

Basturk, Nalan et al.

Working Paper

BayesMultiMode: Bayesian Mode Inference in R

Tinbergen Institute Discussion Paper, No. TI 2023-041/III

Provided in Cooperation with:

Tinbergen Institute, Amsterdam and Rotterdam

Suggested Citation: Basturk, Nalan et al. (2023) : BayesMultiMode: Bayesian Mode Inference in R, Tinbergen Institute Discussion Paper, No. TI 2023-041/III, Tinbergen Institute, Amsterdam and Rotterdam

This Version is available at:

<https://hdl.handle.net/10419/273852>

Standard-Nutzungsbedingungen:

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

Terms of use:

Documents in EconStor may be saved and copied for your personal and scholarly purposes.

You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.

If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.

TI 2023-041/III
Tinbergen Institute Discussion Paper

BayesMultiMode: Bayesian Mode Inference in R

Nalan Basturk¹

Jamie Cross²

Peter de Knijff³

Lennart Hoogerheide⁴

Paul Labonne⁵

Herman K van Dijk⁶

¹ University of Maastricht

² Melbourne Business School

³ Leiden University, Medical Centre

⁴ VU University Amsterdam and Tinbergen Institute

⁵ BI Norwegian Business School

⁶ Erasmus University, Norges Bank and Tinbergen Institute

Tinbergen Institute is the graduate school and research institute in economics of Erasmus University Rotterdam, the University of Amsterdam and Vrije Universiteit Amsterdam.

Contact: discussionpapers@tinbergen.nl

More TI discussion papers can be downloaded at <https://www.tinbergen.nl>

Tinbergen Institute has two locations:

Tinbergen Institute Amsterdam
Gustav Mahlerplein 117
1082 MS Amsterdam
The Netherlands
Tel.: +31(0)20 598 4580

Tinbergen Institute Rotterdam
Burg. Oudlaan 50
3062 PA Rotterdam
The Netherlands
Tel.: +31(0)10 408 8900

BayesMultiMode: Bayesian Mode Inference in R

Nalan Bastürk
Maastricht University

Jamie Cross
Melbourne Business School
University of Melbourne

Peter de Knijff
Leiden University
Medical Centre

Lennart Hoogerheide
Vrije Universiteit Amsterdam
and Tinbergen Institute

Paul Labonne
BI Norwegian
Business School

Herman K. van Dijk
Erasmus University Rotterdam,
Norges Bank and
Tinbergen Institute

Abstract

Multimodal empirical distributions arise in many fields like Astrophysics, Bioinformatics, Climatology and Economics due to the heterogeneity of the underlying populations. Mixture processes are a popular tool for accurate approximation of such distributions and implied mode detection. Using Bayesian mixture models and methods, **BayesMultiMode** estimates posterior probabilities of the number of modes, their locations and uncertainty, yielding a powerful tool for mode inference. The approach works in two stages. First, a flexible mixture with an unknown number of components is estimated using a Bayesian MCMC method due to [Malsiner-Walli, Frühwirth-Schnatter, and Grün \(2016\)](#). Second, suitable detection algorithms are employed to estimate modes for continuous and discrete probability distributions. Given these mode estimates, posterior probabilities for the number of modes, their locations and uncertainties are constructed. **BayesMultiMode** supports a range of mixture processes, complementing and extending existing software for mixture modeling. The mode detection algorithms implemented in **BayesMultiMode** also support MCMC draws for mixture estimation generated with external software. The package uses for illustrative purposes both continuous and discrete empirical distributions from the four listed fields yielding credible multiple mode detection with substantial posterior probability where frequentist tests fail to reject the null hypothesis of unimodality.

Keywords: multimodality, mixture distributions, Bayesian estimation, sparse finite mixtures, R.

1. Introduction

There exist several computer packages for investigating multimodality; see, e.g., the R package **multimode** (Ameijeiras-Alonso, Crujeiras, and Rodriguez-Casal 2021) which implements Silverman’s test and its extensions (Silverman 1981; Fischer, Mammen, and Marron 1994) and the dip test (Hartigan and Hartigan 1985; Hartigan 1985). However, these tests are usually not suitable for discrete data despite their widespread occurrence in areas such as Bioinformatics. Furthermore, most methods that test for the existence of multiple modes do not provide any information about the number of modes, their locations and uncertainty about these estimates. **BayesMultiMode** combines state-of-the-art methods in Bayesian statistics to provide a novel tool for mode inference in both continuous and discrete empirical distributions. We illustrate that **BayesMultiMode** is capable of detecting multiple modes in situations where frequentist methods fail to reject unimodality.

The Bayesian mode inference approach implemented in **BayesMultiMode** works in two stages. First, a mixture model is fitted on the data of the empirical distribution using the sparse finite mixture Markov chain Monte Carlo (SFM MCMC) method due to Malsiner-Walli *et al.* (2016). The SFM MCMC algorithm can be viewed as a two-step process. In step one, the researcher starts by overfitting the mixture model by selecting a larger number of components than is expected to describe the data. In step two, a regularization prior is used which, in combination with likelihood and data information, shrinks the effective number of mixture components towards a credible number with substantial posterior probability. This method of selecting the number of components is particularly appealing because it overcomes well-known problems associated with using marginal likelihoods for component selection, see Frühwirth-Schnatter (2006), and it avoids the computational burden of having to estimate infinite mixture models using non-parametric approaches (Frühwirth-Schnatter and Malsiner-Walli 2019). This approach has the substantial benefit of capturing the uncertainty attached to the number of mixture components inherently present in mixture models.

Once the mixture model is estimated, the second stage consists of detecting the modes. The fixed-point algorithm of Carreira-Perpinan (2000) is used for Gaussian mixtures whereas modes in other continuous mixtures are found with the modal EM (MEM) algorithm of Li, Ray, and Lindsay (2007). Finding modes in discrete mixtures is more straightforward and we follow the recent work of Cross, Hoogerheide, Labonne, and van Dijk (2023) in that instance. Given the estimated modes, posterior probabilities are derived for the number of modes, their locations and associated uncertainty. Thus **BayesMultiMode** provides a comprehensive characterization of the uncertainty surrounding mode inference. Communicating uncertainty effectively contributes to better understanding of science (Van Der Bles, Van Der Linden, Freeman, Mitchell, Galvao, Zaval, and Spiegelhalter 2019) and it is a critical factor in analyzing implied uncertainty properties of forecasting and policy.

Mixture models can be estimated using frequentist or Bayesian methods. When using maximum likelihood, the expectation-maximization (EM) algorithm (see Dempster, Laird, and Rubin 1977) is typically employed because maximizing the likelihood function directly is often difficult. A Bayesian approach using MCMC offers three substantial advantages compared to the EM algorithm. First, computing standard errors using the EM algorithm is non-trivial, whereas estimation uncertainty is readily available from MCMC output. Second, while the exact number of mixture components has to be specified *a priori* using the EM algorithm, the Bayesian framework allows for an unknown number of mixture components. This leads

to the third important advantage of using MCMC methods: their ability to easily quantify the uncertainty associated with the estimated number of components and their locations. **BayesMultiMode** is the first R package to implement the SFM MCMC algorithm for estimating mixtures with an unknown number of components across a comprehensive range of mixture processes for approximating empirical distributions of continuous as well as discrete data. Specifically, we use the Normal and skew-Normal distributions for continuous data and Poisson and shifted-Poisson distributions for discrete data. We emphasize that the mode inference tools provided in the package are also designed to be compatible with external software implementations of Bayesian MCMC mixture modeling, e.g., JAGS, Stan and WinBUGS. Thus the user can make use of the mode inference functions implemented in **BayesMultiMode** even if the mixture is not estimated with the package.

The next section discusses related software. Section 3 sets out the mixture models and priors available in **BayesMultiMode**. Section 4 explains the SFM MCMC and mode detection algorithms. Section 5 provides a brief tutorial on the use of **BayesMultiMode**. Section 6 illustrates the package with continuous data distributions from Astrophysics and Economics and discrete data distributions from Bioinformatics. The results yield credible multiple mode detection with substantial posterior probability where frequentist tests fail to reject the null hypothesis of unimodality. An example describes in a stepwise manner the mode-inference capabilities of **BayesMultiMode** when working with an external MCMC output using continuous data from Climatology.

We end with a remark on notation. Formally, we deal with empirical distributions of observed data (continuous and discrete), probability distributions, probability density functions and probability mass functions (continuous and discrete) in our models. For convenience, we use the short terminology of, respectively, data, distributions, densities and mass functions, when no ambiguity arises.

2. Related software implementations

BayesMultiMode provides tools for both mixture modeling and mode estimation which yield powerful inferential information. As such **BayesMultiMode** contributes to three statistical topics: multimodality tests, mixture modeling and mode estimation. This section discusses how **BayesMultiMode** compares with other open-source software implementing similar methods, in particular, R (R Core Team 2021) packages.

2.1. Mixture estimation

A comprehensive overview of the packages available for estimating mixture models in R can be found in the section on model-based clustering of the CRAN Task View "Cluster Analysis & Finite Mixture Models" (Leisch and Gruen 2023). Many of these packages estimate the model via the EM algorithm. Since **BayesMultiMode** makes use of MCMC, we limit our discussion here to packages that adopt Bayesian estimation methods. Packages for MCMC estimation of mixture models in R are relatively scarce. Among them, **BayesMix** (Gruen and Plummer 2021) provides an interface to finite mixture models coded in JAGS, and **BNPmix** (Corradin, Canale, and Nipoti 2021) provides a very fast and flexible implementation of non-parametric Bayesian mixtures. Besides R, custom mixtures can also be estimated with MCMC techniques using Stan (Carpenter, Gelman, Hoffman, Lee, Goodrich, Betancourt, Brubaker,

Guo, Li, and Riddell 2017), BUGS (Lunn, Thomas, Best, and Spiegelhalter 2000) and JAGS (Plummer 2003). Researchers working with Python can also use the popular **PyMC** package (Wiecki, Salvatier, Vieira, Kochurov, Patil, Osthege, Willard, Engels, Carroll, Martin, Seyboldt, Rochford, Paz, rpgoldman, Meyer, Coyle, Gorelli, Abril-Pla, Kumar, Lao, Andreani, Yoshioka, Ho, Kluyver, Beauchamp, Andorra, Pananos, Spaak, Edwards, and Ma 2023). All of these packages can be used in conjunction with **BayesMultiMode**. The only requirement for running **BayesMultiMode** is a suitably defined MCMC output (details are provided in Subsection 5.2).

2.2. Mode inference

To the best of our knowledge **BayesMultiMode** is the first R package that is capable of mode inference in mixture models using Bayesian methods. Specifically, **BayesMultiMode** implements two efficient algorithms for estimating modes in mixtures of univariate continuous distributions: the fixed-point algorithm of Carreira-Perpinan (2000) for Normal mixtures and the Modal EM (MEM) algorithm of Li *et al.* (2007) for general continuous mixtures. For the case of discrete mixtures we implement a simple and efficient algorithm proposed in Cross *et al.* (2023).

If a mixture model is not needed to characterize the data, then other packages for mode inference in R include: **ModEstM** (Collet 2022) which provides a function to find modes in univariate distributions using a kernel approach following Eddy (1980); **multimode** which implements the mode tree (Minnotte and Scott 1993), mode forest (Minnotte, Marchette, and Wegman 1998) and SiZer map Chaudhuri and Marron (1999) for visualizing modes in kernel density estimates; and **modehunt** implements the multiscale inference of Dümbgen and Walther (2008) which relies on local order statistics instead of mixtures or kernel methods. Unlike the Bayesian approach implemented in **BayesMultiMode**, these methods yield limited information regarding the uncertainty attached to mode estimates.

Conventional frequentist tests for multimodality (Silverman 1981; Hall and York 2001; Hartigan and Hartigan 1985; Cheng and Hall 1998; Fisher and Marron 2001) can all be accessed with **multimode**. Unlike these tests, however, **BayesMultiMode** provides posterior probabilities for the number of modes given a family of mixture distributions.

3. Mixture models and priors

We provide a brief introduction to mixture models intended to provide users with sufficient knowledge to use the package on their own datasets. More general treatments can be found in the textbooks of McLachlan and Peel (2004) and Frühwirth-Schnatter (2006), among others.

Let y_i , $i = 1, \dots, n$, denote random variables that can be either discrete valued, i.e., $y_i \in \mathbf{N}$, or continuous valued, i.e., $y_i \in \mathbf{R}$. A mixture of K distributions from the same parametric family $P(\cdot|\theta_k)$, continuous or discrete, is given by:

$$y_i \sim \sum_{k=1}^K \pi_k P(\cdot|\theta_k), \quad (1)$$

where the mixture component probabilities are constrained such that $\sum_{k=1}^K \pi_k = 1$ and $\pi_k \geq 0$, $k = 1, \dots, K$, and θ_k denotes the parameter vector belonging to the k -th component.

BayesMultiMode estimates the unknown parameters: π_k and θ_k using Bayesian MCMC methods. To that end, a symmetric Dirichlet prior is used for the mixture weights:

$$\pi_k \sim \text{Dirichlet}(e_0, \dots, e_0). \quad (2)$$

The concentration parameter e_0 has traditionally been viewed as a hyperparameter that is selected by the researcher. In a recent article, [Malsiner-Walli, Frühwirth-Schnatter, and Grün \(2016\)](#) show that estimating this parameter allows for simple and efficient inference on the number of mixture components. They recommend to overfit the mixture distribution by selecting a large value of K , and specifying a regularization prior on e_0 to shrink the effective number of mixture components to a credible number with large posterior probability. This is accomplished through a Gamma hyperprior on the concentration parameter e_0 of the form

$$e_0 \sim \text{Gamma}(a_0, A_0), \quad (3)$$

in which the hyperparameters control the unconditional moments of e_0 , e.g., the unconditional mean is $\frac{a_0}{A_0}$ and the unconditional variance is $\frac{a_0}{A_0^2}$. A theoretical justification for this approach stems from asymptotic results in [Rousseau and Mengersen \(2011\)](#), and a comparison of the approach with the popular Dirichlet process prior is given by [Frühwirth-Schnatter and Malsiner-Walli \(2019\)](#). Following [Frühwirth-Schnatter and Malsiner-Walli \(2019, p. 48\)](#) default values of the hyperparameters are selected to be $a_0 = 1$, $A_0 = 200$.

BayesMultiMode supports four parametric mixture distributions: Normal and skew-Normal continuous mixtures, Poisson and shifted-Poisson discrete mixtures. We briefly cover each of these in turn.

Normal model A component of a mixture of Normal densities takes the form:

$$p(y_i | \mu_k, \sigma_k) = \frac{1}{\sqrt{2\pi} \sigma_k} \exp\left(-\frac{1}{2} \left(\frac{y_i - \mu_k}{\sigma_k}\right)^2\right), \quad (4)$$

where μ_k is a location parameter and σ_k^2 a scale parameter. We use independent conjugate priors for μ_k and σ_k^2 (see for instance [Malsiner-Walli *et al.* 2016](#); [Richardson and Green 1997](#)):

$$\mu_k \sim \text{Normal}(b_0, B_0), \quad (5)$$

$$\sigma_k^{-2} \sim \text{Gamma}(c_0, C_0), \quad (6)$$

$$C_0 \sim \text{Gamma}(g_0, G_0). \quad (7)$$

The default values are set to be $b_0 = \text{mean}(y)$, $B_0 = \text{range}(y)^2$, where $\text{range}(y) = \max(y) - \min(y)$, $c_0 = 2.5$, $g_0 = 0.5$, and $G_0 = 100g_0/c_0/R^2$. These values are common in applications based on Normal mixtures, e.g., [Frühwirth-Schnatter \(2006, p. 192\)](#) and [Malsiner-Walli *et al.* \(2016, eq. 5\)](#).

Skew-Normal model We use the skew-Normal density of [Azzalini \(1985\)](#) (see also [Azzalini 2013](#)) which takes the form:

$$p(y_i | \xi_k, \omega_k, \alpha_k) = \frac{1}{\omega_k \sqrt{2\pi}} \exp\left(-\frac{1}{2} \left(\frac{y_i - \xi_k}{\omega_k}\right)^2\right) \left(1 + \text{erf}\left(\alpha_k \left(\frac{y_i - \xi_k}{\omega_k \sqrt{2}}\right)\right)\right), \quad (8)$$

where ξ_k is a location parameter, ω_k a scale parameter, and α_k the shape parameter introducing skewness. To estimate the model we follow [Frühwirth-Schnatter and Pyne \(2010\)](#) and use the following reparameterized random-effects model:

$$z_i \sim TN_{[0,\infty)}(0, 1), \quad (9)$$

$$y_i | (S_i = k) = \xi_k + \psi_k z_i + \epsilon_i, \quad \epsilon_i \sim N(0, \sigma_k^2), \quad (10)$$

where S_i indicates from which component the observation i stems and where the parameters of the skew-Normal density in (8) are recovered as

$$\omega_k = \frac{\psi_k}{\sigma_k}, \quad \omega_k^2 = \sigma_k^2 + \psi_k^2. \quad (11)$$

By defining a vector of regressors $x_i = (1, z_i)'$ with the representation (9), the skew-Normal mixture can be seen as a random-effects model with Normal errors. Its parameters can be sampled from their posterior using standard techniques for Normal mixtures as outlined above. Therefore we specify similar priors for the Normal mixture model:

$$(\xi_k, \psi_k)' \sim \text{Normal}(\mathbf{b}_0, \mathbf{B}_0) \quad (12)$$

$$\sigma_k^{-2} \sim \text{Gamma}(c_0, C_0) \quad (13)$$

$$C_0 \sim \text{Gamma}(g_0, G_0). \quad (14)$$

The only difference in the default choice of hyperparameters is that we set $\mathbf{b}_0 = (\text{median}(y), 0)'$ and $\mathbf{B}_0 = \text{diag}(2)$.

Poisson model A component of a mixture of Poisson mass functions takes the form:

$$p(y_i | \lambda_k) = \frac{1}{y_i!} \lambda_k^{y_i} \exp(-\lambda_k), \quad (15)$$

where

$$\lambda_k \sim \text{Gamma}(l_0, L_0). \quad (16)$$

Following [Viallefont, Richardson, and Green \(2002\)](#), the default shape parameter l_0 is selected to be just above one so as to avoid the exponential shape of the Gamma density while preserving a reasonable amount of variation, while the scale parameter, L_0 , is set such that the prior mean is equal to the arithmetic mean of the observed data ($\mathbb{E}[\lambda_k] = l_0 L_0 = \text{mean}(y)$).

Shifted-Poisson model The shifted-Poisson mass function augments the standard Poisson with a shift (or location) parameter such that:

$$p(y_i | \lambda_k, \kappa_k) = \frac{1}{(y_i - \kappa_k)!} \lambda_k^{(y_i - \kappa_k)} \exp(-\lambda_k). \quad (17)$$

As discussed in [Cross *et al.* \(2023\)](#), there are two key advantages offered by the shifted-Poisson over the regular Poisson. First, it is useful for non-zero count data. Second, the variance of a shifted-Poisson component can be different from its mean which brings additional flexibility to the model. Following [Cross *et al.* \(2023\)](#), we specify a non-informative discrete uniform prior

Parameter	Mixture	Description	Default
a_0	All	Shape parameter of the gamma hyperprior applied to the concentration parameter e_0 of the Dirichlet prior.	1
A_0	All	Scale parameter of the gamma hyperprior applied to the concentration parameter e_0 of the Dirichlet prior.	200
b_0	Normal skew-Normal	Mean parameter of the normal prior applied to the mean parameters μ_k .	median(data) (median(data), 0)'
B_0	Normal skew-Normal	Variance parameter of the normal prior applied to the mean parameters μ_k .	range(y) ² diag(2)
c_0	Normal skew-Normal	Shape parameter of the inverse gamma prior applied to the variance parameters σ_k^2 .	2.5 2.5
C_0	Normal skew-Normal	Scale parameter of the inverse gamma prior applied to the variance parameters σ_k^2 .	Gamma hyperprior
g_0	Normal skew-Normal	Shape parameter of the gamma hyperprior applied to C_0 .	0.5
G_0	Normal skew-Normal	Scale parameter of the gamma hyperprior applied to C_0 .	$100g_0 \sqrt{c_0 \text{range}(y)^2}$ $g_0 / (0.5 \times \text{var}(y))$
l_0	Poisson shifted-Poisson	Shape parameter of the gamma prior applied to λ_k .	1.1 5
L_0	Poisson shifted-Poisson	Scale parameter of the gamma prior applied to λ_k .	1.1/median(y) $1 - l_0$

Table 1: Summary of the prior parameters implemented in BayesMultiMode.

for the shift parameter κ_k , i.e. $U(0, \max(y))$, where a different large value (not depending on the data y) could also be used instead of $\max(y)$.

A summary of the default priors set in **BayesMultiMode** is provided in Table 1.

4. Algorithms

This section discusses the algorithms for mixture estimation and mode detection implemented in **BayesMultiMode**.

4.1. SFM MCMC

Numerous MCMC methods exist for the estimation of the model parameters and number of components of a mixture process. To the best of our knowledge, there does not exist a theorem that shows which MCMC method is uniformly superior in terms of computational efficiency and robustness for the simulation from posterior distributions of mixture processes. That is, model specification, prior information and data information content play an important role in choosing which MCMC method is most suitable in a certain case. **BayesMultiMode** implements the sparse finite mixture MCMC (SFM MCMC) algorithm, see [Malsiner-Walli et al. \(2016\)](#) and [Frühwirth-Schnatter and Malsiner-Walli \(2019\)](#). A major advantage of this

procedure is its simplicity and ease of implementation. It consists of initially overfitting the mixture model by selecting a larger number of components than is expected to describe the data and then using a sparse hierarchical prior on the mixture weights to shrink the model space to a credible number of components with substantial posterior probability. The authors show that the SFM approach is more reliable than Dirichlet process mixtures (DPM) which are known to overfit the number of components (Miller and Harrison 2013) and that the SFM approach is conceptually simpler and substantially easier to implement than the Reversible Jump MCMC (see Green 1995; Richardson and Green 1997). For background we refer to Malsiner-Walli *et al.* (2016) and Frühwirth-Schnatter and Malsiner-Walli (2019).

Label-switching It is notoriously difficult to identify individual components when fitting mixtures with MCMC methods; a problem referred to as label-switching. Label-switching occurs when the likelihood is invariant to permutation of the mixture components, which is often the case when the data are not very informative. This is, however, not an issue for mode inference. Estimation of the modes only requires estimates of the mixture distribution at each MCMC draw, regardless of the components' order. Therefore, while label-switching can be a feature of the MCMC results derived with **BayesMultiMode**, this should not concern the user for the case of mode detection.

4.2. Mode estimation

Given equation (1), the density or mass function consisting of a mixture of K densities or mass functions from the same parametric family $p(\cdot|\theta_k)$ is given by:

$$y_i \sim \sum_{k=1}^K \pi_k p(\cdot|\theta_k), \quad (18)$$

We want to find the local maxima of $p(y_i|\theta)$, with $\theta = (\theta_1, \dots, \theta_k)'$. To that end, we implement two algorithms for detecting modes for a mixture of densities and one algorithm for a mixture of mass functions. The reason for using two algorithms for detecting modes for continuous mixtures is that efficient algorithms exist for the case of Normal mixtures that are not easily transferable to cases of non-Normality.

Fixed-point algorithm for Normal mixtures For a mixture of Normal densities the following algorithm from Carreira-Perpinan (2000, section 4) is used. A mode y is found by iterating the two steps:

$$(i) \quad p(k|y^{(n)}) = \frac{\pi_k p(y^{(n)}|\theta_k)}{p(y^{(n)})}, \quad (19)$$

$$(ii) \quad y^{(n+1)} = f(y^{(n)}), \quad (20)$$

where

$$p(y) = \sum_k \pi_k p(y|\theta_k), \quad (21)$$

$$f(y) = \left(\sum_k p(k|y) \sigma_k^{-2} \right)^{-1} \sum_k p(k|y) \sigma_k^{-2} \mu_k, \quad (22)$$

until numerical convergence to the fixed point of $f(\cdot)$, which is defined by $\text{abs}(y^{(n+1)} - y^{(n)}) < \text{tol}_{\text{conv}}$, where tol_{conv} is an argument with default value $1e - 8$. Following Carreira-Perpinan (2000) we start the algorithm multiple times, once at each component's location. Separately, it is necessary to identify identical modes which diverge only up to a small value. By default **BayesMultimode** will merge modes which are closer than $\text{sd}(y)/10$; this tolerance value can be controlled with the argument `tol_y`.

The Modal EM (MEM) For continuous mixtures other than the Normal the following algorithm from Li *et al.* (2007) is used. The MEM algorithm consists of iterating the two steps:

$$(i) \quad p(k|y^{(n)}) = \frac{\pi_k p(y^{(n)}|\theta_k)}{p(y^{(n)})}, \quad (23)$$

$$(ii) \quad y^{(n+1)} = \arg \max_y \sum_k p(k|y^{(n)}) \log p(y|\theta_k), \quad (24)$$

until convergence, that is, until $\text{abs}(y^{(n+1)} - y^{(n)}) < \text{tol}_{\text{conv}}$. The maximization is carried out with a limited-memory BFGS algorithm (specifically the L-BFGS-B method with the `optim` function in R). Iterating these two steps is typically more reliable than maximizing (18), or its logarithm, directly. Like with the fixed-point algorithm, the convergence tolerance is set to $1e - 8$ and the MEM algorithm is repeatedly started at each component's location.

While it is also possible to use the MEM algorithm for Normal mixtures, this is not recommended because the algorithm is less efficient than the fixed-point method in this particular case.

Mode-detection in discrete mixtures Detecting modes in discrete mixtures is easier than in continuous mixtures. Here we follow Cross *et al.* (2023) and implement the following simple and efficient method. By definition, modes must satisfy either:

1. $p_k(y_m - 1) < p_k(y_m) > p_k(y_m + 1)$,
2. $p_k(y_m - 1) < p_k(y_m) = p_k(y_m + 1) = \dots = p_k(y_m + l - 1) > p_k(y_m + l)$.

In the first case there is a unique mode which is clearly identified. In the second case in which l consecutive values of the posterior predictive probability mass function are equally high, we consider all values as mode locations (flat modes), but only count the first as a single mode when computing the posterior probability of the number of modes. From a practical perspective, this amounts to a simple five-step procedure:

1. Compute the posterior mixture at a given draw of parameters
2. Evaluate the posterior mass function at each (discrete) observation
3. Take the first difference of the resulting series
4. Find the points where the series of first differences have a non-positive value after a positive value

5. Extract all of the locations of modes of these mass function estimates

4.3. Mode inference

Each draw from the MCMC output after burnin, $\theta^{(d)}$, $d = 1, \dots, D$, leads to a posterior predictive density/mass function:

$$p(y|\theta^{(d)}) = \sum_{k=1}^K \pi_k^{(d)} p(y|\theta_k^{(d)}). \quad (25)$$

Using this function, the modes in draw d , $y_m^{(d)}$, $m = 1, \dots, M^{(d)}$, where $M^{(d)}$ is the number of modes in draw d , are estimated using the algorithms mentioned above.

After running this procedure across all retained posterior draws, we compute the posterior probability for the number of modes being M as:

$$P(\#\text{modes} = M) = \frac{1}{D} \sum_{d=1}^D 1(M^{(d)} = M), \quad (26)$$

with $M^{(d)}$ the number of modes in a given draw from MCMC estimation (after burnin), $d = 1, \dots, D$. Similarly, posterior probabilities for mode locations are:

$$P(y = \text{mode}) = \frac{1}{D} \sum_{d=1}^D 1(y = y_m^{(d)}), \quad (27)$$

for each local mode $y_m^{(d)}$ in the range $[\min(y), \max(y)]$. As an example if $y = 0$ is a mode at each draw i.e., $1(y = y_m^{(d)})$, $d = 1, \dots, D$, then $P(0 = \text{mode}) = 1$. Alternatively, if $y = 1$ is detected as a mode at 75% of draws then $P(1 = \text{mode}) = 0.75$.

Computing this posterior probability over the range $[\min(y), \max(y)]$ yields a full characterization of the uncertainty attached to mode locations. Obviously, empirical distributions of continuous variables are not defined on a discrete support; it is therefore necessary to choose a rounding decimal to discretize their support. The computation of the posterior probabilities of mode locations requires a discrete range (or equivalently a width of intervals/bins). A rounding decimal to discretize their support can be chosen by the user in **BayesMultiMode**.

5. Using BayesMultiMode

BayesMultiMode offers a flexible approach for estimating mixtures given its range of supported density/mass functions and its implementation of an SFM MCMC algorithm where the number of mixture components does not have to be known *a priori*. However, the algorithms for mode inference implemented in the package are also available for users who prefer using external software for MCMC estimation. This section details the use of **BayesMultiMode** in both cases.

5.1. Using BayesMultiMode for both estimation and mode inference

The typical workflow using **BayesMultiMode** is:

1. Estimate the mixture model with `bayes_estimation(data, K, dist, priors, nb_iter, burnin)` where the user selects
 - (a) A vector of observations to investigate with the `data` argument;
 - (b) Family of the mixture with the `dist` argument i.e. "normal", "skew_normal", "poisson" or "shifted_poisson";
 - (c) The maximum number of components with the `K` argument;
 - (d) Priors (optional) with the `priors = list()` argument;
 - (e) The number of MCMC draws with the `nb_iter` argument;
 - (f) The number of draws used as burnin with the `burnin` argument;
2. Visualize the mixture alongside the data with `plot()`;
3. Conduct mode inference with `bayes_mode()`;
4. Visualize the mode inference with `plot()`;
5. Summarize the mode inference with `summary()`.

5.2. BayesMultiMode for mode inference with external MCMC output

Alternatively, if the user prefers carrying out MCMC estimation with an external software, steps one and two presented above are skipped. The user needs to convert the external MCMC results into an object of class `BayesMixture` using the function `new_BayesMixture(mcmc, data, K, burnin, dist = "NA", pars_names, pdf_func = NULL, dist_type)` where the arguments are

1. `mcmc`: MCMC output matrix with parameters as columns and MCMC draws as rows;
2. `data`: the data used to estimate the model;
3. `K`: the number of mixture components;
4. `burnin`: the number of MCMC draws to be discarded as burnin;
5. `dist`: the name of the components' distribution, i.e. "normal", "skew_normal", "poisson" or "shifted_poisson";
6. `pars_names`: the names of the parameters featuring in the MCMC output;
7. `pdf_func`: the density or mass function of the components;
8. `dist_type`: the type of distribution, either continuous or discrete.

The argument `dist` should be provided only if it is a distribution supported by `bayes_estimation()`; if it is not supported then this argument should be ignored and the density or mass function should be provided with the argument `pdf_func`. The function given in `pdf_func` should take a vector as first argument and a named vector as second argument, with the names corresponding to the parameters of the function. Importantly these names should match the variable

Function	Description
<code>bayes_estimation()</code>	Estimate a mixture distribution using SFM MCMC methods.
<code>bayes_mode()</code>	Estimate modes and retrieve posterior probabilities for their number and locations.
<code>bayes_trace()</code>	Trace plots of MCMC draws using package bayesplot .
<code>new_BayesMixture()</code>	Take external MCMC results and return an object of class BayesMixture which can then be used with <code>bayes_mode()</code> .
<code>fixed_point()</code>	Estimate the modes of a mixture of Normal distributions using the fixed-point algorithm of Carreira-Perpinan (2000) .
<code>MEM()</code>	Estimate the modes of a mixture of continuous distributions using the Modal EM algorithm of Li <i>et al.</i> (2007) .
<code>discrete_MF()</code>	Find the modes of a mixture of discrete distributions.

Table 2: Overview of the functions available in BayesMultiMode.

names of the MCMC output; see the example in the documentation of the `new_BayesMixture` function.

Table 2 provides a summary of the functions exported in **BayesMultiMode**.

Given that the methods and main functions of **BayesMultiMode** have been described, we now illustrate its use.

6. Case studies

To facilitate the illustration and use of the capabilities of **BayesMultiMode**, we have included the galaxy data used in [Richardson and Green \(1997\)](#) downloaded from <https://people.maths.bris.ac.uk/~mapjg/mixdata> in the package. This is a widely used empirical distribution for illustrating estimating continuous mixtures and mode inference. Next, we make use of data discussing the potential convergence in per capita income across economies, see, e.g., [Bianchi \(1997\)](#), [Henderson, Parmeter, and Russell \(2008\)](#) and [Paap and van Dijk \(1998\)](#). For this application we make use of the average GDP per capita over 10 year intervals, collected from 170 countries, in the Penn World Tables ([Feenstra, Inklaar, and Timmer 2015](#)). We also included a dataset on cyclone lifetime maximum intensity. The galaxy and cyclone data are lazy-loaded and can be retrieved with `galaxy` and `cyclone`.

Finally, we included a DNA set of discrete data to show the benefit of analyzing discrete empirical distributions with **BayesMultiMode**. These data are obtained from 270 unrelated human DNA samples from Asian, African and Caucasian origin, see [Schaap, Lemmers, Maassen, van der Vliet, Hoogerheide, van Dijk, Baştürk, de Knijff, and van der Maarel \(2013\)](#). It is of substantial interest to analyze the number and location of modes in the data, since differences in these values may be linked to genetic diseases. Like for the continuous data, the DNA data are lazy-loaded and can be retrieved with `d4z4` and `ct47`.

6.1. Galaxy data

We use the `galaxy` data to illustrate the use of the package with continuous data. These data, introduced in [Roeder \(1990\)](#), show the velocity at which 82 galaxies in the Corona

Borealis region are moving away from our galaxy. The data are commonly scaled by 1000 and have been analyzed extensively to illustrate mixture methods, see for instance Roeder (1990), Roeder and Wasserman (1997), Richardson and Green (1997) and more recently Grün, Malsiner-Walli, and Frühwirth-Schnatter (2022). Studies analyzing these data typically find evidence about at least three modes (Roeder 1990; Roeder and Wasserman 1997; Richardson and Green 1997; Grün *et al.* 2022).

```
R> library(BayesMultiMode)
R> y = galaxy
```

First, we analyze the data using the frequentist tests for unimodality from Silverman (1981) (SI), Hall and York (2001), Hartigan and Hartigan (1985) (HH or dip test), Cheng and Hall (1998) (CH) and Fisher and Marron (2001) (FM). They can all be accessed with the R package **multimode**. The authors of **multimode** have also introduced a new test (ACR) (see Ameijeiras-Alonso, Crujeiras, and Rodriguez-Casal 2019) which is the default method in their package. We perform all these tests together and show their p-values:

```
R> set.seed(123)
R> library(multimode)
R> tests_mode <- function(y) {
+
+   tests = c(modetest(y, method = "SI")$p.value,
+             modetest(y, method = "HY")$p.value,
+             modetest(y, method = "FM")$p.value,
+             modetest(y, method = "HH")$p.value,
+             modetest(y, method = "CH")$p.value,
+             modetest(y, method = "ACR")$p.value)
+
+   names(tests) = c("SI", "HY", "FM", "HH", "CH", "ACR")
+
+   return(tests)
+ }
R> tests_mode(y)
```

```
      SI    HY    FM    HH    CH    ACR
0.208 0.012 0.000 0.696 0.262 0.188
```

The null hypothesis in all these tests is unimodality and the alternative hypothesis is at least two modes. The only tests rejecting unimodality at a 95% confidence level are the FM and HY tests, the other four tests fail to detect multimodality. This goes against the consensus view that these data include at least three modes. We obtain similar results with other packages, such as **diptest**.

Next we analyze the galaxy data with **BayesMultiMode**. The first step consists of fitting a mixture; here we choose Normal components, which is a common choice for these data (Grün *et al.* 2022). We choose a maximum number of 10 components and run 2000 draws with half of them used as burnin.


```
R> plot(mix_mcmc, max_size = 100, transparency = 0.8)
```

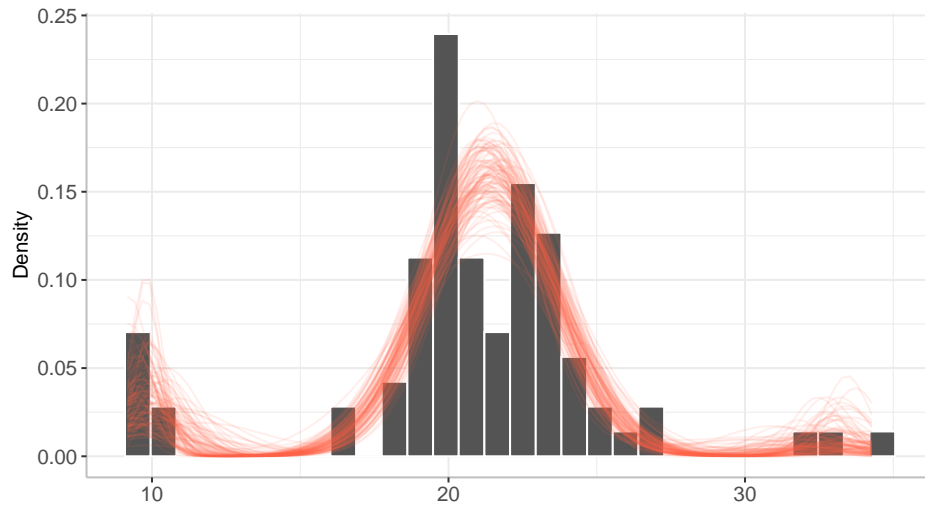


Figure 1: Empirical distribution of galaxy data and mixture estimate.

```
R> mix_mcmc = bayes_estimation(data = y,
+                             K = 10,
+                             dist = "normal",
+                             nb_iter = 2000,
+                             burnin = 1000,
+                             printing = FALSE)
```

The function `bayes_estimation` generates an object of class `BayesMixture` which can be visualized with `plot()`. Figure 1 shows the estimated density for 100 draws alongside the data. We observe estimates of three modes, one on each extremity and one in the center. The `mcmc` element of the `BayesMixture` object can be used as input to most functions in `bayesplot` if diagnostic plots are required.

Now that the mixture model has been estimated, we use the MCMC output for mode inference:

```
R> mode_mcmc = bayes_mode(mix_mcmc)
```

The function `bayes_mode()` returns an object of class `BayesMode` which can be visualized using `plot()` and summarized with `summary()`. Here the fixed-point algorithm of [Carreira-Perpinan \(2000\)](#) is used for mode detection because we have estimated a Gaussian mixture. A summary of the results yields:

```
R> summary(mode_mcmc)
```

The posterior probability of the data being multimodal is 0.989

```
Number of estimated modes and their posterior probabilities:
Number of modes Posterior probability
```

[1,]	1	0.011
[2,]	2	0.152
[3,]	3	0.828
[4,]	4	0.009

The results are clearly in favor, in a credible way, of multimodality, with three modes being the number of modes with the highest posterior probability. Next we investigate the location

```
R> plot(mode_mcmc, max_size = 100, transparency = 0.8)
```

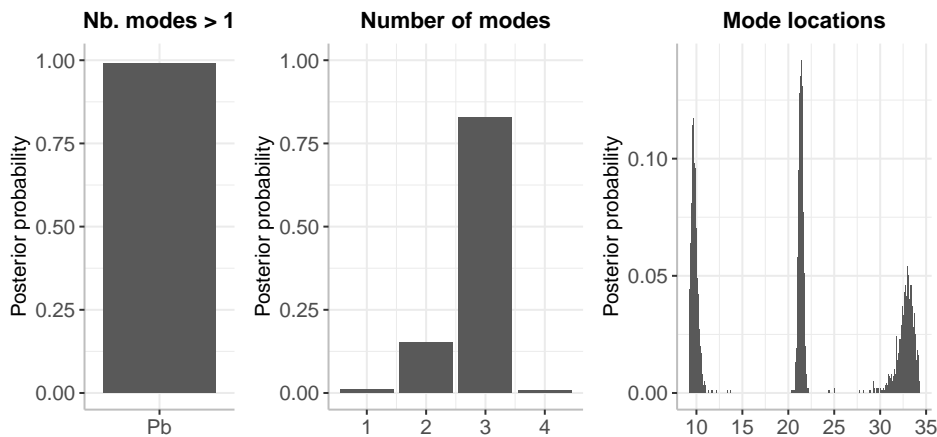


Figure 2: Posterior probabilities for the number of modes and their locations using the galaxy data.

of the modes with the third panel of Figure 2.

BayesMultiMode provides a full characterization of the uncertainty attached to mode locations. We observe, for instance, that the location of the right mode is subject to more uncertainty than the other two modes.

Note that, when working with continuous data, it is necessary to choose a rounding precision to compute the posterior probabilities of mode locations (or equivalently to choose a width of the intervals/bins when displaying histograms of draws or graphs of the estimated probability of mode locations). The default in **BayesMultiMode** is to round the location axis at the first decimal. Figure 3 compares the implication of using different rounding precision. Here we make use of the **graphs** argument of the plot function which allows to select a subset of the three graphs plotted by default. We use **ggplot2** (Wickham 2016) to add features to the plots generated by the `plot()` method from **BayesMultiMode** (which also makes use of **ggplot2**). Combining individual plots can be done with the **ggpubr** package.

It is possible to investigate graphically estimated modes using the argument `show_plot` of `bayes_mode` with argument `nb_iter` determining the number of draws to use. For example Figure 4 shows the estimated mixture at one MCMC iteration together with the estimated modes shown as vertical bars. Choosing different priors than the default priors can affect MCMC estimation and can thus also affect the estimates of the modes. The choice of priors can be seen as similar to choosing the bandwidth parameter in kernel estimation; the estimation technique used in nonparametric tests. But unlike bandwidth parameters, we can rely

```

R> library(ggpubr)
R> library(ggplot2)
R> mode_mcmc_a = bayes_mode(mix_mcmc, rd = 1)
R> mode_mcmc_b = bayes_mode(mix_mcmc, rd = 0)
R> plot_a = plot(mode_mcmc_a, graphs = "loc") +
+   ggtitle("Rounding at the first decimal")
R> plot_b = plot(mode_mcmc_b, graphs = "loc") +
+   ggtitle("Rounding at the unit level")
R> ggarrange(plot_a, plot_b,
+           ncol = 1, nrow = 2, widths = c(0.5, 0.5))

```

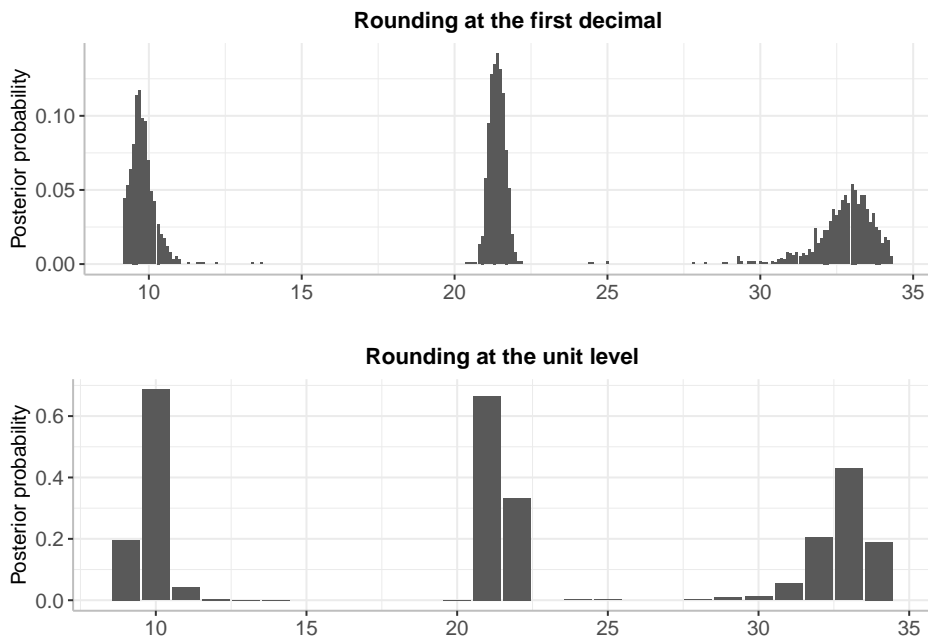


Figure 3: Illustration of mode inference for the galaxy data using different rounding precision.

on extensive research for specifying priors. In the case of the galaxy data, [Grün *et al.* \(2022\)](#) provide a comprehensive discussion on the effect that prior specifications have on the final mixture estimates.

```
R> par(mar = c(2, 2, 2, 2))  
R> mode_mcmc = bayes_mode(mix_mcmc, nb_iter = 1, show_plot = TRUE)
```

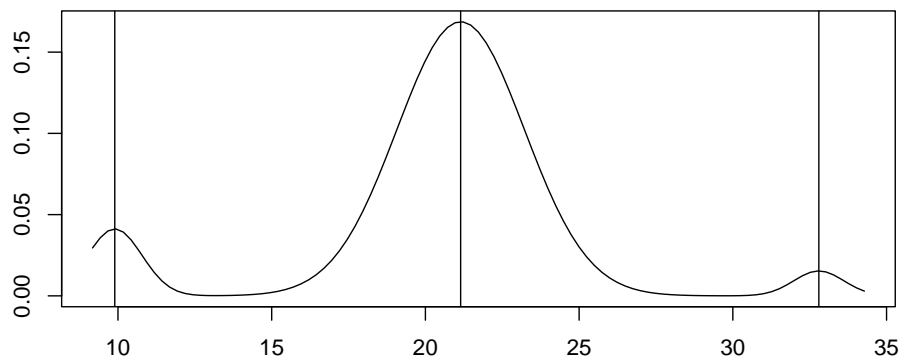


Figure 4: Estimated mixture (curve) and modes (vertical bars) from one draw of the MCMC algorithm for the galaxy data.

6.2. International economic growth data

There exists a rich literature in economics discussing the potential convergence in per capita income across economies. Mode analysis is a useful tool to tackle this highly debated topic. The data used are the average GDP per capita over 10 year intervals, collected from 170 countries, in the Penn World Tables. These empirical distributions are made available in R through the package `pwt10` (Zeileis 2023). We apply a mixture of normal densities to approximate the data and to credibly estimate the number of modes over time.¹

```
R> library(pwt10)
R> library(dplyr)
R> estimation_growth <- function(start, end) {
+   y = pwt10.0 %>%
+     select(year, country, rgdpe, pop) %>%
+     filter(year %in% rep(start:end, 1)) %>%
+     group_by(country) %>%
+     summarise(rgdpe = as.numeric(mean(rgdpe/pop, na.rm=T)/1000)) %>%
+     na.omit() %>%
+     select(rgdpe) %>%
+     unlist()
+
+   y = sort(y, decreasing = T)
+   y = y[-c(1:5)]
+
+   mix = bayes_estimation(y, dist = "normal", K = 10, printing = F)
+   plot_mix = plot(mix, max_size = 100, transparency = 0.1) + ylab(NULL) +
+     ggtitle(paste0(start, "'s"))
+   modes = bayes_mode(mix, rd = 0)
+   tests = tests_mode(y)
+
+   return(list(mix = mix,
+               plot_mix = plot_mix,
+               modes = modes,
+               tests = tests))
+ }
R> res_60s = estimation_growth(1960, 1969)
R> res_70s = estimation_growth(1970, 1979)
R> res_80s = estimation_growth(1980, 1989)
R> res_90s = estimation_growth(1990, 1999)
R> res_00s = estimation_growth(2000, 2009)
R> res_10s = estimation_growth(2010, 2019)
```

The posterior probability of multimodality and the p-values of conventional tests are shown in Table 3. The posterior probabilities of the number of modes are shown in Table 4. The support of the data is continuous but the computation of the posterior probabilities of mode

¹The five largest economies in terms of per capita income (usually Middle East economies and tax haven islands) are excluded because they are large outliers.

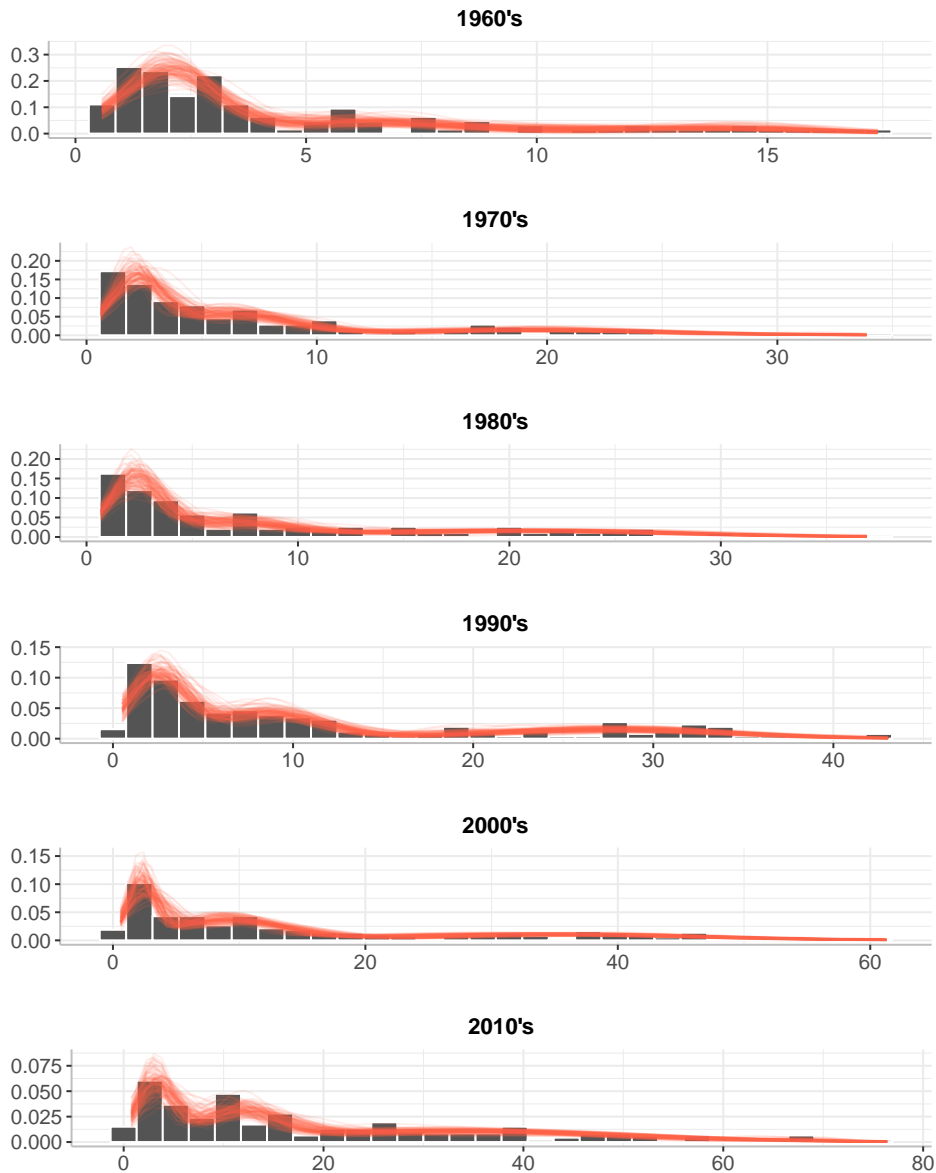


Figure 5: Estimated densities at 1000 MCMC iteration and empirical distributions of average real GDP per capita from the Penn World Tables.

locations in Figure 6 requires a discrete range; for this purpose we have rounded the support to unity levels. Our results suggest that the density of per capita income across countries has been multimodal since the 1960's. There are three credible modes, with the left-side mode being the most concentrated; there is greater uncertainty regarding the location of the other two modes. The uncertainty attached to the mode location has increased over time while the modes have gradually diverged. There is a fourth mode emerging from the 1990's.

For possible GDP convergence analysis over time, we consider the estimated densities over different periods. Estimates at the beginning of the period have higher peaks and almost no

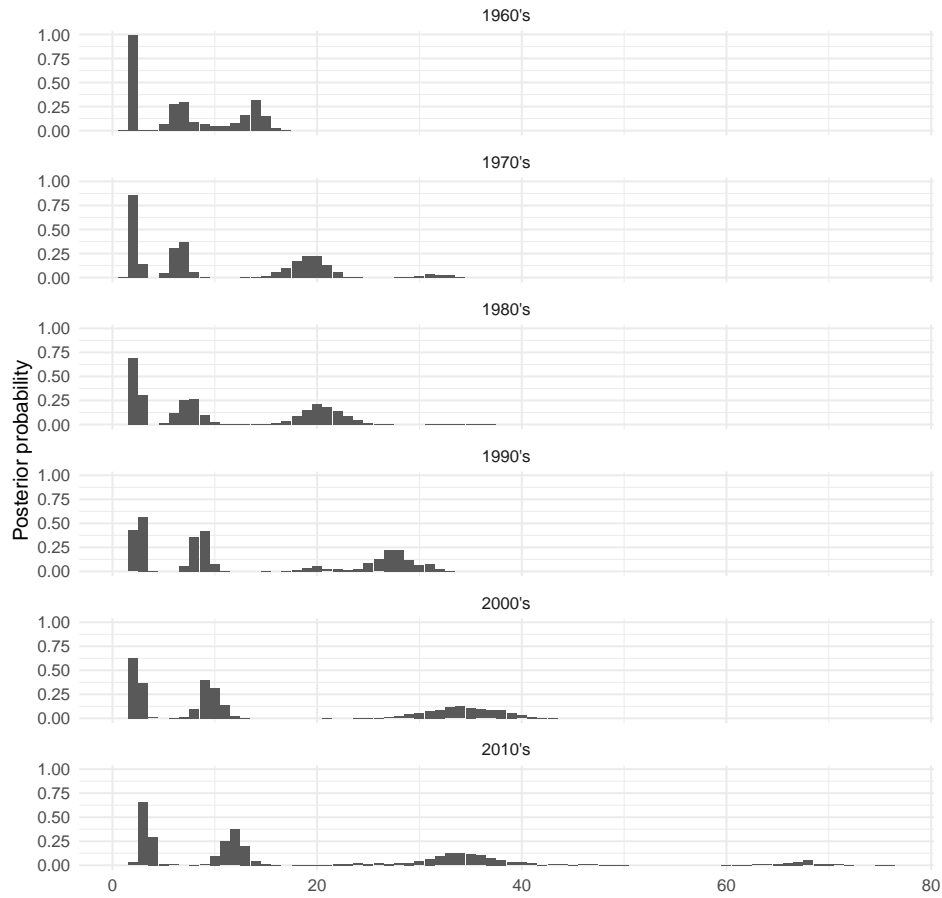


Figure 6: Posterior probabilities attached to mode locations using the Penn World Tables data.

probability mass in the mid-point of the data range. At the end of the period, despite the decreasing number of modes, the probability mass at the mid-point increases while the peaks at the tails of the densities are less pronounced. We therefore conclude that the dynamic behavior of the number of modes over time do not necessarily indicate ‘convergence’ and according to these results a ‘middle income’ category seems to be emerging over time.

	SI	HY	FM	HH	CH	ACR	BayesMultiMode
1960's	0.53	0.25	0.01	0.93	0.63	0.46	0.01
1970's	0.53	0.15	0.00	0.99	0.82	0.73	0.00
1980's	0.35	0.09	0.00	0.98	0.74	0.63	0.00
1990's	0.08	0.00	0.00	0.51	0.10	0.04	0.00
2000's	0.19	0.01	0.00	0.94	0.52	0.38	0.00
2010's	0.69	0.36	0.09	0.26	0.01	0.00	0.00

Table 3: P-values from conventional tests of multimodality for the Penn World Tables data. Null hypothesis of unimodality with alternative hypothesis of at least two modes.

decade	1	2	3	4	5
1960's	0.01	0.35	0.63	0.01	0.00
1970's	0.00	0.19	0.72	0.08	0.00
1980's	0.00	0.20	0.76	0.04	0.00
1990's	0.00	0.07	0.80	0.14	0.00
2000's	0.00	0.01	0.97	0.02	0.00
2010's	0.00	0.04	0.72	0.23	0.01

Table 4: Posterior probabilities attached to the number of modes for the Penn World Tables data.

6.3. DNA count data

Count data are a specific type of discrete empirical distributions that are collections of non-negative integers representing the number of times a discrete event is observed. They are prevalent in many areas of Bioinformatics. Examples of such data are counts of macrosatellite repeats (MSRs), hospital admissions, and rates of cardiac arrest. Despite the widespread occurrence, there is no unified framework for detecting multimodality in discrete data; **BayesMultiMode** fills an important gap in that respect.

Here we use macrosatellite repeats (MSRs) copy number variation in worldwide populations as examined by [Schaap *et al.* \(2013\)](#), among others see, e.g., [Giacalone, Friedes, and Francke \(1992\)](#) and [Jones, King, Himeda, Homma, Chen, Beermann, Yan, Emerson, Miller, Wagner *et al.* \(2015\)](#). MSRs typically span hundreds of kilobases of genomic DNA and have a highly polymorphic nature that has been linked to the detection of genetic disease (see, e.g. [van Overveld, Lemmers, Sandkuijl, Enthoven, Winokur, Bakels, Padberg, van Ommen, Frants, and van der Maarel 2003](#); [Bruce, Sachs, Rudnicki, Lin, Willour, Cowell, Conroy, McQuaid, Rossi, Gaile, Nowak, Holmes, Sklar, Ross, Delisi, and Margolis 2009](#); [Balog, Miller, Sanchez-Curtailles, Carbo-Marques, Block, Potman, De Knijff, Lemmers, Tapscott, and Van Der Maarel 2012](#); [Mitsuhashi, Nakagawa, Takahashi Ueda, Imanishi, Frith, and Mitsuhashi 2017](#)). Reliable statistical approaches for testing unimodal against multimodal size distributions in MSRs are consequently of great practical importance.

We analyze two data sets with different modal features. One data set (sequence d4z4) exhibits a clear sign of multimodality while the other (sequence ct47) does not. Testing for multimodality using the frequentist tests specified above yields

```
R> table_tests = rbind(tests_mode(d4z4),
+                       tests_mode(ct47))
R> row.names(table_tests) = c("d4z4", "ct47")
R> table_tests
```

	SI	HY	FM	HH	CH	ACR
d4z4	0.652	0.404	0.182	0.956	0.562	0.584
ct47	0.718	0.586	0.050	0.946	0.474	0.600

While sequence d4z4 is likely to be multimodal, none of the tests rejects the null hypothesis of unimodality, and this by a comfortable margin. Regarding sequence ct47, two tests reject the null hypothesis of unimodality with a 90% confidence level whereas we expect these data to have a unimodal distribution. Thus the frequentist tests provide results which are at odds with our results as well as with previous studies using these data (see [Schaap *et al.* 2013](#)). They fail to spot evident multimodality while falsely rejecting unimodality in data likely to be unimodal. Frequentist tests are apparently unreliable when applied to count data, of which DNA data are a typical example, partially because these tests assume continuous underlying data.

We now use **BayesMultiMode** to analyze the DNA sequences. First, we estimate a mixture of shifted-Poisson distributions. We use 2000 draws and discard half of those as burnin. We use the default priors of the package.

```
R> mcmc_d4z4 = bayes_estimation(data = d4z4,
+                               K = 10,
```

```

+                               dist = "shifted_poisson",
+                               nb_iter = 2000,
+                               burnin = 1000,
+                               printing = FALSE)
R> mcmc_ct47 = bayes_estimation(data = ct47,
+                               K = 10,
+                               dist = "shifted_poisson",
+                               nb_iter = 2000,
+                               burnin = 1000,
+                               printing = FALSE)

```

Figure 7 shows the estimated mass functions for 100 draws together with the data for both sequences. Visually, we notice that the MCMC estimation yields a mass function with about four modes for the d4z4 sequence and a unique mode for the ct47 sequence. We can formally

```

R> p1 = plot(mcmc_d4z4, max_size = 100, transparency = 0.8) +
+   ggtitle("d4z4")
R> p2 = plot(mcmc_ct47, max_size = 100, transparency = 0.8) +
+   ggtitle("ct47")
R> ggarrange(p1, p2,
+           ncol = 2, nrow = 1, widths = c(0.5, 0.5))

```

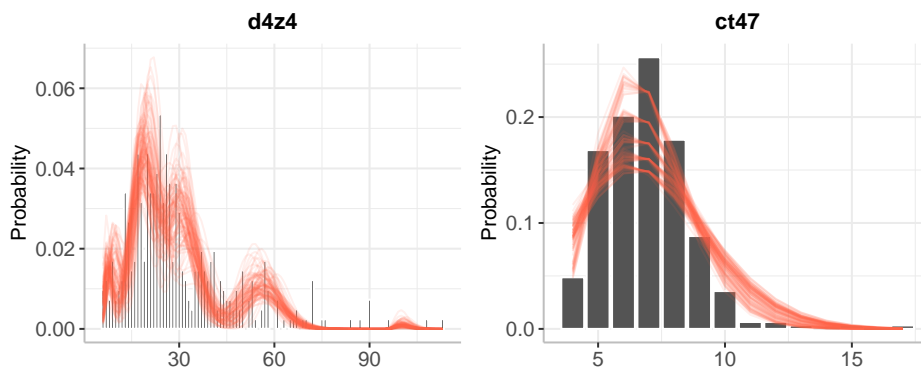


Figure 7: Two DNA empirical distributions and their mixture estimates.

investigate the modes in the estimated probability mass functions with the **BayesMultiMode**'s `bayes_mode` function; graphical outputs are shown in Figure 8. Posterior probabilities for the number of modes favor four modes for the d4z4 sequence and one mode (with posterior probability of one) for ct47 sequence.

We have illustrated how **BayesMultiMode** can correct the failure of frequentist tests analyzing discrete data.

```
R> p1 = plot(bayes_mode(mcmc_d4z4)) +
+   ggtitle("d4z4")
R> p2 = plot(bayes_mode(mcmc_ct47)) +
+   ggtitle("ct47")
R> ggarrange(p1, p2, labels = c("d4z4", "ct47"),
+           ncol = 1, nrow = 2, widths = c(0.5, 0.5))
```

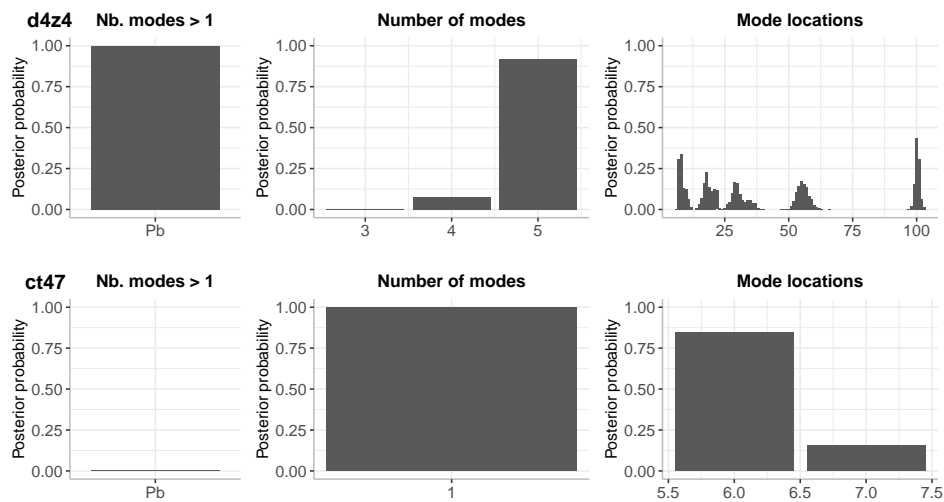


Figure 8: Posterior probabilities of the number of modes and their locations using the DNA count data d4z4 and ct47.

6.4. BayesMultiMode with external MCMC output using cyclone data

BayesMultiMode provides a self-contained tool for exploring modal features by first fitting a Bayesian mixture model and subsequently estimating modes using the MCMC draws. The choice of available distributions in the package, which can be either continuous or discrete, and the implementation of an SFM MCMC algorithm for estimating the number of mixture components makes **BayesMultiMode** a competitive tool for estimating mixture distributions. However, given that Bayesian methods for fitting mixtures are diverse and evolving rapidly, users might prefer an alternative MCMC approach. Therefore, **BayesMultiMode** is deliberately designed to be compatible with MCMC outputs generated using external software.

To illustrate how the package can be used in conjunction with external software we make use of the package **BNPmix**. The latter provides nonparametric Bayesian methods for estimating mixtures. It is notably very fast because it is partly written in C++, which makes it useful for analyzing large datasets.

In this section we analyze data on tropical cyclone intensity, an ongoing subject of interest in climate studies (Lee, Tippett, Sobel, and Camargo 2016; Kossin, Olander, and Knapp 2013; Manganello, Hodges, Kinter, Cash, Marx, Jung, Achuthavarier, Adams, Altshuler, Huang, Jin, Stan, Towers, and Wedi 2012; Lee *et al.* 2016; Elsner, Kossin, and Jagger 2008; Webster, Holland, Curry, and Chang 2005; Song, Klotzbach, and Duan 2021). Data on tropical cyclones are made available at the International Best Track Archive for Climate Stewardship (IBTrACS) (Knapp, Kruk, Levinson, Diamond, and Neumann 2010; Knapp, Diamond, Kossin, Kruk, and Schreck 2018). Intensity is typically measured as the lifetime maximum intensity (LMI) which we derive as the maximum wind speed of each cyclone. The dataset `cyclone` in **BayesMultiMode** shows cyclones' year, identification and LMI; the code to generate it is available in the source code of the package.

Figure 9 shows LMI data of cyclones in the Eastern North Pacific basin from 1981 with an estimated mixture.²

The data of tropical cyclone intensity across the planet are known to have a bimodal distribution which has important implications for climate modeling. Tropical cyclones are thus a singular natural event in that their likelihood of occurrence can increase with intensity past a certain threshold. Lee *et al.* (2016) show that this can be explained by a phenomenon called rapid intensification which affects most cyclones with high intensity forming the right mode of the estimated density.

Estimating a mixture on these data with **BNPmix** can be done as follows.

```
R> library(BNPmix)
R> y = cyclone %>%
+   filter(BASIN == "EP",
+         SEASON > "1981") %>%
+   select(max_wind) %>%
+   unlist()
R> PY_result = PYdensity(y,
+                         mcmc = list(niter = 2000,
```

²We do not use the entire global data for all basins because estimation and generating the plots would take more computing time, but it is entirely feasible. We follow other studies on these data in using cyclones only from the 1980's.

```

+                               nburn = 1000,
+                               print_message = FALSE),
+                               output = list(out_param = TRUE))

```

Most software for Bayesian mixture modeling generate an output in the form of a matrix of MCMC draws with one column per variable. However, **BNPmix** returns results in a different format. Thus before using the MCMC output from **BNPmix** into the `bayes_mode()` function we must manipulate the result slightly, which gives us the opportunity to illustrate how this can be done for similar MCMC output formats.

```

R> mcmc_py = list()
R> for (i in 1:length(PY_result$p)) {
+   k = length(PY_result$p[[i]][, 1])
+
+   draw = c(PY_result$p[[i]][, 1],
+           PY_result$mean[[i]][, 1],
+           sqrt(PY_result$sigma2[[i]][, 1]),
+           i)
+
+   names(draw)[1:k] = paste0("eta", 1:k)
+   names(draw)[(k+1):(2*k)] = paste0("mu", 1:k)
+   names(draw)[(2*k+1):(3*k)] = paste0("sigma", 1:k)
+   names(draw)[3*k + 1] = "draw"
+
+   mcmc_py[[i]] = draw
+ }
R> mcmc_py = bind_rows(mcmc_py)

```

Now that we have a MCMC matrix with one column per variable (e.g. `eta1`, `eta2`, ..., `mu1`, `mu2`, ... etc) we can create an object of class `BayesMixture`; the necessary input type for function `bayes_mode()`.

```

R> pars_names = c(eta = "eta",
+               mu = "mu",
+               sigma = "sigma")
R> py_BayesMix = new_BayesMixture(mcmc = mcmc_py,
+                               data = y,
+                               K = (ncol(mcmc_py)-1)/3,
+                               burnin = 0, # the burnin has already been discarded
+                               dist = "normal",
+                               pars_names = pars_names,
+                               dist_type = "continuous")

```

Figure 9 shows the estimated mixture in 100 MCMC draws. The estimated density is credibly multimodal with two clear modes emerging.

The remaining steps are similar to those described above when estimating the mixture density with **BayesMultiMode**. We retrieve the mode estimates with

```
R> plot(py_BayesMix, max_size = 100, transparency = 0.8)
```

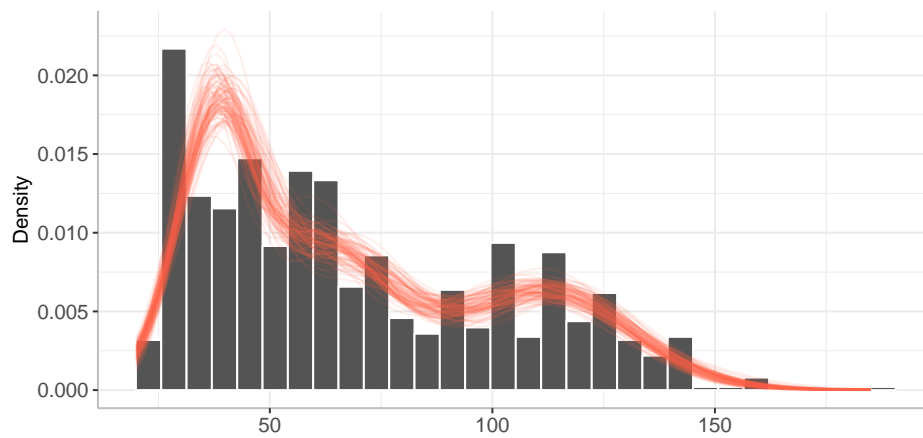


Figure 9: Mixture estimates using package BNPmix alongside the empirical distribution of the cyclone data.

```
R> bayesmode = bayes_mode(py_BayesMix)
```

Figure provides a graphical representation of the Bayesian mode inference. The posterior

```
R> plot(bayesmode, max_size = 100, transparency = 0.8)
```

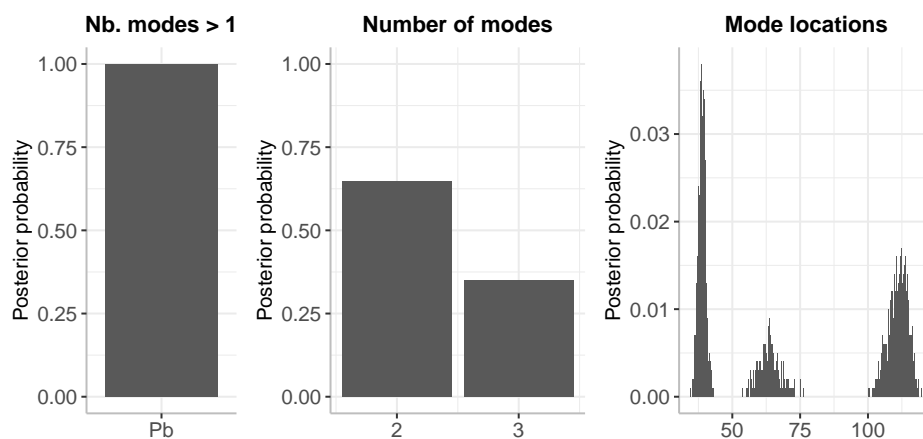


Figure 10: Posterior probabilities for the number of modes and their locations using cyclone data.

probability of the density being multimodal is

```
R> 1-bayesmode$p1
```

```
[1] 1
```

with approximately a 3/4 posterior probability of two modes and 1/4 posterior probability of three modes. The first mode emerges around 40ks while there is a small mode around 60ks and a stronger mode around 120ks. The latter mode corresponds to cyclones with very large intensity. This result is consistent with the global results of Lee *et al.* (2016).

7. Concluding remarks

BayesMultiMode fills an important gap in statistical software by providing a robust mode inference approach which (i) yields information on the number of modes, their locations and associated uncertainties and (ii) is also suitable for discrete empirical distributions contrary to frequentist methods. Using a set of data ranging from the fields of Astrophysics, Bioinformatics, Climatology and Economics as well as a set of mixture models, we have shown that **BayesMultiMode** spots multiple modes in a credible way with substantial posterior probability where frequentist methods fail; conversely our method can credibly indicate unimodality where frequentist methods can only reject or not reject this hypothesis. The mode inference tools of **BayesMultiMode** are also compatible with external software using other MCMC methods for mixture estimation. Thus, while **BayesMultiMode** provides a unique implementation of the SFM MCMC algorithm using several families of probability distributions, the user is not limited to this choice and can integrate our mode inference tool in her statistical workflow easily.

Finally, we emphasize that our results may be used as a first step in forecasting and policy analysis of issues where the heterogeneity of the underlying population distribution is important.

References

- Ameijeiras-Alonso J, Crujeiras RM, Rodriguez-Casal A (2021). “multimode: An R Package for Mode Assessment.” *Journal of Statistical Software*, **97**, 1–32. ISSN 1548-7660. doi:10.18637/jss.v097.i09. URL <https://doi.org/10.18637/jss.v097.i09>.
- Ameijeiras-Alonso J, Crujeiras RM, Rodriguez-Casal A (2019). “Mode testing, critical bandwidth and excess mass.” *TEST*, **28**(3), 900–919. ISSN 1863-8260. doi:10.1007/s11749-018-0611-5. URL <https://doi.org/10.1007/s11749-018-0611-5>.
- Azzalini A (1985). “A Class of Distributions Which Includes the Normal Ones.” *Scandinavian Journal of Statistics*, **12**(2), 171–178. ISSN 0303-6898. Publisher: [Board of the Foundation of the Scandinavian Journal of Statistics, Wiley], URL <https://www.jstor.org/stable/4615982>.
- Azzalini A (2013). *The Skew-Normal and Related Families*. Institute of Mathematical Statistics Monographs. Cambridge University Press. doi:10.1017/CB09781139248891.
- Balog J, Miller D, Sanchez-Curtailles E, Carbo-Marques J, Block G, Potman M, De Knijff P, Lemmers RJ, Tapscott SJ, Van Der Maarel SM (2012). “Epigenetic regulation of the X-chromosomal macrosatellite repeat encoding for the cancer/testis gene CT47.” *European Journal of Human Genetics*, **20**(2), 185–191.

- Bianchi M (1997). “Testing for convergence: evidence from non-parametric multimodality tests.” *Journal of Applied Econometrics*, **12**(4), 393–409.
- Bruce H, Sachs N, Rudnicki D, Lin S, Willour V, Cowell J, Conroy J, McQuaid D, Rossi M, Gaile D, Nowak N, Holmes S, Sklar P, Ross C, Delisi L, Margolis R (2009). “Long tandem repeats as a form of genomic copy number variation: structure and length polymorphism of a chromosome 5p repeat in control and schizophrenia populations.” *Psychiatric Genetics*, **19**(2), 64–71.
- Carpenter B, Gelman A, Hoffman MD, Lee D, Goodrich B, Betancourt M, Brubaker M, Guo J, Li P, Riddell A (2017). “Stan: A Probabilistic Programming Language.” *Journal of Statistical Software*, **76**, 1–32. ISSN 1548-7660. doi:10.18637/jss.v076.i01. URL <https://doi.org/10.18637/jss.v076.i01>.
- Carreira-Perpinan M (2000). “Mode-finding for mixtures of Gaussian distributions.” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **22**(11), 1318–1323. ISSN 1939-3539. doi:10.1109/34.888716. Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence.
- Chaudhuri P, Marron JS (1999). “SiZer for exploration of structures in curves.” *Journal of the American Statistical Association*, **94**(447), 807–823.
- Cheng MY, Hall P (1998). “Calibrating the Excess Mass and Dip Tests of Modality.” *Journal of the Royal Statistical Society Series B: Statistical Methodology*, **60**(3), 579–589. ISSN 1369-7412. doi:10.1111/1467-9868.00141. URL <https://doi.org/10.1111/1467-9868.00141>.
- Collet J (2022). “ModEstM: Mode Estimation, Even in the Multimodal Case.” URL <https://CRAN.R-project.org/package=ModEstM>.
- Corradin R, Canale A, Nipoti B (2021). “BNPmix: An R Package for Bayesian Nonparametric Modeling via Pitman-Yor Mixtures.” *Journal of Statistical Software*, **100**, 1–33. ISSN 1548-7660. doi:10.18637/jss.v100.i15. URL <https://doi.org/10.18637/jss.v100.i15>.
- Cross JL, Hoogerheide L, Labonne P, van Dijk HK (2023). “Credible Mode Determination in Multimodal Economic and Financial Data Distributions.” *Tinbergen Institute Discussion Paper TI 2023-038/III*.
- Dempster AP, Laird NM, Rubin DB (1977). “Maximum Likelihood from Incomplete Data via the EM Algorithm.” *Journal of the Royal Statistical Society. Series B (Methodological)*, **39**(1), 1–38. ISSN 0035-9246. Publisher: [Royal Statistical Society, Wiley], URL <https://www.jstor.org/stable/2984875>.
- Dümbgen L, Walther G (2008). “Multiscale inference about a density.” *The Annals of Statistics*, **36**(4), 1758–1785. ISSN 0090-5364, 2168-8966. doi:10.1214/07-AOS521. Publisher: Institute of Mathematical Statistics, URL <https://projecteuclid.org/journals/annals-of-statistics/volume-36/issue-4/Multiscale-inference-about-a-density/10.1214/07-AOS521.full>.

- Eddy WF (1980). “Optimum Kernel Estimators of the Mode.” *The Annals of Statistics*, **8**(4), 870–882. ISSN 0090-5364, 2168-8966. doi:[10.1214/aos/1176345080](https://doi.org/10.1214/aos/1176345080). Publisher: Institute of Mathematical Statistics, URL <https://projecteuclid.org/journals/annals-of-statistics/volume-8/issue-4/Optimum-Kernel-Estimators-of-the-Mode/10.1214/aos/1176345080.full>.
- Elsner JB, Kossin JP, Jagger TH (2008). “The increasing intensity of the strongest tropical cyclones.” *Nature*, **455**(7209), 92–95. ISSN 1476-4687. doi:[10.1038/nature07234](https://doi.org/10.1038/nature07234). Number: 7209 Publisher: Nature Publishing Group, URL <https://www.nature.com/articles/nature07234>.
- Feenstra RC, Inklaar R, Timmer MP (2015). “The Next Generation of the Penn World Table.” *American Economic Review*, **105**(10), 3150–3182.
- Fischer NI, Mammen E, Marron JS (1994). “Testing for multimodality.” *Computational Statistics & Data Analysis*, **18**(5), 499–512.
- Fisher NI, Marron JS (2001). “Mode Testing via the Excess Mass Estimate.” *Biometrika*, **88**(2), 499–517. ISSN 0006-3444. Publisher: [Oxford University Press, Biometrika Trust], URL <https://www.jstor.org/stable/2673496>.
- Frühwirth-Schnatter S (2006). *Finite mixture and Markov switching models: Modeling and applications to random processes*. Springer Series in Statistics. Springer, New York/Berlin/Heidelberg.
- Frühwirth-Schnatter S, Malsiner-Walli G (2019). “From here to infinity: sparse finite versus Dirichlet process mixtures in model-based clustering.” *Advances in Data Analysis and Classification*, **13**, 33–64.
- Frühwirth-Schnatter S, Pyne S (2010). “Bayesian inference for finite mixtures of univariate and multivariate skew-normal and skew-t distributions.” *Biostatistics*, **11**(2), 317–336. ISSN 1465-4644. doi:[10.1093/biostatistics/kxp062](https://doi.org/10.1093/biostatistics/kxp062). URL <https://doi.org/10.1093/biostatistics/kxp062>.
- Giacalone J, Friedes J, Francke U (1992). “A novel GC-rich human macrosatellite VNTR in Xq24 is differentially methylated on active and inactive X chromosomes.” *Nature genetics*, **1**(2), 137–143.
- Green PJ (1995). “Reversible jump Markov chain Monte Carlo computation and Bayesian model determination.” *Biometrika*, **82**(4), 711–732.
- Gruen B, Plummer M (2021). “bayesmix: Bayesian Mixture Models with JAGS.” URL <https://CRAN.R-project.org/package=bayesmix>.
- Grün B, Malsiner-Walli G, Frühwirth-Schnatter S (2022). “How many data clusters are in the Galaxy data set?” *Advances in Data Analysis and Classification*, **16**(2), 325–349. ISSN 1862-5355. doi:[10.1007/s11634-021-00461-8](https://doi.org/10.1007/s11634-021-00461-8). URL <https://doi.org/10.1007/s11634-021-00461-8>.
- Hall P, York M (2001). “On the calibration of Silverman’s test for multimodality.” *Statistica Sinica*, **11**(2), 515–536.

- Hartigan JA, Hartigan PM (1985). “The DIP Test of Unimodality.” *The Annals of Statistics*, **13**(1), 70–84. ISSN 00905364.
- Hartigan PM (1985). “Computation of the DIP Statistic to Test for Unimodality.” *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, **34**(3), 320–325. ISSN 00359254.
- Henderson DJ, Parmeter CF, Russell RR (2008). “Modes, weighted modes, and calibrated modes: evidence of clustering using modality tests.” *Journal of Applied Econometrics*, **23**(5), 607–638.
- Jones TI, King OD, Himeda CL, Homma S, Chen JC, Beermann ML, Yan C, Emerson CP, Miller JB, Wagner KR, *et al.* (2015). “Individual epigenetic status of the pathogenic D4Z4 macrosatellite correlates with disease in facioscapulohumeral muscular dystrophy.” *Clinical epigenetics*, **7**(1), 1–22.
- Knapp KR, Diamond HJ, Kossin J, Kruk MC, Schreck C (2018). “International Best Track Archive for Climate Stewardship (IBTrACS) Project, Version 4.” *NOAA National Centers for Environmental Information*. doi:10.1175/2009BAMS2755.1.
- Knapp KR, Kruk MC, Levinson DH, Diamond HJ, Neumann CJ (2010). “The International Best Track Archive for Climate Stewardship (IBTrACS): Unifying Tropical Cyclone Data.” *Bulletin of the American Meteorological Society*, **91**(3), 363–376. ISSN 0003-0007, 1520-0477. doi:10.1175/2009BAMS2755.1. Publisher: American Meteorological Society Section: Bulletin of the American Meteorological Society, URL https://journals.ametsoc.org/view/journals/bams/91/3/2009bams2755_1.xml.
- Kossin JP, Olander TL, Knapp KR (2013). “Trend Analysis with a New Global Record of Tropical Cyclone Intensity.” *Journal of Climate*, **26**(24), 9960–9976. ISSN 0894-8755, 1520-0442. doi:10.1175/JCLI-D-13-00262.1. Publisher: American Meteorological Society Section: Journal of Climate, URL <https://journals.ametsoc.org/view/journals/clim/26/24/jcli-d-13-00262.1.xml>.
- Lee CY, Tippett MK, Sobel AH, Camargo SJ (2016). “Rapid intensification and the bimodal distribution of tropical cyclone intensity.” *Nature Communications*, **7**(1), 10625. ISSN 2041-1723. doi:10.1038/ncomms10625. Number: 1 Publisher: Nature Publishing Group, URL <https://www.nature.com/articles/ncomms10625>.
- Leisch F, Gruen B (2023). “CRAN Task View: Cluster Analysis & Finite Mixture Models.” Publisher: Comprehensive R Archive Network (CRAN), URL <https://CRAN.R-project.org/view=Cluster>.
- Li J, Ray S, Lindsay BG (2007). “A Nonparametric Statistical Approach to Clustering via Mode Identification.” *Journal of Machine Learning Research*, **8**, 1687–1723.
- Lunn DJ, Thomas A, Best N, Spiegelhalter D (2000). “WinBUGS - A Bayesian modelling framework: Concepts, structure, and extensibility.” *Statistics and Computing*, **10**(4), 325–337. ISSN 1573-1375. doi:10.1023/A:1008929526011. URL <https://doi.org/10.1023/A:1008929526011>.
- Malsiner-Walli G, Frühwirth-Schnatter S, Grün B (2016). “Model-based clustering based on sparse finite Gaussian mixtures.” *Statistics and Computing*, **26**(1), 303–324.

- ISSN 1573-1375. doi:10.1007/s11222-014-9500-2. URL <https://doi.org/10.1007/s11222-014-9500-2>.
- Malsiner-Walli G, Frühwirth-Schnatter S, Grün B (2016). “Model-based clustering based on sparse finite Gaussian mixtures.” *Statistics and Computing*, **26**, 303–324.
- Manganello JV, Hodges KI, Kinter JL, Cash BA, Marx L, Jung T, Achuthavarier D, Adams JM, Altshuler EL, Huang B, Jin EK, Stan C, Towers P, Wedi N (2012). “Tropical Cyclone Climatology in a 10-km Global Atmospheric GCM: Toward Weather-Resolving Climate Modeling.” *Journal of Climate*, **25**(11), 3867–3893. ISSN 0894-8755, 1520-0442. doi:10.1175/JCLI-D-11-00346.1. Publisher: American Meteorological Society Section: Journal of Climate, URL <https://journals.ametsoc.org/view/journals/clim/25/11/jcli-d-11-00346.1.xml>.
- McLachlan G, Peel D (2004). *Finite mixture models*. John Wiley & Sons.
- Miller JW, Harrison MT (2013). “A simple example of Dirichlet process mixture inconsistency for the number of components.” *Advances in neural information processing systems*, **26**, 199–206.
- Minnotte MC, Marchette DJ, Wegman EJ (1998). “The Bumpy Road to the Mode Forest.” *Journal of Computational and Graphical Statistics*, **7**(2), 239–251. ISSN 1061-8600. doi:10.2307/1390816. Publisher: [American Statistical Association, Taylor & Francis, Ltd., Institute of Mathematical Statistics, Interface Foundation of America], URL <https://www.jstor.org/stable/1390816>.
- Minnotte MC, Scott DW (1993). “The Mode Tree: A Tool for Visualization of Nonparametric Density Features.” *Journal of Computational and Graphical Statistics*, **2**(1), 51–68. ISSN 1061-8600. doi:10.2307/1390955. Publisher: [American Statistical Association, Taylor & Francis, Ltd., Institute of Mathematical Statistics, Interface Foundation of America], URL <https://www.jstor.org/stable/1390955>.
- Mitsuhashi S, Nakagawa S, Takahashi Ueda M, Imanishi T, Frith MC, Mitsuhashi H (2017). “Nanopore-based single molecule sequencing of the D4Z4 array responsible for facioscapulothoracic muscular dystrophy.” *Scientific reports*, **7**(1), 1–8.
- Paap R, van Dijk HK (1998). “Distribution and mobility of wealth of nations.” *European Economic Review*, **42**(7), 1269–1293.
- Plummer M (2003). “JAGS: A Program for Analysis of Bayesian Graphical Models Using Gibbs Sampling.” *Proceedings of the 3rd International Workshop on Distributed Statistical Computing (DSC 2003)*. URL <https://mcmc-jags.sourceforge.io>.
- R Core Team (2021). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL <https://www.R-project.org/>.
- Richardson S, Green PJ (1997). “On Bayesian Analysis of Mixtures with an Unknown Number of Components.” *Journal of the Royal Statistical Society. Series B (Methodological)*, **59**(4), pp. 731–792. ISSN 00359246.

- 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. ISSN 0162-1459. doi:10.2307/2289993. Publisher: [American Statistical Association, Taylor & Francis, Ltd.], URL <https://www.jstor.org/stable/2289993>.
- Roeder K, Wasserman L (1997). “Practical Bayesian Density Estimation Using Mixtures of Normals.” *Journal of the American Statistical Association*, **92**(439), 894–902. ISSN 0162-1459. doi:10.2307/2965553. Publisher: [American Statistical Association, Taylor & Francis, Ltd.], URL <https://www.jstor.org/stable/2965553>.
- Rousseau J, Mengersen K (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.
- Schaap M, Lemmers RJ, Maassen R, van der Vliet PJ, Hoogerheide LF, van Dijk HK, Bastürk N, de Knijff P, van der Maarel SM (2013). “Genome-wide analysis of macrosatellite repeat copy number variation in worldwide populations: evidence for differences and commonalities in size distributions and size restrictions.” *BMC Genomics*, **14**(1), 1–12.
- Silverman BW (1981). “Using kernel density estimates to investigate multimodality.” *Journal of the Royal Statistical Society. Series B (Methodological)*, **41**(1), 97–99.
- Song J, Klotzbach PJ, Duan Y (2021). “Increasing lifetime maximum intensity of rapidly intensifying tropical cyclones over the western North Pacific.” *Environmental Research Letters*, **16**(3), 034002. ISSN 1748-9326. doi:10.1088/1748-9326/abdbf1. URL <https://iopscience.iop.org/article/10.1088/1748-9326/abdbf1>.
- Van Der Bles AM, Van Der Linden S, Freeman AL, Mitchell J, Galvao AB, Zaval L, Spiegelhalter DJ (2019). “Communicating uncertainty about facts, numbers and science.” *Royal Society open science*, **6**(5), 181870. Publisher: The Royal Society.
- van Overveld PG, Lemmers RJ, Sandkuijl LA, Enthoven L, Winokur ST, Bakels F, Padberg GW, van Ommen GJB, Frants RR, van der Maarel SM (2003). “Hypomethylation of D4Z4 in 4q-linked and non-4q-linked facioscapulohumeral muscular dystrophy.” *Nature Genetics*, **35**(4), 315–317.
- Viallefont V, Richardson S, Green PJ (2002). “Bayesian analysis of Poisson mixtures.” *Journal of Nonparametric Statistics*, **14**(1-2), 181–202.
- Webster PJ, Holland GJ, Curry JA, Chang HR (2005). “Changes in Tropical Cyclone Number, Duration, and Intensity in a Warming Environment.” *Science*, **309**(5742), 1844–1846. doi:10.1126/science.1116448. Publisher: American Association for the Advancement of Science, URL <https://www.science.org/doi/10.1126/science.1116448>.
- Wickham H (2016). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York. ISBN 978-3-319-24277-4. URL <https://ggplot2.tidyverse.org>.
- Wiecki T, Salvatier J, Vieira R, Kochurov M, Patil A, Osthege M, Willard BT, Engels B, Carroll C, Martin OA, Seyboldt A, Rochford A, Paz L, rpgoldman, Meyer K, Coyle P, Gorelli ME, Abril-Pla O, Kumar R, Lao J, Andreani V, Yoshioka T, Ho G, Kluyver T, Beauchamp K, Andorra A, Pananos D, Spaak E, Edwards B, Ma E (2023). “pymc-devs/pymc: v5.3.1.” doi:10.5281/zenodo.7868623. URL <https://doi.org/10.5281/zenodo.7868623>.

Zeileis A (2023). *pwt10: Penn World Table (Version 10)*. R package version 10.01-0, URL <https://CRAN.R-project.org/package=pwt10>.

Affiliation:

Paul Labonne

BI Norwegian Business school, Nydalsveien 37, 0484 Oslo, Norway