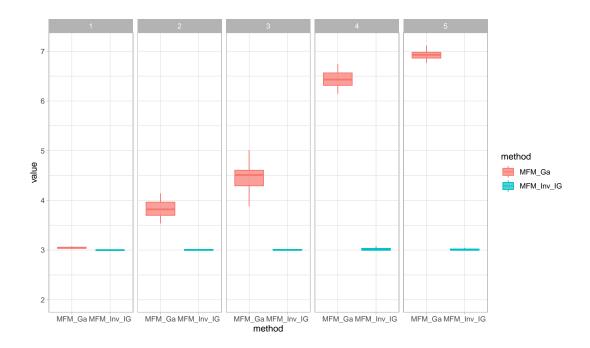


Figure 4: Box plots of the posterior mean of the number of communities M. From left to right, shape parameters are set as $\gamma, \alpha = 1, 0.2, 10^{-1}, 10^{-2}, 10^{-3}$.

Table 7: Results of the rand index $\widehat{\mathrm{RI}}_i$ and posterior probability that the number of empty components is equal to zero $\widehat{\mathbb{P}}(M_{na}=0)$ (their standard deviations are shown in parentheses) averaged over 50 Monte Carlo replications.

			$\widehat{\mathrm{RI}}_i$			
	$\alpha = \gamma = 1$	$\alpha = \gamma = 0.2$	$\alpha = \gamma = 10^{-1}$	$\alpha = \gamma = 10^{-2}$	$\alpha = \gamma = 10^{-3}$	
MFM-Inv-Ga	0.997	0.997	0.995	0.991	0.991	
	(0.010)	(0.019)	(0.032)	(0.044)	(0.044)	
MFM-Ga	0.995	0.996	0.980	0.987	0.910	
	(0.032)	(0.026)	(0.059)	(0.054)	(0.044)	
$\widehat{\mathbb{P}}(M=0)$						

$\mathbb{P}(M_{na}=0)$						
	$\alpha = \gamma = 1$	$\alpha = \gamma = 0.2$	$\alpha = \gamma = 10^{-1}$	$\alpha = \gamma = 10^{-2}$	$\alpha = \gamma = 10^{-3}$	
MFM-Inv-Ga	1.000	0.994	0.988	0.975	0.962	
	(0.001)	(0.009)	(0.029)	(0.044)	(0.094)	
MFM-Ga	0.960	0.500	0.311	0.089	0.068	
	(0.011)	(0.071)	(0.064)	(0.024)	(0.018)	

Table 8: Posterior probabilities and their standard deviations (shown in parentheses) of the number of components M averaged over 50 Monte Carlo replications. The highest value is bolded.

		MF	FM-Inv-G	a		
	M = 1	M=2	M = 3	M = 4	M = 5	$M \ge 6$
$\alpha = 1$	0.000	0.000	0.927	0.073	0.000	0.000
	(0.000)	(0.000)	(0.243)	(0.243)	(0.000)	(0.000)
$\alpha = 0.2$	0.000	0.012	0.966	0.022	0.000	0.000
	(0.000)	(0.084)	(0.111)	(0.073)	(0.001)	(0.000)
$\alpha = 10^{-1}$	0.000	0.020	0.950	0.027	0.002	0.000
	(0.000)	(0.141)	(0.178)	(0.106)	(0.009)	(0.002)
$\alpha = 10^{-2}$	0.000	0.036	0.936	0.023	0.004	0.001
	(0.000)	(0.179)	(0.184)	(0.035)	(0.008)	(0.004)
$\alpha = 10^{-3}$	0.000	0.028	0.938	0.023	0.007	0.004
	(0.000)	(0.143)	(0.180)	(0.044)	(0.021)	(0.017)
		N	AFM-Ga			
$\gamma = 1$	0.000	0.020	0.930	0.049	0.002	0.000
	(0.000)	(0.139)	(0.135)	(0.028)	(0.002)	(0.000)
$\gamma = 0.2$	0.000	0.010	0.480	0.307	0.133	0.070
	(0.000)	(0.069)	(0.085)	(0.036)	(0.035)	(0.038)
$\gamma = 10^{-1}$	0.000	0.035	0.300	0.288	0.186	0.191
	(0.000)	(0.104)	(0.059)	(0.042)	(0.040)	(0.070)
$\gamma = 10^{-2}$	0.000	0.010	0.091	0.156	0.175	0.566
	(0.000)	(0.041)	(0.036)	(0.014)	(0.015)	(0.092)
$\gamma = 10^{-3}$	0.000	0.006	0.069	0.131	0.160	0.634
	(0.000)	(0.028)	(0.023)	(0.017)	(0.012)	(0.085)

Table 9: Comparison of CPU times in seconds (standard deviations are shown in parentheses) averaged over 50 replications.

	MFM-Inv-Ga	MFM-Geng
$\alpha = \gamma = 1$	38.793	87.519
	(2.894)	(3.719)
$\alpha = \gamma = 10^{-3}$	38.361	83.423
	(4.431)	(4.673)

indicates that MFM-Inv-Ga is on average more than twice as efficient as MFM-Geng in terms of computation time. The reason is why our algorithm does not update the label variable based on the restaurant process and MFM-Inv-Ga has a structure that is unlikely to produce empty components.

The MFM-Geng can estimate the number of clusters k, but cannot directly estimate the number of communities M. Furthermore, it is difficult in the MFM-Geng to estimate the hyperparameter of q_M , because it is necessary to perform complex series calculations, including its parameter. In summary, MFM-Inv-Ga (and MFM-Ga) are superior to MFM-Geng in terms of computation time, direct estimation of M, and estimation of an essential parameter.

4.3.2 Dolphins social network data

The dolphins social network data is often used as a benchmark. Data can be obtained in http://www-personal.umich.edu/mejn/netdata/. The data is constructed as an undirected graph and expresses a small-scale animal social network with 64 bottlenose dolphins off Doubtful Sound, New Zealand. Each node represents a dolphin, and an edge is drawn if two dolphins appear to be closely related to each other. In previous studies, it is well-known that the network has two communities. The details of the data are found in Lusseau et al. (2003).

As before, we compare the posterior distribution of the number of communities between MFM-Inv-Ga and MFM-Ga, and create a co-clustering matrix of MFM-Inv-Ga as quantifying uncertainty of clustering. In this analysis, we set $M-1 \sim \text{Poisson}(1)$ and

Table 10: Posterior probabilities of the number of communities M for dolphins social network data

	MFM-Inv-Ga							
	M = 1	M=2	M = 3	M=4	M = 5	$M \ge 6$		
$\alpha = 1$	0.000	0.997	0.003	0.000	0.000	0.000		
$\alpha = 0.2$	0.000	0.972	0.025	0.002	0.001	0.000		
$\alpha = 10^{-1}$	0.000	0.961	0.034	0.004	0.000	0.000		
$\alpha = 10^{-2}$	0.000	0.941	0.048	0.010	0.001	0.000		
$\alpha = 10^{-3}$	0.000	0.854	0.136	0.009	0.002	0.000		
		\mathcal{N}	IFM-Ga					
$\gamma = 1$	0.000	0.949	0.049	0.002	0.000	0.000		
$\gamma = 0.2$	0.000	0.605	0.307	0.074	0.012	0.001		
$\gamma = 10^{-1}$	0.000	0.514	0.340	0.117	0.025	0.004		
$\gamma = 10^{-2}$	0.000	0.619	0.303	0.069	0.000	0.000		
$\gamma = 10^{-3}$	0.370	0.400	0.172	0.048	0.007	0.001		

 $a_Q = b_Q = 3.0$. From Table 10, the MFM-Inv-Ga is able to successfully estimate the number of communities M = 2 regardless of the values of the shape parameter, while the MFM-Ga is not. We report a co-cluster matrix of MFM-Inv-Ga for $\alpha = 1.0$ and MFM-Geng for $\gamma = 1.0$, and the results are shown in Figure 5. The result of MFM-Inv-Ga is almost identical to the results reported in Geng et al. (2019). Furthermore, we confirmed that changing the value of α does not change the co-cluster matrices and the clustering solution based on MAP estimation. This shows the efficiency of the proposed method.

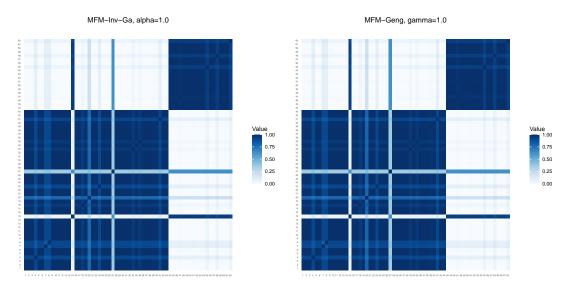


Figure 5: Co-clustering matrix with the MFM-Inv-Ga with $\alpha = 1.0$ (left) and the MFM-Geng with $\gamma = 1.0$ (right) for dolphins social network data.

5 Concluding remarks

We proposed a mixture of finite mixtures (MFM) model based on the normalized inverse Gaussian distribution and constructed an efficient posterior sampling algorithm based on Argiento and De Iorio (2022). The proposed method is a finite analog of the inverse Gaussian processes proposed by Lijoi et al. (2005). We illustrate the performance of the proposed method for clustering, density estimation, and community detection, compared to existing MFM models based on Dirichlet distribution (e.g., Miller and Harrison, 2018; Geng et al., 2019; Argiento and De Iorio, 2022). The proposed method is robust against the choice of hyper-parameter α , and provided reasonable estimates of the number of components and communities compared to the MFM based on the Dirichlet prior distribution. Moreover, the proposed method also has a reasonable predictive performance in the sense of density estimation by suppressing the appearance of empty components.

The drawbacks of the proposed method are as follows. Some parameters involved in the model do not have closed-form marginal distributions because it is not easy to marginalize out U_n . For example, when we focus on clustering, obtaining the interpretable prior distribution for the number of clusters is very important (see e.g., Zito et al., 2023) to incorporate subjective prior information. However, the proposed model cannot lead to a tractable marginal prior distribution for the number of clusters. In the mixture of finite mixtures, the model consists of a distribution over a simplex based on the normalization of independent random variables. Therefore, the correlation between categories cannot be properly modeled. The normalized inverse Gaussian distribution has a negative covariance as well as the Dirichlet distribution. It may be inappropriate for data with positive correlations between categories, such as the proportion of symbiotic organisms present, disease complication data, or gene expression data. To address such a problem, it may be necessary to construct the MFM in a more general framework that removes the assumption of independence in the Norm-IFPP by Argiento and De Iorio (2022). The construction of MFM models using a more flexible distribution over a simplex that can also express positive correlations such as the generalized Dirichlet distribution (Wong, 1998) is an interesting future topic. Furthermore, the proposed model can be applied to spatial data

(e.g., Geng et al., 2021) and functional data (e.g., Hu et al., 2023). For network data, it is expected to extend the MFM to network with weighted edges, degree-corrected stochastic block models, and mixed membership stochastic block models.

Acknowledgement

This work was supported by Japan Society for the Promotion of Science, the establishment of university fellowships towards the creation of science technology innovation Grant Number JPMJFS2129. This work is partially supported by the Japan Society for the Promotion of Science (grant number: 21K13835).

References

- Argiento, R. and M. De Iorio (2022). Is infinity that far? a bayesian nonparametric perspective of finite mixture models. *The Annals of Statistics* 50(5), 2641–2663.
- Escobar, M. D. and M. West (1995). Bayesian density estimation and inference using mixtures. *Journal of the American Statistical Association* 90(430), 577–588.
- Frühwirth-Schnatter, S. (2006). Finite mixture and Markov switching models. Springer.
- Frühwirth-Schnatter, S., G. Celeux, and C. P. Robert (2019). *Handbook of mixture analysis*. CRC press.
- Frühwirth-Schnatter, S. and S. Kaufmann (2008). Model-based clustering of multiple time series. *Journal of Business & Economic Statistics* 26(1), 78–89.
- Frühwirth-Schnatter, S., G. Malsiner-Walli, and B. Grün (2021). Generalized mixtures of finite mixtures and telescoping sampling. *Bayesian Analysis* 16(4), 1279–1307.
- Frühwirth-Schnatter, S. and S. Pyne (2010). Bayesian inference for finite mixtures of univariate and multivariate skew-normal and skew-t distributions. *Biostatistics* 11(2), 317–336.

- Geng, J., A. Bhattacharya, and D. Pati (2019). Probabilistic community detection with unknown number of communities. *Journal of the American Statistical Associ*ation 114(526), 893–905.
- Geng, J., W. Shi, and G. Hu (2021). Bayesian nonparametric nonhomogeneous poisson process with applications to usgs earthquake data. *Spatial Statistics* 41, 100495.
- Ghosal, S. and A. W. van der Vaart (2017). Fundamentals of nonparametric Bayesian inference, Volume 44. Cambridge University Press.
- Green, P. J. (1995). Reversible jump markov chain monte carlo computation and bayesian model determination. *Biometrika* 82(4), 711–732.
- Handcock, M. S., A. E. Raftery, and J. M. Tantrum (2007). Model-based clustering for social networks. Journal of the Royal Statistical Society Series A: Statistics in Society 170(2), 301–354.
- Henze, N. (1986). A probabilistic representation of the skew-normal distribution. Scandinavian Journal of Statistics 13(4), 271–275.
- Hu, G., J. Geng, Y. Xue, and H. Sang (2023). Bayesian spatial homogeneity pursuit of functional data: an application to the us income distribution. *Bayesian Analysis* 18(2), 579–605.
- James, L. F., A. Lijoi, and I. Prünster (2009). Posterior analysis for normalized random measures with independent increments. *Scandinavian Journal of Statistics* 36(1), 76–97.
- Lijoi, A., R. H. Mena, and I. Prünster (2005). Hierarchical mixture modeling with normalized inverse-gaussian priors. *Journal of the American Statistical Association* 100 (472), 1278–1291.
- Lusseau, D., K. Schneider, O. J. Boisseau, P. Haase, E. Slooten, and S. M. Dawson (2003).

 The bottlenose dolphin community of doubtful sound features a large proportion of

- long-lasting associations: can geographic isolation explain this unique trait? Behavioral Ecology and Sociobiology 54, 396–405.
- Malsiner-Walli, G., S. Frühwirth-Schnatter, and B. Grün (2016). Model-based clustering based on sparse finite gaussian mixtures. *Statistics and Computing* 26(1), 303–324.
- McLachlan, G. (2000). Finite mixture models. A wiley-interscience publication.
- McLachlan, G. J., R. W. Bean, and D. Peel (2002). A mixture model-based approach to the clustering of microarray expression data. *Bioinformatics* 18(3), 413–422.
- McLachlan, G. J., S. X. Lee, and S. I. Rathnayake (2019). Finite mixture models. Annual review of Statistics and its Application 6(1), 355–378.
- Miller, J. W. and M. T. Harrison (2018). Mixture models with a prior on the number of components. *Journal of the American Statistical Association* 113(521), 340–356.
- Newman, M. E. (2004). Detecting community structure in networks. *The European Physical Journal B* 38, 321–330.
- Nobile, A. (1994). Bayesian analysis of finite mixture distributions. Carnegie Mellon University.
- Nobile, A. and A. T. Fearnside (2007). Bayesian finite mixtures with an unknown number of components: The allocation sampler. *Statistics and Computing* 17, 147–162.
- Nowicki, K. and T. A. B. Snijders (2001). Estimation and prediction for stochastic block-structures. *Journal of the American statistical association* 96 (455), 1077–1087.
- Richardson, S. and P. J. Green (1997). On bayesian analysis of mixtures with an unknown number of components (with discussion). *Journal of the Royal Statistical Society Series*B: Statistical Methodology 59(4), 731–792.
- Roeder, K. (1990). Density estimation with confidence sets exemplified by superclusters and voids in the galaxies. *Journal of the American Statistical Association* 85(411), 617–624.

- Rousseau, J. and K. Mengersen (2011). Asymptotic behaviour of the posterior distribution in overfitted mixture models. *Journal of the Royal Statistical Society Series B:*Statistical Methodology 73(5), 689–710.
- Shi, J. and J. Malik (2000). Normalized cuts and image segmentation. *IEEE Transactions* on Pattern Analysis and Machine Intelligence 22(8), 888–905.
- White, S. and P. Smyth (2005). A spectral clustering approach to finding communities in graphs. In *Proceedings of the 2005 SIAM international conference on data mining*, pp. 274–285. SIAM.
- Wong, T.-T. (1998). Generalized dirichlet distribution in bayesian analysis. Applied Mathematics and Computation 97(2-3), 165–181.
- Zito, A., T. Rigon, and D. B. Dunson (2023). Bayesian nonparametric modeling of latent partitions via stirling-gamma priors. arXiv preprint arXiv:2306.02360.