Bayesian clustering for continuous-time hidden Markov models

Yu LUO\(^1\)*, David A. STEPHENS\(^2\)*, and David L. BUCKERIDGE\(^3\)

\(^1\)Department of Mathematics, King’s College London, London, U.K.
\(^2\)Department of Mathematics and Statistics, McGill University, Montreal, Canada
\(^3\)Department of Epidemiology, Biostatistics and Occupational Health, McGill University, Montreal, Canada

Key words and phrases: Continuous-time hidden Markov models; mixture models; model-based clustering; nonparametric Bayesian inference; reversible-jump MCMC; split–merge proposal.

MSC 2020: Primary 60J22, 62F15; secondary 62H30.

Abstract: We develop clustering procedures for longitudinal trajectories based on a continuous-time hidden Markov model (CTHMM) and a generalized linear observation model. Specifically, in this article we carry out finite and infinite mixture model-based clustering for a CTHMM and achieve inference using Markov chain Monte Carlo (MCMC). For a finite mixture model with a prior on the number of components, we implement reversible-jump MCMC to facilitate the trans-dimensional move between models with different numbers of clusters. For a Dirichlet process mixture model, we utilize restricted Gibbs sampling split–merge proposals to improve the performance of the MCMC algorithm. We apply our proposed algorithms to simulated data as well as a real-data example, and the results demonstrate the desired performance of the new sampler.

1. INTRODUCTION

Continuous-time Markov processes on a finite state space have been widely used to represent longitudinal data, especially if the times between successive observations are irregular. Likelihood inference for the infinitesimal generator of a continuous-time Markov jump process was studied by Billingsley (1961). However, in reality the process is observed only at discrete time points,
and inference for the generator of the process becomes difficult. This problem arises in a variety of practical settings ranging from public health surveillance (Luo et al., 2021) to molecular dynamics (Hobolth & Stone, 2009). Inference for discretely observed continuous-time Markov processes has been explored by Bladt & Sørensen (2005) in both the likelihood and Bayesian frameworks.

In most settings, either the observed data are not direct observations of the Markov process, or the process is observed with measurement error. In such cases, we may consider a latent process as an unobserved “trajectory” recording the true but unobserved state of nature. In this setting, a hidden Markov model (HMM) is suitable, and it is assumed that the Markov property is imposed on the unobserved process. There is a broad interest in the application of continuous-time HMMs (CTHMMs) (see, e.g., Jackson et al., 2003; Lange et al., 2015), with the majority of research devoted to frequentist approaches; however, Luo et al. (2021) and Williams et al. (2020) recently constructed Bayesian CTHMMs using different missing data likelihood formulations for the underlying Markov chain.

If the cohort is assumed homogeneous with respect to the stochastic properties of the latent model, a common continuous-time model may be fitted to the entire dataset. However, it is plausible that the study base comprises different subcohorts that have distinct stochastic properties. One possibility for dealing with potential heterogeneity is to add random effects in the outcome model, taking into account individual- and outcome-specific effects due to unobserved heterogeneity (Maruotti & Rydén, 2009; Zucchini, MacDonald & Langrock, 2017); however, the computation is demanding, as possible integration is required for continuous-valued random effects, especially for a large sample size. Many of the other methods may be classified as model-based clustering procedures, where clustering is achieved by adopting parametric likelihood- or density-based calculations. Clustering longitudinal data using discrete HMMs has been proposed by Panuccio, Bicego & Murino (2002) and Bicego, Murino & Figueiredo (2003), who used model-based and similarity-based approaches, respectively, but these methods are suitable only for small-scale problems because calculating the distance matrix based on the likelihood of the HMMs is computationally intensive. Smyth (1997) proposed a likelihood-based distance matrix of observed sequences using a hierarchical clustering approach, and Jebara, Song & Thadani (2007) used a spectral clustering method with a probability kernel. Crayen et al. (2012) applied a mixture latent Markov model for clustering dynamics in mood regulation. However, in those methods the number of clusters had to be prespecified. The primary focus of this article is to identify meaningful subgroups of latent trajectories in a large dataset where each trajectory is represented by a CTHMM, allowing the number of clusters to be inferred during the analysis. This problem has not been studied previously, since Bayesian inference for CTHMMs was developed only recently. In this article, we specifically consider finite and infinite process mixture model-based clustering approaches for longitudinal observations based on CTHMMs with an unknown number of components.

In Section 2, we review the development of a CTHMM/generalized linear model (CTHMM-GLM) for nonequidistant longitudinal data. Section 3 discusses model-based clustering approaches; Section 3.1 compares the prior distribution of finite and infinite mixture models that are commonly used for clustering models. A reversible-jump MCMC approach to clustering under finite mixture models is described in Section 4. Section 5 presents Dirichlet process mixture model-based clustering implemented using restricted Gibbs sampling split–merge proposals. In Sections 6 and 7, we report the analysis of both simulated data and a case study involving chronic obstructive pulmonary disease (COPD) to illustrate the performance of our model-based clustering inference. We discuss possible extensions of our methods in Section 8.
2. CTHMM-GLM MODEL

We presume that for a single individual, a sequence \( \{O_1, \ldots, O_T\} \) of health-status-related variables is observed at time points \( \{\tau_1, \ldots, \tau_T\} \). A latent process \( \{X_s\} \) \((s \in \mathbb{R}^+)\), representing the health status for a condition of interest, is assumed to be a continuous-time Markov chain (CTMC) with parameters \((\pi, Q)\), where \(\pi\) is the initial distribution, and \(Q\) is the infinitesimal generator, taking values on the finite state space \(\{1, 2, \ldots, K\}\). The assumption of a finite state space for the latent process is justified in many practical instances, such as the case study discussed in Luo et al. (2021), and corresponds to conceptual stages in the progression of the underlying disease process. However, such discrete models are undoubtedly an approximation, in general. The observation process \(O|X_s, B\), parameterized by a coefficient matrix \(B\) containing all GLM coefficients for each latent state, i.e., \(B = (\beta_{d,k}), d = 1, \ldots, D\) and \(k = 1, \ldots, K\), is presumed to follow an exponential family, with

\[
f(O_t|X_{\tau_t} = k) = \exp \left[ \{O_t \theta_{t,k} - b(\theta_{t,k})\} / \phi^2 + c(O_t, \phi) \right].
\] (1)

A GLM with corresponding link function specified as \(g(u(\theta_{t,k})) = Z_t^T \beta_k\) can be incorporated if there are explanatory variables \(Z \in \mathbb{R}^D\), where \(u(\theta_{t,k}) = E(O_t|X_{\tau_t} = k)\), and \(\beta_k\) is a coefficient vector for state \(k\). Let \(S_t = (S_{t,1}, \ldots, S_{t,K})^T\) be an indicator random vector with \(S_{t,k} = 1\) if \(X_{\tau_t} = k\) and 0 otherwise. We can rewrite the linear predictor into a matrix form as \(g(u(\theta_{t,k})) = Z_t^T BS_t\).

Luo et al. (2021) developed likelihood and Bayesian inference procedures for this model using the expectation–maximization (EM) algorithm and MCMC approaches. In that formulation, observations are indexed using an integer index \((i.e., O_t)\) and the latent process is indexed using a continuous-valued index \((i.e., X_{\tau_t})\). In this article, we assume that the measurement process itself \((i.e., the\ collection\ of\ times\ \tau_{t}, t = 1, \ldots, T)\) is not informative about the system either in its hidden or observed components. The following diagram provides a schematic of the presumed data generating structure for one subject.

In this diagram, outcome \(O_s\) is observed at \(s = \tau_t, t = 1, \ldots, T\). The underlying trajectory determines that the subject begins in state 1, then progresses through the states 2, 3, 2, and so on, until finally reverting to state 1 after the final observation. Observations are made at times that do not coincide with the transition times between states, and \(\Delta_t = \tau_{t+1} - \tau_t\) records the interval between observations \(O_{t+1}\) and \(O_t\). The variable \(C\) represents the cluster label of this individual. In the case involving a single cluster, \(C\) will be the same across all the subjects.

This model, developed in Luo et al. (2021) (the corresponding Supplementary Material provides full details regarding the likelihood construction and the Bayesian hierarchical model), is a parametric one for which the likelihood is reasonably complex, albeit one that is simplified in its representation using a latent process. The model is presumed to apply to all individuals in the study, who constitute a random sample from the target population. In this article, we consider an extension to allow for systematic heterogeneity to be exhibited by subpopulations of subjects. If there are \(N\) subjects, let \(O_{n,t} = (t = 1, \ldots, T_n)\) be the \(r\)th observation for subject \(n\)
with the associated observation time $\tau_{n,t}$, with the corresponding hidden state $X_{n,\tau_{n,t}}$; then $O_n = \{O_{n,1}, \ldots, O_{n,T_n}\}$ and $X_n = \{X_{n,\tau_{n,1}}, \ldots, X_{n,\tau_{n,T_n}}\}$ represent the collection of data for subject $n$. The variables $O = \{O_{n,t}\}$ and $X = \{X_{n,\tau_{n,t}}\}$ for $n = 1, \ldots, N, t = 1, \ldots, T_n$ represent the entire data for $N$ subjects. To illustrate the challenging nature of clustering trajectories, Figure 1 shows simulations from three groups with different parameters (taken from the Gaussian example in Section 6.1); typically, Cluster 1 exhibits less variation, but it is not easy to distinguish between Clusters 2 and 3. Since differences among the clusters are also originating from the underlying continuous-time Markov process $X_s$, it is necessary to extend the basic model to allow for heterogeneity in observation and latent processes.

3. MODEL-BASED CLUSTERING

The principal contribution of this article is to develop model-based clustering for data presumed to be generated by the complex model introduced in Section 2; specifically, we develop clustering approaches for the latent trajectories based on the observed data and the presumed hidden Markov structure.

Model-based clustering is typically achieved via parametric likelihood- or density-based calculations, with the number of clusters selected using information criteria, such as Akaike or Bayesian information criterion (AIC or BIC) (Fraley & Raftery, 1998). This approach was explored extensively by Luo (2019) in the context of CTHMMs for modelling health trajectories. However, in such calculations the number of clusters must be prespecified. Bayesian
model-determination approaches have been a longstanding focus of interest in Bayesian inference (see, e.g., Carlin & Chib, 1995; Green, 1995; Godsill, 2001). Two approaches are typically adopted to address this issue. First, reversible-jump Markov chain Monte Carlo (MCMC) (Green, 1995) exploits trans-dimensional Metropolis–Hastings (MH) moves, allowing movement across parameter spaces of different dimensions. Second, Bayesian nonparametric procedures based on the Dirichlet process are also widely used—these models are often identified as infinite mixture models, where a prior is placed on the space of discrete random measures, and where the models are limiting versions of exchangeable finite mixture models. Dirichlet process models are now widely used, with implementation facilitated via MCMC; key relevant references in a well-established literature include Escobar & West (1995), MacEachern & Müller (1998), Neal (2000), Ishwaran & James (2001), and Jain & Neal (2004).

The principal challenge in our setting is that the core model on which the clustering will be based is driven by the unobserved continuous-time trajectory represented by \( \{ X_s \} \). We meet this challenge by implementing MCMC algorithms based on the complete data likelihood introduced in Section 2.

3.1. Clustering via Mixture Models

The general principle behind model-based clustering is to cluster individuals based on the component model parameters that determine the mixture form. The basic formulation of the model envisages that the population is composed of distinct subpopulations each with distinct stochastic properties. In the case of the CTHMM formulation, this corresponds to each subpopulation having a potentially different parameter \( \Theta = (\pi, Q, B) \). The estimated parameter for each cluster will provide a subpopulation-level summary and allow comparisons within and between clusters.

Let \( C_n \) be the cluster membership indicator for individual \( n \) with cluster-specific model parameters \( \Theta_{C_n} = \{ \pi_{C_n}, Q_{C_n}, B_{C_n} \} \). The likelihood contribution for this individual is \( \mathcal{L} \left( O_n, X_n | \Theta_{C_n} \right) \). With \( M \) denoting the number of mixture components, a prior probability, \( p_0 (C | M) \), would be assigned to each cluster membership partition \( C = \{ C_1, \ldots, C_N \} \), resulting in the posterior form

\[
 p ( C, \Theta | O, X, M ) \propto p_0 ( C | M ) \prod_{n=1}^{N} \mathcal{L} \left( O_n, X_n | \Theta_{C_n} \right),
\]

where \( C \) is a specific partition of the \( N \) individuals into \( M^* \) nonempty clusters. For practical inference, the key is whether the form of \( \mathcal{L} \left( O_n, X_n | \Theta_{C_n} \right) \) can be easily computed, and the cluster-specific parameter \( \Theta_{C_n} \) can be integrated out. This would greatly simplify the MCMC algorithm. In our model specification, conditional on a proposed clustering model, we will adopt the formulation of Luo et al. (2021) for the CTHMM, so the differences in inference will be driven by the specific clustering model adopted. In the next sections, we compare aspects of the finite mixture and Dirichlet process mixture models, noting their similarities and where they differ.

3.2. Mixture of Finite Mixtures

Conventionally, in a finite mixture model, the number of clusters is determined by fitting an \( M \)-component mixture model with various choices for \( M \) and then using information criteria to choose among these choices. A natural extension is to regard the number of components, \( M \), as a random variable, specifying the prior as \( M \sim p_0 (M) \), a mass function on \( \{ 1, 2, 3, \ldots \} \), resulting in the joint prior distribution of the form \( p_0 (C, M) = p_0 (C|M) p_0 (M) \). However, there is a crucial distinction between the number of components \( M \) in the mixture model and the
number of clusters $M^*$ in the data, which is defined as the number of components used to generate the observed data, or the number of “filled” mixture components. By specifying a prior on the number of components $M$, we implicitly place a prior on $M^*$. We typically specify mixture weights
\[
\pi_1, \ldots, \pi_M | M \sim \text{Dirichlet} (\delta, \ldots, \delta)
\]
with $\delta = 1$, making the weight distribution uniform, and then draw cluster labels $C$ independently from the multinomial distribution with $P ( C_n = m | M, \pi_1, \ldots, \pi_M ) = \pi_m$. The resulting conditional distribution of $C$, given that there are $M$ components, has the form
\[
p_0 ( C | M, \delta ) = \frac{\Gamma ( M\delta )}{\Gamma ( N + M\delta )} \prod_{m=1}^{M} \frac{\Gamma ( N_m + \delta )}{\Gamma ( \delta )},
\]
where $N_m$ is the number of subjects placed in component $m$ by this procedure.

In Section 4, we will explicitly introduce this model in the context of CTHMM and the corresponding inference based on reversible-jump MCMC (Green, 1995) with an efficient proposal.

3.3. Dirichlet Mixture Models

In the finite mixture formulation, allowing $M \rightarrow \infty$, given a finite sample size $N$, will still yield a valid prior distribution, albeit with some $N_m$’s equal zero. This leads to another choice of a prior distribution, namely the Dirichlet process (Ferguson, 1973; Antoniak, 1974; Lo, 1984). Specifically, for a Dirichlet process prior, the prior distribution of the partition has the form
\[
p_0 ( \mathcal{Z} | \alpha ) = \frac{\alpha^M \Gamma ( \alpha )}{\Gamma ( N + \alpha )} \prod_{m=1}^{M} \Gamma ( N_m ).
\]
The probability mass function of $M$ and the conditional distribution of $C$ given $M$ are (Green & Richardson, 2001)
\[
p_0 ( M ) = | S_{N, M} | \frac{\alpha^M \Gamma ( \alpha )}{\Gamma ( N + \alpha )}, \quad p_0 ( \mathcal{Z} | M ) = \frac{\prod_{m=1}^{M} \Gamma ( N_m )}{| S_{N, M} |},
\]
where $| S_{N, M} | = \sum_{\mathcal{Z}} \prod_{m=1}^{M} \Gamma ( N_m )$ denotes the absolute value of a Stirling number of the first kind, the number of ways of partitioning $N$ items into $M$ nonempty subsets.

In Section 5, we will use the Dirichlet process prior in the context of CTHMMs. This representation marginalizes out the parameters of the model, which is particularly useful in terms of the computation. Specifically, a split–merge MCMC for this model will be introduced to facilitate the computation of updating cluster memberships.

3.4. Calibrating the Prior on the Number of Clusters

In order to make our analysis on the cluster structures under the finite and Dirichlet mixture models as comparable as possible, we aim to match their prior specification on the number of clusters. Miller & Harrison (2018) discussed similarities and differences between the Dirichlet mixture model and a mixture of finite mixtures (MFMs) in terms of the prior distribution on the number of components. For MFMs, they showed that when $N \rightarrow \infty$, the distribution of the
number of clusters $M^*$ exhibits behaviour similar to that of $M$, the number of components, and under the posterior they also behave similarly. In addition, these researchers also illustrated that this prior for MFMs is a simple exchangeable partition distribution that closely parallels that of the Dirichlet process, with the distribution of the random partition

$$p_0(\mathcal{Z} | \delta) = V_N(M^*) \prod_{m=1}^{M} \delta^{N_m},$$

where

$$V_N(M^*) = \sum_{M=1}^{\infty} \frac{M (M-1) \cdots (M-M^*+1)}{\delta M (\delta M + 1) \cdots (\delta M + N - 1)} p_0(M), \quad \mathcal{Z} = \{ E_m : |E_m| > 0 \}$$

and $E_m = \{ n : C_n = m \}$ for $m \in \{1, 2, \ldots \}$. Therefore, posterior samples can be generated via a direct application of MCMC for the Dirichlet process mixture, and the only difference is that the new element is placed in an existing cluster $m$ with probability proportional to $(N_m + \delta)$ or a new cluster with probability proportional to $\delta V_N(M^* + 1)/V_N(M^*)$. Therefore, we can also directly apply the algorithm developed for Dirichlet mixture models to MFMs by replacing $\alpha$ with $\delta V_N(M^* + 1)/V_N(M^*)$ and $N_m$ with $N_m + \delta$.

The Dirichlet process prior on $M$, given by the first expression in Equation (4), can be computed directly for any $N$. This is numerically challenging when $N$ is large; so rather than compute its value exactly, we adopt a Monte Carlo strategy, as sampling the number of clusters is straightforward: under this prior, $M^* = \sum_{i=1}^{N} 1 \left\{ U_i > \frac{\alpha}{\alpha + i - 1} \right\}$, where $U_1, \ldots, U_N$ are independent $\text{Uniform}(0, 1)$ random variables, and $1 \{ \cdot \}$ is the indicator function. Figure 2 shows $p_0(M^*)$ computed for $\alpha = 0.5$ and $\mathbf{N} = 1000, 3000, 5000, 10,000$, and 25,000, with Monte Carlo simulation being used to compute for the two largest sample sizes.

The expected number of clusters under this prior is

$$\sum_{i=1}^{N} \frac{\alpha}{\alpha + i - 1} = \alpha (\psi(\alpha + N) - \psi(\alpha)) \approx \alpha \log \left( 1 + \frac{N}{\alpha} \right),$$

where $\psi(\bullet)$ is the Digamma function. Thus the bulk of the mass in this distribution is concentrated on low values of $M^*$ even when $N$ is very large, and the expectation increases very slowly with $N$. To make approaches comparable, in the simulation and real data analysis, we will use this $p_0(M^*)$ as the prior in the finite mixture model that we discussed in Section 3.2.

While under MFMs, we can use a similar procedure to sample the number of clusters by replacing $\alpha$ with $\delta V_N(M^* + 1)/V_N(M^*)$ and $N_m$ with $N_m + \delta$. We use Monte Carlo simulation to generate the empirical distribution of $p_0(M^*)$ (shown in Figure 2). Under the prior $M \sim \text{Poisson}(\alpha \log(N)) + 1$, the prior of $M^*$ is similar to $\text{Poisson}(\alpha \log(N)) + 1$ as well as with Dirichlet mixture models. In terms of cluster sizes, as is evident from the form in Equation (4), the Dirichlet process prior favours a few large clusters (see, e.g., Green & Richardson, 2001; Miller & Harrison, 2018), whereas under the symmetric prior with $\delta = 1$, the finite mixture model specification assigns greater mass to clusters of the same order of magnitude.

4. COMPUTATION FOR THE MIXTURE OF FINITE MIXTURES CTHMM

We deploy reversible-jump MCMC procedures to allow the number of clusters to be inferred via the posterior distribution under a finite mixture model formulation. Specifically, we use

\[\text{DOI: 10.1002/cjs.11671}\]
a split–combine move to update the number of clusters and then implement fixed-dimension MCMC.

In this section, we start with finite mixture model-based clustering for CTHMMs. We have the following hierarchy:

\[ M \sim p_0(M), \; M \in \{1, 2, 3, \ldots\}, \]
\[ \omega_1, \ldots, \omega_M | M \sim \text{Dirichlet}(\delta, \ldots, \delta), \]
\[ \mathbb{P}(C_n = m | \omega_1, \ldots, \omega_M, M) = \omega_m, \; m = 1, \ldots, M; \; n = 1, \ldots, N, \]
\[ X_n | \Theta, C_n \sim \text{CTMC}(\pi_{C_n}, Q_{C_n}), \]
\[ O_n | X_n, \Theta, C_n \sim \text{Exponential family}(B_{C_n}). \]

The complete-data likelihood for subject \( n \) is

\[ \mathcal{L}(O_n, X_n | C_n, \Theta) = \prod_{m=1}^{M} [\omega_m \mathcal{L}(O_n, X_n | C_n = m, \Theta_m)] I(C_n = m). \]

A subject is assigned to cluster \( m \) according to the probability

\[ \mathbb{P}(C_n = m | O_n, X_n, \Theta) = \frac{\omega_m \mathcal{L}(O_n, X_n | C_n = m, \Theta_m)}{\sum_{l=1}^{M} \omega_l \mathcal{L}(O_n, X_n | C_n = l, \Theta_l)}. \] (5)

![Figure 2: Prior distribution for number of clusters, \( p_0(M^*) \) for Dirichlet mixture models (DMM) with \( \alpha = 0.5 \) and mixture of finite mixtures (MFMs) with \( \delta = 1 \) and \( M \sim \text{Poisson}(\alpha \log(N)) + 1. \)](image)
In reality, the model parameters $\Theta$ and the values of the latent states, $X_n$, are not known, and they must be inferred from the observed data.

### 4.1. Reversible-Jump MCMC

The reversible-jump algorithm (Green, 1995) serves to implement trans-dimensional MCMC for the finite mixture model with $M^*$ unknown. We study split/combine moves for pairs of states similar in spirit to the split/merge moves of Richardson & Green (1997) and Dellaportas & Papageorgiou (2006). If the model consists of $M$ components with $K$ latent states ($K$ is assumed fixed) in each component, then the model can be viewed as a CTHMM-GLM with $K \times M$ states. The infinitesimal generator $Q$ can be expressed as a block diagonal matrix with diagonal blocks $Q_1, \ldots, Q_m, \ldots, Q_M$, where $Q_m$ is the $K \times K$ infinitesimal generator matrix for cluster $m$ ($m = 1, \ldots, M$). This constraint prevents the transition of subjects between clusters across time.

One iteration of the algorithm includes

1. a move that considers splitting a cluster into two, or combining two clusters into one;
2. an update of the cluster label for each individual according to the posterior probability specified in Equation (5), given the parameters in each cluster, and the latent states;
3. an update of the model parameters using standard MCMC moves for each cluster with $M$, the number of clusters, fixed, specifically proposals that
   - update the latent state indicators $S_{n,t}$,
   - update the parameters associated with the observation process $B$,
   - update the initial distribution $\pi$, and
   - update the infinitesimal generator $Q$.

For any empty cluster resulting from Step 2, we generate model parameters from prior distributions.

For the split and combine moves, we will implement a reversible-jump algorithm (Green, 1995). We carry out this move on the marginalized model, where the cluster labels and latent processes are marginalized out from the calculation, and use the likelihood

$$
L(o|\Theta, M) = \prod_{n=1}^{N} \left\{ \sum_{m=1}^{M} \sigma_m L(o_n|m) \right\},
$$

where $\Theta$ denotes the collection of $\sigma$ and CTMC parameters.

Consider a proposal from the current state $(M, \Theta)$ to a new state $(M', \Theta')$ using the proposal density $q(M', \Theta'; M, \Theta) = q_1(M'; M) q_2(\Theta'; \Theta)$, that is, using independent proposals for the two components. The acceptance probability for this proposal is given by

$$
a(M', \Theta'; M, \Theta) = \min \left( 1, \frac{q_1(M'; M') q_2(\Theta; \Theta') p(M', \Theta' | o)}{q_1(M'; M) q_2(\Theta'; \Theta) p(M, \Theta | o)} \right),
$$

where $p(M, \Theta | o)$ is the posterior distribution of $(M, \Theta)$ given the observed data $o$, which can, up to proportionality, be decomposed into the marginal (or “incomplete data”) likelihood of the data $L(o|\Theta, M)$ times the prior distribution for $(M, \Theta)$, with prior distribution as $p_0$; thus

$$
p(M, \Theta | o) \propto L(o|\Theta, M)p_0(\Theta | M)p_0(M).
$$

The Canadian Journal of Statistics / La revue canadienne de statistique DOI: 10.1002/cjs.11671
Our algorithm relies upon the ability to compute the marginal likelihood efficiently for any $\Theta$; however, this is a standard “forward” calculation for CTHMMs. The reversible-jump algorithm is for the most part standard, and the only noteworthy elements are the trans-dimensional moves. We discuss these aspects in more detail below.

### 4.2. Split and Combine Move

To construct efficient split and combine moves, we adopt the idea of centred proposals (Brooks, Giudici & Roberts, 2003) designed to produce similar likelihood contributions for the current and proposed parameters. The combine move is designed to choose a cluster, $m$ say, at random, and select another cluster $i$ such that $||B_i - B_m||_2$ is smallest for $i \neq m$, where $||\cdot||_2$ is the Euclidean norm. The reverse-split move is to randomly select a cluster $m$ to split into two clusters, say $m$ and $m^*$, and check if the condition $||B_{ms} - B_m||_2 < ||B_j - B_m||_2$ for $j \neq m$. If this condition is not met, then the split move is rejected directly.

#### 4.2.1. Split move

We consider an update that changes the number of clusters from $M \to M + 1$. Without loss of generality, we aim to split the $M$th cluster with parameters $\Theta_M = \{\pi_M, Q_M, B_M\}$ into two clusters, with corresponding parameters $\Theta' = \{\pi', Q', B'\}$ and $\Theta'' = \{\pi'', Q'', B''\}$. To implement the idea of centring proposals, we use a deterministic proposal for $Q$ and $\pi$, and let $Q' = Q'' = Q_M$ and $\pi' = \pi'' = \pi_M$. For the observation model parameter $B$, we can use a similar proposal: for $k = 1, \ldots, K$, let $\beta_{1,k}^j = \beta_{M,1,k}$ and $\beta_{1,k}' = \mathcal{N}(\beta_{M,1,k}, c^2)$, with $\beta_{j,k}' = \beta_{j,k}'' = \beta_{M,j,k}$ for $j = 2, \ldots, D$. For $\sigma$, let $w \sim \text{Beta}(2, 2)$ and set $\sigma' = w \sigma_M$ and $\sigma'' = (1 - w) \sigma_M$. The proposed move is from $M$ to $M + 1$ clusters, with the new parameters $\Theta' = \{\pi', Q', B'\}$ with

$$
\pi' = (\pi', \pi''), \quad Q' = \begin{pmatrix} Q' & 0 \\ 0 & Q'' \end{pmatrix}, \quad B' = (B', B'').
$$

If we denote the posterior ratio as

$$
r(M+1, (\Theta', \sigma', \sigma''); M, (\Theta_M, \sigma_M)|o) = \frac{p(M+1, (\Theta', \sigma', \sigma'')|o)}{p(M, (\Theta_M, \sigma_M)|o)},
$$

then the acceptance probability for this proposal is

$$
\min \left(1, \frac{q(Q'_M; \pi') q(B'_M; B') q(\pi_M; \pi') q(\sigma_M; \sigma') q(\sigma'_M; \sigma'')}{q(Q'_M; Q'_M) q(B'_M; B'_M) q(\pi'_M; \pi'_M) q(\sigma'_M; \sigma'_M)} \times r(M+1, (\Theta', \sigma', \sigma''); M, (\Theta_M, \sigma_M)|o) \right) = \min \left(1, \frac{d_{M+1} \sigma_M}{b_M p_{\sigma}(w) p_{\beta}(\beta_{1,k})} r(M+1, (\Theta', \sigma', \sigma''); M, (\Theta_M, \sigma_M)|o) \right),
$$

where $b_M$ is the probability of choosing the split move, $d_{M+1} = 1 - b_M$ is the probability of choosing the combine move, $p_{\beta}(\cdot)$ is the Normal density with mean $\beta_{1,k}$ and variance $c^2$, and $p_{\sigma}(\cdot)$ is the Beta($2, 2$) density.
4.2.2. Combine move

For the combine move, we consider an update from \( M + 1 \rightarrow M \) clusters. Again, without loss of generality, we consider combining the \( (M + 1) \)th and \( M \)th clusters into one cluster. We first find the stationary probabilities, \( s_M \) and \( s_{M+1} \), associated with \( Q_M \) and \( Q_{M+1} \), respectively. To combine \( Q_M \) and \( Q_{M+1} \) into \( Q' \), the operation is as follows:

\[
q'_{i,k} = \frac{s_{M,i}}{s_{M,i} + s_{M+1,i}} \times q_{M,i,k} + \frac{s_{M+1,i}}{s_{M,i} + s_{M+1,i}} \times q_{M+1,i,k}, \quad i \neq k = 1, \ldots, K,
\]

and \( q_{M,k,k} = -\sum_{i \neq k} q_{M,i,k} \) for \( k = 1, \ldots, K \). For the observation process parameter \( B \)

\[
\beta'_{i,k} = \frac{s_{M,i}}{s_{M,i} + s_{M+1,i}} \times \beta_{M,i,k} + \frac{s_{M+1,i}}{s_{M,i} + s_{M+1,i}} \times \beta_{M+1,i,k}, \quad i, k = 1, \ldots, K.
\]

For the initial distribution \( \pi \)

\[
\pi'_k = \frac{s_{M,i}}{s_{M,i} + s_{M+1,i}} \times \pi_{M,k} + \frac{s_{M+1,i}}{s_{M,i} + s_{M+1,i}} \times \pi_{M+1,k}, \quad k = 1, \ldots, K,
\]

and we rescale the values such that they sum to 1. For the mixture weight update, we set \( \varpi' = \varpi_M + \varpi_{M+1} \). The acceptance probability for the move from \( M + 1 \) to \( M \) clusters is

\[
\min \left( 1, \frac{\beta_M M M (w)}{d_{M+1} \varpi} r (M, \varpi' ; M + 1, \Theta_M, \Theta_{M+1} | \theta) \right). \tag{7}
\]

5. DIRICHLET PROCESS MIXTURE CTHMM

In the Dirichlet process mixture model, for some \( \alpha > 0 \) and distribution \( G_0 \), we assume that \( \Theta_m = \{ \pi_m, Q_m, B_m \} \sim G_0 \) for \( m = 1, 2, \ldots \) again represent the component-specific model parameters. Next we assume that \( \tilde{\Theta}_n \sim G(\cdot) \), and if \( \nu_m \sim Beta(1, \alpha), m = 1, 2, \ldots \)

\[
G(\tilde{\Theta}_n) = \sum_{m=1}^{\infty} \sigma_m \delta (\tilde{\Theta}_n = \Theta_m) \quad \text{with} \quad \sigma_m = \nu_m \prod_{k<m} (1 - \nu_k).
\]

We then suppose that cluster label \( C_n \) is defined so that \( C_n = m \) implies that \( \tilde{\Theta}_n = \Theta_m \). Within cluster \( m \), for the latent process, \( X_n | \tilde{\Theta}_n, C_n = m \sim \text{CTMC}(\pi_m, Q_m) \), and for the observations \( O_n | X_n, \Theta_n, C_n = m \sim \text{Exponential family}(B_m) \). We assume a priori for the equilibrium probabilities, \( \pi \sim \text{Dirichlet}(\alpha_1, \ldots, \alpha_K) \), and for the off-diagonal elements in \( Q, q_{lm} \sim \text{Gamma}(a_{lm}, b_l) \) for \( 1 \leq l \neq m \leq K \); this prior is conjugate with the complete data likelihood—the representation using the latent trajectory as auxiliary data. Conjugacy can be relaxed using the approaches outlined in Neal (2000), but here we restrict our attention to the conjugate case.

5.1. Split–Merge MCMC Algorithm for DMMs

Inference for the Dirichlet process mixture model is often carried out using MCMC (Escobar & West, 1995; MacEachern & Müller, 1998; Neal, 2000; Jain & Neal, 2004, 2007). For this model, Gibbs sampling via a Polya urn scheme is often used, but this strategy can be slow in mixing when clustering a large volume of data, and it is nontrivial to parallelize. This limitation has motivated the development of MCMC algorithms, which partially address the inherently sequential nature of the updates. Celeux, Hurn & Robert (2000) noted that the standard MCMC sampler tends to
stay within the neighbourhood of the local mode. As a consequence, it is less likely to move to a
new mixture component even with well-separated components because of the low probability of
moving to an intermediate state: updating a group of subjects simultaneously can help resolve this
problem. Green & Richardson (2001) introduced a split–merge update under the reversible-jump
MCMC framework and also showed how to construct the split proposal. Subsequently, Jain
& Neal (2004, 2007) extended this to an MH sampling scheme with split–merge updates:
their extended approach suggested splitting the component in a deterministic manner by
employing restricted Gibbs sampling, which would increase the probability of forming a new
component.

We will use the split–merge approach to obtain the posterior samples. We first focus on
the case where the observation model is covariate-free and then the time-varying covariates
are categorical only, and use the original parameterization of \( \theta \) in the exponential family as
in Equation (1) instead of \( B \). The algorithm proceeds as follows, with superscript \( i = 0, 1, \ldots \)
denoting the iteration number:

- **Initialization:** Randomly sample cluster labels \( C^0 \) from \( \{ 1, \ldots, M^0 \} \) for each subject, where
  \( M^0 \) is an arbitrary positive integer with \(| C^0 | = M^0 \). Starting with initial values (e.g.,
those values can be obtained from the EM algorithm) \( \pi^0_c, Q^0_c, \theta^0_c, \) and \( \phi^0_e \), compute the “forward”
and “backward” quantities \( a_{n,t,k} \) and \( b_{n,t,k,j} \) using the forward–backward algorithm (see the
Supplementary Material for the definitions of these quantities).

- **Update the latent state indicators:** For each \( n \) and \( t \), generate the random vector \( S^t_{n,t} \)
  from the multinomial distribution with the parameter set \( \Delta^t_{n,t,k} \), where
  \( S_{n,t} = (S_{n,t,1}, \ldots, S_{n,t,K}) \) is an indicator random vector with \( S_{n,t,k} = 1 \) if \( X_{t} = k \)
and 0 otherwise.

- **Simulate the path of the latent process:** For each individual, the path simulation follows
in two steps. First, draw the current and next state \( (X_{n,t}, X_{n,t+1}) \) from a multinomial
distribution with the parameter matrix \( b_{n,t,k,j} \), then simulate \( N_{n,t,m} (\Delta_{n,t}) \) and \( R_{n,t} (\Delta_{n,t}) \)
from the Markov processes with infinitesimal generator \( Q^{-1}_{n,m} \) through the intervals \([\tau_{n,t}, \tau_{n,t+1}]\)
initiated at \( X_{n,t} \) and end point \( X_{n,t+1} \) sampled previously.

- **Update C by split–merge** (Jain & Neal, 2004): Denote by \( M^{-1} \) the number of components
  in the current label configuration. Select two distinct subjects \( d \) and \( e \) and denote their cluster
labels by \( j_d = C^{-1}_{d} \) and \( j_e = C^{-1}_{e} \).

1. Let \( \mathcal{M} = \{ f : f \neq d, e, \text{but } C^{-1}_f = j_d \text{ or } C^{-1}_f = j_e \} \subseteq \{ 1, 2, \ldots, N \} \).
2. Define the launch state, \( C^l \), by running a Gibbs sampler scan restricted to the labels of
subjects \( f \in \mathcal{M} \).

   (i) If \( j_d \neq j_e \): for \( f \in \mathcal{M}, P \left( C_f = j \mid C_{-f} \right) \) for \( j \in \{ j_d, j_e \} \) is given by

   \[
   \frac{N^{-}_{f,j} \int \mathcal{L}_f (\Theta) \mathit{d}H^{-}_{f,j} (\Theta)}{N^{-}_{f,j,d} \int \mathcal{L}_f (\Theta) \mathit{d}H^{-}_{f,j,d} (\Theta) + N^{-}_{f,j,e} \int \mathcal{L}_f (\Theta) \mathit{d}H^{-}_{f,j,e} (\Theta)},
   \]

   where \( N^{-}_{f,j} = \sum_{m \neq j} \mathbb{I} (C_m = j) \). For MFM, \( N^{-}_{f,j} \) will be replaced by \( N^{-}_{f,j} + \delta \), and
   \( \mathcal{L}_f (\Theta) \) is defined in Equation (11). The function \( H^{-}_{f,j} (\Theta) \) is the posterior distribution
   of \( \Theta \) based on the prior \( G_0 \) and subjects \( g \in \mathcal{M} \cup \{ d, e \} \) but \( g \neq f \), such that their
   labels \( C_g = j \).
The prior distribution of $C$ is calculated in the same manner as identified in Equation (8). The value $M^{i-1} + 1$ is a new component label that is not represented in the current label set.

3. Split–merge step:

(i) If $j_d = j_e$, propose a split configuration, $C_{\text{split}}$. Set $C_{\text{split}}^{d} = M^{i-1} + 1$ and $C_{\text{split}}^{e} = C_{e}^{i-1} = j_e$. For $f \notin M$, let $C_{f}^{\text{split}} = C_{f}^{i-1}$; for $f \in M$, set $C_{f}^{\text{split}}$ by performing one more Polya urn scan from the launch state. Perform an MH update with acceptance probability $a(C_{\text{split}}, C_{f}^{i-1})$. If the proposal is accepted, then set $C^{i} = C_{\text{split}}^{i}$; if it is rejected, set $C^{i} = C^{i-1}$.

(ii) If $j_d \neq j_e$, propose the merge configuration, $C_{\text{merge}}$. Set $C_{\text{merge}}^{d} = e$; for $f \notin M$, let $C_{f}^{\text{merge}} = C_{e}^{i-1} = j_e$. For $f \in M$, set $C_{f}^{\text{merge}} = C_{f}^{i-1}$. Perform an MH update with acceptance probability $a(C_{\text{merge}}, C_{f}^{i-1})$. If the proposal is accepted, then set $C^{i} = C_{\text{merge}}^{i}$; if it is rejected, set $C^{i} = C^{i-1}$.

The acceptance probability for this proposal is computed in the following subsection.

• **Update the component parameters $\theta$, $Q$, and $\pi$:** Assume that $C^{i} = \{ C^{i}_{1}, \ldots, C^{i}_{N} \}$.

1. Update $Q$: Update the $N_{l,m}(\Delta_{n,t})$ and $R_{f}(\Delta_{n,t})$ from the updated label component generator $Q_{C}$, and then $q_{l,m|C}$ associated with component $C$ from a Gamma distribution with shape parameter $\Lambda_{l,m}^{C}$ and rate parameter $\Upsilon_{l}^{C}$, where $\Lambda_{l,m}^{C} = \sum_{n:C_{n}=C} T_{n} \sum_{t=1}^{T_{n}} N_{l,m}(\Delta_{n,t}) + \alpha_{l,m}$ and $\Upsilon_{l}^{C} = \sum_{n:C_{n}=C} T_{n} \sum_{t=1}^{T_{n}} R_{f}(\Delta_{n,t}) + b_{l}$.

2. Update $\theta$: For the prior $\pi_{0}(\theta)$, generate the estimate $\theta^{i}_{\text{r}}$ via the conditional posterior distribution using only the individuals with cluster label $C$ via a Gibbs/MH step.

3. Update $\pi$: For the conjugate Dirichlet $(\alpha_{1}, \ldots, \alpha_{K})$ prior,

$$
\pi^{i}_{C} \sim \text{Dirichlet}\left( \sum_{n:C_{n}=C} S_{n,1,1}^{i} + \alpha_{1}, \ldots, \sum_{n:C_{n}=C} S_{n,1,K}^{i} + \alpha_{K} \right).
$$

5.2. Acceptance Probability for the Split–Merge Proposal

The acceptance probability for the proposal to update $C$ for split and merge moves takes the form

$$
a(C^{i}, C) = \min \left\{ 1, \frac{q(C|C^{*}) \pi_{0}(C^{*}) L(C^{*})}{q(C|C^{i}) \pi_{0}(C^{i}) L(C^{i})} \right\}. \tag{9}
$$

The prior distribution of $C$ is the product over the partition of $\{ C_{1}, \ldots, C_{N} \}$. Therefore

$$
\frac{\pi_{0}(C_{\text{split}})}{\pi_{0}(C)} = \alpha \frac{(N_{C_{d}^{\text{split}}}-1)! (N_{C_{e}^{\text{split}}}-1)!}{(N_{C_{d}}-1)!}, \quad \frac{\pi_{0}(C_{\text{merge}})}{\pi_{0}(C)} = \frac{1}{\alpha} \frac{(N_{C_{d}^{\text{merge}}}-1)!}{(N_{C_{d}}-1)! (N_{C_{e}}-1)!},
$$
where $N_j$ denotes the count of the number of subjects with label $j$ in the configuration. For MFMs, $\alpha$ and $N_j$ will be replaced by $\delta \frac{V_N(M^*+1)}{V_N(M^*)}$ and $N_j + \delta$, respectively.

As there is only one way to assign all $k \in S$ into one component, the proposal density $q(C_{\text{merge}} | C) = q(C) | C_{\text{split}} = 1$. For $q(C_{\text{split}} | C)$, the probability is the product of transition probabilities from the last launch state $C'$ to the final proposed state $C_{\text{split}}$, that is, $q(C_{\text{split}} | C) = \prod_{k \in S} q(C_{\text{split}} | C_{-k})$, which is calculated by the final Pólya urn scan in the split update identified in Equation (8). Each time, $C_k$ is incrementally modified during the Pólya urn scan and the updated $C_k$ is used in the subsequent restricted Gibbs sampling computation.

Given the current labels for $f \in \mathcal{M} \cup \{d,e\}$, the likelihood contribution equals

$$
\mathcal{L}(C_f) = \prod_{f : C_f = j_d} \int \mathcal{L}_f(\Theta) dH_{f,j_d}(\Theta) \times \prod_{f : C_f = j_e} \int \mathcal{L}_f(\Theta) dH_{f,j_e}(\Theta),
$$

where $H_{f,j}$ is the posterior distribution of $\Theta$ based on $G_0$ and all subjects such that their label $C_g = j$ but $g < f$. As the parameters $\theta$, $\pi$, and $Q$ are separable in $\mathcal{L}_f(\Theta)$, $\mathcal{L}_f(\Theta)$ can be written as

$$
\mathcal{L}_f(\Theta) = \prod_{t=1}^{T_f} \prod_{k=1}^{K} f \left( O_{f,t} | S_{f,t,k} \right) \prod_{k=1}^{K} \pi_k^{S_{f,t,k}} \prod_{t=1}^{T_f} \prod_{l \neq m} \mathcal{L}(q_{lm} | \Delta_{f,t}).
$$

In the split step, the likelihood of the cluster label configuration is

$$
\mathcal{L}(C_{\text{split}}) = \prod_{f : C_f = C_{\text{split}}} \int \mathcal{L}_f(\Theta) dH_{f,c_{\text{split}}}(\Theta) \times \prod_{f : C_f = C_{\text{split}}} \int \mathcal{L}_f(\Theta) dH_{f,j_e}(\Theta).
$$

For the merge step, the likelihood of the cluster label configuration is

$$
\mathcal{L}(C_{\text{merge}}) = \prod_{f : C_f = j_e} \int \mathcal{L}_f(\Theta) dH_{f,j_e}(\Theta).
$$

The term $q(C | C_{\text{merge}})$ is the product of the transition probabilities from the last launch state to the original “split” state, but there is no actual sampling step since the “split” states are already known. Full details of the calculation of a Poisson example are given in the Supplementary Material.

6. EXAMPLES

We examined the performance of our proposed algorithms via simulation studies. In all examples, data were generated from the finite mixture model with a fixed number of clusters. We used $\text{Gamma}(2.5, 5)$ priors on the off-diagonal elements of $Q$, $\mathcal{N}(-2, 1)$, $\mathcal{N}(0, 1)$, and $\mathcal{N}(2, 1)$ priors for states 1, 2, and 3, respectively, in the Gaussian case and $\text{Gamma}(5, 10)$, $\text{Gamma}(10, 10)$, and $\text{Gamma}(20, 10)$, respectively, in the Poisson case for the components of $B$, and a $\text{Dirichlet}(10, \ldots, 10)$ prior for $\pi$. For MFMs, we set $\delta = 1$ and $M \sim \text{Poisson}(0.5 \log(N)) + 1$. DOI: 10.1002/cjs.11671
6.1. Example 1

In the first example, we considered a three-cluster CTHMM-GLM, with each cluster having three latent states and transition matrices

\[
Q_1 = \begin{bmatrix}
-2.5 & 2.0 & 0.5 \\
0.5 & -1.5 & 1.0 \\
0.1 & 0.9 & -1
\end{bmatrix}, \quad Q_2 = \begin{bmatrix}
-1.20 & 1.00 & 0.20 \\
1.40 & -1.50 & 0.10 \\
0.05 & 0.20 & -0.25
\end{bmatrix},
\]

\[
Q_3 = \begin{bmatrix}
-0.50 & 0.49 & 0.01 \\
0.25 & -0.30 & 0.05 \\
0.01 & 0.10 & -0.11
\end{bmatrix},
\]

with the following associated coefficient matrices:

- Gaussian case: \(B_1 = (-4, 0, 5), B_2 = (-5.5, 0.5, 5.5), B_3 = (-5, 1, 4.8)\);
- Poisson case: \(B_1 = (-2, 1.2, 3), B_2 = (-1, 1, 2.5), B_3 = (-1.5, 1.1, 2.8)\).

The initial distributions for all three clusters were \(\pi_1 = (0.5, 0.4, 0.1)^T, \pi_2 = (0.3, 0.5, 0.2)^T, \) and \(\pi_3 = (0.45, 0.45, 0.1)^T\). In the Gaussian outcome model, we set \(\sigma\), the residual error standard deviation, equal to 1. In the split–merge Gibbs sampler, the intermediate restricted sampling scanned three times before performing the actual split or merge update. We initiated the model with one cluster. Data were generated by constructing the continuous Markov chain from the generator \(Q_i\) for cluster \(i = 1, 2, 3\), a continuous-time realization \(\{X_s, 0 \leq s \leq 15\}\), and uniformly sampled \(T - 1\) time points between 0 and 15. Data were generated for 1000 subjects in each cluster.

Results are shown in Table 1 (an extended version is provided in the Supplementary Material) and are based on the three-cluster iterations taken from a total of 2000 iterations. Observed results in the rows labelled MFM-RJ represent MFMs using reversible-jump MCMC that we described in Section 4, whereas those in the rows labelled MFM-SM represent MFMs using the split–merge algorithm described in Section 5.1 by replacing \(\alpha\) and \(N_j\) by \(\delta V_N(M^* + 1) / V_N(M^*)\) and \(N_j + \delta\), respectively (Miller & Harrison, 2018). For DMMs, in all cases, the posterior modal number of clusters was 3; this result was especially clear for the cases \(T = 50, 100\) when over 90% of the iterations resided in the three-cluster model. For a summary output, we assigned cluster membership to the subject according to its posterior mode conditional on the three-cluster iterations. When \(T = 30\) for the Normal case, there were 23.70% and 14.85% instances of two
TABLE 2: Example 2: Simulation study with three clusters.

| No. of cluster | \( \sigma = 0.5 \) | \( \sigma = 1 \) | \( \sigma = 2 \) |
|---------------|-------------------|-------------------|-------------------|
|               | MFM- RJ           | DMM               | MFM- RJ           | DMM               | MFM- RJ           | DMM               | MFM- RJ           | DMM               |
| 1             | 1.20%             | 0.05%             | 0.10%             | 0.65%             | 0.10%             | 0.10%             | 0.05%             | 0.15%             | 0.25%             |
| 2             | 38.50%            | 0.20%             | 26.05%            | 0.55%             | 0.05%             | 5.70%             | 0.75%             | 18.15%            | 13.95%            |
| 3             | **58.05%**        | **80.90%**        | **54.10%**        | **61.05%**        | **95.75%**        | **46.30%**        | **26.15%**        | **37.40%**        | **37.20%**        |
| 4             | 1.45%             | 17.05%            | 17.25%            | 31.95%            | 4.10%             | 32.60%            | 20.95%            | 38.25%            | 28.55%            |
| 5             | 0.80%             | 1.80%             | 2.55%             | 5.80%             | 0.00%             | 11.95%            | 19.80%            | 6.05%             | 12.80%            |
| 6             | 0.00%             | 0.00%             | 2.30%             | 0.00%             | 0.00%             | 7.80%             | 1.80%             | 0.00%             | 6.10%             |
| ≥7            | 0.00%             | 0.00%             | 0.00%             | 0.00%             | 0.00%             | 1.60%             | 14.20%            | 0.00%             | 1.15%             |

% Misclassification | 4.90% | 19.20% | 15.40% | 9.50% | 22.13% | 19.30% | 24.20% | 24.70% | 25.30% |

Note: Each cluster has three latent states, the same \( B \) and \( \pi \) parameters, but different \( Q \) matrices. The bold values represent the percentages of three-cluster iterations.

and four clusters, respectively, while for the Poisson case, 43.40% of the iterations involved four clusters. In general, restricted Gibbs sampling using the split–merge proposals performed well. The norm differences in parameters between true values and the posterior means were small, and misclassification rates were low. As sample size decreased, the misclassification rate increased and it was more difficult to cluster trajectories; also, the Gaussian cases had higher misclassification rates than the Poisson cases. For MFMs, the posterior mode of the number of clusters was three for all cases. The proposed algorithm MFM-SM exhibited a lower percentage of three-cluster iterations than its MFM-SM counterpart, MFM-RJ, since split–merge encourages more trans-dimensional moves to explore the parameter space while our reversible-jump MCMC yielded greater accuracy in terms of the cluster membership because it is updated in each sweep. In Miller & Harrison (2018), they included both the split–merge algorithm and the Pólya urn scheme in each MCMC step to encourage trans-dimensional moves as well as to increase the accuracy of clustering. For a small sample size, this approach is feasible, but in our case, when the sample size is large and the model structure is complex, including both steps would be computationally intensive.

6.2. Example 2

In the second example, data were generated from a finite mixture model with different infinitesimal generators \( Q \) within each cluster; the parameters \( B \) and \( \pi \) were identical to the values assigned to Cluster 1 from Example 1. We assumed that the observation process involved only the Gaussian distribution. Data were generated for 1000 subjects in each cluster with 3000 subjects in total. For the elements of \( Q \), we implemented the restricted Gibbs sampling procedure with split–merge proposals, involving five intermediate Gibbs steps (see the Supplementary Material for the algorithm).

Observed results for this simulation study are reported in Table 2; again, an extended version may be found in the Supplementary Material. The tabulated values are conditional on three-cluster iterations with total 2000 iterations. As this example represents a more difficult clustering problem compared to the previous example, the percentage of three-cluster iterations decreased. However, for DMMs, posterior modes of the number of clusters were still three,
apart from the case when $\sigma = 2$, in which case the posterior was more diffuse. For the DMM also, the posterior distribution of the number of clusters was also more dispersed when $\sigma = 2$. For $\sigma = 0.5$, 17.05% of the iterations had four clusters, while for $\sigma = 1$, the corresponding percentage was 4.10. When $\sigma = 2$, 18.15%, 38.25%, and 6.05% of iterations involved two, four, and five clusters, respectively. The misclassification rate for $\sigma = 2$ was roughly 40%, but the posterior mean estimates were still close to the corresponding true values. For MFMs, the posterior modes of the number of clusters were three for all cases. In the particular case involving MFM-RJ when $\sigma = 0.5$, the model spent more iterations in two clusters before stabilizing, while the algorithm stabilized faster to three clusters because, when $\sigma = 0.5$, the cluster separation is greater. When $\sigma = 2$, the algorithm tended to move to higher dimensions, with 20.95%, 19.80%, and 10.15% of iterations involving four to seven clusters, respectively. Like the observed results in the previous example, MFM-RJ had the smallest misclassification rates conditional on three-cluster iterations, but MFM-SM yielded more trans-dimensional moves. This example illustrates that separating clusters via $Q$ is more challenging than via all other parameters.

6.3. Example 3

In the third example, which once again involved using a three-cluster generating model, we added a three-level, time-varying factor covariate $Z_1 \sim \text{Multinomial}(1; 1/4, 1/4, 1/2)$ at each observation time point to modify a Poisson outcome model, with the coefficient matrices

$$B_1 = \begin{pmatrix} -2 & 1.2 & 3 \\ -0.3 & 0 & 0 \\ 0.5 & -0.1 & -0.1 \end{pmatrix}, \quad B_2 = \begin{pmatrix} -1 & 1 & 2.5 \\ 0.4 & -0.2 & -0.5 \\ -0.1 & 0 & -0.4 \end{pmatrix},$$

$$B_3 = \begin{pmatrix} -1.5 & 1.1 & 2.8 \\ 1 & 0.1 & -0.1 \\ -0.5 & 0.1 & -0.5 \end{pmatrix}.$$

The parameters $Q$ and $\pi$ were assigned the same values as in Example 1. Data were generated with 300, 500, and 200 subjects in Clusters 1, 2, and 3, respectively, with 1000 subjects in total. The intermediate restricted Gibbs sampling scanned twice before performing the actual split or merge update. In this case, we used $\text{Gamma}(11, 10)$, $\text{Gamma}(55, 10)$, and $\text{Gamma}(165, 10)$ as prior distributions for $B$ in states 1, 2, and 3, respectively.

The observed results that summarize the three-cluster iterations for this example are reported in Table 3; a full version may be found in the Supplementary Material. In terms of misclassification rates, we observed results that were similar to those obtained for the Poisson cases in Example 1. For DMMs, as we have more parameters in this case, the posterior distribution of the number of clusters was more dispersed; see the various plots provided in the Supplementary Material. For the case $T = 30$, the number of clusters fluctuated between 3 and 4, involving 37.2% and 56.8% of the total iterations, respectively. When $T$ increased to 50 and 100, the mode became three with less posterior variation observed—for $T = 50, 100$, only 13.55% and 2.95% of 2000 iterations resulted in four clusters, respectively. For $T = 100$, there were a few iterations with more than five clusters initially, but the number of clusters soon stabilized at 3. For MFMs, all cases had a posterior mode of three. MFM-RJ cases tended to remain in one model for a longer time, which led to more three-cluster iterations, whereas cases involving MFM-SM had more dimension changes but exhibited higher misclassification rates, an observation that is consistent with our experience in the two previous examples.
Table 3: Example 3: Simulation study with three clusters with one time-varying factor covariate.

|                  | $T = 30$ | $T = 50$ | $T = 100$ |
|------------------|----------|----------|-----------|
| % Three-cluster iterations | MFM-RJ 97.55% | 98.65% | 98.55% |
|                   | MFM-SM 65.90% | 65.90% | 87.50% |
|                   | DMM 37.20% | 85.05% | 95.60% |
| % Misclassification | MFM-RJ 3.50% | 1.90% | 0.60% |
|                   | MFM-SM 12.90% | 7.00% | 3.60% |
|                   | DMM 11.50% | 6.20% | 2.90% |

Note: Each cluster has three latent states and Poisson observation process.

7. CASE STUDY: HEALTH SURVEILLANCE OF COPD PATIENTS

Our health trajectories case study relates to healthcare surveillance for COPD in greater Montreal, Québec, Canada. In 1998, a 25% random sample was drawn from the registry of the Régie de l’assurance maladie du Québec (RAMQ, the Québec provincial health authority) with a residential postal code in the census metropolitan area of Montreal. In subsequent years, a 25% random sample of residents new to Montreal within the previous year was sampled to maintain the representative cohort. If people died or moved away from Montreal, their follow-up ended. The data include outpatient diagnoses and procedures submitted through RAMQ billing claims, and procedures and diagnoses from inpatient claims. Using established case definitions based on medical diagnostic codes (Blais et al., 2014; Lix et al., 2018), 76,888 COPD patients were enrolled with an incident event occurring after a minimum of 2 years at risk with no events. Patients were followed from January 1998, starting from the time of their first diagnosis, until December 2014. Physicians observed these patients only during medical visits, which occurred when patients chose to interact with the healthcare system, and at which information, including the number of prescribed medications, was collected. However, as this information was available only for patients with drug insurance, we restricted the cohort to patients over 65 years old with COPD, since prescription data were available for all of these patients. In addition, the types of healthcare utilization—hospitalization (HOSP), specialist visit (SPEC), general practitioner visit (GP), and emergency department visit (ED)—were also recorded.

In our analysis, we fitted and compared MFMs and DMMs. Specifically, we used reversible-jump MCMC for MFMs; from our simulation studies, MFM-RJ yielded more accurate results with a reasonable amount of trans-dimensional moves. The number of states in each cluster was fixed at 4 to match the convention in the COPD literature (GOLD Executive Committee, 2017), with the states labelled mild, moderate, severe, and very severe. We fitted the model with parameters $Q$, $\theta_{\text{HOSP}}$, $\theta_{\text{SPEC}}$, $\theta_{\text{GP}}$, $\theta_{\text{ER}}$, and $\pi$ in each cluster, where $\theta_U$ ($U = \text{HOSP, SPEC, GP, ER}$) represents the log mean parameter for the number of drugs prescribed in a Poisson model for each healthcare utilization. Each algorithm was initiated in the one-cluster model. Using elicitation via simulating trajectories, subjective prior distributions for off-diagonal elements of $Q$ were $\text{Gamma}(20, 500)$, chosen with consideration for plausible holding and transition times within each state, and $\text{Gamma}(1.5, 10)$, $\text{Gamma}(30, 10)$, $\text{Gamma}(60, 10)$, and $\text{Gamma}(100, 10)$ for states 1–4, respectively, for $B$ and $\text{Dirichlet}(10, \ldots, 10)$ for $\pi$. Finally, we chose $p_0(M)$ in Table 2 with $N = 25,000$ as the prior distribution on the number of clusters for the finite mixture model.

In our analysis, to compare the performance of the two models and demonstrate the feasibility of clustering trajectories, we implemented this method on a dataset comprised of 24,712 COPD patients. We report the observed results for both models based on 2000 MCMC samples with

DOI: 10.1002/cjs.11671 The Canadian Journal of Statistics / La revue canadienne de statistique
100 burn-in iterations after initialization using an EM algorithm fit of the one-cluster model as described in Luo et al. (2021). For both models, the mode of the posterior distribution of the number of clusters was four, and hence the results presented are from the four-cluster iterations. The Supplementary Material gives details of the effective sample sizes of the posterior samples for $Q$ and $B$ in each cluster; these were satisfactory. Parameters $\theta_{\text{HOSP}}$, $\theta_{\text{SPEC}}$, $\theta_{\text{GP}}$, and $\theta_{\text{ER}}$ are converted to a contrast (relative-risk) parameterization in each cluster for both models. The posterior mean of the exponential of the estimated coefficients for the four-cluster model with the number of patients in each cluster may be found in the tables in the Supplementary Material. The numbers of drugs across different healthcare utilizations are different within each state. All four clusters have distinct numbers of prescribed drugs in each state. In general, the average number of drugs prescribed following a visit to a specialist was smaller than the corresponding average following a typical visit to a general practitioner; Cluster 1 had on average a greater number of drugs prescribed. Cluster 2 was the largest cluster, and had a greater number of drugs prescribed, on average, than the other two clusters. The posterior distribution for the DMM varied slightly more than for finite mixture models due to the nature of the assumptions on cluster structure.

Recall that one of the principal computational challenges in this problem is that at each iteration we essentially need to sample and store complete trajectories for each of the almost 25,000 patients in the study; however, these posterior sampled trajectories can themselves be useful in gathering inference concerning the time spent in each state, time required to transition into more severe states, and so on. In the Supplementary Material, we provide plots of posterior mean transition patterns over 5 years for the four clusters. In both figures, we observed a similar transition pattern, which we summarize as follows: for State 1, patients in Cluster 2 are those most likely to remain in State 1, while Cluster 4 has the greatest probability for patients to exit this state. For State 2, all the clusters have a similar transition pattern. In State 3, patients in Cluster 2 have a relatively high probability of transitioning to State 4 compared to other clusters; Cluster 2 has notably different patterns from other clusters, as patients tend to stay in State 4 but are more likely to transition to State 4 from State 3—it is believed that those patients are likely in a progression from middle to end-stage COPD.

Because of the data structure, with different start times and observations times for each subject, and the latent nature of the modelled process, it is challenging to demonstrate the clustering of individual trajectories by using conventional plots of longitudinal curves, so other summaries must suffice. The plots of transition probabilities that we previously discussed reveal some but not all of the differences between clusters. To examine further the distinction between clusters, Figure 3 displays the real component of the three (nontrivial) eigenvalues of the posterior sampled $Q$ matrices in the four-cluster realizations. In both plots, it is clear that although there is some overlap for the second and third eigenvalues between Clusters 1 and 3, they are nonetheless distinct. This evidence, coupled with the differences in outcome response levels indicated by the results in tables of the posterior mean of the exponential of the coefficients—on average, Cluster 1 has a greater number of drugs prescribed, indicating that patients in Cluster 1 suffer from more severe illness than those in Cluster 3—confirms the presence of population substructure.

Table 4 displays the co-clustering matrix for our two methods of clustering, which consistently grouped the majority of patients into the same cluster. Clusters 1 and 3 showed some overlap in Figure 3, while the DMM resulted in a more distinct separation. This is because a high percentage of the subjects in Cluster 1 from the DMM was grouped in Cluster 3 in the finite mixture model, which resulted in a less distinct separation in the left panel of Figure 3. As we previously reported in the simulation studies, the discrepancy in cluster memberships of two methods mainly arises from the updating mechanism for the number of clusters and the cluster membership of the two algorithms. Reversible-jump MCMC updates the cluster membership in each iteration after the split/combine move of the number of clusters. However, split–merge MCMC updates both
Table 4: COPD data analysis: Co-clustering matrix for the two clustering methods.

|                  | Finite mixture model | Dirichlet mixture model |
|------------------|----------------------|-------------------------|
|                  | Cluster 1 | Cluster 2 | Cluster 3 | Cluster 4 | Cluster 1 | Cluster 2 | Cluster 3 | Cluster 4 |
| Cluster 1        | 1002      | 563       | 584       | 697       | 1002      | 563       | 584       | 697       |
| Cluster 2        | 322       | 9950      | 2421      | 1005      | 322       | 9950      | 2421      | 1005      |
| Cluster 3        | 535       | 1077      | 2903      | 847       | 535       | 1077      | 2903      | 847       |
| Cluster 4        | 249       | 654       | 613       | 1290      | 249       | 654       | 613       | 1290      |

Note: Cluster membership determined by the posterior modal cluster.

8. DISCUSSION

We developed and implemented model-based clustering procedures for health trajectories based on CTHMMs involving both finite and infinite mixture models. The methodology was applied to simulated examples where the Markov transition rate matrices dictated that each cluster had its own transition characteristics and observation process. The posterior distribution for the cluster labels was computed by reversible-jump MCMC (Green, 1995) for finite mixture models and via Pólya urn schemes (Neal, 2000) and more efficient split–merge updates (Jain & Neal, 2004) for DMMs. These simulation studies demonstrated that both types of mixture models could identify the correct number of clusters and sample the target posterior distribution. Only the conjugate DP mixture case was explored in this article; for handling the nonconjugate case, an auxiliary variable method (Neal, 2000, Algorithm 8) or nonconjugate split–merge proposals (Jain & Neal, 2007) could be used. This would allow us to incorporate continuous time-varying covariates in the observation process and baseline covariates of general form in \( Q \). Alternatively, if log-linear specifications are used in latent and observation models, an approximate conjugate
analysis could be carried out using Gaussian approximations. In our illustrative analyses, we have examined a cohort of over 24,000 patients involving nearly 1 million records. This analysis demonstrates the computing capability of our proposed algorithm. The main computational obstacle is the imputation of the latent process, which is needed to facilitate all aspects of the likelihood calculation; this computational hurdle may, in part, be overcome by parallelization of the individual likelihood calculations.

Focusing on the number of clusters, the Dirichlet precision parameter $\alpha$ and the prior distribution on the component-specific parameters could both influence the number of inferred clusters—this phenomenon is inevitable in the context of the unsupervised learning problem to which model-based clustering corresponds. For example, a more concentrated prior on the components of $Q$ will encourage more clusters for the same observed data. Similarly, larger values of $\alpha$ will encourage more clusters; it is also straightforward to treat $\alpha$ as an unknown parameter in the MCMC algorithm, although we did not pursue that approach here. In this article, we have not explored an approach that allows the number of states to vary, but that is also possible. An efficient construction of proposal distributions is required in order to allow the sampler to move around and explore the parameter space for both the number of clusters and the number of states (Brooks, Giudici & Roberts, 2003). We have extended our algorithm to allow the number of states to be inferred using the reversible-jump approach, which we have reported elsewhere (see Luo & Stephens, 2021).

In Luo et al. (2021), other potential extensions of the model are discussed, and we recap them here. We have not addressed issues that involve informative dropout; in principle, dropout can be handled using standard Bayesian missing data procedures once a suitable missingness mechanism has been proposed, although the specification of a realistic model may be challenging to construct. In the case of the COPD data, the influence of informative dropout is likely to be minimal, as the principal cause of dropout is subjects leaving the province, which mainly affects younger subjects who were excluded from our analysis. Second, censoring due to death of the subject can be handled within the framework of the latent HMM by including an absorbing state; however, information on the time of death is not available in our case study dataset since the cohort contains data involving interactions with the healthcare system. This shortcoming could be addressed by linking the RAMQ data to provincial death records. In this article, we assumed that the sequence of observation times $t_t, t = 1, \ldots, T$ was not informative about the underlying Markov chain or the outcome process, but the model could be extended to account for informative observation times by introducing a further stochastic process, for example, an inhomogeneous Poisson process with rate dependent on the latent HMM. This extension would complicate the computation and very likely be prohibitive for large datasets, but would still be feasible in principle.

ACKNOWLEDGEMENTS

We would like to thank two anonymous referees for their constructive comments, which have greatly improved the contents and presentation of this article. This research was supported by the Canadian Institutes of Health Research and the Natural Sciences and Engineering Research Council of Canada.

REFERENCES

Antoniak, C. E. (1974). Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems. *The Annals of Statistics*, 2, 1152–1174.

Bicego, M., Murino, V., & Figueiredo, M. A. (2003). Similarity-based clustering of sequences using hidden Markov models. In Pernier, P. & Rosenfeld, A. (Eds.) *Machine Learning and Data Mining in Pattern Recognition*, Springer, Berlin, 86–95.

Billingsley, P. (1961). Statistical methods in Markov chains. *The Annals of Mathematical Statistics*, 32, 12–40.
Bladt, M. & Sørensen, M. (2005). Statistical inference for discretely observed Markov jump processes. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67, 395–410.

Blais, C., Jean, S., Sirois, C., Rochette, L., Plante, C., Larocque, I., & Doucet, M. (2014). Quebec integrated chronic disease surveillance system (QICDSS), an innovative approach. *Chronic Diseases and Injuries in Canada*, 34, 226–235.

Brooks, S. P., Giudici, P., & Roberts, G. O. (2003). Efficient construction of reversible jump Markov chain Monte Carlo proposal distributions (with discussion). *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 65, 3–55.

Carlin, B. P. & Chib, S. (1995). Bayesian model choice via Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 57, 473–484.

Celeux, G., Hurn, M., & Robert, C. P. (2000). Computational and inferential difficulties with mixture posterior distributions. *Journal of the American Statistical Association*, 95, 957–970.

Crayen, C., Eid, M., Lischetzke, T., Courvoisier, D. S., & Vermunt, J. K. (2012). Exploring dynamics in mood regulation-mixture latent Markov modeling of ambulatory assessment data. *Psychosomatic Medicine*, 74, 366–376.

Dellaportas, P. & Papageorgiou, I. (2006). Multivariate mixtures of normals with unknown number of components. *Statistics and Computing*, 16, 57–68.

Escobar, M. D. & West, M. (1995). Bayesian density estimation and inference using mixtures. *Journal of the American Statistical Association*, 90, 577–588.

Ferguson, T. S. (1973). A Bayesian analysis of some nonparametric problems. *The Annals of Statistics*, 1, 209–230.

Fraley, C. & Raftery, A. E. (1998). How many clusters? Which clustering method? Answers via model-based cluster analysis. *The Computer Journal*, 41, 578–588.

Godsill, S. J. (2001). On the relationship between Markov chain Monte Carlo methods for model uncertainty. *Journal of Computational and Graphical Statistics*, 10, 230–248.

GOLD Executive Committee. (2017). *Pocket guide to COPD diagnosis, management and prevention: A guide for health care professionals*, 2017 report. http://goldcopd.org/wp-content/uploads/2016/12/wms-GOLD-2017-Pocket-Guide.pdf.

Green, P. J. (1995). Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82, 711–732.

Green, P. J. & Richardson, S. (2001). Modelling heterogeneity with and without the Dirichlet process. *Scandinavian Journal of Statistics*, 28, 355–375.

Hobolth, A. & Stone, E. A. (2009). Simulation from endpoint-conditioned, continuous-time Markov chains on a finite state space, with applications to molecular evolution. *The Annals of Applied Statistics*, 3, 1204–1231.

Ishwaran, H. & James, L. F. (2001). Gibbs sampling methods for stick-breaking priors. *Journal of the American Statistical Association*, 96, 161–173.

Jackson, C. H., Sharples, L. D., Thompson, S. G., Duffy, S. W., & Couto, E. (2003). Multistate Markov models for disease progression with classification error. *Journal of the Royal Statistical Society: Series B (The Statistician)*, 52, 193–209.

Jain, S. & Neal, R. M. (2004). A split-merge Markov chain Monte Carlo procedure for the Dirichlet process mixture model. *Journal of Computational and Graphical Statistics*, 13, 158–182.

Jain, S. & Neal, R. M. (2007). Splitting and merging components of a non-conjugate Dirichlet process mixture model. *Bayesian Analysis*, 2, 445–472.

Jebara, T., Song, Y., & Thadani, K. (2007). Spectral clustering and embedding with hidden Markov models. *European Conference on Machine Learning*, Springer, Berlin, 164–175.

Lange, J. M., Hubbard, R. A., Inoue, L. Y., & Minin, V. N. (2015). A joint model for multistate disease processes and random informative observation times, with applications to electronic medical records data. *Biometrics*, 71, 90–101.

Lix, L., Puchtinger, R., Reimer, K., Robitaille, C., Smith, M., Svenson, L., Tu, K., et al. (2018). The Canadian chronic disease surveillance system: A model for collaborative surveillance. *International Journal of Population Data Science*, 3. doi:10.23889/ijpds.v3i3.433

Lo, A. Y. (1984). On a class of Bayesian nonparametric estimates: I. Density estimates. *The Annals of Statistics*, 12, 351–357.
Luo, Y. (2019). *Latent multi-state models for non-equidistant longitudinal observations with finite and infinite mixture model-based clustering* (PhD thesis). McGill University.

Luo, Y. & Stephens, D. A. (2021). Bayesian inference for continuous-time hidden Markov models with an unknown number of states. *Statistics and Computing*, 31, 57.

Luo, Y., Stephens, D. A., Verma, A., & Buckeridge, D. L. (2021). Bayesian latent multi-state modeling for non-equidistant longitudinal electronic health records. *Biometrics*, 77, 78–90.

MacEachern, S. N. & Müller, P. (1998). Estimating mixtures of Dirichlet process models. *Journal of Computational and Graphical Statistics*, 7, 223–238.

Maruotti, A. & Rydén, T. (2009). A semiparametric approach to hidden Markov models under longitudinal observations. *Statistics and Computing*, 19, 381–393.

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

Neal, R. M. (2000). Markov chain sampling methods for Dirichlet process mixture models. *Journal of Computational and Graphical Statistics*, 9, 249–265.

Panuccio, A., Bicego, M., & Murino, V. (2002). A hidden Markov model-based approach to sequential data clustering. In Caelli, T., Amin, A., Duin, R. P. W., de Ridder, D., & Kamel, M. (Eds.) *Structural, Syntactic, and Statistical Pattern Recognition*, Springer, Berlin, 734–743.

Richardson, S. & Green, P. J. (1997). On Bayesian analysis of mixtures with an unknown number of components (with discussion). *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 59, 731–792.

Smyth, P. (1997). Clustering sequences with hidden Markov models. *Advances in Neural Information Processing Systems*, 648–654.

Williams, J. P., Storlie, C. B., Therneau, T. M., Jack, C. R., Jr., & Hannig, J. (2020). A Bayesian approach to multistate hidden Markov models: Application to dementia progression. *Journal of the American Statistical Association*, 115, 16–31.

Zucchini, W., MacDonald, I. L., & Langrock, R. (2017). *Hidden Markov Models for Time Series: An Introduction Using R*. CRC Press, New York.

Received 1 March 2021
Accepted 6 September 2021