Stat Comput
DOI 10.1007/s11222-017-9774-2

Relabelling in Bayesian mixture models by pivotal units

Leonardo Egidi[1] · **Roberta Pappadà**[2] · **Francesco Pauli**[2] · **Nicola Torelli**[2]

Received: 7 January 2017 / Accepted: 21 August 2017

Abstract Label switching is a well-known and fundamental problem in Bayesian estimation of finite mixture models. It arises when exploring complex posterior distributions by Markov Chain Monte Carlo (MCMC) algorithms, because the likelihood of the model is invariant to the relabelling of mixture components. If the MCMC sampler randomly switches labels, then it is unsuitable for exploring the posterior distributions for component-related parameters. In this paper, a new procedure based on the post-MCMC relabelling of the chains is proposed. The main idea of the method is to perform a clustering technique on the similarity matrix, obtained through the MCMC sample, whose elements are the probabilities that any two units in the observed sample are drawn from the same component. Although it cannot be generalized to any situation, it may be handy in many applications because of its simplicity and very low computational burden.

Electronic supplementary material The online version of this article (doi:10.1007/s11222-017-9774-2) contains supplementary material, which is available to authorized users.

✉ Leonardo Egidi
egidi@stat.unipd.it

Roberta Pappadà
rpappada@units.it

Francesco Pauli
francesco.pauli@deams.units.it

Nicola Torelli
nicola.torelli@deams.units.it

[1] Dipartimento di Scienze Statistiche, Università degli Studi di Padova, Padua, Italy

[2] Dipartimento di Scienze Economiche, Aziendali, Matematiche e Statistiche 'Bruno de Finetti', Università degli Studi di Trieste, Trieste, Italy

Keywords Label switching · Complex posterior distributions · MCMC · Finite mixture model

1 Introduction

Label switching is a well-known and fundamental problem in Bayesian estimation of finite mixture models (McLachlan and Peel 2000). The label switching problem arises when exploring complex posterior distributions by Markov Chain Monte Carlo (MCMC) algorithms, because the likelihood of a G-component mixture model is invariant to the relabelling of mixture components. Because of this invariance, the likelihood has as many global maxima as there are permutations of the indices ($G!$). This is a minor problem (if a problem at all) when we perform classical inference, since any maximum leads to a valid solution and inferential conclusions are the same regardless of which one is chosen. However, invariance with respect to labels is a major problem when Bayesian inference is used: if the prior distribution is invariant with respect to the labelling as well as the likelihood, then the posterior distribution is multimodal.

A suitable MCMC sampler should, then, in order to explore the different modes, randomly switch labels. As a consequence, it would be unsuitable to make inference on a parameter specific of a component of the mixture.

Most of the existing approaches to perform inferences in the presence of label switching are based on the relabelling of the MCMC chain. A recent comprehensive review can be found in Papastamoulis (2016). Relabelling means permuting the labels at each iteration of the Markov chain in such a way that the relabelled chain can be used to draw inferences on component-specific parameters. Loosely speaking, we may say that the relabelled chain can be seen as a chain

Published online: 28 August 2017

where no label switching has occurred or, in other words, the new labels are such that different labels do refer to distinct components of the mixture. It is worth noting that relabelling strategies may act during the MCMC sampling, and/or they may be used to post-process the chains. Those solutions that post-process the chains are particularly convenient (since the issue can be ignored in performing the MCMC and then dealt with later).

In this paper, a new procedure based on the post-MCMC relabelling of the chains is proposed. The main idea of the method is to perform a clustering technique on the similarity matrix, obtained through the MCMC sample, whose elements are the probabilities that any two units in the observed sample are drawn from the same component. Starting from the obtained partition, G units—called pivots—one for each group are identified that belong to the same group with negligible (posterior) probability. Evidence from simulation studies shows that our procedure, although simpler and in most cases less computationally demanding than some competitors, has comparable performances even when dealing with relatively complex models.

The paper is organized as follows. In Sect. 2, the label switching problem in MCMC sampling is introduced in a general setting. In Sect. 3, a relabelling method based on pivotal units is introduced and discussed, and a range of criteria for pivot identification is given. In Sect. 4, a short overview of a selection of existing solutions to the label switching problem is given, including both deterministic and probabilistic relabelling approaches. In Sect. 5, a suitable simulation study is performed in order to investigate and evaluate the performance of the method introduced in this paper. A comparison with the relabelling algorithms described in the previous section is also provided. Section 6 illustrates the application of the proposed methodology to a real dataset and compares its performance with other available methods, in terms of both relabelling efficiency and computational effort. Section 7 concludes.

2 The label switching problem

Prototypical models in which the label switching problem arises are mixture models, where for a sample $\boldsymbol{y} = (y_1, \ldots, y_n)$ we assume

$$(Y_i | Z_i = g) \sim f(y; \mu_g, \phi),$$

where the Z_i, $i = 1, \ldots, n$, are i.i.d. random variables, $g = 1, \ldots, G$, ϕ is a parameter which is common to all components, $Z_i \in \{1, \ldots, G\}$, and

$$P(Z_i = g) = \pi_g.$$

The likelihood of the model is then

$$L(\boldsymbol{y}; \boldsymbol{\mu}, \boldsymbol{\pi}, \phi) = \prod_{i=1}^{n} \sum_{g=1}^{G} \pi_g f(y_i; \mu_g, \phi), \tag{1}$$

with $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_G)$ component-specific parameters and $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_G)$ mixture weights. Let ν denote a permutation of $\{1, \ldots, G\}$, and let $\nu(\boldsymbol{\mu}) = (\mu_{\nu(1)}, \ldots, \mu_{\nu(G)})$, $\nu(\boldsymbol{\pi}) = (\pi_{\nu(1)}, \ldots, \pi_{\nu(G)})$ be the corresponding permutations of $\boldsymbol{\mu}$ and $\boldsymbol{\pi}$. Denote by $\mathcal{V}$ the set of all the permutations of the indexes $\{1, \ldots, G\}$, Eq. (1) is invariant under any permutation $\nu \in \mathcal{V}$, that is

$$L(\boldsymbol{y}; \boldsymbol{\mu}, \boldsymbol{\pi}, \phi) = L(\boldsymbol{y}; \nu(\boldsymbol{\mu}), \nu(\boldsymbol{\pi}), \phi). \tag{2}$$

As a consequence, the model is unidentified with respect to an arbitrary permutation of the labels.

When Bayesian inference for the model is performed, if the prior distribution $p_0(\boldsymbol{\mu}, \boldsymbol{\pi}, \phi)$ is invariant under a permutation of the indices, then so is the posterior. That is, if $p_0(\boldsymbol{\mu}, \boldsymbol{\pi}, \phi) = p_0(\nu(\boldsymbol{\mu}), \nu(\boldsymbol{\pi}), \phi)$, then

$$p(\boldsymbol{\mu}, \boldsymbol{\pi}, \phi | \boldsymbol{y}) \propto p_0(\boldsymbol{\mu}, \boldsymbol{\pi}, \phi) L(\boldsymbol{y}; \boldsymbol{\mu}, \boldsymbol{\pi}, \phi) \tag{3}$$

is multimodal with (at least) $G!$ modes. This implies that all simulated parameters should be switched to one among the $G!$ symmetric areas of the posterior distribution, by applying suitable permutations of the labels to each MCMC draw.

In what follows, we assume that an MCMC sample is obtained from the posterior distribution for model (1) with a prior distribution which is labelling invariant. We denote as $\{[\theta]_h : h = 1, \ldots, H\}$ the sample for the parameter $\theta = (\boldsymbol{\mu}, \boldsymbol{\pi}, \phi)$, H being the number of MCMC iterations. We assume that also a MCMC sample for the variable Z is obtained and denote it by $\{[Z]_h : h = 1, \ldots, H\}$.

In principle, a perfectly mixing chain should visit the points $(\boldsymbol{\mu}, \boldsymbol{\pi}, \phi)$ and $(\nu(\boldsymbol{\mu}), \nu(\boldsymbol{\pi}), \phi)$ with the same frequency. A chain with less than perfect mixing may either concentrate on one mode of the posterior distribution or exhibit random switches [see Celeux et al. (2000) where several MCMC approaches for exploring the posterior distribution of mixture models and related inference problems are discussed].

It has been suggested that using a sampler which is inefficient with respect to the labelling—that is, unlikely to switch labels—but otherwise efficient, may be a solution to the label switching issue (an example is in Puolamäki and Kaski 2009). We do not want to discuss the aptness of such a solution in detail, but we note that it is tricky to justify it theoretically and that this solution is impractical in general terms since it is difficult to tune a sampler so that it is

inefficient enough to avoid label switches but not too inefficient.

It is well known that the presence of label switches (or the whole issue of relabelling) is totally irrelevant if the quantities of interest are invariant with respect to the labels. A particularly relevant example of invariant quantity is the probability of two units being in the same group, $c_{ij} = P(Z_i = Z_j|\mathcal{D})$, $i, j = 1, \ldots, n$, where $\mathcal{D}$ denotes the set of the data.

Relabelling becomes relevant when we are interested, directly or indirectly, in the features of G groups, such as the posterior (and predictive) distributions of component-related quantities such as the probability of each unit belonging to each group. Regarding this aim, we introduce here the $n \times G$ matrix Q, whose generic element q_{ig} is the probability that the i-th unit belongs to group g, $q_{ig} = P(Z_i = g|\mathcal{D})$, for $i = 1, \ldots, n$, $g = 1, \ldots, G$.

3 A relabelling method based on pivotal units

The starting point for the pivotal methods we propose is a partition of the observations. This can easily be obtained by maximizing the posterior distribution, notwithstanding the fact that the maximum is not unique (there are $G!$ modes); since the maxima are equivalent, any would be suitable. Alternatively, the estimates of the probabilities c_{ij} based on the MCMC sample

$$\hat{c}_{ij} = \frac{1}{H} \sum_{h=1}^{H} \left| [Z_i]_h = [Z_j]_h \right|, \tag{4}$$

where $|\cdot|$ denotes the indicator function of an event, can be used to derive a partition of observations through a suitable clustering technique. In fact, the $n \times n$ matrix $\hat{C}$ with elements $\hat{c}_{ij}$ can be seen as an estimated similarity matrix between units, and the complement to one $\hat{s}_{ij} = 1 - \hat{c}_{ij}$ as a dissimilarity matrix (note that it is not a distance metric as $s_{ij} = 0$ does not imply that the units i and j are the same).

Let, then, $\mathcal{G}_1, \ldots, \mathcal{G}_G$ be a partition. Furthermore, suppose that we can find G units, $i_1, \ldots, i_G$, one for each group, which are (pairwise) separated with (posterior) probability one (that is, the posterior probability of any two of them being in the same group is zero). In terms of the matrix C with elements $c_{ij} = P(Z_i = Z_j|\mathcal{D})$, the $G \times G$ submatrix with only the rows and columns corresponding to $i_1, \ldots, i_G$ will be the identity matrix. It is then straightforward to use the G units, called pivots in what follows, to identify the groups and to relabel the chains: for each $h = 1, \ldots, H$ and $g = 1, \ldots, G$, set

$$[\mu_g]_h = [\mu_{[Z_{i_g}]_h}]_h; \tag{5}$$

$$[Z_i]_h = g \quad \text{for } i : [Z_i]_h = [Z_{i_g}]_h. \tag{6}$$

The applicability of this strategy is limited by the existence of the pivots, which is not guaranteed (see the discussion in Sect. 3.1). Moreover, even when the pivots exist, they may be difficult to find, and the methods to detect them are central to the procedure. Some proposals are given and discussed in Sects. 3.2 and 3.3.

It is worth noting that the idea of solving the relabelling issue by fixing the group for some units dates back to Chung et al. (2004), who, however, gave no indication on how to choose the units. Also, since they suggest imposing such a restriction in the MCMC, there is no measure of the extent to which it influences the result (that is, the extent to which it is informative if we interpret it as a prior information). We note, however, that Chung et al. (2004) may be very interesting when a set of units which are to be attributed to different groups can be defined exogenously.

The idea of using some pivotal quantities for performing the relabelling can also be found in the `ECR` algorithm by Papastamoulis and Iliopoulos (2010) via the definition of equivalence classes representatives, and in Marin et al. (2005) and Marin and Robert (2007), where the pivotal reordering algorithm (`PRA`) is introduced.

Another related idea is put forward by Yao and Li (2014), who propose finding a reference labelling, that is, a clustering for the sample (for example, the posterior mode), and then relabel each iteration by minimizing some distance from the reference labelling. The general idea is similar to the one we suggest, but it is more computationally demanding because of the required minimizations. On the other hand, it avoids the need to condition on the pivots being separated. We can argue, however, that the latter is not a major drawback of our proposal since its effects can be measured and are likely to be small in many practical instances, as discussed in Sect. 3.1.

3.1 Existence of pivots

The existence of the pivots is a requirement of the method, meaning that its use is restricted to those chains—or those parts of a chain—for which the pivots are present. This is not always the case, and it is worth discussing some circumstances in which the pivots do not exist.

First, although the model is based on a mixture of G components, each iteration of the chain may imply a different number of non-empty groups (that is, it may be that $[Z_i]_h \neq g \quad \forall i$ for some g, h); let then $[G]_h \leq G$ be the number of non-empty groups at iteration h,

$$[G]_h = \#\{g : [Z_i]_h = g \quad \text{for some } i\},$$

where $\#A$ is the cardinality of the set A. If $[G]_h < G$ for some h, there cannot be G perfectly separated units,

and so there cannot be G pivots. Hence, the relabelling procedure outlined above can be used only for the subset of the chain for which $[G]_h = G$; let it be $\mathcal{H}_G = \{h : [G]_h = G\}$. This means that the resulting relabelled chain is not a sample (of size H) from the posterior distribution, but a sample (of size $\#\mathcal{H}_G$) from the posterior distribution conditional on there being (exactly) G non-empty groups.

In fact, we can also consider the posterior distribution conditional on $G' < G$ groups for each G' such that $\mathcal{H}_{G'} \neq \emptyset$ ($[G]_h = G'$ for some h). Although the procedure has been described above for G groups, where G is the number of components, it can be implemented for any $G' < G$ such that $\mathcal{H}_{G'} \neq \emptyset$ starting from a partition of the observations into G' groups (and, we note in passing, it may even be meaningless for $G' = G$ since $\mathcal{H}_G$ may be empty).

We do not see this restriction as a major limitation of the procedure since it is reasonable to see the issue of determining the number of groups as a separate one; that is, it is reasonable to study the characteristics of the groups conditional on the number of groups, which entails performing the relabelling for those sections of the chain where the number of non-empty groups is constant.

Even if G non-empty groups are available, however, there may not be G perfectly separated units. Let us define

$$\mathcal{H}_G^* = \{h \in \mathcal{H}_G : \exists k, s \quad \text{s.t. } [Z_{i_k}]_h = [Z_{i_s}]_h\}$$

that is, the set of iterations where (at least) two pivots are in the same group. In order for the pivot method to be applicable, we need to exclude iterations $\mathcal{H}_G^*$; that is, we can perform the pivot relabelling on $\mathcal{H}_G - \mathcal{H}_G^*$. Exclusion of $\mathcal{H}_G^*$ does not have a clear interpretation in terms of conditioning; thus, we may see the restricted chain $\mathcal{H}_G - \mathcal{H}_G^*$ as a sample from an approximation of the posterior conditional to being G non-empty groups, where the quality of the approximation is loosely given by $k^* = 1 - \#\mathcal{H}_G^*/\#\mathcal{H}_G$. Such proportion, as clarified later, could be used for selecting pivot identification criteria.

3.2 Pivot identification based on dissimilarity measures

Assuming that G pivotal units do exist (possibly after enacting the restrictions outlined in Sect. 3.1), identifying them is not straightforward, since the set of all possible choices is too large to be fully searched.

The general method we put forward is to select a unit for each group according to some criterion, conceived so that the chosen unit is as far as possible from units that might belong to the other groups and/or as close as possible to units that belong to the same group. Many criteria could be proposed; for instance, for group g containing units $\mathcal{G}_g$, we may chose $i^* \in \mathcal{G}_g$ that maximizes one of the quantities

$$\begin{aligned}
&\text{(a)} \max_{j \in \mathcal{G}_g} c_{i^* j};\\
&\text{(b)} \sum_{j \in \mathcal{G}_g} c_{i^* j};\\
&\text{(c)} \sum_{j \in \mathcal{G}_g} c_{i^* j} - \sum_{j \notin \mathcal{G}_g} c_{i^* j}.
\end{aligned} \tag{7}$$

These, respectively, give (a) the less distant unit among the members that are the closest (most similar), (b) the unit that maximizes the global within similarity, (c) the unit that maximizes the difference between global within and between similarities. Alternatively, we may choose $i^* \in \mathcal{G}_g$, which minimizes one of the quantities

$$\begin{aligned}
&\text{(d)} \min_{j \in \mathcal{G}_g} c_{i^* j};\\
&\text{(e)} \min_{j \notin \mathcal{G}_g} c_{i^* j};\\
&\text{(f)} \sum_{j \notin \mathcal{G}_g} c_{i^* j},
\end{aligned} \tag{8}$$

obtaining (d) the most distant unit among the members that are the closest (most similar), (e) the most distant unit among the members that are farthest apart (most dissimilar), (f) the most distant unit among the members that minimize the global dissimilarity between one group and all the others.

3.3 The MUS algorithm

We introduce a further method for detecting pivotal units, which we call Maxima Units Search (hereafter, MUS), which turns out to be suitable in case of a low number of mixture components, e.g. $G = 3, 4$.

The MUS algorithm does not rely upon a maximization/minimization step, like the procedures in Sect. 3.2, but searches for G pivots that identify submatrices with a simple structure within the estimated similarity matrix C. The underlying idea is to choose as pivots those units $j_1, \ldots, j_G$ such that the $G \times G$ submatrix of C with only the $j_1, \ldots, j_G$ rows and columns has few, possibly zero, nonzero elements off the diagonal (that is, this submatrix is identical or nearly identical). Note that an identity submatrix of the given dimension may not exist. It is worth stressing that for a small number of groups (e.g. $G = 4$) and a sample size n ranging between 100 and 1000, this search can be computationally demanding. Furthermore, the existence of such identity submatrices is not always guaranteed. For a more technical illustration of the method and an overview of possible applications, we refer to Egidi et al. (2016).

4 A review of selected alternative methods

Relabelling strategies may be divided into two main categories. The first includes *deterministic* procedures, which select a relabelling that minimizes the posterior expectation of some loss function at each MCMC iteration; the second consists of *probabilistic* procedures, where the parameters' permutations are considered parameters with associated uncertainty. In this section, a short description of some existing relabelling methods is provided. Most of deterministic algorithms have been implemented in the **label.switching** `R` package (Papastamoulis 2016) and will be considered in Sects. 5 and 6 in comparison with our proposal.

Deterministic relabelling strategies have been reviewed in Rodríguez and Walker (2014). According to them, most of these algorithms have the objective of finding the permutation of the parameters that minimizes an appropriate loss function. The general decision theoretic framework proposed by Stephens (2000) is an excellent framework for presenting and justifying most of the methods.

This approach translates the problem to that of choosing an action a from a set of actions $\mathcal{A}$ into the parameter space Θ, where a loss function $\mathcal{L} : \mathcal{A} \times \Theta \to \mathbb{R}$ represents the loss we incur if we choose the action a and the true value of the parameter is θ. The loss function makes sense if it is permutation invariant (remember that if we permute the parameter components, the model remains the same). The action a is then chosen by minimizing the posterior expected loss $\mathcal{R}(a) = E(\mathcal{L}(a;\theta)|\mathcal{D})$.

According to our specific problem, the action a can be the estimation of the parameter (or part of it), the clustering allocation into groups, or a particular summary from the posterior distribution; the loss function can be a distribution to be fitted or an estimation error. The choice should be driven by the objective of inference. It is worth noting that a possible problem with this general class of methods might be the choice of the appropriate loss function.

If the objective is the clustering of n units into G groups, a reasonable action is to report the $n \times G$ matrix Q defined in Sect. 2. A corresponding loss is then the distance, somehow measured, between Q and its true value $P(\theta) = [p_{ig}(\theta)]$ where (for the toy example)

$$p_{ig}(\theta) = P(Z_i = g|\boldsymbol{y},\theta) = \frac{\pi_g f(y_i;\mu_g,\theta)}{\sum_j \pi_j f(y_i;\mu_j,\theta)}.$$

In particular, Stephens' method (2000) employs the Kullback–Leibler distance.

Algorithm 1: `STEPHENS`

Start: choose H initial permutations $\nu^{(t)}$, $t = 1,\ldots,H$ (usually set to the identity).
Step 1: for $t = 1,\ldots,H$, $g = 1,\ldots,G$, $i = 1,\ldots,n$ calculate $q_{ig} = H^{-1}\sum_{t=1}^{H} p^{(t)}_{i\nu(g)}$.
Step 2: for $t = 1,\ldots,H$ find a permutation $\nu^{(t)}$ which minimizes

$$\mathcal{L}^{(t)}(Q;\theta) = \sum_{i=1}^{n}\sum_{g=1}^{G} p^{(t)}_{i\nu(g)} \log\left(\frac{p^{(t)}_{i\nu(g)}}{q_{ig}}\right).$$

Step 3: if an improvement is made to $\sum_{t=1}^{H}\mathcal{L}^{(t)}(Q;\theta)$, go to Step 2; otherwise, stop.

Note that step 2 entails n minimizations with respect to all the permutations ($G!$). The method is computationally expensive and requires storing the $H \times n \times G$ array p of classification probabilities.

The `ECR` algorithm (Algorithm 2) introduced by Papastamoulis and Iliopoulos (2010) partitions the set of allocation vectors $Z = (Z_1,\ldots,Z_n)$ into equivalence classes and then selects a representative from each class. In order to find these equivalence sets, the procedure involves the definition of a pivotal allocation $Z^* = (Z_1^*,\ldots,Z_n^*)$, generally selected by choosing an high-posterior density point, e.g. the Maximum A Posteriori (MAP) estimate. Thus, a natural action a here is the allocation vector Z.

Algorithm 2: `ECR`

Start: define a pivot allocation $Z^* = (Z_1^*,\ldots,Z_n^*)$.
Step 1: for $t = 1,\ldots,H$, find $\nu^{(t)}$ that minimizes

$$\mathcal{L}^{(t)}(Z;\theta) = \sum_{i=1}^{n} |\nu(Z_i^{(t)}) \neq Z_i^*|.$$

Rodríguez and Walker (2014) implemented two iterative versions for the `ECR`, named `ECR-iterative-1` and `ECR-iterative-2`, respectively (see Algorithms 3 and 4 below); according to these modified versions, the pivot is selected via an iterative procedure, and this makes these methods computationally more expensive than the basic `ECR` algorithm. The first procedure uses as inputs only the allocation vector Z, while the second one requires also the array of classification probabilities across the MCMC sample p (also used in Stephens 2000). It is worth stressing that storing this array is often not feasible in terms of CPU time.

Algorithm 3: `ECR-iterative-1`

Start: initialize H initial permutations $\nu^{(t)}$, $t = 1, \ldots, H$ (usually set to the identity).
Step 1: for $i = 1, \ldots, n$ calculate $Z_i^* = \text{mode}\{\nu(Z_i^{(t)}),\ t = 1, \ldots, H\}$.
Step 2: for $t = 1, \ldots, H$, find $\nu^{(t)}$ that minimizes

$$\mathcal{L}^{(t)}(Z; \theta) = \sum_{i=1}^{n} |\nu(Z_i^{(t)}) \neq Z_i^*|.$$

Step 3: if an improvement in $\sum_{t=1}^{H} \mathcal{L}^{(t)}(Z; \theta)$ has been achieved, go back to Step 2; otherwise, finish.

Algorithm 4: `ECR-iterative-2`

Start: initialize H initial permutations $\nu^{(t)}$, $t = 1, \ldots, H$ (usually set to the identity).
Step 1: for $i = 1, \ldots, n$ calculate $Z_i^* = \text{argmax}\{p_{i\nu(g)}^{(t)},\ t = 1, \ldots, H\}$.
Step 2: for $t = 1, \ldots, H$, find $\nu^{(t)}$ that minimizes

$$\mathcal{L}^{(t)}(Z, p; \theta) = \sum_{i=1}^{n} |\nu(Z_i^{(t)}) \neq Z_i^*|.$$

Step 3: if an improvement in $\sum_{t=1}^{H} \mathcal{L}^{(t)}(Z, p; \theta)$ has been achieved, go back to Step 2; otherwise, finish.

The `DATA-BASED` algorithm (Algorithm 5) developed in Rodríguez and Walker (2014) aims at defining a simple loss function by using a data-driven approach. Here, the intuition is that, if the MCMC has converged, the labels of the clusters may change, but the clusters should be the same from iteration to iteration. For example, the clusters may be characterized by their centres μ_g and dispersions σ_g, $g = 1, \ldots, G$. An estimate of μ_g and σ_g may be used as pivots for the relabelling procedure.

Algorithm 5: `DATA-BASED`

Start: find estimates m_g and s_g for cluster centres and dispersions, $g = 1, \ldots, G$.
Step 1: for $t = 1, \ldots, H$, find a permutation $\nu^{(t)}$ that minimizes

$$\mathcal{L}^{(t)}(m, s; \theta) = \sum_{k=1}^{G} \sum_{l=1}^{G} |Z_i^{(t)} = \nu(l)| \sum_i \left(\frac{y_i - m_k}{s_k} \right)^2.$$

Marin et al. (2005) and Marin and Robert (2007) propose the `PRA` algorithm (Algorithm 6), where the MCMC sample is permuted in order to minimize its distance from a pivot parameter vector, as the MAP estimate.

Algorithm 6: `PRA`

Start: choose a pivot parameter $\boldsymbol{\theta}^* = (\theta_g^*)$, $g = 1, \ldots, G$.
Step 1: for $t = 1, \ldots, H$, find a permutation $\nu^{(t)}$ that maximizes $\sum_{g=1}^{G} \theta_{\nu(g)}^{(t)} \theta_g^*$.

Finally, the `aic` method (namely Algorithm 7) imposes an artificial constraint on the MCMC sample, which is then permuted according to the ordering of a specific parameter. Note that this is the simplest approach for dealing with the label switching, but it is often not feasible to find a natural ordering for the parameters.

Algorithm 7: `aic`

Start: choose a component-specific parameter, for example the group means μ_g, $g = 1, \ldots, G$.
Step 1: for $t = 1, \ldots, H$ find the permutation $\nu^{(t)}$ such that $\mu_{\nu(1)}^{(t)} < \cdots < \mu_{\nu(G)}^{(t)}$.

The probabilistic relabelling approach first appears in Jasra (2006). Probabilistic relabelling methods do not minimize the distance of the permuted MCMC from a suitable loss function. In order to make inferences on component-specific parameters, a function of the parameters is estimated, which may also depend on an allocation vector $Z^{(t)} = (Z_1^{(t)}, \ldots, Z_n^{(t)})$ at MCMC iteration t ($t = 1, \ldots, H$):

$$u(Z, \theta) = H^{-1} \sum_{t=1}^{H} \sum_{\nu \in \mathcal{V}} p(\nu | Z^{(t)}, y) u(\nu(Z^{(t)}), \nu(\theta^{(t)})), \tag{9}$$

where $p(\nu | Z^{(t)}, y)$ is the posterior distribution for each permutation at MCMC iteration t. From a computational point of view, two tasks of this class of algorithms turn out to be crucial: the choice of $u(\cdot)$ and the estimation of $p(\nu | Z^{(t)}, y)$, also called the permutation distribution. For the first purpose, a natural choice suggested by Sperrin et al. (2010) is the identity function for the components of the parameters' vector, as $u(\boldsymbol{\pi}) = \boldsymbol{\pi}$ for the mixture weights. For the second task, they apply an EM-type algorithm, where the missing data are the permutations $\{\nu^{(t)}, t = 1, \ldots, H\}$ and these densities are estimated by conditioning only on the data the current parameter estimate for θ and the current allocation

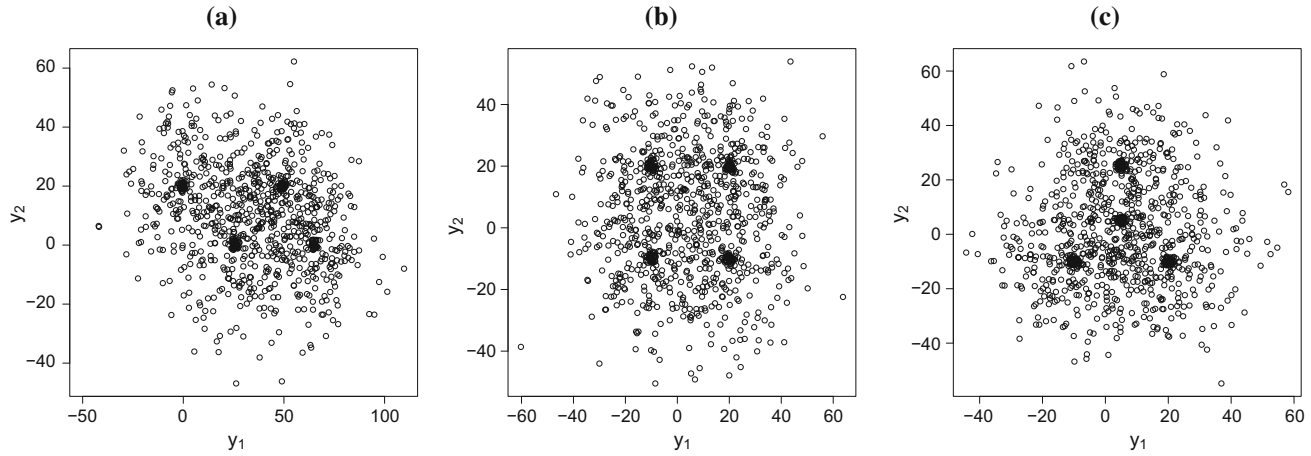

Fig. 1 Illustration of a simulated sample of size $n = 1000$ from model (10) with $G = 4$ components, according to **a** scenario A, **b** scenario B and **c** scenario C, with means as in Table 1

vector $Z^{(t)}$. That is, at the expectation step the permutation densities are estimated using the current parameter estimate for θ, and the estimate for θ is updated by using Eq. (9) at the maximization step.

The approach proposed by Puolamäki and Kaski (2009) is slightly different. They consider the matrix Q, an estimate of which may be obtained by maximizing a Bernoulli likelihood through the EM algorithm. In their method, the permutation density does not depend on the entire set of data, but only on Q. Once Q has been estimated, the authors compute the permutation distributions for each sample t, $t = 1, \ldots, H$, as

$$p(\nu|Z^{(t)}, Q) \propto \sum_{g=1}^{G} \frac{1}{G} \prod_{i=1}^{n} [q_{ig}]^{|Z_i^{(t)}=\nu(g)|} [1 - q_{ig}]^{1-|Z_i^{(t)}=\nu(g)|}.$$

$p(\nu|Z^{(t)}, Q)$ is used to obtain the distribution of (Z, θ) by plugging it into Eq. (9). The latter expression highlights the computational burden of this probabilistic approach, which requires averaging over all possible permutations of the MCMC sample. Despite the appeal of computing the posterior distribution for all possible permutations, such a method is suitable only when applied to simple cases, with a small enough n and G.

5 Evidence from a simulation study

The aim of this section is to investigate the behaviour of the proposed solution for dealing with label switching based on pivot identification in different simulated scenarios. Being interested in the ability of our method to detect the pivots, we need to define a challenging scenario in terms of both relabelling issue and pivotal choice.

For this purpose, we focus on data simulated from a mixture of non-equally weighted mixtures of bivariate Gaussian distributions with unequal covariance matrices, so that the generated components may result in overlapping clusters. Specifically, the simulation scheme consists of the following steps.

Table 1 Two-dimensional mean vectors of scenarios A, B and C adopted in the simulation study

	Scenario A	Scenario B	Scenario C
$\boldsymbol{\mu}_{1s}$	(25, 0)	(−10, −10)	(−10, −10)
$\boldsymbol{\mu}_{2s}$	(60, 0)	(20, −10)	(20, −10)
$\boldsymbol{\mu}_{3s}$	(0, 20)	(−10, 20)	(5, 5)
$\boldsymbol{\mu}_{4s}$	(50, 20)	(20, 20)	(5, 25)

Table 2 Estimated proportion k^* of relabelled iterations (see Sect. 3.1), over 100 macro-replications, for scenarios A, B and C

	(a)	(b)	(c)	(d)	(e)	(f)	MUS
A	0.475	0.993	0.993	0.124	0.506	0.993	0.313
B	0.519	0.998	0.998	0.101	0.707	0.998	0.995
C	0.139	0.300	0.507	0.079	0.267	0.368	0.374

(i) Simulate n values $Y_1, \ldots, Y_n$, from a mixture of mixtures of bivariate Gaussian distributions, where

$$(Y_i|Z_i = g) \sim \sum_{s=1}^{2} p_{gs}\, \mathcal{N}_2(\boldsymbol{\mu}_g, \Sigma_s). \tag{10}$$

That is, conditional on being in group $g \in \{1, \ldots, G\}$, Y_i is picked out from one of two possible Gaussian distributions with weights p_{gs}, means $\boldsymbol{\mu}_g$ and covariances Σ_s, $s = 1, 2$. The likelihood of the model is then

$$L(\boldsymbol{y}; \boldsymbol{\mu}, \boldsymbol{\pi}, \Sigma) = \prod_{i=1}^{n} \sum_{g=1}^{G} \pi_g \left(\sum_{s=1}^{2} p_{gs}\, \mathcal{N}_2(\boldsymbol{\mu}_g, \Sigma_s) \right).$$

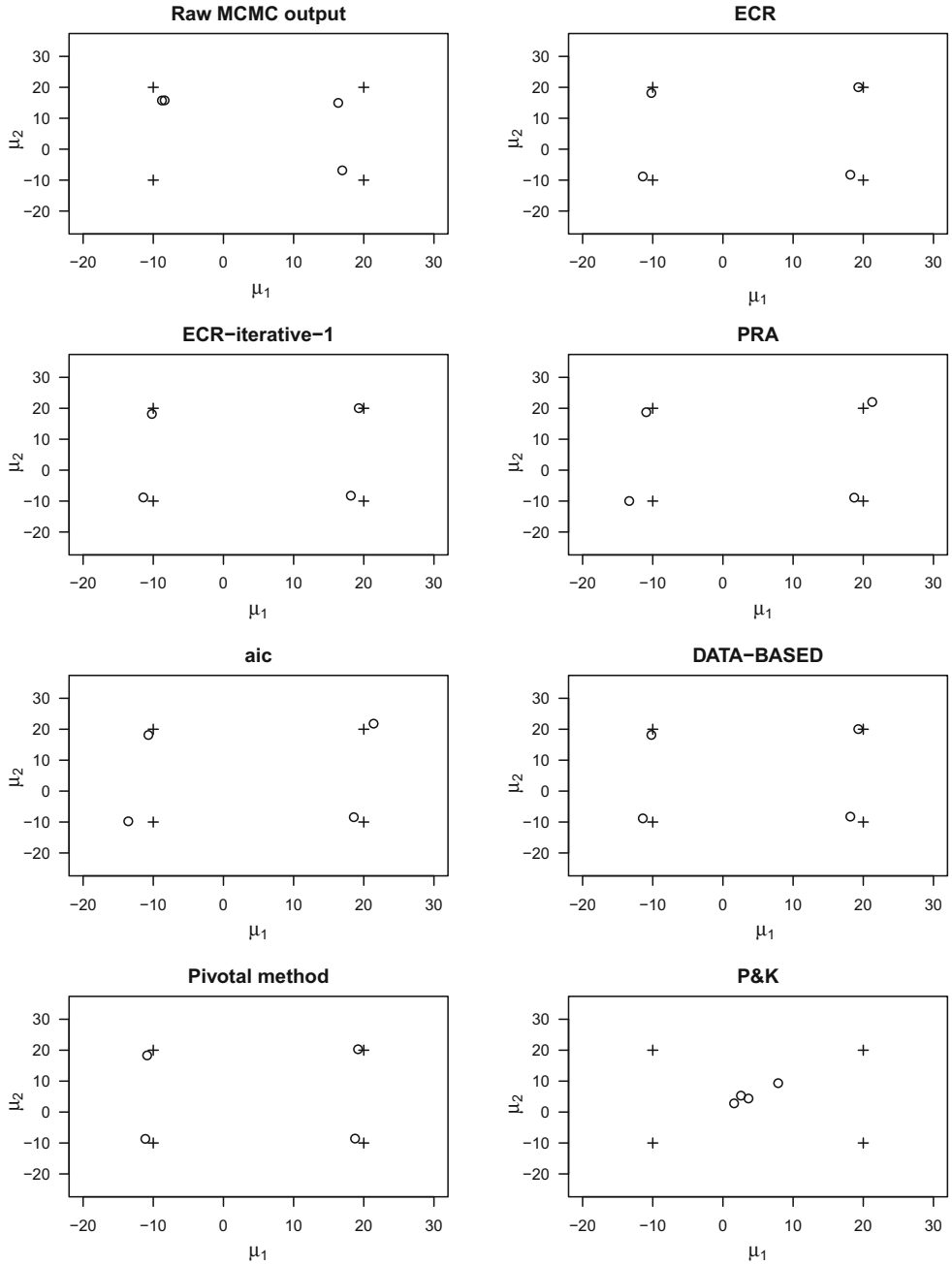

Fig. 2 *Crosses* are group means, and *circles* are the median values of relabelled estimates. Here, the pivotal method is implemented using agglomerative hierarchical clustering and MUS algorithm—see text

(ii) Obtain an MCMC sample that effectively explores all modes of the posterior distribution.
(iii) Estimate the $n \times n$ similarity matrix C with elements $c_{ij} = P(Z_i = Z_j|\mathcal{D})$, $i, j = 1, \ldots, n$, by Eq. (4).
(iv) Apply a suitable clustering technique based on the estimated dissimilarity matrix with elements $\hat{s}_{ij} = 1 - \hat{c}_{ij}$ and obtain a partition of the observations in G groups with units $\mathcal{G}_g$, $g = 1, \ldots, G$.
(v) Detect the pivots, one for each group, according to one criterion among the ones discussed before.
(vi) If necessary, discard those iterations of the chains belonging to $\mathcal{H}_G^*$ (see Sect. 3.1) and relabel the resulting chain with iterations in the restricted chain $\mathcal{H}_G - \mathcal{H}_G^*$ via Eqs. (5) and (6).

In the simulation study presented here, a sample size of $n = 1000$ and $G = 4$ components are considered. For $g = 1, \ldots, 4$, we set $\pi_g = 1/4$, $p_{g1} = 0.2$, $p_{g2} = 0.8$ and $\Sigma_1 = \mathbf{I}_2$, $\Sigma_2 = 200\,\mathbf{I}_2$, $\mathbf{I}_2$ being the 2×2 identity matrix. We generate simulated data from model (10) (see Fig. 1) accord-

Table 3 Mean squared error $\text{MSE}_g = (1/B)\sum_{j=1}^{B}||\boldsymbol{\mu}_g^{(j)} - \hat{\boldsymbol{\mu}}_g^{(j)}||$ of the median values of relabelled estimates of individual group means $\boldsymbol{\mu}_g$, $g = 1, \ldots, 4$ ($B = 100$)

	MSE_1	MSE_2	MSE_3	MSE_4	Overall error
Scenario A					
Pivotal method					
(b)	13.7064	1.6104	1.9814	9.1846	6.6207
(c)	13.7794	1.6723	1.8979	9.2897	6.6598
(e)	14.0215	1.6619	1.9951	11.2910	7.2424
(f)	13.7301	1.6264	1.8889	9.2900	6.6338
MUS	12.5787	1.5531	1.7919	9.6220	6.3864
Other methods					
`aic`	1.6657	1.6074	2.0269	2.1553	1.8638
`DATA-BASED`	13.6077	1.6779	1.9031	8.8071	6.4985
`ECR`	13.6281	1.6589	2.0588	9.0821	6.6069
`ECR-iterative-1`	13.6403	1.6605	1.9015	8.8085	6.5027
`PRA`	1.6733	1.6096	2.0459	2.1366	1.8660
P&K	25.5940	15.5229	15.1522	27.2411	20.8775
Scenario B					
Pivotal method					
(b)	1.4123	1.6005	1.5737	1.5419	1.5321
(c)	1.4121	1.5982	1.6192	1.5420	1.5429
(e)	1.4096	1.5961	1.5729	1.5403	1.5297
(f)	1.4127	1.6003	1.5736	1.5417	1.5321
MUS	1.4070	1.5877	1.5728	1.5437	1.5278
Other methods					
`aic`	2.0131	2.1765	2.0098	2.0270	2.0566
`DATA-BASED`	1.4128	1.5985	1.5720	1.5428	1.5315
`ECR`	1.4112	1.5967	1.5700	1.5417	1.5299
`ECR-iterative-1`	1.4129	1.5984	1.5717	1.5429	1.5314
`PRA`	1.8782	2.0972	1.9197	1.9259	1.9552
P&K	18.4657	18.6185	18.6796	19.0404	18.7010
Scenario C					
Pivotal method					
(b)	6.9196	7.8994	8.7700	14.1766	9.4414
(c)	7.1992	7.1643	9.4728	15.2713	9.7769
(e)	7.7730	9.1701	9.1987	16.4153	10.6393
(f)	7.6160	7.1054	10.2073	13.2589	9.5469
MUS	6.7458	7.5579	9.7924	14.8356	9.7329
Other methods					
`aic`	3.2628	3.5866	10.2319	3.8349	5.2290
`DATA-BASED`	5.7496	5.9469	8.4893	8.7063	7.2230
`ECR`	6.1148	6.5128	8.4971	8.8926	7.5043
`ECR-iterative-1`	6.4891	6.7234	8.4472	9.3649	7.7561
`PRA`	3.1679	3.4210	10.3873	2.9754	4.9879
P&K	17.5726	16.8717	3.4988	20.2620	14.5512

The last column contains the values of the overall error given by $(1/G)\sum_g \text{MSE}_g$

ing to the three scenarios with means reported in Table 1, and obtain an MCMC sample of $H = 3000$ iterations.

We proceed according to points (i)–(vi). As a remark, two different clustering strategies are applied to the dissimilarities $\hat{s}_{ij}$ in order to obtain G clusters of observations, namely agglomerative and partitioning hierarchical clustering. Both methods only require a distance or a dissimilarity matrix as input and return a set of nested clusters that are organized

as a tree structure. We note that the two algorithms provide very similar clusters; thus, we observe that the choice of the clustering technique does not affect the performance of the relabelling procedure. Therefore, for the sake of illustration, we restrict to agglomerative hierarchical clustering, where the so-called *complete linkage* is adopted as a criterion for the computation of the dissimilarity between two clusters, since it is less susceptible to noise and outliers.

Table 2 shows the proportions k^* of relabelled iterations based on 100 simulated samples according to the three scenarios illustrated in Fig. 1, for criteria (a)–(f) in Eqs. (7) and (8) and the MUS algorithm. As can be noticed, methods labelled (b), (c) and (f) register very high values of chain proportions (less than 1% of the iterations is discarded) for both scenarios A and B. Concerning the third scenario (C), criteria (c), (f) and the MUS algorithm yield better results than the others. Method (d) seems to have the worst performance regardless of the considered setting; in particular, in scenario C the algorithm discards about 92% of the original iterations. The fact that the third simulated scenario shows globally less satisfactory results is not surprising. In fact, the means are so close to each other that the clustering algorithm may fail in recognizing the true data partition, thus impairing the quality of the choice of the pivotal units. Additional figures and comments to these results not shown here are available in the "Supplementary Material" file.

In order to compare the performance in estimating the means of the mixture components of the proposed methodology with other relabelling algorithms, we consider the Puolamäki and Kaski (P&K) procedure and the following methods briefly reviewed in Sect. 4: `ECR`, `ECR-iterative-1`, `DATA-BASED`, `PRA` and `aic`. As discussed in Papastamoulis (2016), the need to store the array p of classification probabilities makes Stephens' method and the `ECR-iterative-2` algorithm demanding in terms of computational burden; for this reason, we do not include these procedures in the simulation study. Figure 2 displays the median estimates of relabelled group means (scenario B) according to the pivotal method and the six relabelling algorithms mentioned above. As can be seen, our relabelling procedure seems to provide quite accurate estimates of the group means. Similar results are achieved by `ECR-iterative-1`, `ECR` and `DATA-BASED`, while the algorithm by Puolamäki and Kaski does not appear to yield reliable estimates for the group means. `PRA` and `aic` perform dramatically worse than the other deterministic algorithms and our proposal in scenario B, even if they are very efficient in scenarios A and C. As already mentioned, `aic` may be not considered as a general relabelling algorithm due to a prior imposition of an ordering constraint to a specific parameters' vector. Indeed, it is not always possible to specify a general geometrical ordering; an example in two dimensions is given in Fig. 2. Concerning the `PRA` method, a MAP estimate for the parameters' vector is required, which means optimizing a bivariate mixture log-likelihood for the determination of a MCMC iteration that corresponds to a high density area. Due to the dimensions of our simulation study, this appeared to be computationally demanding and we chose as MAP estimate an arbitrary MCMC iteration. Hence, we consider the `PRA` algorithm to be unfeasible when the dimension of the parameters' vector is large.

Table 4 CPU time (in seconds) for different methods and for scenarios A, B and C

Method	A	B	C
`ECR`	2.84	2.84	2.84
`ECR-iterative-1`	16.86	16.97	16.83
`PRA`	0.22	0.22	0.20
`aic`	0.03	0.05	0.03
`DATA-BASED`	8.22	8.20	8.11
Pivotal	4.32	4.36	4.00

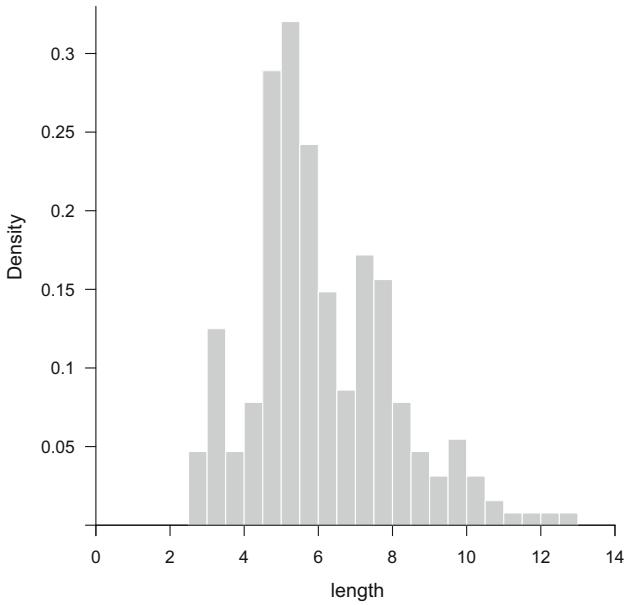

Fig. 3 Histogram of fishery data. Values on x-axis are snapper length measurements

Table 3 reports the mean squared errors for the median values of relabelled estimates of individual group means (computed from $B = 100$ macro-replications) corresponding to the pivotal methods listed in Sect. 3. For each setting, Table 3 also displays a measure of the global error (so-called overall error) in the estimation of the four group means, computed by averaging the mean squared errors of the single components. Motivated by the estimated proportions k^* in Table 2, we only consider criteria (b), (c), (e) and (f), which give overall good results, and the MUS algorithm which, in some cases, outperforms all the others. In summary, all meth-

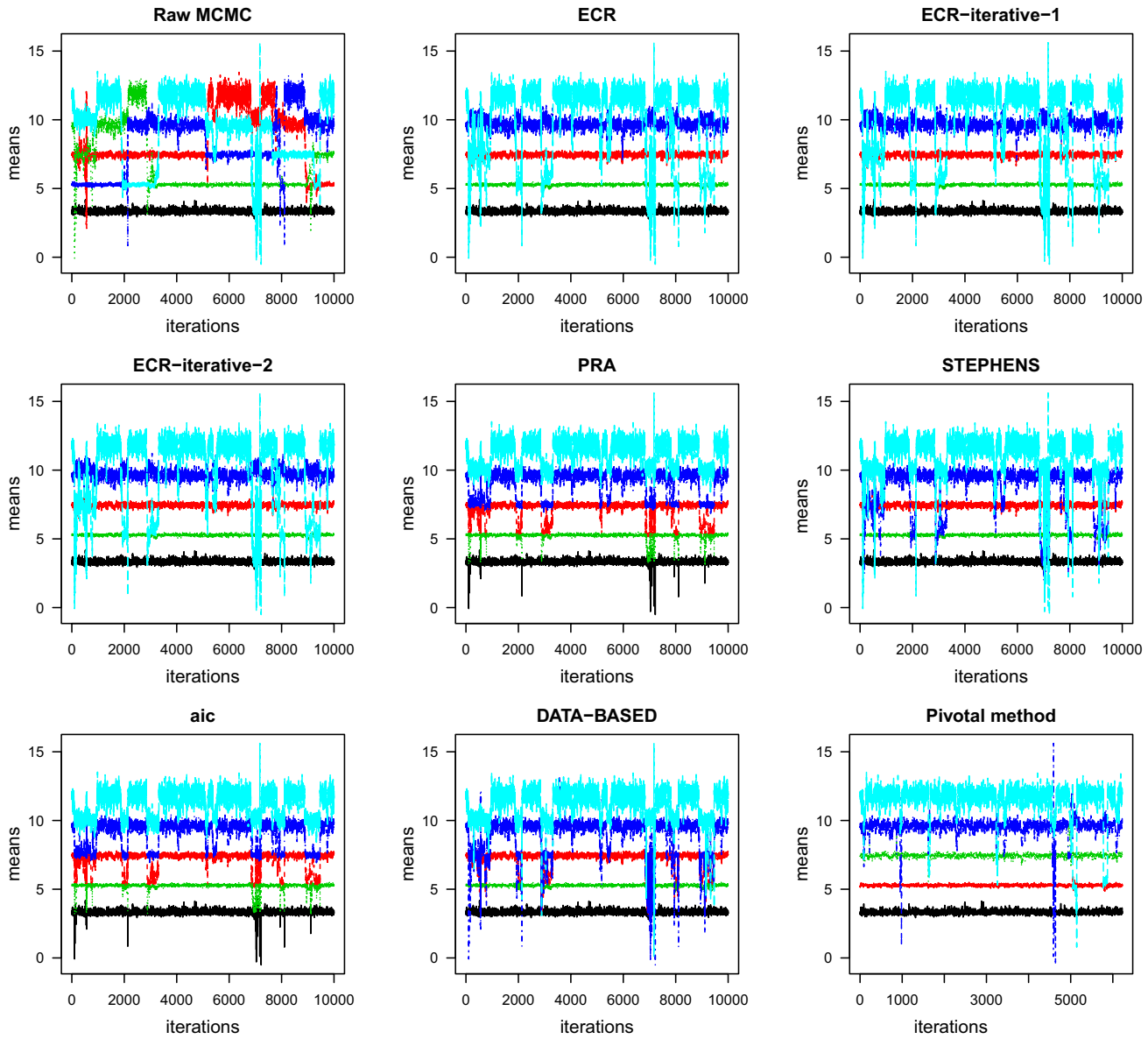

Fig. 4 MCMC traces for fishery data. Raw MCMC sample (*top left*) for μ_g, $g = 1, \ldots, 5$. Reordered MCMC samples according to `ECR`, `ECR-iterative-1`, `ECR-iterative-2`, `PRA`, `STEPHENS`, `aic`, `DATA-BASED` and pivotal method, according to pivotal criterion (f)—see text

ods perform better than P&K under all scenarios. `PRA` and `aic` are the best performers in two out of three scenarios, but, as already noted, they are unfeasible in many situations. The proposed pivotal method performs similarly to competitors in scenarios A and B, while in scenario C the pivotal method implies a slightly larger MSE.

In Table 4, the required CPU times for execution of the different relabelling methods are reported (all computations were performed using R version 3.3.0 on an Intel (R) Core (TM) i7-4790 machine with 3.60 GHz). It is worth mentioning that only the single relabelling step is taken into account (for our method, this means considering the relabelling task in Eqs. (5)–(6) solely). Preliminary steps, such as the log-likelihood optimization in the `PRA` method or the selection of the pivotal allocation vector in `ECR`, are not considered. However, from the discussion in Papastamoulis (2016, Sect. 4), we can argue that for some of the reviewed methods such preliminary computations may be demanding. As for pivotal method, the identification of pivots required for the relabelling task turns out to be not particularly time-consuming, except for the MUS procedure. Finally, note that we do not report the CPU times for P&K algorithm (which were larger than 200), due to its inefficiency and high computational burden.

Table 5 CPU time (in seconds) for different methods applied to fishery dataset

Method	CPU time
ECR	8.66
ECR-iterative-1	60.83
ECR-iterative-2	28.39
PRA	3.96
STEPHENS	344.50
aic	0.08
DATA-BASED	22.68
Pivotal	9.57

Ruling out the comparison of PRA and aic for the reasons outlined above, as far as the other methods are concerned, ECR and our pivotal methods appear to have some advantage in terms of computational time with comparable precision.

6 A case study

The fishery dataset, originally taken from Titterington et al. (1985) and used by Papastamoulis (2016) for comparing different relabelling procedures, consists of $n = 256$ snapper length measurements. In Fig. 3, the histogram of the lengths is shown. The proposed methodology is applied to this dataset, and the results are compared with the five algorithms already considered in the previous section available in the **label.switching** package; additionally, ECR-iterative-2 and STEPHENS are included in our study. We use a Gaussian mixture with $G = 5$ components as suggested by Papastamoulis (2016), that is:

$$y_i \sim \sum_{g=1}^{G} p_g \mathcal{N}(\mu_g, \sigma_g^2), \qquad i = 1, \ldots, n. \tag{11}$$

We set up a Gibbs sampling through the **bayesmix** R package (Grün 2011), with $H = 11{,}000$ iterations and a burn-in period of 1000.

In Fig. 4, the raw MCMC sample and the reordered MCMC samples for μ_g, $g = 1, \ldots, 5$, for different methods are shown (the label.switching function of the same package is used to reorder the obtained chains according to the resulting permutations). Despite an ordering constraint for components' means (the priors are chosen according to the independence option, which favours a natural ordering of the means), label switching occurs, and the raw sampler is unable to yield useful means estimates for the single components (see the top left panel of Fig. 4).

In general, we can see that the procedures from the **label.switching** package seem to perform similarly. In particular, for the greatest mean (light blue trace) there is a global tendency of switching. We note that the same happens also for the second mean (blue trace) in most procedures, especially for DATA-BASED, PRA and aic. Our pivotal method seems to work better in isolating the five high-posterior density regions. We recall that the reordering for our method is explained by Eq. (5).

Table 5 reports the CPU times (in seconds) for the compared procedures. As can be seen, aic and PRA are the fastest methods. We observe that ECR is slightly less intensive than our method, while the pivotal algorithm is faster than four relabelling procedures.

7 Concluding remarks

We propose a simple procedure for dealing with label switching in Bayesian mixture models, based on the identification of as many pivots as mixtures components, used for relabelling the resulting MCMC chains. The main novelty of our contribution is to provide some useful indications of how to choose the pivots, since, as mentioned in Sect. 3, the idea of solving the relabelling issue by fixing the groups for some units is not new. We suggest and evaluate alternative criteria based on a suitably defined similarity matrix obtained through the MCMC sample.

The proposed pivotal method is quite easy to implement and is computationally less demanding than other relabelling methods, since it does not involve a maximization/minimization step at each iteration but only requires a permutation of the labels induced by the pivots membership.

The simulation study presented, although limited, shows that the proposed solution yields overall good performances. A case study on a real dataset is also presented, showing the advantage of using the proposed method. Moreover, an evaluation of the computational complexity of our algorithm compared with other competing procedures (for instance, those available in the **label.switching** R package) confirms that our methodology represents a valid approach to dealing with the label switching problem.

References

Celeux, G., Hurn, M., Robert, C.P.: Computational and inferential difficulties with mixture posterior distributions. J. Am. Stat. Assoc. **95**(451), 957–970 (2000)

Chung, H., Loken, E., Schafer, J.L.: Difficulties in drawing inferences with finite-mixture models. Am. Stat. **58**(2), 152–158 (2004)

Egidi, L., Pappadà, R., Pauli, F., Torelli, N.: Maxima units search (MUS) algorithm: methodology and applications (2016). ArXiv e-prints arXiv:1611.01069

Grün, B.: Bayesmix: bayesian mixture models with JAGS. R package version 0.7-2. http://CRAN.R-project.org/package=bayesmix (2011)

Jasra, A.: Bayesian inference for mixture models via Monte Carlo computation. Ph.D. thesis, Imperial College London (University of London) (2006)

Marin, J.M., Robert, C.P.: Bayesian Core: A Practical Approach to Computational Bayesian Statistics. Springer, New York (2007)

Marin, J.M., Mengersen, K., Robert, C.P.: Bayesian modelling and inference on mixtures of distributions. Handb. Stat. **25**, 459–507 (2005)

McLachlan, J., Peel, D.: Finite Mixture Models. Wiley, New York (2000)

Papastamoulis, P.: Label.switching: an R package for dealing with the label switching problem in MCMC outputs. J. Stat. Soft. **69**(1), 1–24 (2016). doi:10.18637/jss.v069.c01

Papastamoulis, P., Iliopoulos, G.: An artificial allocations based solution to the label switching problem in Bayesian analysis of mixtures of distributions. J. Comput. Graph. Stat. **19**(2), 313–331 (2010)

Puolamäki, K., Kaski, S.: Bayesian solutions to the label switching problem. In: Adams, N.M., Robardet, C., Siebes, A., Boulicaut, J.-F. (Eds.) Advances in Intelligent Data Analysis VIII 8th International Symposium on Intelligent Data Analysis, IDA 2009, Lyon, France, pp. 381–392. Springer, Berlin (2009). http://www.springer.com/gp/book/9783642039140

Rodríguez, C.E., Walker, S.G.: Label switching in Bayesian mixture models: deterministic relabeling strategies. J. Comput. Graph. Stat. **23**(1), 25–45 (2014)

Sperrin, M., Jaki, T., Wit, E.: Probabilistic relabelling strategies for the label switching problem in Bayesian mixture models. Stat. Comput. **20**(3), 357–366 (2010)

Stephens, M.: Dealing with label switching in mixture models. J. R. Stat. S.: Ser. B (Stat. Methodol.) **62**(4), 795–809 (2000)

Titterington, D.M., Smith, A.F., Makov, U.E.: Statistical Analysis of Finite Mixture Distributions. Wiley, New York (1985)

Yao, W., Li, L.: An online Bayesian mixture labelling method by minimizing deviance of classification probabilities to reference labels. J. Stat. Comput. Simul. **84**(2), 310–323 (2014)