Journal of Machine Learning Research 27 (2026) 1-99 Submitted 3/25; Revised 10/25; Published 2/26

Covariate-dependent Hierarchical Dirichlet Processes

Huizi Zhang H.Zhang-144@sms.ed.ac.uk
School of Mathematics and Maxwell Institute for Mathematical Sciences
University of Edinburgh
Edinburgh, EH9 3FD, UK

Sara Wade sara.wade@ed.ac.uk
School of Mathematics and Maxwell Institute for Mathematical Sciences
University of Edinburgh
Edinburgh, EH9 3FD, UK

Natalia Bochkina N.Bochkina@ed.ac.uk
School of Mathematics and Maxwell Institute for Mathematical Sciences
University of Edinburgh
Edinburgh, EH9 3FD, UK

Editor: Brian Kulis

Abstract

Bayesian hierarchical modeling is a natural framework to effectively integrate data and borrow information across groups. In this paper, we address problems related to density estimation and identifying clusters across related groups, by proposing a hierarchical Bayesian approach that incorporates additional covariate information. To achieve flexibility, our approach builds on ideas from Bayesian nonparametrics, combining the hierarchical Dirichlet process with dependent Dirichlet processes. The proposed model is widely applicable, accommodating multiple and mixed covariate types through appropriate kernel functions as well as different output types through suitable component-specific likelihoods. This extends our ability to discern the relationship between covariates and clusters, while also effectively borrowing information and quantifying differences across groups. By employing a data augmentation trick, we are able to tackle the intractable normalized weights and construct a Markov chain Monte Carlo algorithm for posterior inference. The proposed method is illustrated on simulated data and two real data sets on single-cell RNA sequencing (scRNA-seq) and calcium imaging. For scRNA-seq data, we show that the incorporation of cell dynamics facilitates the discovery of additional cell subgroups. On calcium imaging data, our method identifies interpretable clusters of time frames with similar neural activity, aligning with the observed behavior of the animal.

Keywords: clustering, hierarchical model, dependent mixture model, Bayesian nonparametrics, Markov chain Monte Carlo

1. Introduction

In many studies, multiple data sets are collected across groups, where each group may represent multiple experiments, geographical sites, time points, and more. To effectively integrate and borrow information across groups, the Bayesian hierarchical framework is a natural choice. This paper delves into the problems of density estimation and clustering

across related groups, proposing a Bayesian approach that extends existing approaches in the presence of additional covariate information.

With advancements in data acquisition, the need for such tools is growing. For example, in biological studies, complex data are typically collected across multiple groups, which may correspond to repeated experiments or different treatment conditions, tissues, or time points. Moreover, additional side information is usually also available for each observation, such as patient characteristics in bulk RNA studies or cellular dynamics (La Manno et al., 2018) in single-cell RNA sequencing (scRNA-seq). Indeed, the lack of tools for integrating and quantifying differences across multiple single-cell data sets was emphasized as one of the grand challenges in single-cell data science (Lähnemann et al., 2020). Another example is information retrieval scenarios dealing with raw documents from multiple corpora, with externally observed categorical information (Kim and Oh, 2014).

In such settings, we focus on two problems. On one hand, clustering is important to uncover inherent structure. For example, in scRNA-seq, clustering is used to disentangle the heterogeneous gene expression measurements across cells and discover cell subtypes with similar expression patterns. By including cellular-level covariate information, such as dynamics, we can quantify the relationship between cell subtypes and dynamics. On the other hand, covariate-dependent density estimation (also known as density regression) allows modeling the whole density of the response, not only the mean, to change with the covariates. For example, in bulk-RNA studies, we may be interested in understanding how the distribution of expression for certain genes changes across different patients characteristics, while also borrowing information across multiple sites.

Mixture models (Fruhwirth-Schnatter et al., 2019) arise as a natural choice in this context, providing both a probabilistic framework for clustering as well as a useful tool for density estimation due to their attractive balance between smoothness and flexibility. As an alternative to parametric mixture models, Bayesian nonparametric (BNP) methods (Ghosal and van der Vaart, 2017) are widely used to avoid pre-specifying the number of clusters, instead allowing it to grow unboundedly with the number of observations, by placing a nonparametric prior on the unknown mixing measure. Moreover, BNP mixture models are further supported by strong theoretical properties that provide frequentist validation (e.g. Ghosal et al., 1999; Ghosal and van der Vaart, 2001; Wu and Ghosal, 2010; Shen et al., 2013). The Dirichlet process (DP) (Ferguson, 1973) is unarguably the most common prior choice in BNP literature and has many desirable properties including easy elicitation of its parameters, conjugacy, large support, and posterior consistency (Ghosal and van der Vaart, 2017, Chapter 4).

To effectively model more complex data structures, a number of extensions of the Dirichlet process have been proposed. Recent reviews and comparisons of dependent extensions to accommodate covariates are provided in Quintana et al. (2022) and Wade and Inácio (2025). Two of the most widely-used proposals include the dependent Dirichlet process (DDP) (MacEachern, 1999) and the hierarchical Dirichlet process (HDP) (Teh et al., 2006). The latter concentrates exclusively on *partially exchangeable data*, when covariates represent groups and exchangeability holds within group and across group labels (e.g. for observations on patients across different hospitals, the ordering of the patients and hospitals can be shuffled, as long as the same patients belong within each hospital). Thus, the HDP enables clustering and density estimation across related groups, and moreover allows prediction for

new groups. In contrast, the DDP incorporates fixed effects of covariates for conditional density estimation and quantifying covariate effects on clustering.

In this article, we propose a covariate-dependent hierarchical Dirichlet process (C-HDP), combining the hierarchical Dirichlet process with the dependent Dirichlet process. Our focus is the DDP that uses normalized weights and kernels to construct covariate-dependent weights (Foti and Williamson, 2012; Antoniano-Villalobos et al., 2014), as they have the flexibility to recover a variety of complex data-generating scenarios and enhanced interpretability through the normalized constructions (Wade and Inácio, 2025). External covariates can be flexibly incorporated to facilitate clustering across groups, as well as density regression, through the use of various kernel functions. Additionally, the C-HDP can account for different response types through the choice of component-specific likelihoods. The proposed method holds utility in various settings. For efficient inference, we construct a novel Markov Chain Monte Carlo (MCMC) algorithm that employs latent variables to cope with the intractable normalized weights. We demonstrate that our model can capture the relationship between clusters and covariates, and identify meaningful clusters across groups in both simulated and real data.

The paper is organized as follows. We commence by providing a review of the DP and its extensions DDP and HDP in Section 2. Section 3 outlines the definition of the covariate-dependent HDP, examples of common component-specific likelihood and kernel functions. The details of posterior inference are presented in Section 4. Section 5 provides a simulation study highlighting the advantages of combining the HDP and DDP. In Section 6, we showcase the application of C-HDP to two real-world data sets from scRNA-seq and calcium imaging, respectively. Section 7 concludes the paper and discusses potential future directions.

2. Review of Dirichlet Processes and Extensions

Our model is based on the Dirichlet process, which is reviewed in this section along with relevant extensions.

2.1 Dirichlet Processes

Let P denote a random probability measure on the parameter space Θ. The Dirichlet process (Ferguson, 1973) is the most commonly used prior for an unknown probability measure P in the Bayesian nonparametric literature, as it satisfies several desirable properties such as conjugacy, posterior consistency and large support (Ghosal and van der Vaart, 2017, Chapter 4). By definition, P is said to follow a DP prior with baseline probability measure P_0 and concentration parameter α, denoted by $P \sim \mathrm{DP}(\alpha, P_0)$, if for any finite measurable partition $\{A_1, \ldots, A_k\}$ of Θ,

$$(P(A_1), P(A_2), \ldots, P(A_k)) \sim \mathrm{Dir}(\alpha P_0(A_1), \alpha P_0(A_2), \ldots, \alpha P_0(A_k)),$$

where $\mathrm{Dir}(\alpha_1, \ldots, \alpha_k)$ denotes the Dirichlet distribution with concentration parameters $\alpha_1, \ldots, \alpha_k$. Beyond the DP, a number of other nonparametric priors have also been proposed and studied (Hjort, 1990; De Blasi et al., 2013; Lijoi and Prünster, 2011).

An important property of DP is the discrete nature of P (almost surely) such that P can be written as a combination of weights and point masses, $P = \sum_{j=1}^{\infty} p_j \delta_{\theta_j^*}$, where

p_j is the weight associated with component j and $\delta_{\theta_j^*}$ is the Dirac measure at the atom θ_j^*. The atoms are independent and identically distributed from P_0, and independent of the weights. The weights can be defined as normalized jumps of a Gamma process $\xi(t)$ for $0 \leq t \leq 1$ with shape parameter α, defined with $\xi(0) = 0$ such that the increments on disjoint intervals are independent and such that $\xi(t_2) - \xi(t_1)$ has distribution Gamma$(\alpha(t_2 - t_1), 1)$ for $0 \leq t_1 < t_2 \leq 1$. Thus, $P \sim \text{DP}(\alpha, P_0)$ can be represented as

$$P = \sum_{j=1}^{\infty} \frac{\Gamma_j}{\sum_{h=1}^{\infty} \Gamma_h} \delta_{\theta_j^*}, \quad \theta_j^* \overset{i.i.d}{\sim} P_0, \tag{1}$$

where Γ_j are the jumps of the Gamma process and $\sum_{j=1}^{\infty} \Gamma_j = \xi(1) - \xi(0) \sim \text{Gamma}(\alpha, 1)$ is finite almost surely. An alternative representation is given by the stick-breaking construction Sethuraman (1994) of the DP as follows:

$$\begin{gathered} P = \sum_{j=1}^{\infty} p_j \delta_{\theta_j^*}, \\ p_1 = v_1, \quad p_j = v_j \prod_{l<j} (1 - v_l) \quad \text{for } j > 1, \quad v_j \overset{i.i.d}{\sim} \text{Beta}(1, \alpha), \end{gathered} \tag{2}$$

where again $\theta_j^* \overset{i.i.d}{\sim} P_0$, and v_j is independent of θ_j^*. It can be shown that $\sum_{j=1}^{\infty} p_j = 1$ almost surely (Ishwaran and James, 2001).

If $\theta_1, \ldots, \theta_n$ represent draws from P, i.e. $\theta_i \overset{i.i.d}{\sim} P$ $(i = 1, \ldots, n)$, then the discreteness of P implies there are ties among $\theta_1, \ldots, \theta_n$. Thus, the DP itself is typically not directly used to model the unknown data-generating distribution, but instead is convoluted with a parametric likelihood $f(y|\theta)$ on the sample space $\mathcal{Y}$, to produce a DP mixture model (Lo, 1984):

$$y_i | P \overset{i.i.d}{\sim} \int f(y_i|\theta) dP(\theta), \quad P \sim \text{DP}(\alpha, P_0).$$

Hierarchically, this can be expressed as

$$y_i | \theta_i \overset{ind}{\sim} f(y_i|\theta_i), \quad \theta_i | P \overset{i.i.d}{\sim} P, \quad P \sim \text{DP}(\alpha, P_0).$$

This model not only allows flexible density estimation, but ties among the observation-specific parameters $(\theta_1, \ldots, \theta_n)$ induce a latent clustering of the data, where two points belong to the same cluster if they are generated from the same mixture component, i.e. i and i' are in the same cluster if $\theta_i = \theta_{i'}$. Different choices of component-specific likelihoods can be used, reflecting the shape and interpretation of a cluster for the task at hand (e.g. Frühwirth-Schnatter and Pyne, 2010; Blei et al., 2003; Wu and Luo, 2022). The DP does not require the specification of the number of clusters, which is instead data-driven and can grow with the sample size.

2.2 Dependent Dirichlet Processes

When there is an exogenous covariate x, the random probability measure can be augmented to depend on x, denoted as P_x. A popular method is the dependent Dirichlet process (DDP)

which was first introduced by MacEachern (1999). The random probability measure P_x is constructed following a similar stick-breaking representation to Equation (2), which in full generality employs stochastic processes to model the covariate-dependent atoms $\theta_j^*(x)$ and stick-breaking proportions $v_j(x)$. The DP can be considered as a special case when the weights and atoms are independent of the covariate x. In the context of mixture models, the response y has the following conditional density

$$f(y|x, P_x) = \int f(y|x, \theta) dP_x(\theta) = \sum_{j=1}^{\infty} p_j(x) f(y|x, \theta_j^*(x)),$$

where $P_x = \sum_{j=1}^{\infty} p_j(x)\delta_{\theta_j^*(x)}$.

A common simplified DDP model is the "single-weights" DDP where only the atoms depend on the covariates, $P_x = \sum_{j=1}^{\infty} p_j \delta_{\theta_j^*(x)}$. It has been used in the context of ANOVA (De Iorio et al., 2004) for categorical covariates (ANOVA-DDP). It can be generalized to linear combinations of general types of covariates, referred to as the linear DDP (LDDP) (De Iorio et al., 2009).

In contrast, the "single-atoms" DDP assumes only covariate-dependent weights, $P_x = \sum_{j=1}^{\infty} p_j(x)\delta_{\theta_j^*}$. Constructing the covariate-dependent weights is not trivial, as one must ensure they are positive and sum to one almost surely for all $x \in \mathcal{X}$. The stick-breaking construction is the most common approach (e.g. Dunson and Park, 2008; Rodriguez and Dunson, 2011), but other methods are available including normalization (e.g. Foti and Williamson, 2012; Antoniano-Villalobos et al., 2014). The approach of Foti and Williamson (2012), which has been applied in Rao and Teh (2009) for spatial applications, is based on the normalized gamma process representation of the DP and the covariate-dependent weights are constructed using bounded kernel functions on the unit interval. Instead, Antoniano-Villalobos et al. (2014) use a parametric density function and the stick-breaking representation. These two methods allow for different types of kernel functions or density functions to induce dependence without significant modification of the sampler. For a comprehensive review of the DDP, we refer the reader to Quintana et al. (2022) and Wade and Inácio (2025).

2.3 Hierarchical Dirichlet Processes

The hierarchical Dirichlet process (Teh et al., 2006) focuses exclusively on the partially exchangeable setting, where covariates represent groups or data sets. For the i-th observation in the d-th data set $y_{i,d}$ ($i = 1, \ldots, n_d, d = 1, \ldots, D$), and a parametric density $f(y|\theta)$ on the sample space $\mathcal{Y}$ with parameters $\theta \in \Theta$, the HDP assumes the following hierarchical structure

$$\begin{aligned} y_{i,d}|\theta_{i,d} &\overset{ind}{\sim} f(y_{i,d}|\theta_{i,d}), \quad \theta_{i,d}|P_d \overset{ind}{\sim} P_d, \\ P_d|\alpha, P &\overset{ind}{\sim} \mathrm{DP}(\alpha, P), \quad P|\alpha_0, P_0 \sim \mathrm{DP}(\alpha_0, P_0). \end{aligned}$$

Here, another layer of the DP prior is included to borrow strength across groups. Specifically, each group has its own mixing measure P_d which are all apriori centered on the unknown global mixing measure P. The HDP can be defined through hierarchical normalized Gamma processes (Argiento et al., 2020). In particular, letting $\xi_d(t)$ denote the

group-specific Gamma processes, since the base measure P of each group-specific mixing measure P_d is discrete, we have the representation (Kingman, 1975):

$$P_d = \sum_{j=1}^{\infty} \frac{\xi_d(\sum_{l=1}^{j} p_l) - \xi_d(\sum_{l=1}^{j-1} p_l)}{\sum_{h=1}^{\infty} \xi_d(\sum_{l=1}^{h} p_l) - \xi_d(\sum_{l=1}^{h-1} p_l)} \delta_{\theta_j^*} = \sum_{j=1}^{\infty} \frac{\Gamma_{j,d}}{\sum_{h=1}^{\infty} \Gamma_{h,d}} \delta_{\theta_j^*} \tag{3}$$

where $\theta_j^* \overset{i.i.d}{\sim} P_0$; $(\Gamma_{j,d})$ are independent with Gamma$(\alpha p_j, 1)$ distribution; and the p_j are the normalized jumps of the Gamma process (Equation 1) with shape parameter α_0. In the following exposition, we denote a draw from the HDP prior as $P_d \sim \text{HDP}(\alpha_0, \alpha, P_0)$. This construction highlights that different measures P_d share the same atoms θ_j^* but assign different weights to the atoms. In the context of clustering, this is important as it allows clusters (data points that share the same parameters) to be potentially shared across groups, but allows the weight (or size) of the cluster to vary across groups.

Recent research on leveraging predictors in HDP is also available. Dai and Storkey (2014) developed the supervised HDP for topic modelling that can predict continuous or categorical response associated with each document (group) using generalized linear models. The hierarchical Dirichlet scaling process (Kim and Oh, 2014) considers documents with observed labels, and topic proportions are modelled dependent on the distance between the latent locations of the observed labels and topics. Ren et al. (2008) extend HDP to dynamic HDP for time-evolving data, assuming that adjacent groups collected closer in time are more likely to share components. Ren et al. (2011) incorporate spatial-temporal information using a kernel logistic regression. Diana et al. (2020) propose the hierarchical dependent Dirichlet process (HDDP), combining HDP and "single-weights" DDP.

Lastly, while not the focus of this article, we briefly mention the nested DP (Rodriguez et al., 2008; Camerlenghi et al., 2019) which can also be used for clustering data across groups, but uses a multi-level structure with also a latent clustering of groups. Various extensions of both the HDP and nested DP have been proposed, including the semi-HDP (Beraha et al., 2021), the hidden HDP (Lijoi et al., 2022) and the common atoms model (Denti et al., 2023).

3. Covariate-dependent Nonparametric Models

Real-world data sets often encompass various types of covariates for statistical modeling, in addition to collecting data across multiple groups. In order to construct a flexible BNP modeling framework for such data, we first propose a novel covariate-dependent HDP in Section 3.1. To flexibly model the conditional density and cluster observations across groups, the C-HDP can be used as a prior for covariate- and group-dependent mixing measures in mixture models (Section 3.2), where we illustrate examples of kernel functions to introduce dependence on the covariate. Additionally, the C-HDP can be applied to different types of response, depending on the nature of the data and the task at hand, as detailed in Section 3.3 where widely-used component-specific likelihoods are presented.

3.1 Covariate-dependent Hierarchical Dirichlet Processes

We construct the covariate-dependent HDP that borrows ideas from the "single-atoms" DDP and HDP, in order to model an unknown probability measure $P_{x,d}$ that is indexed by

both the covariate x and group index d. Recalling that the HDP assumes $P_d = \sum_{j=1}^{\infty} p_{j,d}\delta_{\theta_j^*}$, with $p_{j,d} = \Gamma_{j,d}/\sum_{k=1}^{\infty}\Gamma_{k,d}$ defined through the normalized construction in Equation (3), we propose to introduce dependence by defining the mixture weights for each group to be functions of the covariate x, leading to

$$P_{x,d} = \sum_{j=1}^{\infty} p_{j,d}(x)\delta_{\theta_j^*}, \quad P_x = \sum_{j=1}^{\infty} p_j(x)\delta_{\theta_j^*}, \tag{4}$$

with $\theta_j^* \overset{i.i.d}{\sim} P_0$. Specifically, the covariate-dependent weights of both the group-specific and global mixing measures are defined based on a normalized construction as

$$p_{j,d}(x) = \frac{\Gamma_{j,d}K(x|\boldsymbol{\psi}_{j,d}^*)}{\sum_{k=1}^{\infty}\Gamma_{k,d}K(x|\boldsymbol{\psi}_{k,d}^*)}, \quad p_j(x) = \frac{\Gamma_j K(x|\boldsymbol{\psi}_j^*)}{\sum_{k=1}^{\infty}\Gamma_k K(x|\boldsymbol{\psi}_k^*)} \tag{5}$$

where $\Gamma_{j,d}$ are i.i.d Gamma$(\alpha p_j, 1)$ as in Equation (3) with $p_j = \Gamma_j/\sum_{k=1}^{\infty}\Gamma_k$, and $K(x|\boldsymbol{\psi})$ is a kernel function relying on kernel parameters $\boldsymbol{\psi}$ which may be group- and component-specific and satisfies $0 \leq K(x|\boldsymbol{\psi}) < c$ for some constant c and for every x, at least one component j satisfies $K(x|\boldsymbol{\psi}_{j,d}^*) > 0$ at the group level and $K(x|\boldsymbol{\psi}_j^*) > 0$ at the global level (ensuring that the normalizing constant is finite almost surely). Note that $\{\theta_j^*\}_{j=1}^{\infty}$, $\{\Gamma_{j,d}\}_{j=1,d=1}^{\infty,D}$ and $\{\boldsymbol{\psi}_{j,d}^*\}_{j=1,d=1}^{\infty,D}$ are independent of each other. In addition, hierarchical priors are assigned to the kernel parameters to borrow strength across groups, such that the $\boldsymbol{\psi}_{j,d}^*$ are independent across components ($j = 1, \ldots, \infty$) and are conditionally independent across data sets ($d = 1, \ldots, D$), centered around the global kernel parameter $\boldsymbol{\psi}_j^*$. An advantage of this hierarchical formulation is that it provides a natural, parsimonious framework to account for differences in the covariate effects on the weights across groups. Examples depend on the choice of component-specific likelihood (e.g. see Section 3.3).

The normalized construction of the covariate-dependent weights is motivated by Foti and Williamson (2012) and Antoniano-Villalobos et al. (2014). Compared to alternative methods (e.g. stick-breaking as shown in Dunson and Park, 2008; Griffin and Steel, 2011; Ren et al., 2011; Rigon and Durante, 2021), the normalized construction has the advantage of enhanced interpretability with $p_{j,d} = \Gamma_{j,d}/\sum_{k=1}^{\infty}\Gamma_{k,d}$ representing the probability that an observation in group d is generated from component j omitting the covariate value, and the kernel represents how likely an observation from group d that is generated from component j will take the value x. Thus, this enhanced interpretability allows for more subjective or empirical specification of parameters (Wade and Inácio, 2025). On the other hand, stick-breaking constructions suffer from difficulty in selecting hyperparameters, which can significantly influence the functional shape of the weights and model performance. Figure 1 illustrates how C-HDP combines the hierarchical framework of the HDP (left) with the covariate dependence of the DDP (middle) to allow for shared clusters across groups with weights that both vary across groups and change smoothly with covariates (right).

We remark that our C-HDP prior differs from the hierarchical dependent Dirichlet process prior in Diana et al. (2020) which combines the "single-weights" DDP and HDP instead. In particular, in Diana et al. (2020) the covariate x is introduced in the global measure P instead of data-specific DPs P_d as we have done, and therefore in Diana et al. (2020) the influence of the covariate is the same across data sets, whilst the effect is allowed to be different in our C-HDP model.

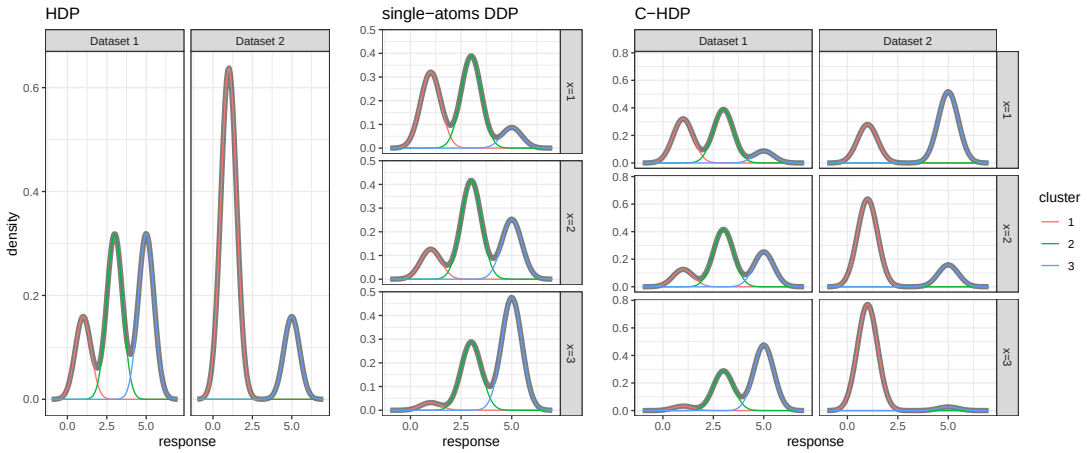


Figure 1: A demonstration of three nonparametric priors: HDP, DDP and C-HDP. The colored lines denote weighted conditional density for each cluster, with marginal density function shown in black. The HDP (left) allows for shared clusters across groups. The "single-atoms" DDP (middle) allows the cluster weights to vary smoothly across covariate values. The C-HDP (right) combines the HDP and DDP for shared clustering across groups, with smoothly varying covariate-dependent weights.

3.1.1 Finite-dimensional Truncation

For computational purposes, a finite-dimensional truncation is useful. First, a finite-dimensional truncation of the global mixing measure P can be obtained from the normalized Gamma process construction. It is defined for truncation level J by considering a discretization $(0, 1/J, 2/J, \ldots, 1)$ of the domain of the Gamma process:

$$P^J = \sum_{j=1}^{J} \frac{\xi(j/J) - \xi((j-1)/J)}{\sum_{h=1}^{J} \xi(h/J) - \xi((h-1)/J)} \delta_{\theta_j^*} = \sum_{j=1}^{J} \frac{q_j^J}{\sum_{h=1}^{J} q_h^J} \delta_{\theta_j^*},$$

where by definition of the Gamma process $q_j^J \overset{i.i.d}{\sim} \text{Gamma}(\alpha_0/J, 1)$ and P^J converges weakly to the DP (Kingman, 1975). Thus, a finite-dimensional truncation in the hierarchical setting, which converges weakly to the HDP (Teh et al., 2006), is similarly obtained as

$$P_d^J = \sum_{j=1}^{J} \frac{q_{j,d}^J}{\sum_{h=1}^{J} q_{h,d}^J} \delta_{\theta_j^*}. \quad q_{j,d}^J \overset{ind}{\sim} \text{Gamma}(\alpha p_j^J, 1),$$

where $p_j^J = q_j^J / \sum_{h=1}^{J} q_h^J$ and $q_j^J \overset{i.i.d}{\sim} \text{Gamma}(\alpha_0/J, 1)$. This allows us to construct a finite-dimensional truncation for C-HDP

$$P_{x,d}^J = \sum_{j=1}^{J} p_{j,d}^J(x) \delta_{\theta_j^*}, \quad P_x^J = \sum_{j=1}^{J} p_j^J(x) \delta_{\theta_j^*}, \tag{6}$$

where

$$p_{j,d}^J(x) = \frac{q_{j,d}^J K(x|\boldsymbol{\psi}_{j,d}^*)}{\sum_{k=1}^J q_{k,d}^J K(x|\boldsymbol{\psi}_{k,d}^*)}, \quad p_j^J(x) = \frac{p_j^J K(x|\boldsymbol{\psi}_j^*)}{\sum_{k=1}^J p_k^J K(x|\boldsymbol{\psi}_k^*)}, \tag{7}$$

with $q_{j,d}^J \sim \text{Gamma}(\alpha p_j^J, 1)$ and $(p_1^J, \ldots, p_J^J) \sim \text{Dir}(\alpha_0/J, \ldots, \alpha_0/J)$. For notational simplicity, we will drop the superscript J, when the context is clear.

The weights $p_{j,d} = \Gamma_{j,d}/\sum_{k=1}^\infty \Gamma_{k,d}$ of the C-HDP in Equation (5) could alternatively be constructed based on the hierarchical stick-breaking representation of the HDP (Teh et al., 2006). This leads to an equivalent nonparametric process with the same law, however, the truncated approximations differ. Our focus is the finite-dimensional approximation, which, in contrast to the stick-breaking truncation, has nice features such as exchangeability of the weights (Ishwaran and Zarepour, 2002); therefore, label-switching moves which are required for posterior exploration of the stochastically ordered stick-breaking weights (Liverani et al., 2015, Section 6) are not necessary. Moreover, as shown in Catalano and Lavenant (2024), the discrepancy between the DP and the stick-breaking truncation depends on the value of α, and for a diverging sequence of α it may fail to converge. Instead, the finite-dimensional approximation has polynomial convergence rate in J which does not depend on α.

3.2 Examples of Kernels for Dependent Weights

The choice of kernel has an important role in defining the dependence structure in the weights. Depending on the characteristics of the data and application, different kernels may be appropriate. Below we provide a few examples of the kernel functions that will be used in the paper along with a description of the type of dependence implied.

Gaussian Kernel.

$$K(x|\boldsymbol{\psi}_{j,d}^*) = \exp\left(-\frac{\left(x - x_{j,d}^*\right)^2}{2{\sigma_{j,d}^*}^2}\right),$$

where $\boldsymbol{\psi}_{j,d}^* = (x_{j,d}^*, {\sigma_{j,d}^*}^2) \in \mathbb{R} \times \mathbb{R}^+$. The parameter $x_{j,d}^*$ describes the value in the covariate space where the j-th component best applies for the d-th group, and ${\sigma_{j,d}^*}^2$ affects the sharpness of the boundary between the covariate regions associated to each component. Smaller values for ${\sigma_{j,d}^*}^2$ lead to more drastic change in the weights (Figure 2 middle). A Gaussian kernel is recommended when we believe the covariate space can be partitioned into well-behaved regions. Notice that when ${\sigma_{j,d}^*}^2 \to \infty$ for all j and d, the C-HDP reduces to the HDP.

Periodic Kernel.

$$K(x|\boldsymbol{\psi}_{j,d}^*) = \exp\left(-\frac{2}{{\sigma_{j,d}^*}^2}\sin^2\left(\frac{x - x_{j,d}^*}{\lambda_{j,d}^*}\right)\right),$$

where $\boldsymbol{\psi}_{j,d}^* = (x_{j,d}^*, {\sigma_{j,d}^*}^2, \lambda_{j,d}^*) \in \mathbb{R} \times \mathbb{R}^+ \times \mathbb{R}^+$. Note that for identifiability, $x_{j,d}^*$ needs to be restricted within one period $(\pi\lambda_{j,d}^*)$. Figure 3 (left) shows the periodic kernel under different parameter values. The parameter $x_{j,d}^*$ represents the value that maximizes the kernel and changing $x_{j,d}^*$ will shift the kernel (red vs. green). The period is determined by

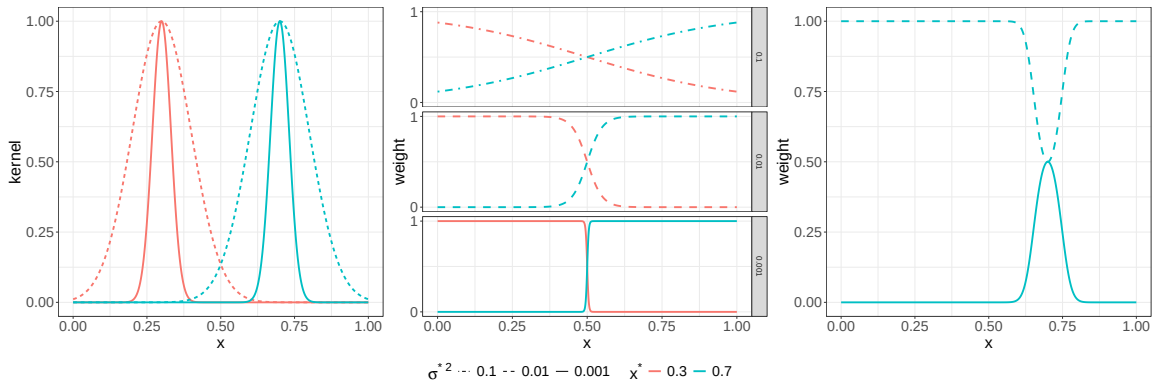


Figure 2: Left: Example of a Gaussian kernel for varying parameter values. Middle: Covariate-dependent weights for two components. In each row, both components have the same ${\sigma^*_{j,d}}^2$ but different $x^*_{j,d}$. When ${\sigma^*_{j,d}}^2$ decreases from top to bottom, the change of weights with respect to the covariate x becomes more abrupt, showing a sharp transition. Right: Covariate-dependent weights for two components. Both clusters have the same $x^*_{j,d}$, but different ${\sigma^*_{j,d}}^2$. The weights can exhibit a bimodal behavior.

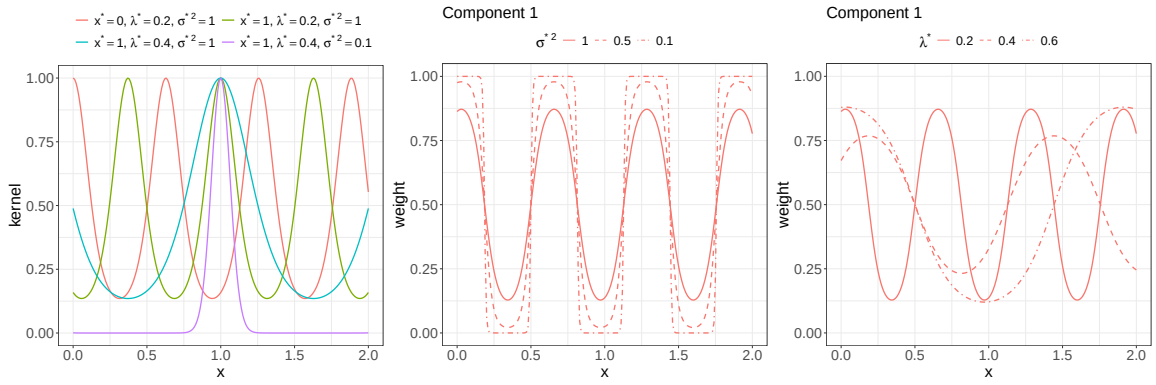


Figure 3: Left: Example of a periodic kernel for varying parameter values. Middle: Covariate-dependent weights for the first component given a total of two components (red and green on the left), for different ${\sigma^*_{j,d}}^2$. When ${\sigma^*_{j,d}}^2$ decreases, the weights show a more abrupt change. Right: Covariate-dependent weights for the first component for different $\lambda^*_{j,d}$.

$\lambda^*_{j,d}$ (green vs. blue), and ${\sigma^*_{j,d}}^2$ is related to the minimum value of the kernel and smooths the kernel (blue vs. purple). The influence of ${\sigma^*_{j,d}}^2$ and $\lambda^*_{j,d}$ on the covariate-dependent weight is shown in Figure 3 (middle and right). A periodic kernel is appropriate when there is repeated behavior over the covariate, e.g. time. Similar to the Gaussian kernel, for periodic kernels, as ${\sigma^*_{j,d}}^2 \to \infty$, the C-HDP reduces to the HDP.

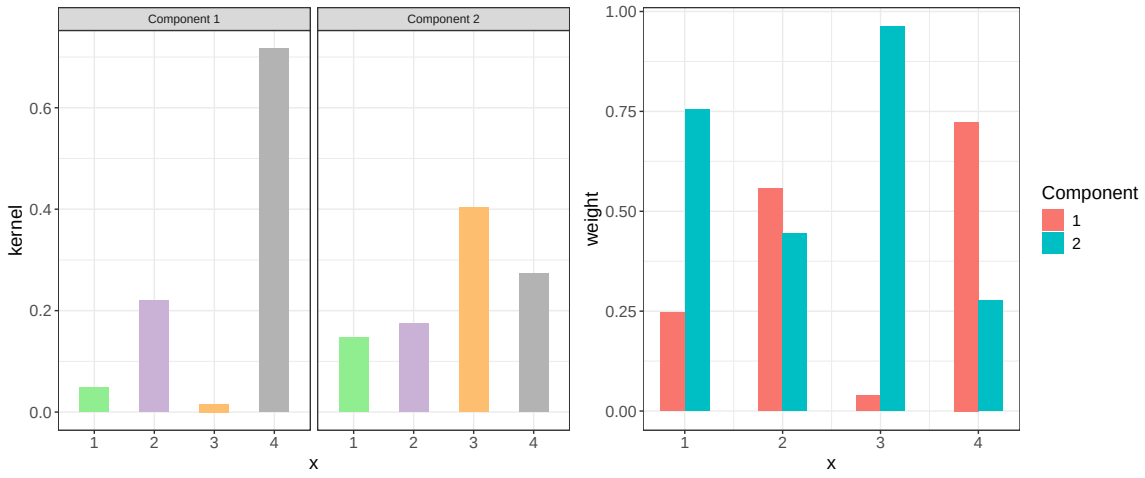


Figure 4: Left: Example of a categorical kernel for two different sets of kernel parameters (probabilities), with $L = 4$. Right: Covariate-dependent weights for two components, with kernel parameters shown in the left panel.

Categorical Kernel.

$$K(x|\boldsymbol{\psi}^*_{j,d}) = \prod_{l=1}^{L} (\rho^*_{j,d,l})^{\mathbb{I}(x=l)},$$

for $x \in \{1, \ldots, L\}$ where $\mathbb{I}(\cdot)$ is the indicator function, $\boldsymbol{\psi}^*_{j,d} = (\rho^*_{j,d,1}, \ldots, \rho^*_{j,d,L})$ and the probabilities $\rho^*_{j,d,l}$ are positive and sum to one. Categorical kernels are a natural choice when we have categorical covariates (see Figure 4 for examples).

The choice of these kernels makes the denominator in Equation (5) finite, ensuring $P_{x,d}$ and P_x are valid probability measures. Following the definition in Equation (4) and Equation (5), a draw from the C-HDP prior is concisely denoted as $P_{x,d} \sim \text{C-HDP}(\alpha_0, \alpha, P_0, \boldsymbol{\Psi}^*)$, where $\boldsymbol{\Psi}^*$ denotes the collection of all kernel parameters across components and groups.

In our work, hierarchical priors are assigned to kernel parameters, similar to the hierarchical prior for $q_{j,d}$, to borrow strength across groups and account for differences in the covariate effects on the weights across groups. For example, for the Gaussian kernel, the following hierarchical priors are considered in our case study later,

$$x^*_{j,d} \overset{ind}{\sim} \text{N}(r_j, s^2), \quad r_j \overset{i.i.d}{\sim} \text{N}(\mu_r, \sigma^2_r), \quad s^2 \sim \text{IG}(\eta_1, \eta_2),$$
$$\sigma^{*\,2}_{j,d} \overset{ind}{\sim} \text{log-N}(h_j, m^2), \quad h_j \overset{i.i.d}{\sim} \text{N}(\mu_h, \sigma^2_h), \quad m^2 \sim \text{IG}(\kappa_1, \kappa_2).$$

The effects of the prior parameters are demonstrated in Figure 5. For large values of the concentration parameter α and small prior variance s^2 and m^2, the differences in the relationship between the weights and covariates across groups are minimal and resemble the global relationship (left panel of Figure 5). On the other hand, for small α and/or large prior variance, the covariate-dependent weights vary across groups. In particular, for small α, the group-level relationship still centers around the global one (middle panel of Figure

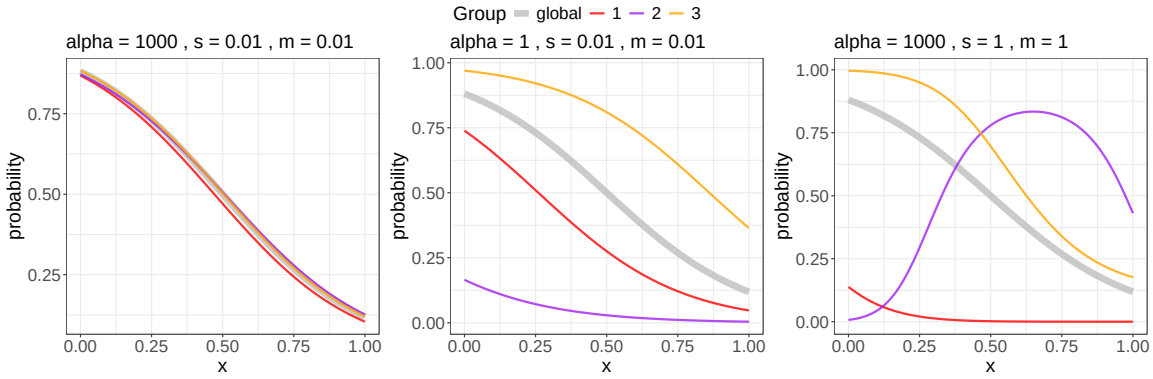


Figure 5: The effects of the concentration parameter α and prior variance parameters s and m on the covariate-dependent weights across groups. Three groups are considered here.

5), while the differences in the shape are more prominent for larger prior variance (right panel of Figure 5).

3.3 Component-specific Likelihood

The mixture model, under C-HDP, is given by

$$f(y|x, P_{x,d}) = \int f(y|\theta, x) dP_{x,d}(\theta),$$

where $P_{x,d} \sim \text{C-HDP}(\alpha_0, \alpha, P_0, \boldsymbol{\Psi}^*)$. This can be hierarchically written as

$$y_{i,d}|\theta_{i,d}, x_{i,d} \overset{ind}{\sim} f(y_{i,d}|\theta_{i,d}, x_{i,d}), \quad \theta_{i,d}|x_{i,d} = x, P_{x,d} \overset{ind}{\sim} P_{x,d}.$$

And, the latent variables $z_{i,d} \in \{1, 2, \ldots\}$ can be introduced, where $z_{i,d} = j$ if $\theta_{i,d} = \theta_j^*$.

Depending on the type of the data, different distributions can be selected for the component-specific likelihood, which should account for the characteristics and nature of the sample space $\mathcal{Y}$. For instance, a normal distribution is the most common for a continuous response, while a (skew) t or skew-normal may be more appropriate when there are outliers or asymmetry (Frühwirth-Schnatter and Pyne, 2010; Lee and McLachlan, 2014). Positive responses are usually modeled from a gamma or log-normal distribution, and a beta distribution is suitable for values on the unit interval. For a categorical response, a Bernoulli (Pan et al., 2024) or multinomial distribution (Shahbaba and Neal, 2009; Blei et al., 2003) is usually selected. Poisson (Karlis and Xekalaki, 2005; Krnjajić et al., 2008) and negative-binomial (Wu and Luo, 2022; Liu et al., 2024) distributions are used for count data. Below are some examples of component-specific likelihoods, with the first used in the simulation study of Section 5 and the latter two employed in the examples of Section 6.

Gaussian Model. For a continuous response $\boldsymbol{y}_{i,d} = (y_{i,1,d}, \ldots, y_{i,G,d})^T \in \mathbb{R}^G$, a normal distribution is commonly used

$$\boldsymbol{y}_{i,d}|z_{i,d} = j, \boldsymbol{\mu}_j^*, \boldsymbol{\Sigma}_j^* \sim \text{N}(\boldsymbol{\mu}_j^*, \boldsymbol{\Sigma}_j^*), \tag{8}$$

where $\boldsymbol{\mu}_j^* \in \mathbb{R}^G$ denotes the mean and $\boldsymbol{\Sigma}_j^*$ is a $G \times G$ covariance matrix, both associated with component j, i.e. $\theta_j^* = (\boldsymbol{\mu}_j^*, \boldsymbol{\Sigma}_j^*)$.

Linear Regression Model. For a continuous response, where we also want to allow each component to reflect different linear relationships between the response and covariates, a linear regression model can be employed,

$$\boldsymbol{y}_{i,d}|z_{i,d} = j, \boldsymbol{x}_{i,d}, \boldsymbol{\beta}_j^*, \boldsymbol{\Sigma}_j^* \sim \mathrm{N}(\boldsymbol{x}_{i,d}^T \boldsymbol{\beta}_j^*, \boldsymbol{\Sigma}_j^*),$$

where $\boldsymbol{\beta}_j^*$ is a p-dimensional (column) vector of coefficients for component j and $\boldsymbol{x}_{i,d}$ is a p-dimensional (column) vector of covariates.

Vector Autoregressive (VAR) Model. An extension of the normal distribution to time-series data, based on a VAR model with lag one in the mean, is of the following form:

$$\boldsymbol{y}_{i,d}|\boldsymbol{y}_{i-1,d}, z_{i,d} = j, \boldsymbol{a}_j^*, \mathbf{B}_j^*, \boldsymbol{\Sigma}_j^* \sim \ \mathrm{N}(\boldsymbol{a}_j^* + \mathbf{B}_j^* \boldsymbol{y}_{i-1,d}, \boldsymbol{\Sigma}_j^*),$$

where $\boldsymbol{a}_j^* \in \mathbb{R}^G$ denotes the intercept and $\mathbf{B}_j^*$ is a real-valued $G \times G$ matrix of the coefficients in VAR, both associated with component j.

Negative-binomial Model. For count data, we consider a negative-binomial distribution for the within-component likelihood, with mean $\mu_{j,g}^* \in \mathbb{R}^+$ and dispersion $\phi_{j,g}^* \in \mathbb{R}^+$ in each dimension g ($g = 1, \ldots, G$) for component j:

$$y_{i,g,d}|z_{i,d} = j, \mu_{j,g}^*, \phi_{j,g}^* \sim \mathrm{NB}(\mu_{j,g}^*, \phi_{j,g}^*).$$

4. Inference

In this section, we describe a Gibbs sampling scheme for the C-HDP mixture model, where we focus on the finite-dimensional truncation (Equation 6-7). The truncation level J should be chosen as a generous upperbound; see the online Appendix, showing how inference is stable for large enough J, while computational run time is approximately linear in J. Gibbs sampling can be applied to draw posterior samples for parameters with full conditionals of a standard form. For non-standard full conditionals, adaptive Metropolis-Hastings (AMH) is used (Griffin and Stephens, 2013). We highlight the key steps in constructing the Gibbs sampler, including an initial data augmentation trick to handle the normalizing constant of the weights.

4.1 A Data Augmentation Trick

For mixture models, the complete data likelihood is widely employed to allow for efficient inference, which for the finite-dimensional C-HDP mixture has the form:

$$f(y_{i,d}, z_{i,d} = j|\boldsymbol{q}_{1:J,d}, \theta_j^*, \boldsymbol{\psi}_{1:J,d}^*, x_{i,d}) = \frac{q_{j,d} K(x_{i,d}|\boldsymbol{\psi}_{j,d}^*)}{\sum_{k=1}^J q_{k,d} K(x_{i,d}|\boldsymbol{\psi}_{k,d}^*)} \times f(y_{i,d}|\theta_j^*, x_{i,d}).$$

However, the normalizing constant in the denominator makes it difficult to obtain standard full conditional densities for $q_{j,d}$ and the kernel parameters. We propose to use a data augmentation trick, introducing a latent variable $\xi_{i,d} \in \mathbb{R}_+$, and the augmented likelihood

is

$$f(y_{i,d}, \xi_{i,d}, z_{i,d} = j | \boldsymbol{q}_{1:J,d}, \theta_j^*, \boldsymbol{\psi}_{1:J,d}^*, x_{i,d}) = \exp\left(-\xi_{i,d} \sum_{j=1}^{J} q_{j,d} K(x_{i,d}|\boldsymbol{\psi}_{j,d}^*)\right) \times q_{j,d} K(x_{i,d}|\boldsymbol{\psi}_{j,d}^*) \times f(y_{i,d}|\theta_j^*, x_{i,d}). \quad (9)$$

Using the fact that $\int_0^\infty \exp(-\xi\lambda)d\xi = \frac{1}{\lambda}$, the complete data likelihood is restored when integrating out $\xi_{i,d}$. It is worth noticing that, unlike $z_{i,d}$, the $\xi_{i,d}$ does not have a physical interpretation. Define $N_{j,d}$ as the number of observations in component j in data set d and n_d the size of data d. The augmented data likelihood yields standard full conditional distributions to enable sampling both $q_{j,d}$ and $\xi_{i,d}$ effectively:

$$\begin{aligned}
\pi(q_{j,d}|\dots) &\propto (q_{j,d})^{N_{j,d}} \times \exp\left(-q_{j,d} \sum_{i=1}^{n_d} \xi_{i,d} K(x_{i,d}|\boldsymbol{\psi}_{j,d}^*)\right) \times (q_{j,d})^{\alpha p_j - 1} \exp(-q_{j,d}) \\
&\propto (q_{j,d})^{N_{j,d} + \alpha p_j - 1} \times \exp\left(-q_{j,d}\left[1 + \sum_{i=1}^{n_d} \xi_{i,d} K(x_{i,d}|\boldsymbol{\psi}_{j,d}^*)\right]\right), \\
\Rightarrow \quad q_{j,d}|\dots &\sim \text{Gamma}\left(N_{j,d} + \alpha p_j, 1 + \sum_{i=1}^{n_d} \xi_{i,d} K(x_{i,d}|\boldsymbol{\psi}_{j,d}^*)\right).
\end{aligned}$$

$$\begin{aligned}
\pi(\xi_{i,d}|\dots) &\propto \exp\left(-\xi_{i,d} \sum_{j=1}^{J} q_{j,d} K(x_{i,d}|\boldsymbol{\psi}_{j,d}^*)\right), \\
\Rightarrow \quad \xi_{i,d}|\dots &\sim \text{Gamma}\left(1, \sum_{j=1}^{J} q_{j,d} K(x_{i,d}|\boldsymbol{\psi}_{j,d}^*)\right).
\end{aligned}$$

In addition to the intricate denominator, the presence of the kernel parameters inside the exponential term in Equation (9) also poses challenges. For kernel parameters, we introduce another latent variable $u_{i,j,d} \in (0,1)$ to facilitate MCMC sampling. The update of $\boldsymbol{\psi}_{j,d}^*$ and θ_j^* depends on the choice of the kernel and component-specific likelihood. If appropriate priors are chosen, it is possible to sample from the full conditionals of $\boldsymbol{\psi}_{j,d}^*$ and θ_j^*.

For example, consider the Gaussian kernel with a hierarchical normal prior $x_{j,d}^* \overset{ind}{\sim} \text{N}(r_j, s^2)$ (for details see online Appendix), the full conditional distribution for $x_{j,d}^*$ is

$$\pi(x_{j,d}^*|\dots) \propto \prod_{i:z_{i,d}=j} K(x_{i,d}|\boldsymbol{\psi}_{j,d}^*) \times \prod_{i=1}^{n_d} \exp\left(-\xi_{i,d} q_{j,d} K(x_{i,d}|\boldsymbol{\psi}_{j,d}^*)\right) \times \text{N}(x_{j,d}^*|r_j, s^2).$$

With the introduction of $u_{i,j,d} \in (0,1)$, the above can be written as

$$\pi(x_{j,d}^*|\dots) \propto \prod_{i:z_{i,d}=j} K(x_{i,d}|\boldsymbol{\psi}_{j,d}^*) \times \prod_{i=1}^{n_d} \mathbb{I}\left(u_{i,j,d} < M_{i,j,d}\right) \times \text{N}(x_{j,d}^*|r_j, s^2),$$

where $M_{i,j,d} = \exp\left(-\xi_{i,d} q_{j,d} K(x_{i,d}|\boldsymbol{\psi}^*_{j,d})\right)$. The full conditional of the latent variable is $u_{i,j,d}|\ldots \sim \text{Unif}\,(0, M_{i,j,d})$, and for the kernel parameter $x^*_{j,d}$ given $u_{i,j,d}$, the full conditional is a truncated normal distribution:

$$\pi(x^*_{j,d}|\ldots) \propto \text{N}(x^*_{j,d}|\hat{r}_{j,d}, \hat{s}^2_{j,d}) \times \mathbb{I}(x^*_{j,d} \in A_{j,d}),$$

where

$$\hat{s}^2_{j,d} = \left(\frac{1}{s^2} + \frac{N_{j,d}}{{\sigma^*_{j,d}}^2}\right)^{-1}, \quad \hat{r}_{j,d} = \frac{r_j/s^2 + \sum_{i:z_{i,d}=j} x_{i,d}/{\sigma^*_{j,d}}^2}{1/s^2 + N_{j,d}/{\sigma^*_{j,d}}^2},$$

and the truncation region is of the form

$$A_{j,d} = \bigcap_{i:-\log u_{i,j,d} < \xi_{i,d} q_{j,d}} A_{i,j,d},$$

where

$$A_{i,j,d} = (-\infty, x_{i,d} - k_{i,j,d}) \bigcup (x_{i,d} + k_{i,j,d}, +\infty), \quad k_{i,j,d} = \sqrt{-2{\sigma^*_{j,d}}^2 \log\left[-\frac{\log u_{i,j,d}}{\xi_{i,d} q_{j,d}}\right]}.$$

The full derivation and details of the Gibbs sampling algorithm for all parameters under different kernel and within-component likelihood choices are presented in online Appendix.

4.2 Clustering

The Bayesian approach provides a collection of posterior samples of the allocation variables. To understand the posterior and uncertainty in the clustering represented by these allocation variables, we construct the posterior similarity matrix (PSM) where each entry is the posterior probability that observations i and i' are co-clustered, which is approximated by $\text{PSM}_{[i,i']} \approx 1/L \sum_{l=1}^{L} \mathbb{I}(z_i^{(l)} = z_{i'}^{(l)})$ where $z_i^{(l)}$ is the l-th MCMC sample. For multiple data sets, we can compute the PSM for observations both within and across data sets to visualize the uncertainty in clustering across groups.

To summarize the posterior with a point estimate, the optimal clustering is obtained that minimizes the posterior expected variation of information (VI) (Wade and Ghahramani, 2018). If interest lies in understanding the patterns within each cluster of this optimal solution and the marginal component-specific likelihood is unavailable in closed form (i.e. the prior on the atoms is not conjugate to the component-specific likelihood), a subsequent MCMC is considered to estimate and quantify uncertainty in cluster-specific parameters given the optimal clustering solution.

For the post-processing step, to understand the uncertainty in allocations, we calculate the posterior allocation probability of each data point conditioned on all others

$$p(z_i = j|\mathcal{D}, \boldsymbol{z}^*_{-i}) = \int p(z_i = j|\boldsymbol{\Theta}, \mathcal{D}) p(\boldsymbol{\Theta}|\mathcal{D}, \boldsymbol{z}^*_{-i}) d\boldsymbol{\Theta}, \quad i = 1, \ldots, n,$$

where $\mathcal{D}$ denotes the observed data, $\boldsymbol{z}^*_{-i}$ denotes the optimal clustering without the i-th observation and $\boldsymbol{\Theta}$ represents all the unknown parameters. If we approximate $p(\boldsymbol{\Theta}|\mathcal{D}, \boldsymbol{z}^*_{-i})$ by

$p(\boldsymbol{\Theta}|\mathcal{D}, \boldsymbol{z}^*_{1:n})$, which corresponds to the posterior distribution of $\boldsymbol{\Theta}$ from the post-processing MCMC, then the above can be approximated by the average over MCMC samples (from the post-processing step),

$$p(z_i = j|\mathcal{D}, \boldsymbol{z}^*_{-i}) \approx \frac{1}{L}\sum_{l=1}^{L} p(z_i = j|\boldsymbol{\Theta}^{(l)}, \mathcal{D}), \tag{10}$$

where $\boldsymbol{\Theta}^{(l)}$ denotes the l-th MCMC sample.

4.3 Covariate-dependent Predictive Quantities of Interest

In the context of density estimation, we can obtain the covariate-dependent conditional density, approximated by averaging over the MCMC samples:

$$f(\tilde{y}|\tilde{x}, \mathcal{D}) = \int f(\tilde{y}|\tilde{x}, \boldsymbol{\Theta})\pi(\boldsymbol{\Theta}|\mathcal{D})d\boldsymbol{\Theta} \approx \frac{1}{L}\sum_{l=1}^{L} f(\tilde{y}|\tilde{x}, \boldsymbol{\Theta}^{(l)}),$$

where $\tilde{x}$ and $\tilde{y}$ denote new data from group d and $f(\tilde{y}|\tilde{x}, \boldsymbol{\Theta}) = \sum_{j=1}^{J} p_{j,d}(\tilde{x}) f(\tilde{y}|\theta^*_j, \tilde{x})$. Similarly, other quantities can be computed, such as the posterior predictive mean

$$\mathbb{E}(\tilde{y}|\tilde{x}, \mathcal{D}) \approx \frac{1}{L}\sum_{l=1}^{L} \mathbb{E}(\tilde{y}|\tilde{x}, \boldsymbol{\Theta}^{(l)}).$$

For instance, in the case of the Gaussian within-component likelihood defined in Section 3.3, for an observation from group d, we have $\mathbb{E}(\tilde{y}|\tilde{x}, \boldsymbol{\Theta}) = \sum_{j=1}^{J} p_{j,d}(\tilde{x})\mu^*_j$.

5. Simulation Study

We now demonstrate our method in a simulation study and compare its performance with other existing Bayesian nonparametric priors, in particular the HDP and DDP. We generate 10 replicated sets of data from a two-dimensional Gaussian mixture, consisting of 3 components. The data-generating process is as follows:

$$\boldsymbol{y}_{i,d}|z_{i,d} = j, \boldsymbol{\mu}^*_j, \boldsymbol{\Sigma}^*_j \overset{ind}{\sim} \mathrm{N}(\boldsymbol{\mu}^*_j, \boldsymbol{\Sigma}^*_j), \quad i = 1, \ldots, n_d, d = 1, \ldots, D,$$
$$z_{i,d} \overset{ind}{\sim} \mathrm{Cat}(p^J_{1,d}(x_{i,d}), \ldots, p^J_{J,d}(x_{i,d})),$$

where $D = 5, n_d = 300$ for all d. The covariate x for all observations in each data set d is equally spaced between 0 and 1. The parameters in each component-specific likelihood are

$$\boldsymbol{\mu}^*_1 = (0, 0),\ \boldsymbol{\mu}^*_2 = (4, 4),\ \boldsymbol{\mu}^*_2 = (0, 4),$$
$$\boldsymbol{\Sigma}^*_1 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \boldsymbol{\Sigma}^*_2 = \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix}, \boldsymbol{\Sigma}^*_3 = \begin{pmatrix} 1 & -0.5 \\ -0.5 & 1 \end{pmatrix}.$$

The relationship between the weights and the covariate is given by

$$p^J_{j,d}(x) = \frac{\exp\left(\beta_{j,d,1} + \beta_{j,d,2}x + \beta_{j,d,3}x^2\right)}{\sum_{k=1}^{3}\exp(\beta_{k,d,1} + \beta_{k,d,2}x + \beta_{k,d,3}x^2)},$$

where each $\boldsymbol{\beta}_{j,d} = (\beta_{j,d,1}, \beta_{j,d,2}, \beta_{j,d,3})$ is generated from a diagonal Gaussian distribution, $\boldsymbol{\beta}_{j,d} \sim \mathrm{N}(\boldsymbol{\beta}_j, 5\mathbf{I})$, with $\boldsymbol{\beta}_1 = (0, 20, 0), \boldsymbol{\beta}_2 = (10, -20, 0), \boldsymbol{\beta}_3 = (0, 0, 40)$.

5.1 Implementation and Results

The Gaussian kernel defined in Section 3.2 is used in the C-HDP and DDP to model the covariate-dependent weights. While the HDP allows for weights to be different across data sets, it does not incorporate covariate effects. Instead, for the DDP, we consider two types of models, with and without the group indicator as an additional covariate, where the latter corresponds to setting when data is simply pooled across groups. In the former, the group variable is incorporated through a categorical kernel defined in Section 3.2. For each model and replicate, the full MCMC algorithm and the post-processing step (which is used to estimate the covariate-dependent allocation probabilities for each cluster in the optimal solution) use 15000 iterations, with a burn-in of 12000 iterations and thinning of 3, leading to 1000 samples for posterior inference. The MCMC setup is the same across all methods and a truncation level of $J = 8$ is applied.

Across all replicates, the estimated clustering from our proposed C-HDP method generally shows better alignment with the truth, with a higher average adjusted rand index (ARI) in Figure 6 (left, red points). The DDP with the group indicator (DDP_1) also provides relatively high ARI, sometimes even higher than the C-HDP, but the difference is small and the performance of the DDP appears more unstable. Further, the ARI drops dramatically if the grouping is not considered in the DDP (DDP_2). As for the HDP, it is noticeably outperformed by the C-HDP and DDP_1, but with much smaller variability in its performance.

Regarding the modelling of the relationship between the weights and the covariate, for an example cluster shown in Figure 6 (middle), we notice that using the Gaussian kernel for the C-HDP can estimate the relationship well, with the truth covered by the posterior samples, while in the DDP (including the groups), the true relationship is only marginally within the samples. This is worse in the simpler DDP (DDP_2) which does not capture the true relationship except for the upward trend. Note that since the HDP does not take into account the covariate, the probability is constant with x, showing much larger uncertainty. More detailed results are shown in online Appendix.

In addition, we also compare the four methods in terms of density estimation (Figure 6 right). The estimated density from the C-HDP appears much closer to the truth with their differences concentrated around zero, followed by the DDP and HDP.

Lastly, one advantage of the C-HDP and HDP over the DDP is the ability to predict the (covariate-dependent) weights in new groups through posterior predictive sampling, which is not available for the DDP as the groups are treated as fixed categories. The relationship between the weights and covariate shows variability across groups (see red lines in Figure 7 and Figure 6 middle). Figure 7 (left) illustrates that the new group's weight can be reasonably estimated by the C-HDP where the posterior median (green solid line) is close to the truth and samples exhibit relatively large uncertainty (green dashed lines) compared to those for the existing groups. As for the HDP, large discrepancy from the truth is observed due to the absence of the covariate in the method.

Overall, the simulation study on Gaussian mixtures shows that C-HDP outperforms DDP and HDP in estimating the clustering, the relationship between the weights and covariate, as well as density estimation. In particular, even though the true relationship between the weights and covariate is not correctly specified in the C-HDP model (a softmax

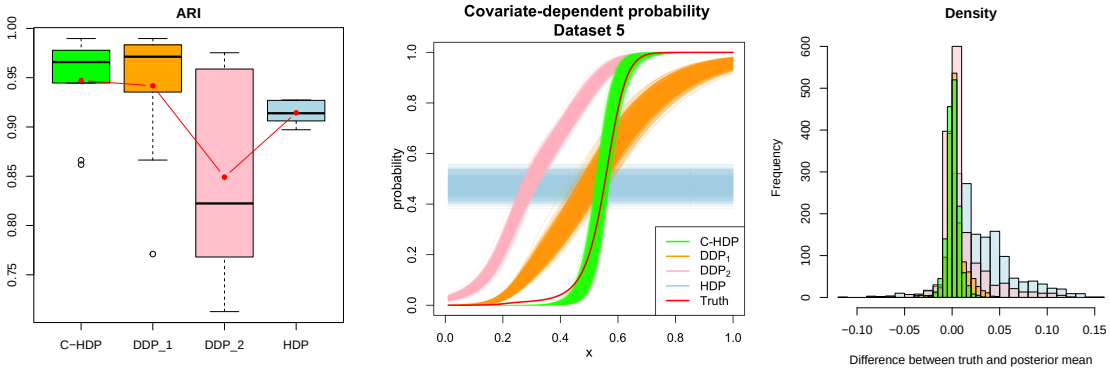


Figure 6: Left: Adjusted rand index (ARI) comparing each method to the truth across all replicates, with the red line showing the average ARI. DDP_1 and DDP_2 refer to the model with and without the group indicator, respectively. Middle: Posterior samples of covariate-dependent probabilities for all models, shown for an example cluster in data set 5 from one replicate. Right: Differences between the posterior mean of the density and the truth for all observations in one replicate, under the four methods.

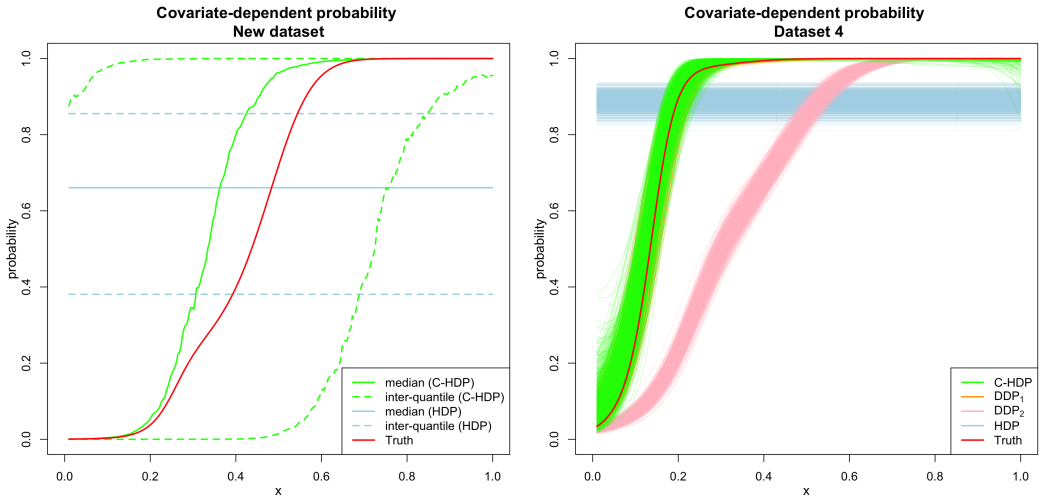


Figure 7: Left: Posterior predictive samples of covariate-dependent probabilities for the C-HDP and HDP, shown for an example cluster in a new group from one replicate. Right: Posterior samples of covariate-dependent probabilities for all models, shown for an example cluster in data set 4 from one replicate. Samples from the C-HDP and DDP_1 overlap.

function as opposed to a Gaussian kernel), a Gaussian kernel is still able to capture the true underlying relationship. Moreover, the hierarchical structure of the C-HDP facilitates understanding of the weights in a new group based on the posterior predictive distribution.

6. Case Studies

We demonstrate the application of the C-HDP prior through experiments on two real data sets, using a Gaussian kernel combined with a negative-binomial component-specific likelihood and a periodic kernel combined with a VAR model, respectively.

6.1 Application to Single-cell RNA Sequencing Data - Pax6

The transcription factor Pax6 is believed to play an important role in the development and fates of embryonic cells. Specifically, Pax6 is critical to regulate gene expressions to avoid patterning defects in the brain and can affect signaling between neighbouring cells during development (Caballero et al., 2014). Mutations in Pax6 can result in eye anomalies (Jordan et al., 1992) and neurodevelopmental disorders, such as intellectual disability and autism spectrum disorder (Davis et al., 2008; Kikkawa et al., 2019). It plays a crucial role in cortical neurogenesis, controlling proliferation and differentiation of neural stem cells and progenitor cells (Estivill-Torrus et al., 2002; Götz et al., 1998).

To study and quantify the effects of Pax6 empirically at the single-cell level, Manuel et al. (2022) collected the scRNA-seq data at 13.5 days since conception (day E13.5) from mouse embryos in two groups: the mutant (HOM) group where both copies of Pax6 gene were deleted, and the control group (HET) where only one copy of Pax6 was deleted, to account for gene changes due to the process of deletion. In previous work, Liu et al. (2024) developed a model using the HDP to discover hidden cell subpopulations with similar gene expression and study how the proportion of cells in each cell group is affected by Pax6 activity. However, additional information is available for cells, such as the developmental trajectory, which may be relevant to identify clusters and quantify the influence of Pax6 across the developmental stages. Our aim is to extend the model of Liu et al. (2024) to incorporate this information and determine how the proportion of cells in each cell group is affected by both Pax6 activity and cellular developmental trajectory.

Each group d ($d = 1, 2$) contains the mRNA counts $y_{c,g,d}$ for gene g ($g = 1, \ldots, G$) in cell c ($c = 1, \ldots, C_d$). Data has been preprocessed in Liu et al. (2024) using the approaches in Hoffman (2023), where low-quality cells and lowly expressed genes are removed and highly variable genes are selected. The HET and HOM data sets contain $C_1 = 3096$ and $C_2 = 5282$ cells, both with $G = 2529$ genes. The covariate of interest is a proxy for cell developmental trajectory, specifically, the cell-specific latent time $t_{c,d} \in [0, 1]$ (Bergen et al., 2020), which empirically shows a relation to the clustering in Liu et al. (2024) (Figure 8). Bergen et al. (2020) argue that the latent time is correlated to the cellular position in the biological process, with a small value corresponding to an earlier developmental stage. The latent time is derived from a per-gene model based on the relative amount of unspliced mRNAs to spliced mRNAs. For each group, the abundances of unspliced and spliced mRNAs are obtained from the *velocyto* pipeline (La Manno et al., 2018) and the latent time is computed from a generalized RNA velocity model (Bergen et al., 2020).

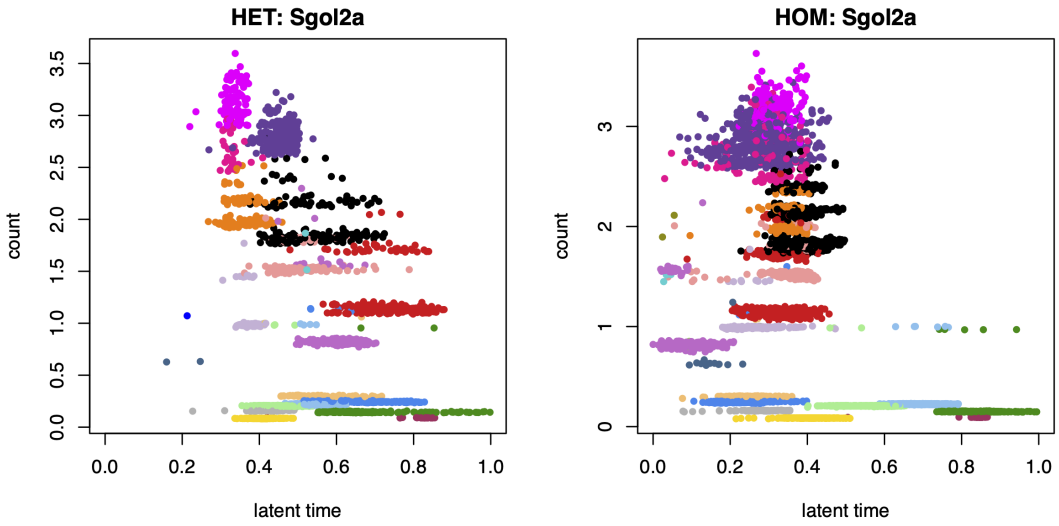


Figure 8: Counts versus latent time for gene *Sgol2a* in scRNA-seq data (Manuel et al., 2022) collected under two experimental conditions. Cells are colored by cluster membership from Liu et al. (2024) without information of latent time.

6.1.1 Bayesian Model for Pax6 data

The model for clustering the Pax6 data extends the work in Liu et al. (2024) to include latent time as a covariate. Liu et al. (2024) employed the HDP prior for shared clustering across the HET and HOM data sets, where the clustering model is built upon the likelihood of *bayNorm* (Tang et al., 2020) that addresses the problem of normalization, imputation and batch effect correction in an integrated manner.

Likelihood. The observed count $y_{c,g,d}$ is assumed to follow a binomial distribution given the latent true count $y^0_{c,g,d}$, with cell-specific capture efficiency $\beta_{c,d}$

$$y_{c,g,d}|y^0_{c,g,d}, \beta_{c,d} \sim \text{Bin}(y^0_{c,g,d}, \beta_{c,d}).$$

The binomial distribution accounts for the case where partial true counts are observed. The latent counts follow a negative-binomial distribution accounting for over-dispersion:

$$y^0_{c,g,d}|\mu_{c,g,d}, \phi_{c,g,d} \sim \text{NB}(\mu_{c,g,d}, \phi_{c,g,d}),$$

with mean expression $\mu_{c,g,d}$ and dispersion $\phi_{c,g,d}$ that are both specific to each gene and cell. The latent counts can be integrated out to obtain:

$$y_{c,g,d}|\mu_{c,g,d}, \phi_{c,g,d}, \beta_{c,d} \sim \text{NB}(\mu_{c,g,d}\beta_{c,d}, \phi_{c,g,d}),$$

where it is noticed that μ and β are not identifiable while only their product is. An informative prior for $\beta_{c,d}$ is applied to mitigate this problem (Liu et al., 2024).

While Liu et al. (2024) assume the cell-specific mean and dispersion

$$(\boldsymbol{\mu}_{c,d}, \boldsymbol{\phi}_{c,d})|P_d \overset{i.i.d}{\sim} P_d,$$

and employ an HDP prior $P_d \sim \text{HDP}(\alpha_0, \alpha, P_0)$, we extend with a C-HDP prior

$$P_{t_{c,d},d} \sim \text{C-HDP}(\alpha_0, \alpha, P_0, \mathbf{\Psi}^*),$$

with the covariate being the latent time. For the base measure P_0, a linear relationship is assumed between log mean expression and dispersion (Brennecke et al., 2013; Vallejos et al., 2015; Eling et al., 2018).

Kernel. The latent time is included via a Gaussian kernel with kernel parameters $\boldsymbol{\psi}^*_{j,d} = (t^*_{j,d}, {\sigma^*_{j,d}}^2)$. The parameter $t^*_{j,d}$ represents the value in the covariate space where the j-th component from the d-th group best applies and ${\sigma^*_{j,d}}^2$ controls the smoothness of the transition between components in the covariate space. The kernel parameters are data-specific, which allows for shared clusters across HET and HOM to be associated with different cellular positions and the degree of separation between clusters that can vary between two conditions.

For full details of the model and prior specifications, Gibbs sampling algorithm and implementation, see the online Appendix. Results for posterior predictive checks (Gelman et al., 1996) that suggest no strong disagreement between the data and model, along with a simulation study, are available in online Appendix.

6.1.2 Results on Pax6 Data

Clustering. The Bayesian C-HDP model identifies 14 clusters using the VI criterion, whose sizes and proportion of HET and HOM cells are summarized in Figure 9. We refer to clusters as over-represented/under-represented in the mutant group if their proportion of HOM cells is greater/less than the overall proportion, and stable if the proportion is similar to the overall proportion. In this case, five clusters (2, 4, 5, 6, 10) are found to be over-represented in the mutant group, three clusters (3, 7, 9) are under-represented in the mutant group, and the remaining clusters (1, 8, 11, 12, 13, 14) show relatively stable proportions. Moreover, while we focus on the clustering estimate, the posterior similarity matrix (Figure 9) highlights some uncertainty in further splitting or merging some clusters. In addition, based on posterior allocation probability (Equation 10), we observe some uncertainty in cell allocations at the boundary between clusters in the covariate space (see online Appendix).

Although all clusters are shared in both groups, suggesting that Pax6 may play a small role at this early stage in the development (day E13.5) (as concluded in Liu et al., 2024), we observe some interesting patterns in the mutant group when connecting the clusters to latent time. Figure 10 displays the first principal component computed from the observed gene expression matrix against latent time. The three under-represented clusters 3, 7, 9 (dark green, yellow, light pink) are associated with larger latent time, particularly for the mutant group, which suggests interesting implications in the role of Pax6 in cellular development, especially later in the biological process. The over-represented clusters 2, 4, 5, 6, 10 (red, dark purple, orange, black, light green) have moderate latent time in the mutant group, whereas the stable clusters have relatively earlier latent time in the mutant group.

Time-dependent Probabilities. To further investigate the differences between HET and HOM, the time-dependent probabilities are visualized for stable, under-represented and over-represented clusters (1, 3, 6) in Figure 11. The under-represented cluster is closely associated to cells with high latent time (probability close to 1) in the mutant group, while

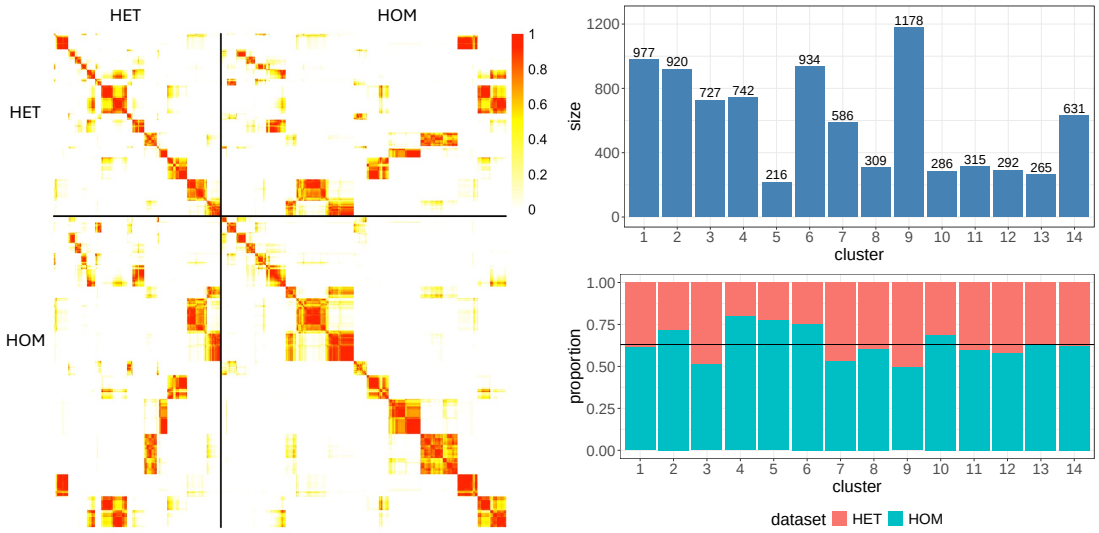


Figure 9: Left: Posterior similarity matrix within and between two experimental conditions. Diagonal blocks correspond to within-group PSM. Top-right: Size of each cluster found in Pax6 data. Bottom-right: Proportions of HET and HOM cells in each cluster. The black horizontal line represents the overall proportion.

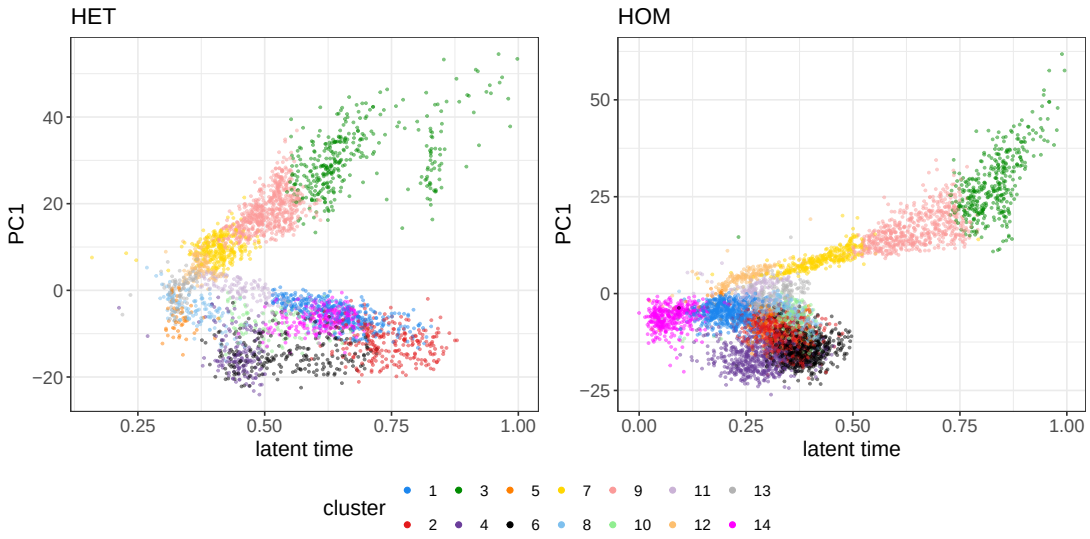


Figure 10: Plot of the first principal component (PC1) against latent time for HET (left) and HOM (right). Cells are colored by cluster membership. The three under-represented clusters 3, 7, 9 (dark green, yellow, light pink) are associated with higher latent time in the mutant group, and the over-represented clusters 2, 4, 5, 6, 10 (red, dark purple, orange, black, light green) have more moderate latent time, while the stable clusters (1, 8, 11, 12, 13, 14) are mainly associated with smaller latent time.

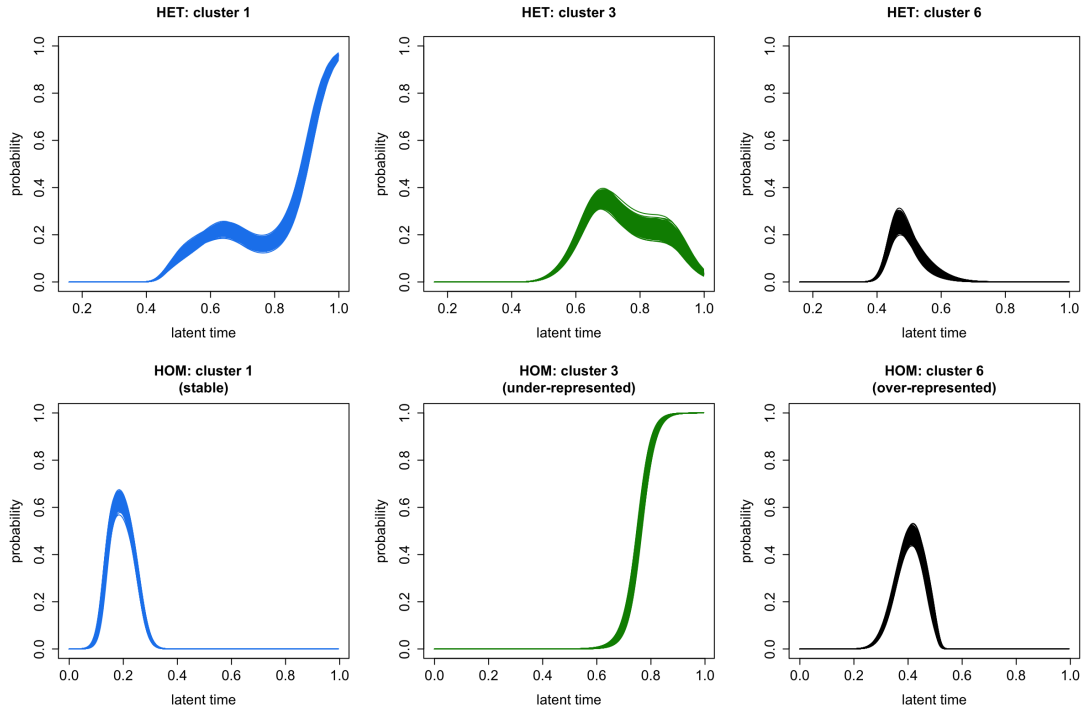


Figure 11: Time-dependent probabilities for clusters 1, 3, 6 (stable, under-represented, over-represented in HOM) in each data set for Pax6. The top row shows results for HET, with HOM shown in the bottom.

for the control group this association is not so pronounced. Specifically, although this cluster is associated with high latent time in the control group, cells with higher latent time may also belong to other clusters (probability is less than 0.5), as observed in Figure 10 (left). The stable cluster is mainly associated to cells with small latent time in the mutant group (probability greater than 0.5), whereas it is more associated with moderate to large latent time in the control group. As for cluster 6 (over-represented in HOM), the difference between two conditions is not evident. The results for full set of clusters are shown in online Appendix.

Latent Counts. Beyond clustering, the C-HDP can be used for nonparametric conditional density estimation and regression (Section 4.3). Specifically, for scRNA-seq, this is useful to understand how gene expression changes over latent time and across conditions. The expected count for a new cell c from data set d as a function of latent time is

$$\mathbb{E}(y^0_{c,g,d}|t_{c,d}=t,\mathcal{D}) = \int \sum_{j=1}^{J} p^J_{j,d}(t)\mu^*_{j,g} d\pi(\boldsymbol{q}^J_{1:J,d}, \boldsymbol{\mu}^*_{1:J,g}, \boldsymbol{\psi}^*_{1:J,d}|\mathcal{D}),$$

which is approximated from the MCMC samples. This is illustrated in Figure 12 for an example gene *3110035E14Rik*. A general decreasing pattern is observed in the control group with a potentially mild increase in the beginning (small latent time), whereas in the mutant

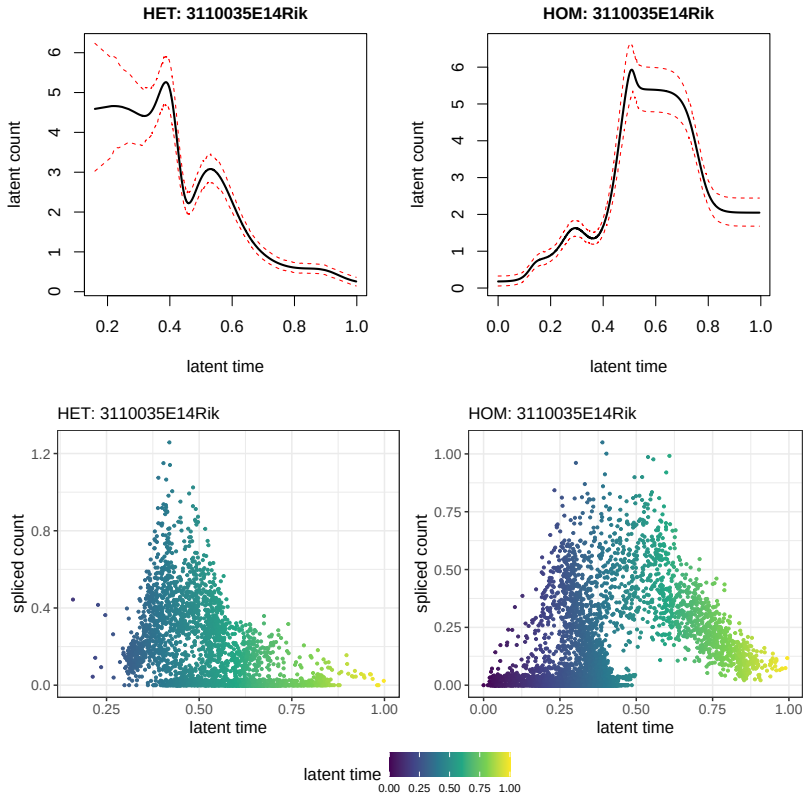


Figure 12: Top: Estimated latent counts against latent time for an example gene *3110035E14Rik*. The 95% highest posterior density interval is given by dashed red lines. Bottom: Spliced mRNA counts against latent time (La Manno et al., 2018). Cells are colored by latent time.

group a clear increasing trend is present followed by a decreasing trend, suggesting Pax6 may influence the expression activity of gene *3110035E14Rik*. In addition, the relationship between the latent counts and latent time appears similar to that between the observed spliced mRNA counts and latent time, which implies the reliability of the nonparametric estimate.

6.2 Application to Calcium Imaging Data

Most neurons interact with other neurons for communicating information rather than receiving direct inputs from the external world, which gives rise to an internal structure (Rubin et al., 2019). Such internal structure differs across brain regions with different computational roles, and can be used to expose unknown encoded variables, facilitating understanding of the functional roles of different brain circuits. Therefore, understanding the internal structure has become a critical goal in neuroscience, elucidating the neural mechanisms important for behavioral variables. This can be empirically studied using the calcium imaging technique which records neural activity of behaving animals over time (Mukamel

et al., 2009; Rubin et al., 2015), with a higher measurement suggesting that the neuron is firing, actively passing or receiving information.

In this setting, clustering is useful to identify time frames when neural activity is similar. In Rubin et al. (2019), it was shown that neural activity can be encoded in a lower-dimensional space and the identified clusters in the latent space correlate with behavioral measurements and positions, shedding light on the neural code associated with behavior and location. However, the authors analyzed multiple experiments individually, which makes it difficult to identify shared patterns across experiments. In addition, algorithmic clustering, e.g. k-means, is used, which is unable to account for the temporal nature of the data. To address these concerns, our aim is to use the C-HDP to integrate multiple experiments, identify shared neural activity patterns, and account for heterogeneous temporal dependence in the data (Figure 14) through a hierarchical Bayesian modelling framework.

We note that related approaches include D'Angelo et al. (2023), who propose a nested Bayesian finite mixture model with common atoms to allow shared clustering across experiments, which takes into account the time dependence. However, the method is designed for neural activity in a single neuron. In addition, D'Angelo et al. (2023) identify clusters characterized by similar spike magnitudes and therefore temporal dependence between successive time frames is assumed homogeneous within each experiment.

We study calcium imaging data collected by the Nolan lab in the Centre for Discovery Brain Sciences at the University of Edinburgh to study neural activity in mice. Data is from the dorsal CA1 region of the hippocampus, where neurons are tuned to spatial positions of the animal (O'Keefe and Dostrovsky, 1971). For a single mouse, two experiments were conducted over two different days inside a linear rig. The mouse was asked to run back and forth in the rig, and each experiment records the calcium activity of hundreds of neurons at fixed time bins. Following the approach in Rubin et al. (2019), the data is projected to a lower-dimensional space to explore their activity patterns. Specifically, we apply kernel PCA (Bishop, 2006) to the first experiment, which is used to transform both experiments from hundreds of neurons into a three-dimensional encoding of the neural activity of the neurons. The encoded neural activity at the i-th time bin from the d-th experiment is denoted as $\boldsymbol{y}_{i,d} \in \mathbb{R}^G$ with $i = 1, \ldots, n_d$, $G = 3$ and $d = 1, 2$. Each experiment consists of $n_d = 800$ time frames, and the covariate of interest is the observed time t which is rescaled to be equally spaced on $[0, 1]$.

6.2.1 Bayesian Model for Calcium Imaging Data

Our aim is to build on the work of Rubin et al. (2019) to develop a clustering model that accounts for the temporal dependence in the data through the model-based approach and integrates data across multiple experiments through the hierarchical framework of the C-HDP.

Likelihood. The three-dimensional encoding of the neural activity at each time point i from experiment d, $\boldsymbol{y}_{i,d}$, is assumed to follow a vector autoregression model with lag one

$$\boldsymbol{y}_{i,d} | \boldsymbol{y}_{i-1,d}, \boldsymbol{a}_{i,d}, \mathbf{B}_{i,d}, \boldsymbol{\Sigma}_{i,d} \sim \text{N}(\boldsymbol{a}_{i,d} + \mathbf{B}_{i,d}\boldsymbol{y}_{i-1,d}, \boldsymbol{\Sigma}_{i,d}),$$

where we define $\mathbf{L}_{i,d} = (\boldsymbol{a}_{i,d} \quad \mathbf{B}_{i,d})^T$ which represent the $(G+1) \times G$ coefficient matrix, and $\boldsymbol{\Sigma}_{i,d}$ is the $G \times G$ covariance matrix. Time frames with similar dependence on the previous

time point and a similar covariance structure are expected to be in the same cluster. We assume the likelihood parameters for each time frame are generated from the covariate-dependent distribution $P_{t,d}$ in group d, which is modelled from the proposed C-HDP prior, with the covariate being the observed time:

$$(\mathbf{L}_{i,d}, \mathbf{\Sigma}_{i,d})|P_{t_{i,d},d} \sim P_{t_{i,d},d}, \quad P_{t_{i,d},d} \sim \text{C-HDP}(\alpha_0, \alpha, P_0, \mathbf{\Psi}^*),$$

Base Measure. For the base measure P_0 of the C-HDP, the component-specific parameters $\mathbf{L}_j^* = (\boldsymbol{a}_j^* \quad \mathbf{B}_j^*)^T$ and $\mathbf{\Sigma}_j^*$ are given conjugate priors:

$$\mathbf{L}_j^*|\mathbf{\Sigma}_j^* \overset{ind}{\sim} \text{MN}(\mathbf{L}_0, \mathbf{V}_0, \mathbf{\Sigma}_j^*), \quad \mathbf{\Sigma}_j^* \overset{i.i.d}{\sim} \text{IW}(\omega_0, \mathbf{\Phi}_0),$$

where MN and IW denote the matrix normal and inverse-Wishart distribution, $\mathbf{L}_0$ is of dimension $(G+1) \times G$, $\mathbf{V}_0$ is of dimension $(G+1) \times (G+1)$, $\mathbf{\Phi}_0$ is of dimension $G \times G$, and $\omega_0 > G - 1$. Empirical estimates are used for prior specification (for details see the online Appendix).

Kernel. As a cyclic pattern over time has been observed in the transformed data (see Figure 14 below), we implement the periodic kernel to account for the recurring pattern, with kernel parameters $\boldsymbol{\psi}_{j,d}^* = (\mu_{j,d}^*, {\sigma_{j,d}^*}^2, \lambda_{j,d}^*)$. The parameter $\mu_{j,d}^*$ represents the value that maximizes the kernel, $\lambda_{j,d}^*$ specifies the period of the kernel and ${\sigma_{j,d}^*}^2$ again smooths the covariate region (Figure 3).

For full details of the model and prior specifications, Gibbs sampling algorithm and a simulated experiment, see the online Appendix. In addition, we provide predictions of the neural activity and covariate-dependent probabilities for future time points and assess model fit through posterior predictive checks in the online Appendix.

6.2.2 Results on Calcium Imaging Data

Clustering. Twenty clusters associated to different activities are identified across both experiments, with the posterior similarity matrix shown in Figure 13 depicting some uncertainty in allocations. There are 15 clusters shared in both experiments, with varying proportions reflecting different amounts of time associated to patterns across the experiments. There are also some unique clusters of neural activity as well as several small clusters with size < 20, which may represent the noise in the data.

To understand the neural activity patterns of the identified clusters, we plot the encoding of activity at each time frame against previous time for a specific cluster (Figure 14 top). It is noticed that cluster 17 (pink) is mainly below the equivalent line $y = x$ in the first dimension, suggesting a decreasing trend in the encoded activity as time increases, whilst an increasing trend is observed in the third dimension. On the other hand, cluster 5 exhibits exactly the opposite patterns to cluster 17. Both clusters show an upward trend in the second dimension. This can be confirmed from the lower panels in Figure 14 which shows a time-series plot for each reduced dimension.

Further, we compute the posterior estimated relationship between consecutive time points for each dimension based on

$$\boldsymbol{y}_{i,d} = \hat{\boldsymbol{a}}_j^* + \hat{\mathbf{B}}_j^* \boldsymbol{y}_{i-1,d},$$

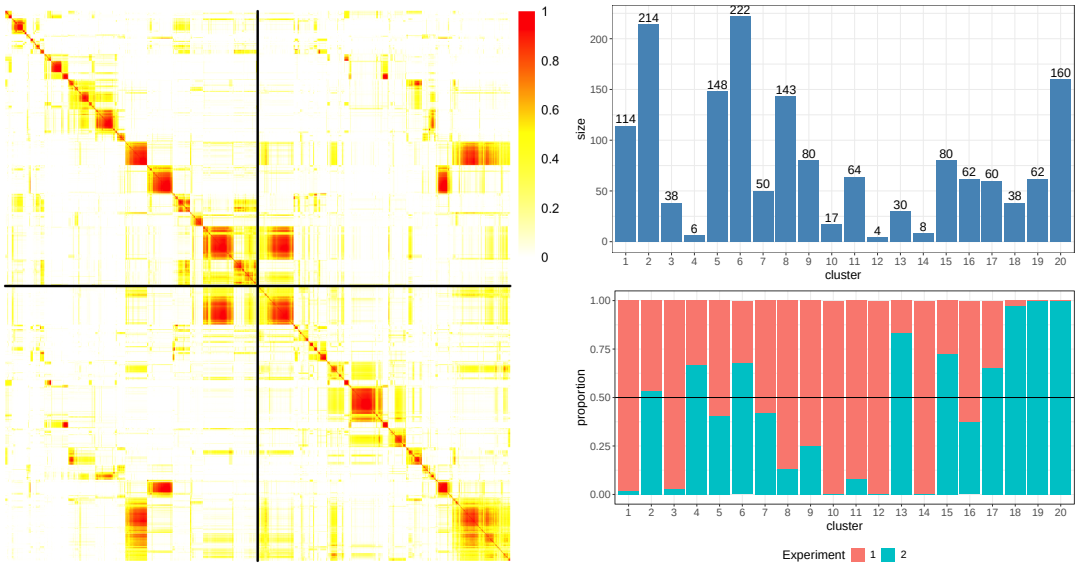


Figure 13: Left: Posterior similarity matrix within and between experiments. Diagonal blocks correspond to within-experiment PSM. Top-right: Size of each cluster in the calcium imaging data. Bottom-right: Proportions of time frames from the first and second experiment in each cluster. The black horizontal line shows the overall proportion.

where $\hat{\boldsymbol{a}}_j^*$ and $\hat{\mathbf{B}}_j^*$ denote the posterior mean of the coefficients for cluster j that $\boldsymbol{y}_{i,d}$ belongs to. The estimated relationship is shown in the colored solid lines in Figure 14. It is worth noting that an almost linear relationship is observed for cluster 5 in the first dimension and cluster 17 in the second dimension, suggesting a dominant role of the past observation in the same reduced dimension. Nonlinear relationships instead indicate interactions between different lower-dimensional embeddings that represent different aspects of summaries of neuronal activities.

For a comparison, we visualize the clusters obtained from fitting a Gaussian mixture model (GMM) with an unconstrained covariance matrix[1] (Equation 8) after pooling the data sets. The pairplots of the encoded activity (Figure 15) display a similar cyclic behavior in both experiments. For the C-HDP method, each shared cluster tends to accumulate at the same edge or vertex. For instance, clusters 5 and 6 (orange and black) are mainly in the top-left corner, shared in both experiments. Clusters 8 and 9 (light blue and light pink) concentrate around the upper-right and lower-right edges, respectively, and such pattern is also similar across two experiments. However, GMM is unable to capture such characteristics. Instead, GMM simply groups time frames into blocks of similar expressions (e.g. light green and light pink clusters with $y_1 > 0.2$ in the first experiment), and clusters are therefore less likely to be shared at the same edge or vertex across experiments. A full set of pairwise scatterplots is shown in online Appendix.

1. GMM is implemented using R package `mclust` (Scrucca et al., 2016).

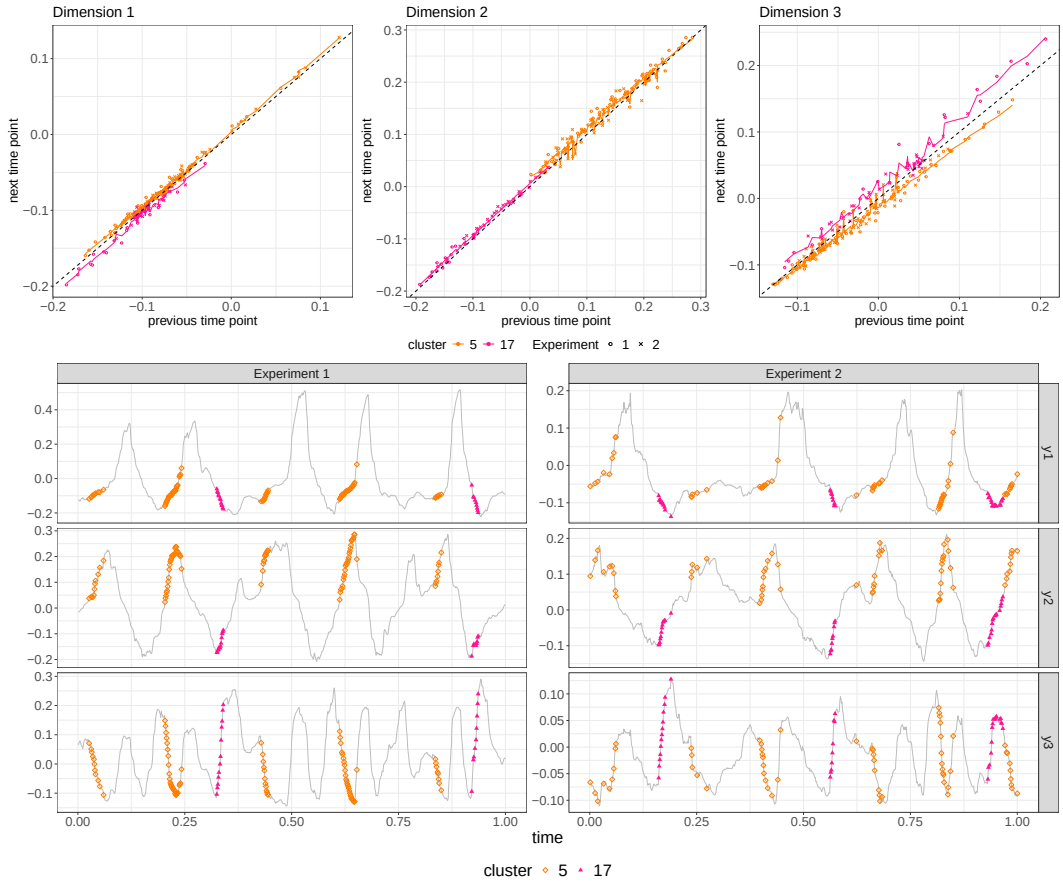


Figure 14: Top: Plot of current time frame against the previous time frame for each dimension, for clusters 5 and 17 in both experiments. The black dashed line denotes $y = x$. The solid colored line denotes the posterior estimated relationship between successive time frames for each cluster. Bottom: Time-series plot for each dimension (row), with clusters 5 and 17 highlighted in color.

Alignment with Behavioral Data. To understand the relationship between clusters and externally recorded behaviors of the mouse, Figure 16 displays the spatial locations of the mouse at each time point belonging to the same cluster. Even though the spatial information is not used for modelling, clusters can still match with the spatial positions and navigation of the mouse, the known variable encoded by the hippocampus. For instance, cluster 2 is likely to correspond to the mouse scurrying around the right end of the linear rig, whereas cluster 15 may represent the mouse moving around the left end. Clusters 5 and 16 coincide with the mouse moving towards the left end and right end, respectively. In particular, cluster 5 spans almost across the whole linear track, while cluster 16 is mainly located at the left half track. This highlights that our model is able to identify meaningful neural activity patterns mapping to the behavioral variables.

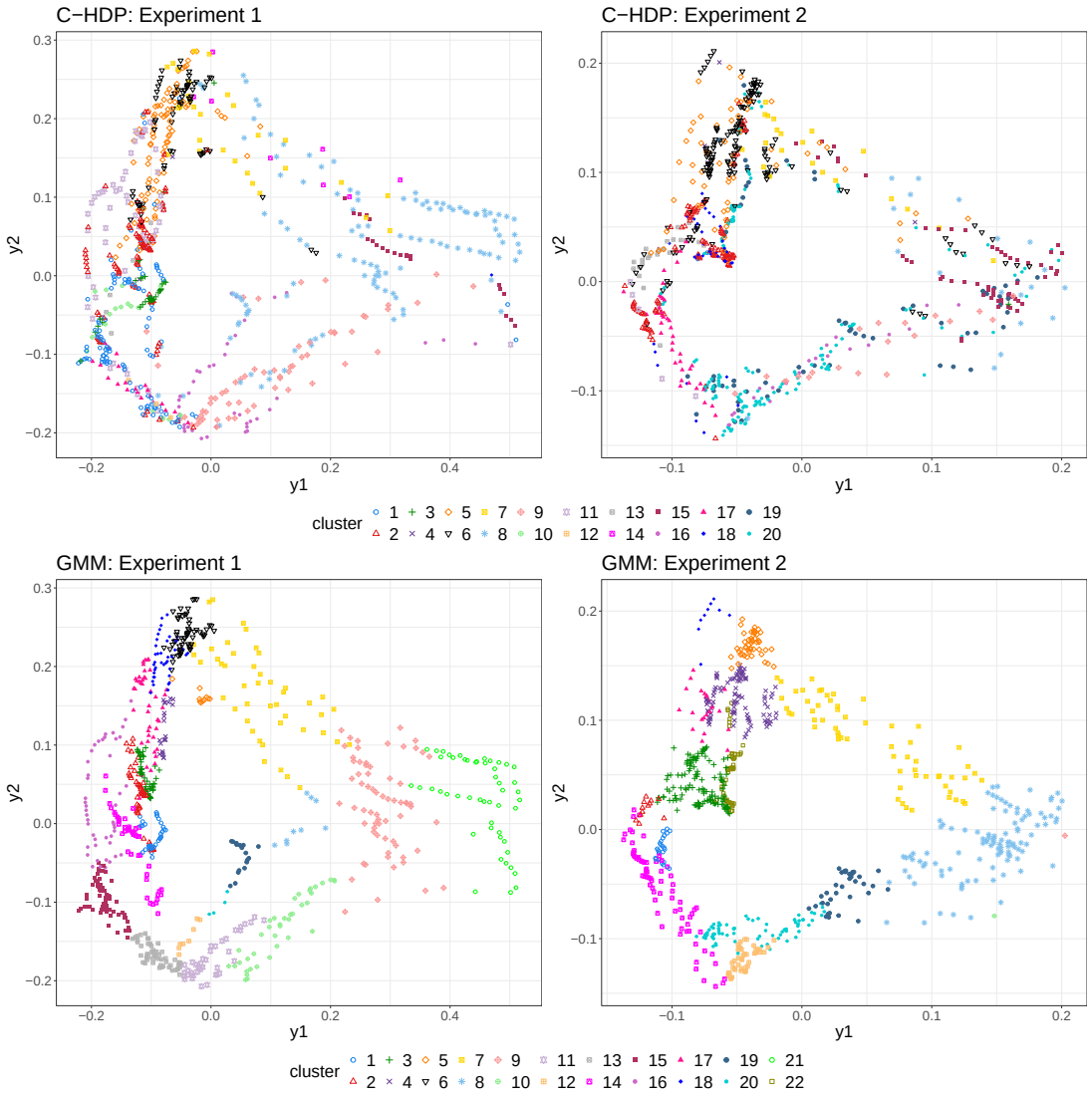


Figure 15: Pairwise scatterplots for the first two dimensions. The top panel displays results from the C-HDP, and the bottom row corresponds to a simple GMM. Columns correspond to different experiments. Time frames are colored by cluster membership.

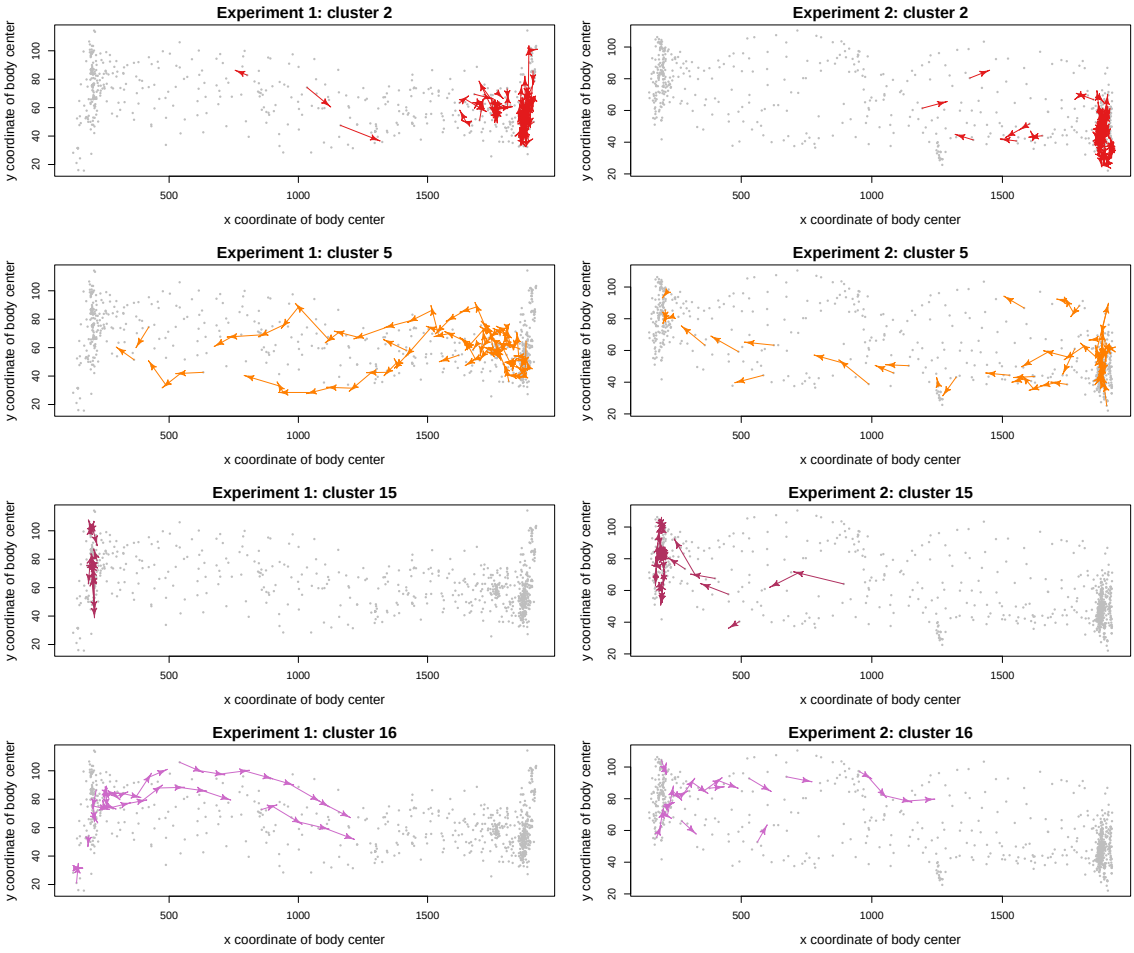


Figure 16: The body locations of the mouse. In each panel, the arrow points from the current point to the next time point, and the color indicates the cluster corresponding to the next time point (arrowhead).

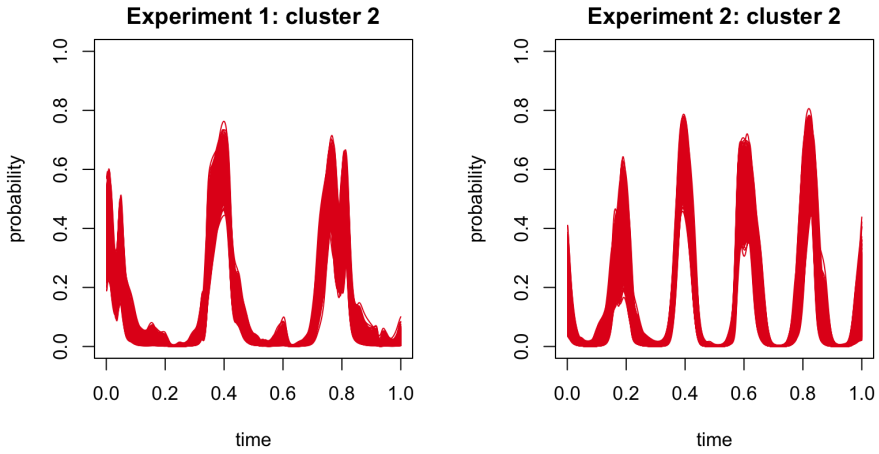


Figure 17: Time-dependent probabilities for cluster 2 for the calcium imaging data.

Time-dependent Probabilities. Figure 17 shows posterior samples of the time-dependent probabilities for cluster 2. The probabilities show an recurring pattern, and there is no substantial difference in the magnitude of uncertainty between two experiments, as they have the same data size. In addition, different experiments can have different periodicity, with cluster 2 happening more frequently in the second experiment, suggesting that the animal probably spent more time moving around the right end of the linear rig in the second experiment (Figure 16). The posterior samples for all clusters are shown in online Appendix.

7. Discussion

In this paper, we have developed a covariate-dependent hierarchical Dirichlet process prior to flexibly integrate external covariates into clustering and density estimation across related groups, combining the strengths from both the HDP and DDP. An efficient MCMC sampling scheme is provided for inference based on data augmentation tricks. The dependence on the covariate is introduced by appropriate kernel functions and covariate-dependent weights are constructed based on normalization for enhanced interpretability. Our approach is fairly general in terms of the choice of the component-specific likelihood and kernel, which can be subjective and depend on the characteristics of the data.

We illustrated the utility of the C-HDP on two real data sets using two examples of kernel functions: a Gaussian kernel and a periodic kernel, and two examples of component-specific likelihood functions: negative-binomial and vector autoregression. The results demonstrate that our C-HDP prior yields meaningful clusters for both data sets. The covariate-dependent probabilities enhance our understanding of the influence of external covariates on clustering and differences between groups. For the scRNA-seq data, the identified clusters reveal separation in the lower dimensional embeddings and latent time. In particular, the under-represented clusters in the mutant group are associated with larger latent time as well as high covariate-dependent probabilities (close to 1), suggesting Pax6 plays an important role in the later stage of the biological process. On the other hand, stable clusters have relatively

smaller latent time in the mutant group with moderately large probabilities (above 0.5). The C-HDP method further enables use to discern the relationship between latent counts and latent time through a nonparametric estimate, which shows a similar pattern to that of the spliced mRNA counts. For the calcium imaging data, time frames from the same cluster exhibit a homogeneous dependence structure with the previous time point, which cannot be achieved by a simple Gaussian mixture model. In addition, without using the spatial information in the model which is known to be influenced by the hippocampus where the data was collected, the clusters are still found closely aligned with the spatial locations and navigation of the animal.

However, while we have constructed an efficient Gibbs sampling algorithm for posterior inference, the algorithm may still face the dilemma of getting trapped in the local modes and a large number of iterations is needed to reach convergence. The problem is even more severe for high-dimensional scRNA-seq data with large number of cells and genes. We have employed consensus clustering (Coleman et al., 2022) to overcome these difficulties to exploit the clustering. In the future, alternative methods will be considered, and one option is the parallelizable posterior bootstrap (Fong et al., 2019) suitable for multimodal posteriors. Moreover, for the Pax6 application, incorporating both the clustering and estimation of latent time into a single model instead of two separate steps would better account for the uncertainty in the scRNA-seq data. As for the calcium imaging application, future work could incorporate dimension reduction within the modeling-based clustering approach (Chandra et al., 2023). Finally, it is worth extending the model to encompass covariate-dependent atoms as well, enhancing its applicability to more complex data sets.

Acknowledgments

The authors are grateful to the Centre for Statistics for funding of the joint workshops with the Centre for Discovery Brain Sciences (CDBS) at the University of Edinburgh. The Pax6 data was provided by Prof. David Price's lab (CDBS) and the calcium imaging data was provided by Dr. Alessandro Di Filippo and Prof. Matt Nolan (CDBS).

The authors declare no competing interests.

An R package to implement the proposed models is available at `https://github.com/huizizhang949/chdp`. Additional examples for the simulation study are available at `https://github.com/huizizhang949/chdp_example`.

Supplementary Materials

This supplement includes the details of the MCMC algorithm for the proposed C-HDP model in the analysis of the single-cell RNA sequencing data and calcium imaging data. Results from the simulation study concerning different types of kernel functions and within-component likelihood functions are provided. Additional results for real data applications are discussed.

Appendix A. Posterior Inference for Pax6 Data

We assume the following likelihood for the mRNA count $y_{c,g,d}$ for gene g ($g = 1, \ldots, G$) in cell c ($c = 1, \ldots, C_d$) from group d ($d = 1, \ldots, D, D = 2$):

$$y_{c,g,d}|\mu_{c,g,d}, \phi_{c,g,d}, \beta_{c,d} \sim \text{NB}(\mu_{c,g,d}\beta_{c,d}, \phi_{c,g,d}).$$

Under the C-HDP mixture model, the cell-specific mean expression and dispersion parameters $\mu_{c,g,d}$ and $\phi_{c,g,d}$ are assumed

$$(\boldsymbol{\mu}_{c,d}, \boldsymbol{\phi}_{c,d})|P_{t_{c,d},d} \sim P_{t_{c,d},d}, \quad P_{t_{c,d},d} \sim \text{C-HDP}(\alpha_0, \alpha, P_0, \boldsymbol{\Psi}^*),$$

where $\boldsymbol{\mu}_{c,d} = (\mu_{c,1,d}, \ldots, \mu_{c,G,d})$ and $\boldsymbol{\phi}_{c,d} = (\phi_{c,1,d}, \ldots, \phi_{c,G,d})$ denote the collection of parameters across all genes.

A.1 Prior Specification

Below we specify the priors for all parameters.

Base Measure. For the base measure P_0, we follow Liu et al. (2024) to model the relationship between $\mu^*_{j,g}$ and $\phi^*_{j,g}$ (component-specific parameters for gene g) as follows:

$$\mu^*_{j,g} \overset{i.i.d}{\sim} \text{log-N}(0, \alpha^2_\mu), \quad \phi^*_{j,g}|\mu^*_{j,g} \overset{ind}{\sim} \text{log-N}(b_0 + b_1 \log(\mu^*_{j,g}), \alpha^2_\phi),$$

where log-N denotes the log-normal distribution. The linear relationship between the logarithmic mean expression and dispersion has been observed in Brennecke et al. (2013), Vallejos et al. (2015) and Eling et al. (2018). The value of α^2_μ is set using the empirical estimates for the mean parameters from *bayNorm* (Tang et al., 2020) (see Section A.12). The mean-dispersion parameters $\boldsymbol{b} = (b_0, b_1)^T$ and α^2_ϕ have hyper-priors as follows

$$\boldsymbol{b}|\alpha^2_\phi \sim \text{N}(\boldsymbol{m}_b, \alpha^2_\phi \mathbf{V}_b), \quad \alpha^2_\phi \sim \text{IG}(\nu_1, \nu_2),$$

where IG is the inverse-gamma distribution and by default $\mathbf{V}_b = \mathbf{I}$, and we use the estimated mean and dispersion parameters from bayNorm to determine $\boldsymbol{m}_b, \nu_1$ and ν_2.

Capture Efficiencies $\beta_{c,d}$. The prior for capture efficiencies $\beta_{c,d}$ is

$$\beta_{c,d} \overset{ind}{\sim} \text{Beta}(a^\beta_d, b^\beta_d),$$

where the values of a^β_d, b^β_d are based on the empirical estimates from bayNorm (see Section A.12). To avoid bimodal and exponentially decaying (increasing) shape of the Beta prior, we set $a^\beta_d, b^\beta_d > 1$. For identifiability of $\beta_{c,d}$, an informative prior is used, where the mean is specified to be an estimate of global mean capture efficiency across cells 0.06 (Tang et al., 2020).

Kernel Parameters. For efficient MCMC sampling, a hierarchical prior for kernel parameters $\boldsymbol{\psi}^*_{j,d} = (t^*_{j,d}, {\sigma^*_{j,d}}^2)$ in the Gaussian kernel is used:

$$t^*_{j,d} \overset{ind}{\sim} \mathrm{N}(r_j, s^2), \quad r_j \overset{i.i.d}{\sim} \mathrm{N}(\mu_r, \sigma_r^2), \quad s^2 \sim \mathrm{IG}(\eta_1, \eta_2),$$
$${\sigma^*_{j,d}}^2 \overset{ind}{\sim} \text{log-N}(h_j, m^2), \quad h_j \overset{i.i.d}{\sim} \mathrm{N}(\mu_h, \sigma_h^2), \quad m^2 \sim \mathrm{IG}(\kappa_1, \kappa_2).$$

The prior means (r_j, h_j) for $t^*_{j,d}$ and ${\sigma^*_{j,d}}^2$ are component-specific, and (r_j, h_j) are given Gaussian hyper-priors with global means (μ_r, μ_h) to allow for borrowing of information across groups, which is similar to the hierarchical prior for $q^J_{j,d}$. The values of hyperparameters are set as $\mu_r = 0.5, \sigma_r = 0.5, \eta_1 = 5, \eta_2 = 1, \mu_h = -5, \sigma_h = 0.5, \kappa_1 = 5, \kappa_2 = 1$.

Concentration Parameters α, α_0. For concentration parameters, weakly informative priors are used

$$\alpha \sim \mathrm{Gamma}(1,1), \quad \alpha_0 \sim \mathrm{Gamma}(1,1).$$

If prior information on the number of clusters is available, we can use this information to set the hyperparameters.

Using a finite-dimensional truncation at J, the complete model is as follows:

$$\begin{aligned}
y_{c,g,d} | z_{c,d} = j, \mu^*_{j,g}, \phi^*_{j,g}, \beta_{c,d} &\overset{ind}{\sim} \mathrm{NB}(\mu^*_{j,g}\beta_{c,d}, \phi^*_{j,g}),\\
z_{c,d} | p^J_{1,d}(t_{c,d}), \ldots, p^J_{J,d}(t_{c,d}) &\overset{ind}{\sim} \mathrm{Cat}(p^J_{1,d}(t_{c,d}), \ldots, p^J_{J,d}(t_{c,d})),\\
p^J_{j,d}(t_{c,d}) &= \frac{q^J_{j,d} K(t_{c,d}|\boldsymbol{\psi}^*_{j,d})}{\sum_{k=1}^J q_{k,d} K(t_{c,d}|\boldsymbol{\psi}^*_{k,d})},\\
q^J_{j,d} &\overset{ind}{\sim} \mathrm{Gamma}(\alpha p^J_j, 1),\\
p^J_1, \ldots, p^J_J &\sim \mathrm{Dir}\left(\frac{\alpha_0}{J}, \ldots, \frac{\alpha_0}{J}\right),\\
t^*_{j,d} &\overset{ind}{\sim} \mathrm{N}(r_j, s^2),\\
r_j &\overset{i.i.d}{\sim} \mathrm{N}(\mu_r, \sigma_r^2),\\
s^2 &\sim \mathrm{IG}(\eta_1, \eta_2),\\
{\sigma^*_{j,d}}^2 &\overset{ind}{\sim} \text{log-N}(h_j, m^2),\\
h_j &\overset{i.i.d}{\sim} \mathrm{N}(\mu_h, \sigma_h^2),\\
m^2 &\sim \mathrm{IG}(\kappa_1, \kappa_2),\\
\mu^*_{j,g} &\overset{i.i.d}{\sim} \text{log-N}(0, \alpha_\mu^2),\\
\phi^*_{j,g} | \mu^*_{j,g} &\overset{ind}{\sim} \text{log-N}(b_0 + b_1 \log(\mu^*_{j,g}), \alpha_\phi^2),\\
\beta_{c,d} &\overset{ind}{\sim} \mathrm{Beta}(a^\beta_d, b^\beta_d),\\
\boldsymbol{b} | \alpha_\phi^2 &\sim \mathrm{N}(\boldsymbol{m}_b, \alpha_\phi^2 \mathbf{V}_b),\\
\alpha_\phi^2 &\sim \mathrm{IG}(\nu_1, \nu_2),\\
\alpha &\sim \mathrm{Gamma}(1,1),\\
\alpha_0 &\sim \mathrm{Gamma}(1,1).
\end{aligned}$$

Define $\mathbf{Z} = \{z_{c,d}\}_{c=1,d=1}^{C_d,D}$, $\mathbf{Y} = \{y_{c,g,d}\}_{c=1,g=1,d=1}^{C_d,G,D}$, $\boldsymbol{t} = \{t_{c,d}\}_{c=1,d=1}^{C_d,D}$, $\boldsymbol{q}^J = \left\{q_{j,d}^J\right\}_{j=1,d=1}^{J,D}$, $\boldsymbol{p}^J = (p_1^J, \ldots, p_J^J)$, $\boldsymbol{\mu}_j^* = (\mu_{j,1}^*, \ldots, \mu_{j,G}^*)$, $\boldsymbol{\phi}_j^* = (\phi_{j,1}^*, \ldots, \phi_{j,G}^*)$, $\boldsymbol{\beta} = \{\beta_{c,d}\}_{c=1,d=1}^{C_d,D}$, $\boldsymbol{\xi} = \{\xi_{c,d}\}_{c=1,d=1}^{C_d,D}$, $\boldsymbol{b} = (b_0, b_1)^T$, $\boldsymbol{t}^* = \left\{t_{j,d}^*\right\}_{j=1,d=1}^{J,D}$, $\boldsymbol{\sigma}^{*2} = \left\{{\sigma_{j,d}^*}^2\right\}_{j=1,d=1}^{J,D}$, $\boldsymbol{r} = (r_1, \ldots, r_J)$, $\boldsymbol{h} = (h_1, \ldots, h_J)$. The posterior distribution is

$$
\begin{aligned}
&\pi(\mathbf{Z}, \boldsymbol{q}^J, \boldsymbol{p}^J, \boldsymbol{\mu}^*, \boldsymbol{\phi}^*, \boldsymbol{\beta}, \boldsymbol{\xi}, \alpha, \alpha_0, \boldsymbol{b}, \alpha_\phi^2, \boldsymbol{t}^*, \boldsymbol{\sigma}^{*2}, \boldsymbol{r}, s^2, \boldsymbol{h}, m^2 | \mathbf{Y}, \boldsymbol{t}) \\
\propto &\prod_{j=1}^{J} \prod_{(c,d):z_{c,d}=j} \prod_{g=1}^{G} \mathrm{NB}(y_{c,d,g} | \mu_{j,g}^* \beta_{c,d}, \phi_{j,g}^*) \\
&\times \prod_{j=1}^{J} \prod_{d=1}^{D} \prod_{c:z_{c,d}=j} K(t_{c,d} | \boldsymbol{\psi}_{j,d}^*) \times \prod_{j=1}^{J} \prod_{d=1}^{D} \left(q_{j,d}^J\right)^{N_{j,d}} \\
&\times \prod_{j=1}^{J} \prod_{d=1}^{D} \prod_{c=1}^{C_d} \exp\left(-\xi_{c,d} q_{j,d}^J K(t_{c,d} | \boldsymbol{\psi}_{j,d}^*)\right) \\
&\times \prod_{j=1}^{J} \prod_{d=1}^{D} \mathrm{Gamma}(q_{j,d}^J | \alpha p_j^J, 1) \times \mathrm{Dir}\left(\boldsymbol{p}^J | \frac{\alpha_0}{J}, \ldots, \frac{\alpha_0}{J}\right) \\
&\times \prod_{j=1}^{J} \prod_{g=1}^{G} \left[\text{log-N}\left(\mu_{j,g}^* | 0, \alpha_\mu^2\right) \times \text{log-N}\left(\phi_{j,g}^* | b_0 + b_1 \log\left(\mu_{j,g}^*\right), \alpha_\phi^2\right)\right] \\
&\times \prod_{d=1}^{D} \prod_{c=1}^{C_d} \mathrm{Beta}(\beta_{c,d} | a_d^\beta, b_d^\beta) \times \mathrm{Gamma}(\alpha | 1, 1) \times \mathrm{Gamma}(\alpha_0 | 1, 1) \\
&\times \mathrm{N}(\boldsymbol{b} | m_b, \alpha_\phi^2 \mathbf{V}_b) \times \mathrm{IG}(\alpha_\phi^2 | \nu_1, \nu_2) \\
&\times \prod_{j=1}^{J} \prod_{d=1}^{D} \left[\mathrm{N}(t_{j,d}^* | r_j, s^2) \times \text{log-N}({\sigma_{j,d}^*}^2 | h_j, m^2)\right] \times \prod_{j=1}^{J} \left[\mathrm{N}(r_j | \mu_r, \sigma_r^2) \times \mathrm{N}(h_j | \mu_h, \sigma_h^2)\right] \\
&\times \mathrm{IG}(s^2 | \eta_1, \eta_2) \times \mathrm{IG}(m^2 | \kappa_1, \kappa_2),
\end{aligned}
\tag{11}
$$

where $N_{j,d} = \sum_{c=1}^{C_d} \mathbb{I}(z_{c,d} = j)$ is the number of cells in component j in the d-th group, $\mathbb{I}(\cdot)$ is the indicator function that takes the value 1 if the condition inside the bracket holds, and is 0 otherwise. Note that the first three lines in Equation (11) comes from the augmented data likelihood after introducing the latent variables $\xi_{c,d}$. The MCMC algorithm (Gibbs sampling) iteratively samples from the full conditional distributions of (blocked) parameters. For standard full conditional densities, we can draw samples directly, while adaptive Metropolis-Hastings (AMH) is used for non-standard forms. Define $\boldsymbol{C} = (C_1, \ldots, C_D)$. The time complexity for each parameter is provided below.

- parameters in covariate-dependent weights $\boldsymbol{q}^J$: $\mathcal{O}(\text{sum}(\boldsymbol{C})J)$.
- latent cell-specific parameters $\boldsymbol{\xi}$: $\mathcal{O}(\text{sum}(\boldsymbol{C})J)$.
- latent parameters $\{u_{c,j,d}\}_{c=1,j=1,d=1}^{C_d,J,D}$ for efficient sampling of kernel parameters: $\mathcal{O}(\text{sum}(\boldsymbol{C})J)$.

- kernel parameters (centre) $\boldsymbol{t}^*$: $\mathcal{O}(\text{sum}(\boldsymbol{C})J)$.
- kernel parameters (bandwidth) $\boldsymbol{\sigma}^{*2}$: $\mathcal{O}(\text{sum}(\boldsymbol{C})J)$.
- concentration parameter α: $\mathcal{O}(JD)$.
- concentration parameter α_0: $\mathcal{O}(J)$.
- allocation variables $\mathbf{Z}$: $\mathcal{O}(\text{sum}(\boldsymbol{C})JG)$.
- component probabilities $\boldsymbol{p}^J$: $\mathcal{O}(JD)$.
- mean-dispersion parameters $\boldsymbol{b}, \alpha_\phi^2$: $\mathcal{O}(JG)$.
- component-specific parameters $\boldsymbol{\mu}^*_{1:J,1:G}, \boldsymbol{\phi}^*_{1:J,1:G}$: $\mathcal{O}(\text{sum}(\boldsymbol{C})JG)$.
- capture efficiencies $\boldsymbol{\beta}$: $\mathcal{O}(\text{sum}(\boldsymbol{C})G)$.
- hyperparameters $\boldsymbol{r}, s^2, \boldsymbol{h}, m^2$ for the kernel parameters: $\mathcal{O}(JD)$.

Next we provide the details of sampling each parameter.

A.2 Group-specific Parameters for Component-specific Likelihood $q^J_{j,d}$

For each j and d, the full conditional distribution is

$$\begin{aligned}
&\pi(q^J_{j,d}|\left\{z_{c,d}\right\}_{c=1}^{C_d}, \alpha, p^J_j, \left\{\xi_{c,d}\right\}_{c=1}^{C_d}, \left\{t_{c,d}\right\}_{c=1}^{C_d}, t^*_{j,d}, {\sigma^*_{j,d}}^2) \\
\propto &\left(q^J_{j,d}\right)^{N_{j,d}} \times \exp\left(-q^J_{j,d}\sum_{c=1}^{C_d}\xi_{c,d}K(t_{c,d}|\boldsymbol{\psi}^*_{j,d})\right) \\
&\times \left(q^J_{j,d}\right)^{\alpha p^J_j - 1}\exp\left(-q^J_{j,d}\right) \\
\propto &\left(q^J_{j,d}\right)^{N_{j,d}+\alpha p^J_j - 1} \times \exp\left(-q^J_{j,d}\left[1+\sum_{c=1}^{C_d}\xi_{c,d}K(t_{c,d}|\boldsymbol{\psi}^*_{j,d})\right]\right),
\end{aligned}$$

i.e.

$$q^J_{j,d}|\ldots \sim \text{Gamma}\left(N_{j,d}+\alpha p^J_j, 1+\sum_{c=1}^{C_d}\xi_{c,d}K(t_{c,d}|\boldsymbol{\psi}^*_{j,d})\right).$$

A.3 Latent Cell-specific Parameters $\xi_{c,d}$

For each c and d, the full conditional distribution is

$$\pi(\xi_{c,d}|\boldsymbol{q}^J_{1:J,d}, t_{c,d}, \boldsymbol{t}^*_{1:J,d}, {\boldsymbol{\sigma}^*_{1:J,d}}^2) \propto \exp\left(-\xi_{c,d}\sum_{j=1}^{J}q^J_{j,d}K(t_{c,d}|\boldsymbol{\psi}^*_{j,d})\right),$$

i.e.

$$\xi_{c,d}|\ldots \sim \text{Gamma}\left(1, \sum_{j=1}^{J}q^J_{j,d}K(t_{c,d}|\boldsymbol{\psi}^*_{j,d})\right).$$

A.4 Kernel Parameters $t^*_{j,d}$ and ${\sigma^*_{j,d}}^2$

The joint full conditional distribution for $\boldsymbol{t}^*$ and $\boldsymbol{\sigma}^{*2}$ is

$$\begin{aligned}
&\pi(\boldsymbol{t}^*, \boldsymbol{\sigma}^{*2} | \mathbf{Z}, \boldsymbol{t}, \boldsymbol{\xi}, \boldsymbol{q}^J, \boldsymbol{r}, s^2, \boldsymbol{h}, m^2) \\
&\propto \prod_{j=1}^{J} \prod_{d=1}^{D} \prod_{c: z_{c,d}=j} K(t_{c,d} | \boldsymbol{\psi}^*_{j,d}) \times \prod_{j=1}^{J} \prod_{d=1}^{D} \prod_{c=1}^{C_d} \exp\left(-\xi_{c,d} q^J_{j,d} K(t_{c,d} | \boldsymbol{\psi}^*_{j,d})\right) \\
&\quad \times \prod_{j=1}^{J} \prod_{d=1}^{D} \left[\mathrm{N}(t^*_{j,d} | r_j, s^2) \times \text{log-N}({\sigma^*_{j,d}}^2 | h_j, m^2) \right].
\end{aligned}$$

Due to the presence of the exponential term, it is impossible to obtain standard distributions. We introduce latent variables $\boldsymbol{u} = \{u_{c,j,d}\}_{c=1,j=1,d=1}^{C_d,J,D} \in (0,1)$, and the joint full conditional distribution becomes

$$\begin{aligned}
\pi(\boldsymbol{t}^*, \boldsymbol{\sigma}^{*2}, \boldsymbol{u} | \ldots) \propto &\prod_{j=1}^{J} \prod_{d=1}^{D} \prod_{c: z_{c,d}=j} K(t_{c,d} | \boldsymbol{\psi}^*_{j,d}) \times \prod_{j=1}^{J} \prod_{d=1}^{D} \prod_{c=1}^{C_d} \mathbb{I}\left(u_{c,j,d} < M_{c,j,d}\right) \\
&\times \prod_{j=1}^{J} \prod_{d=1}^{D} \left[\mathrm{N}(t^*_{j,d} | r_j, s^2) \times \text{log-N}({\sigma^*_{j,d}}^2 | h_j, m^2) \right],
\end{aligned} \tag{12}$$

where $M_{c,j,d} = \exp\left(-\xi_{c,d} q^J_{j,d} K(t_{c,d} | \boldsymbol{\psi}^*_{j,d})\right)$. Integrating out $u_{c,j,d}$ on $(0,1)$ restores the joint full conditional distribution for $\boldsymbol{t}^*$ and $\boldsymbol{\sigma}^{*2}$.

A.4.1 Latent Parameters $u_{c,j,d}$

For each c, j and d, the full conditional distribution of the latent variable is

$$\pi(u_{c,j,d} | \xi_{c,d}, q^J_{j,d}, t_{c,d}, t^*_{j,d}, {\sigma^*_{j,d}}^2) \propto \mathbb{I}\left(u_{c,j,d} < M_{c,j,d}\right),$$

i.e.

$$u_{c,j,d} | \ldots \sim \text{Unif}\left(0, \exp\left(-\xi_{c,d} q^J_{j,d} K(t_{c,d} | \boldsymbol{\psi}^*_{j,d})\right)\right).$$

A.4.2 Centre $t^*_{j,d}$

For each j and d, the full conditional distribution is

$$\begin{aligned}
&\pi(t^*_{j,d} | r_j, s^2, \{z_{c,d}\}_{c=1}^{C_d}, \{\xi_{c,d}\}_{c=1}^{C_d}, \{u_{c,j,d}\}_{c=1}^{C_d}, \{t_{c,d}\}_{c=1}^{C_d}, q^J_{j,d}, {\sigma^*_{j,d}}^2) \\
&\propto \prod_{c: z_{c,d}=j} K(t_{c,d} | \boldsymbol{\psi}^*_{j,d}) \times \mathrm{N}(t^*_{j,d} | r_j, s^2) \times \mathbb{I}(t^*_{j,d} \in A_{j,d}).
\end{aligned}$$

Let $I_{j,d} = \{c : z_{c,d} = j\}$. The first two terms are proportional to

$$\begin{aligned}
&\exp\left[-\frac{1}{2{\sigma^*_{j,d}}^2}\sum_{I_{j,d}}\left(t^*_{j,d}-t_{c,d}\right)^2\right]\times\exp\left[-\frac{1}{2s^2}\left(t^*_{j,d}-r_j\right)^2\right]\\
\propto&\exp\left[-\frac{1}{2{\sigma^*_{j,d}}^2}\left(N_{j,d}{t^*_{j,d}}^2-2t^*_{j,d}\sum_{I_{j,d}}t_{c,d}\right)-\frac{1}{2s^2}\left({t^*_{j,d}}^2-2t^*_{j,d}r_j\right)\right]\\
\propto&\exp\left(-\frac{1}{2{\sigma^*_{j,d}}^2s^2}\left[\left({\sigma^*_{j,d}}^2+N_{j,d}s^2\right){t^*_{j,d}}^2-2t^*_{j,d}\left(r_j{\sigma^*_{j,d}}^2+s^2\sum_{I_{j,d}}t_{c,d}\right)\right]\right)\\
\propto&\mathrm{N}(t^*_{j,d}|\hat{r}_{j,d},\hat{s}^2_{j,d}),
\end{aligned}$$

where

$$\hat{s}^2_{j,d}=\left(\frac{1}{s^2}+\frac{N_{j,d}}{{\sigma^*_{j,d}}^2}\right)^{-1},\quad \hat{r}_{j,d}=\frac{r_j/s^2+\sum_{I_{j,d}}t_{c,d}/{\sigma^*_{j,d}}^2}{1/s^2+N_{j,d}/{\sigma^*_{j,d}}^2}.$$

The indicator function $\mathbb{I}(t^*_{j,d}\in A_{j,d})$ results from the one in Equation (12), leading to a truncated normal distribution. The truncation region $A_{j,d}$ is

$$\begin{aligned}
A_{j,d}&=\bigcap_{c=1}^{C_d}A_{c,j,d}=\bigcap_{c=1}^{C_d}\left\{t^*_{j,d}:u_{c,j,d}<\exp\left(-\xi_{c,d}q^J_{j,d}K(t_{c,d}|\psi^*_{j,d})\right)\right\}\\
&=\bigcap_{c=1}^{C_d}\left\{t^*_{j,d}:-\frac{\log u_{c,j,d}}{\xi_{c,d}q^J_{j,d}}>\exp\left(-\frac{1}{2{\sigma^*_{j,d}}^2}\left(t_{c,d}-t^*_{j,d}\right)^2\right)\right\}\\
&=\bigcap_{c=1}^{C_d}\left\{t^*_{j,d}:\log\left[-\frac{\log u_{c,j,d}}{\xi_{c,d}q^J_{j,d}}\right]>-\frac{1}{2{\sigma^*_{j,d}}^2}\left(t_{c,d}-t^*_{j,d}\right)^2\right\}.
\end{aligned}$$

Since the right-hand side is always negative, if $-\frac{\log u_{c,j,d}}{\xi_{c,d}q^J_{j,d}}\geq 1$, i.e. $-\log u_{c,j,d}\geq\xi_{c,d}q^J_{j,d}$, we will have $A_{c,j,d}=\mathbb{R}$ and hence there is no truncation. Otherwise,

$$A_{c,j,d}=\left(-\infty,t_{c,d}-\sqrt{-2{\sigma^*_{j,d}}^2\log\left[-\frac{\log u_{c,j,d}}{\xi_{c,d}q^J_{j,d}}\right]}\right)\bigcup\left(t_{c,d}+\sqrt{-2{\sigma^*_{j,d}}^2\log\left[-\frac{\log u_{c,j,d}}{\xi_{c,d}q^J_{j,d}}\right]},+\infty\right).$$

Hence the region $A_{j,d}$ is given by

$$A_{j,d}=\bigcap_{c:-\log u_{c,j,d}<\xi_{c,d}q^J_{j,d}}A_{c,j,d}.$$

Note that if there is no cell in group d that belongs to component j, we will sample $t^*_{j,d}$ from the prior truncated to $A_{j,d}$. Furthermore, if it satisfies that $\left\{c:-\log u_{c,j,d}<\xi_{c,d}q^J_{j,d}\right\}=\emptyset$, there is no truncation. Therefore, there are four possible cases, based on truncation or not and whether or not the component j is empty in group d.

A.4.3 Bandwidth ${\sigma^*_{j,d}}^2$

For each j and d, the full conditional distribution is

$$\begin{aligned}&\pi({\sigma^*_{j,d}}^2|h_j, m^2, \{z_{c,d}\}_{c=1}^{C_d}, \{\xi_{c,d}\}_{c=1}^{C_d}, \{u_{c,j,d}\}_{c=1}^{C_d}, \{t_{c,d}\}_{c=1}^{C_d}, q^J_{j,d}, t^*_{j,d})\\ \propto &\prod_{c:z_{c,d}=j} K(t_{c,d}|\boldsymbol{\psi}^*_{j,d}) \times \text{log-N}({\sigma^*_{j,d}}^2|h_j, m^2) \times \mathbb{I}({\sigma^*_{j,d}}^2 \in B_{j,d}).\end{aligned}$$

The first two terms are proportional to

$$\exp\left[-\frac{1}{2{\sigma^*_{j,d}}^2}\sum_{I_{j,d}}\left(t^*_{j,d}-t_{c,d}\right)^2\right] \times \frac{1}{{\sigma^*_{j,d}}^2}\exp\left[-\frac{1}{2m^2}\left(\log\left({\sigma^*_{j,d}}^2\right)-h_j\right)^2\right], \tag{13}$$

which is not a standard form when component j is occupied, and hence we will apply adaptive Metropolis-Hastings. The region $B_{j,d}$ is given by

$$\begin{aligned}B_{j,d} &= \bigcap_{c=1}^{C_d} B_{c,j,d} = \bigcap_{c=1}^{C_d}\left\{{\sigma^*_{j,d}}^2 : u_{c,j,d} < \exp\left(-\xi_{c,d}q^J_{j,d}K(t_{c,d}|\boldsymbol{\psi}^*_{j,d})\right)\right\}\\ &= \bigcap_{c=1}^{C_d}\left\{{\sigma^*_{j,d}}^2 : -\frac{\log u_{c,j,d}}{\xi_{c,d}q^J_{j,d}} > \exp\left(-\frac{1}{2{\sigma^*_{j,d}}^2}\left(t_{c,d}-t^*_{j,d}\right)^2\right)\right\}\\ &= \bigcap_{c=1}^{C_d}\left\{{\sigma^*_{j,d}}^2 : \log\left[-\frac{\log u_{c,j,d}}{\xi_{c,d}q^J_{j,d}}\right] > -\frac{1}{2{\sigma^*_{j,d}}^2}\left(t_{c,d}-t^*_{j,d}\right)^2\right\}.\end{aligned}$$

Similar to $t^*_{j,d}$, if $-\frac{\log u_{c,j,d}}{\xi_{c,d}q^J_{j,d}} \geq 1$, there will be no truncation. Otherwise,

$$B_{c,j,d} = \left(0, -\frac{\left(t_{c,d}-t^*_{j,d}\right)^2}{2\log\left[-\frac{\log u_{c,j,d}}{\xi_{c,d}q^J_{j,d}}\right]}\right).$$

Hence the region $B_{j,d}$ is

$$B_{j,d} = \bigcap_{c:-\log u_{c,j,d} < \xi_{c,d}q^J_{j,d}} B_{c,j,d} = \left(0, \sigma^+_{j,d}\right),$$

where

$$\sigma^+_{j,d} = \min_{c:-\log u_{c,j,d} < \xi_{c,d}q^J_{j,d}} -\frac{\left(t_{c,d}-t^*_{j,d}\right)^2}{2\log\left[-\frac{\log u_{c,j,d}}{\xi_{c,d}q^J_{j,d}}\right]}.$$

Adaptive Metropolis-Hastings for ${\sigma^*_{j,d}}^2$ For notation simplicity, we will drop the subscript j, d in this section.

1. Apply the following transformation to ${\sigma^*}^2$:

$$x = g({\sigma^*}^2) = -\log\left(\frac{1}{{\sigma^*}^2} - \frac{1}{\sigma^+}\right) \in \mathbb{R}.$$

The Jacobian term is

$$J_x = \frac{dx}{d{\sigma^*}^2} = \frac{\sigma^+}{{\sigma^*}^2(\sigma^+ - {\sigma^*}^2)}.$$

The inverse transformation is

$${\sigma^*}^2 = \frac{1}{\exp(-x) + 1/\sigma^+} \in \left(0, \sigma^+\right).$$

2. Let d denote the dimension of x ($\mathrm{d} = 1$ for the case of ${\sigma^*_{j,d}}^2$). At the current iteration n, denote the sampled ${\sigma^*}^2$ from iteration $n-1$ as ${\sigma^*_{old}}^2$. Conditional on σ^+ at the current iteration, define $x_{old} = g({\sigma^*_{old}}^2)$.

We use random walk to sample x_{new}. For $n \leq 100$, we draw

$$x_{new} \sim \mathrm{N}(x_{old}, 0.01 \times \mathbf{I}_{\mathrm{d}}).$$

For $n > 100$, letting $s_{\mathrm{d}} = 2.4^2/\mathrm{d}$, we propose

$$x_{new} \sim \mathrm{N}(x_{old}, s_{\mathrm{d}} \times (\Sigma_{n-1} + \epsilon \mathbf{I}_d)),$$

where $\epsilon = 0.01$, and Σ_{n-1} is the sample covariance based on the past iterations, which needs to be updated at each iteration. We obtain ${\sigma^*_{new}}^2$ through the inverse transformation.

3. Next, we compute the acceptance probability. Let $\pi({\sigma^*}^2)$ denote the posterior distribution, and Q_n the proposal distribution at step n. The acceptance probability is

$$\begin{aligned}
\alpha({\sigma^*_{new}}^2, {\sigma^*_{old}}^2) &= \min\left(1, \frac{\pi({\sigma^*_{new}}^2) Q_n({\sigma^*_{old}}^2 | {\sigma^*_{new}}^2)}{\pi({\sigma^*_{old}}^2) Q_n({\sigma^*_{new}}^2 | {\sigma^*_{old}}^2)}\right) \\
&= \min\left(1, \frac{\pi({\sigma^*_{new}}^2)|J_{x_{old}}|}{\pi({\sigma^*_{old}}^2)|J_{x_{new}}|}\right) \\
&= \min\left(1, \exp\left[\log \pi({\sigma^*_{new}}^2) - \log \pi({\sigma^*_{old}}^2) + \log|J_{x_{old}}| - \log|J_{x_{new}}|\right]\right),
\end{aligned}$$

where $\pi({\sigma^*_{new}}^2)$ and $\pi({\sigma^*_{old}}^2)$ are given by Equation (13) evaluated at the new and old ${\sigma^*}^2$, and $|J_x|$ is provided in step 1, conditional on σ^+ in the current iteration:

$$\frac{|J_{x_{old}}|}{|J_{x_{new}}|} = \frac{{\sigma^*_{new}}^2\left(\sigma^+ - {\sigma^*_{new}}^2\right)}{{\sigma^*_{old}}^2\left(\sigma^+ - {\sigma^*_{old}}^2\right)}. \tag{14}$$

Taking the logarithm of Equation (13) and Equation (14) yields

$$\log \pi\left({\sigma^*}^2 \middle| \ldots\right) = -\frac{1}{2{\sigma^*}^2} \sum_{I_{j,d}} \left(t^*_{j,d} - t_{c,d}\right)^2 - \log\left({\sigma^*}^2\right) - \frac{1}{2m^2}\left(\log\left({\sigma^*}^2\right) - h_j\right)^2 + \text{const.},$$

$$\log\left(\frac{|J_{x_{old}}|}{|J_{x_{new}}|}\right) = \log\left({\sigma^*_{new}}^2\right) + \log\left(\sigma^+ - {\sigma^*_{new}}^2\right) - \log\left({\sigma^*_{old}}^2\right) - \log\left(\sigma^+ - {\sigma^*_{old}}^2\right).$$

4. After making the decision to accept the proposed value or not, we update the sample covariance/variance Σ_n. For computational purposes, Liu et al. (2024) use a recursive formulae to update Σ_n sequentially at each iteration, which is described below.

   For $\mathrm{d} = 1$, the variance Σ_n is computed based on two statistics: $M_2(n)$ and $\bar{x}_n$, which are defined as

$$M_2(n) = \sum_{i=1}^{n}(x_i - \bar{x}_n)^2, \quad \bar{x}_n = \frac{1}{n}\sum_{i=1}^{n} x_i.$$

   The following relationship is observed between $\bar{x}_n$ and $\bar{x}_{n-1}$, and between $M_2(n)$ and $M_2(n-1)$:

$$\begin{aligned}
\bar{x}_n &= \left(1 - \frac{1}{n}\right)\bar{x}_{n-1} + \frac{x_n}{n}, \\
\Sigma_n &= \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x}_n)^2 = \frac{1}{n-1}M_2(n) \\
&= \frac{1}{n-1}[M_2(n-1) + (x_n - \bar{x}_{n-1})(x_n - \bar{x}_n)]. \qquad (15)
\end{aligned}$$

   The proof of Equation (15) is as follows:

$$\begin{aligned}
(n-1)\Sigma_n - (n-2)\Sigma_{n-1} &= \sum_{i=1}^{n}(x_i - \bar{x}_n)^2 - \sum_{i=1}^{n-1}(x_i - \bar{x}_{n-1}) \\
&= (x_n - \bar{x}_n)^2 + \sum_{i=1}^{n-1}((x_i - \bar{x}_n)^2 - (x_i - \bar{x}_{n-1})^2) \\
&= (x_n - \bar{x}_n)^2 + \sum_{i=1}^{n-1}(x_i - \bar{x}_n + x_i - \bar{x}_{n-1})(\bar{x}_{n-1} - \bar{x}_n) \\
&= (x_n - \bar{x}_n)^2 + (\bar{x}_n - x_n)(\bar{x}_{n-1} - \bar{x}_n) \\
&= (x_n - \bar{x}_n)(x_n - \bar{x}_n - \bar{x}_{n-1} + \bar{x}_n) \\
&= (x_n - \bar{x}_n)(x_n - \bar{x}_{n-1}).
\end{aligned}$$

   Hence we first compute $\bar{x}_n$ from $\bar{x}_{n-1}$ and x_n. Then $\bar{x}_n, \bar{x}_{n-1}, x_n$ and $M_2(n-1)$ are used for the calculation of Σ_n.

   For $\mathrm{d} > 1$, we will compute $\boldsymbol{\Sigma}_n$ based on two statistics: $\tilde{\mathbf{S}}(n)$ and $\boldsymbol{m}(n)$ defined as

$$\tilde{\mathbf{S}}(n) = \begin{pmatrix} \sum_{i=1}^{n}(x_{i,1})^2 & \sum_{i=1}^{n} x_{i,1}x_{i,2} & \cdots & \sum_{i=1}^{n} x_{i,1}x_{i,\mathrm{d}} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{i=1}^{n} x_{i,\mathrm{d}}x_{i,1} & \sum_{i=1}^{n} x_{i,\mathrm{d}}x_{i,2} & \cdots & \sum_{i=1}^{n} x_{i,\mathrm{d}}x_{i,\mathrm{d}} \end{pmatrix},$$

   where $x_{i,l}$ is the i-th posterior sample of the l-th unknown parameter, and $\boldsymbol{m}(n)$ is a d-dimension row vector, with the l-th element $m_l(n) = \sum_{i=1}^{n} x_{i,l}/n$ $(l = 1, \ldots, \mathrm{d})$.

The element in the sample covariance matrix after n iterations is given by

$$\begin{aligned}\boldsymbol{\Sigma}_n(u,v) &= \frac{1}{n-1}\sum_{i=1}^{n}(x_{i,u}-m_u(n))(x_{i,v}-m_v(n)) \\ &= \frac{1}{n-1}\left[\sum_{i=1}^{n}x_{i,u}x_{i,v} - m_v(n)\sum_{i=1}^{n}x_{i,u} - m_u(n)\sum_{i=1}^{n}x_{i,v} + n\times m_v(n)m_u(n)\right] \\ &= \frac{1}{n-1}\left[\sum_{i=1}^{n}x_{i,u}x_{i,v} - n\times m_v(n)m_u(n)\right] \\ &= \frac{1}{n-1}\sum_{i=1}^{n}x_{i,u}x_{i,v} - \frac{n}{n-1}m_v(n)m_u(n).\end{aligned}$$

Hence the covariance matrix could be written in the following form:

$$\boldsymbol{\Sigma}_n = \frac{1}{n-1}\tilde{\mathbf{S}}(n) - \frac{n}{n-1}\left(\boldsymbol{m}(n)^T\boldsymbol{m}(n)\right),$$

and we note the following relationship:

$$\begin{aligned}\tilde{\mathbf{S}}(n) &= \tilde{\mathbf{S}}(n-1) + \boldsymbol{x}_n^T\boldsymbol{x}_n, \\ \boldsymbol{m}(n) &= \left(1-\frac{1}{n}\right)\boldsymbol{m}(n-1) + \frac{1}{n}\boldsymbol{x}_n.\end{aligned}$$

Therefore, we first compute $\tilde{\mathbf{S}}(n)$ and $\boldsymbol{m}(n)$ based on $\tilde{\mathbf{S}}(n-1), \boldsymbol{m}(n-1)$ and the new value $\boldsymbol{x}_n$. Then the covariance matrix can be updated given $\tilde{\mathbf{S}}(n)$ and $\boldsymbol{m}(n)$.

Lastly, we note that the above AMH will be applied when component j is occupied. In this case, if there is no truncation (the upper bound $\sigma_{j,d}^{+}=\infty$), the transformation defined in step 1 will reduce to a simple log-transformation, and the Jacobian is simply $1/{\sigma_{j,d}^{*}}^2$. When component j is empty at one iteration, we will draw a new sample from the log-normal prior (may or may not be truncated). In this case, the sample is always accepted and transformed to x to update the covariance/variance.

A.5 Concentration Parameters α and α_0

The full conditional distribution of α is

$$\begin{aligned}\pi(\alpha|\boldsymbol{q}^J,\boldsymbol{p}^J) &\propto \prod_{j=1}^{J}\prod_{d=1}^{D}\text{Gamma}(q_{j,d}^J|\alpha p_j^J,1)\times\text{Gamma}(\alpha|1,1) \\ &\propto \prod_{j=1}^{J}\prod_{d=1}^{D}\left[\frac{1}{\Gamma\left(\alpha p_j^J\right)}\left(q_{j,d}^J\right)^{\alpha p_j^J}\right]\times\exp(-\alpha).\end{aligned}$$

The distribution is not of a standard form and we apply the AMH scheme as described in Section A.4.3. Specifically, we use the log-transformation

$$x=\log(\alpha)\in\mathbb{R}.$$

The Jacobian is

$$J_x = \frac{dx}{d\alpha} = \frac{1}{\alpha}.$$

and the inverse transformation is $\alpha = \exp(x)$.

The logarithm of the full conditional density is

$$\log \pi(\alpha|\boldsymbol{q}^J, \boldsymbol{p}^J) = -\alpha + \sum_{j=1}^{J}\sum_{d=1}^{D}\left[\alpha p_j^J \log\left(q_{j,d}^J\right) - \log\left(\Gamma\left(\alpha p_j^J\right)\right)\right] + \text{const.}$$

Hence the acceptance probability of the new sample is

$$\begin{aligned}
\alpha(\alpha_{new}, \alpha_{old}) &= \min\left(1, \frac{\pi(\alpha_{new})Q_n(\alpha_{old}|\alpha_{new})}{\pi(\alpha_{old})Q_n(\alpha_{new}|\alpha_{old})}\right) \\
&= \min\left(1, \frac{\pi(\alpha_{new})\alpha_{new}}{\pi(\alpha_{old})\alpha_{old}}\right) \\
&= \min\left(1, \exp\left[\log \pi(\alpha_{new}) - \pi(\alpha_{old}) + \log(\alpha_{new}) - \log(\alpha_{old})\right]\right).
\end{aligned}$$

After the decision of rejection or acceptance, we update the sample variance following step 4 of Section A.4.3 (d = 1).

As for α_0, the full conditional distribution is

$$\begin{aligned}
\pi(\alpha_0|\boldsymbol{p}^J) &\propto \text{Gamma}(\alpha_0|1,1) \times \text{Dir}\left(\boldsymbol{p}^J|\frac{\alpha_0}{J}, \ldots, \frac{\alpha_0}{J}\right) \\
&\propto \exp(-\alpha_0) \times \frac{\Gamma(\alpha_0)}{[\Gamma(\frac{\alpha_0}{J})]^J}\prod_{j=1}^{J}\left(p_j^J\right)^{\frac{\alpha_0}{J}}.
\end{aligned}$$

This distribution does not have a closed form and hence we apply AMH (Section A.4.3). Same log transformation as α is applied to transform α_0, and hence details are omitted here. The logarithm of the full conditional distribution is

$$\log \pi(\alpha_0|\boldsymbol{p}^J) = -\alpha_0 + \log(\Gamma(\alpha_0)) - J\log\left(\Gamma\left(\frac{\alpha_0}{J}\right)\right) + \frac{\alpha_0}{J}\sum_{j=1}^{J}\log\left(p_j^J\right) + \text{const.}$$

A.6 Allocation Variables $z_{c,d}$

The full conditional distribution is

$$\pi(z_{c,d} = j|\boldsymbol{\mu}^*_{1:J,1:G}, \boldsymbol{\phi}^*_{1:J,1:G}, \boldsymbol{\beta}, \mathbf{Y}, \boldsymbol{t}, \boldsymbol{q}^J) \propto \prod_{g=1}^{G}\text{NB}(y_{c,g,d}|\mu^*_{j,g}\beta_{c,d}, \phi^*_{j,g}) \times q_{j,d}^J K(t_{c,d}|\boldsymbol{\psi}^*_{j,d}).$$

Let $\tilde{p}_{c,d,j}$ denote the term on the right-hand side. We have

$$\pi(z_{c,d} = j|\boldsymbol{\mu}^*_{1:J,1:G}, \boldsymbol{\phi}^*_{1:J,1:G}, \boldsymbol{\beta}, \mathbf{Y}, \boldsymbol{t}, \boldsymbol{q}^J) = \frac{\tilde{K}\tilde{p}_{c,d,j}}{\tilde{K}\sum_{l=1}^{J}\tilde{p}_{c,d,l}},$$

where we remove the most extreme probability to avoid numerical errors,

$$\log\left(\tilde{K}\right) = -\max_j \log\left(\tilde{p}_{c,d,j}\right).$$

In all, we sample $z_{c,d}$ from $\{1, \ldots, J\}$ according to $\pi(z_{c,d} = j|\boldsymbol{\mu}^*_{1:J,1:G}, \boldsymbol{\phi}^*_{1:J,1:G}, \boldsymbol{\beta}, \mathbf{Y}, \boldsymbol{t}, \boldsymbol{q}^J)$. This is repeated for every c and d.

A.7 Component Probabilities p_j^J

The full conditional distribution is

$$\pi(p_1^J,\ldots,p_J^J|\boldsymbol{q}^J,\alpha,\alpha_0) \propto \prod_{j=1}^{J}\prod_{d=1}^{D}\text{Gamma}(q_{j,d}^J|\alpha p_j^J,1)\times \text{Dir}\left(p_1^J,\ldots,p_J^J|\frac{\alpha_0}{J},\ldots,\frac{\alpha_0}{J}\right)$$
$$\propto \prod_{j=1}^{J}\prod_{d=1}^{D}\left[\frac{1}{\Gamma\left(\alpha p_j^J\right)}\left(q_{j,d}^J\right)^{\alpha p_j^J}\right]\times\prod_{j=1}^{J}\left(p_j^J\right)^{\frac{\alpha_0}{J}-1},$$

which has no closed-form, and hence AMH is applied (Section A.4.3). Since p_j^J sum to one, the following transformation is applied yielding $\boldsymbol{x}\in\mathbb{R}^{J-1}$:

$$x_j=\log\left(\frac{p_j^J}{p_J^J}\right),\quad j=1,\ldots,J-1.$$

The inverse transformation is given by

$$p_j^J=\frac{\exp(x_j)}{1+\sum_{j=1}^{J-1}\exp(x_j)},\quad j=1,\ldots,J-1,$$
$$p_J^J=1-\sum_{j=1}^{J-1}p_j^J=\frac{1}{1+\sum_{j=1}^{J-1}\exp(x_j)}.$$

The Jacobian matrix is

$$J_{\boldsymbol{x}}=\begin{pmatrix}\frac{dx_1}{dp_1} & \frac{dx_2}{dp_1} & \cdots & \frac{dx_{J-1}}{dp_1}\\ \vdots & \vdots & \ddots & \vdots\\ \frac{dx_1}{dp_{J-1}} & \frac{dx_2}{dp_{J-1}} & \cdots & \frac{dx_{J-1}}{dp_{J-1}}\end{pmatrix}$$
$$=\begin{pmatrix}\frac{1}{p_1}+\frac{1}{p_J} & \cdots & \frac{1}{p_J}\\ \vdots & \ddots & \vdots\\ \frac{1}{p_J} & \cdots & \frac{1}{p_{J-1}}+\frac{1}{p_J}\end{pmatrix}$$
$$=\begin{pmatrix}\frac{1}{p_J} & \cdots & \frac{1}{p_J}\\ \vdots & \ddots & \vdots\\ \frac{1}{p_J} & \cdots & \frac{1}{p_J}\end{pmatrix}+\begin{pmatrix}\frac{1}{p_1} & 0 & \cdots & 0\\ 0 & \frac{1}{p_2} & \vdots & 0\\ \vdots & \vdots & \ddots & 0\\ 0 & 0 & \cdots & \frac{1}{p_{J-1}}\end{pmatrix}$$
$$=B+A.$$

Because $\det(A+B)=\det(A)+\det(B)+Tr(A^{-1}B)\det(A)$, $\det(B)=0$ and $\det(A)=\prod_{j=1}^{J-1}\frac{1}{p_j}$, it follows that $\det(A+B)=\prod_{j=1}^{J-1}\frac{1}{p_j}+(1-p_J)\prod_{j=1}^{J}\frac{1}{p_j}=\prod_{j=1}^{J}\frac{1}{p_j}$. Therefore,

$$\log|J_{\boldsymbol{x}}|=\log\left[\prod_{j=1}^{J}\frac{1}{p_j}\right]=-\sum_{j=1}^{J}\log(p_j).$$

The log full conditional distribution is

$$\log \pi\left(\boldsymbol{p}^J|\ldots\right) = \sum_{j=1}^{J}\sum_{d=1}^{D}\left[\alpha p_j^J \log\left(q_{j,d}^J\right) - \log\Gamma(\alpha p_j^J)\right] + \sum_{j=1}^{J}\left[\left(\frac{\alpha_0}{J}-1\right)\log\left(p_j^J\right)\right] + \text{const.}$$

Combining all terms together, the acceptance probability is

$$\begin{aligned}
\alpha(\boldsymbol{p}_{new}^J, \boldsymbol{p}_{old}^J) &= \min\left(1, \frac{\pi(\boldsymbol{p}_{new}^J)Q_n(\boldsymbol{p}_{old}^J|\boldsymbol{p}_{new}^J)}{\pi(\boldsymbol{p}_{old}^J)Q_n(\boldsymbol{p}_{new}^J|\boldsymbol{p}_{old}^J)}\right) \\
&= \min\left(1, \frac{\pi(\boldsymbol{p}_{new}^J)|J_{\boldsymbol{x}_{old}}|}{\pi(\boldsymbol{p}_{old}^J)|J_{\boldsymbol{x}_{new}}|}\right) \\
&= \min\left(1, \exp\left[\log\pi(\boldsymbol{p}_{new}^J) - \log\pi(\boldsymbol{p}_{old}^J) + \sum_{j=1}^{J}\left(\log\left(p_{j,new}^J\right) - \log\left(p_{j,old}^J\right)\right)\right]\right).
\end{aligned}$$

We note that the sampling of a new transformed variable $\boldsymbol{x}_{new}$ is slightly different from step 2 in Section A.4.3. Following Algorithm 6 in Griffin and Stephens (2013), instead of a fixed scale parameter $s_{\text{d}} = 2.4^2/\text{d}$ ($\text{d} = J-1$ for the case of $\boldsymbol{p}^J$), s_{d} is also updated at each iteration. The idea is to adapt s_{d} to achieve a particular average acceptance probability $\bar{\alpha}$, e.g. 0.234, which has been shown to be optimal in the multivariate target distribution (Roberts et al., 1997).

We use an initial value $s_{\text{d}}^{(1)} = 0.001$. At the current iteration n, let $\boldsymbol{x}_{new}$ denote the new sample after the decision of rejection or not. Define

$$\omega^{(n)} = \exp\left(\log\left(s_{\text{d}}^{(n)}\right) + n^{-0.7} \times \left(\alpha(\boldsymbol{p}_{new}^J, \boldsymbol{p}_{old}^J) - \bar{\alpha}\right)\right),$$

then

$$s_{\text{d}}^{(n+1)} = \begin{cases} \omega^-, & \text{if } \omega^{(n)} < \omega^-, \\ \omega^{(n)}, & \text{if } \omega^{(n)} \in [\omega^-, \omega^+], \\ \omega^+, & \text{if } \omega^{(n)} > \omega^+, \end{cases}$$

where $\omega^- = \exp(-50)$ and $\omega^+ = \exp(50)$. The update of the covariance matrix follows from step 4 (multivariate case) in Section A.4.3.

A.8 Mean-dispersion Parameters $\boldsymbol{b}$ and α_ϕ^2

The joint distribution of $\boldsymbol{b} = (b_0, b_1)^T$ and α_ϕ^2 is

$$\begin{aligned}
\pi(\boldsymbol{b}, \alpha_\phi^2|\boldsymbol{\mu}_{1:J,1:G}^*, \boldsymbol{\phi}_{1:J,1:G}^*) \propto& \text{N}(\boldsymbol{b}|\boldsymbol{m}_b, \alpha_\phi^2\mathbf{I}) \times \text{IG}(\alpha_\phi^2|\nu_1, \nu_2) \times \prod_{j=1}^{J}\prod_{g=1}^{G}\text{log-N}(\phi_{j,g}^*|b_0 + b_1\log\left(\mu_{j,g}^*\right), \alpha_\phi^2) \\
\propto& \left(\alpha_\phi^2\right)^{-\left(\nu_1+2+\frac{JG}{2}\right)} \\
&\times \exp\Bigg(-\frac{1}{\alpha_\phi^2}\Bigg[\frac{1}{2}\sum_{j=1}^{J}\sum_{g=1}^{G}\left(\log\left(\phi_{j,g}^*\right) - b_0 - b_1\log\left(\mu_{j,g}^*\right)\right)^2 + \frac{1}{2}\boldsymbol{b}^T\boldsymbol{b} \\
&- \boldsymbol{b}^T\boldsymbol{m}_b + \frac{1}{2}\boldsymbol{m}_b^T\boldsymbol{m}_b + \nu_2\Bigg]\Bigg).
\end{aligned}$$

For $\boldsymbol{b}$, we have

$$\pi(\boldsymbol{b}|\boldsymbol{\mu}^*, \boldsymbol{\phi}^*, \alpha_\phi^2) \propto \exp\left(-\frac{1}{2\alpha_\phi^2}\left[\sum_{j=1}^{J}\sum_{g=1}^{G}(\log(\phi_{j,g}^*) - b_0 - b_1\log(\mu_{j,g}^*))^2 + \boldsymbol{b}^T\boldsymbol{b} - 2\boldsymbol{b}^T\boldsymbol{m}_b\right]\right).$$

We have

$$\sum_{j=1}^{J}\sum_{g=1}^{G}(\log(\phi_{j,g}^*) - b_0 - b_1\log(\mu_{j,g}^*))^2 = \sum_{j=1}^{J}(\log(\boldsymbol{\phi}_j^*) - \tilde{\boldsymbol{\mu}}_j\boldsymbol{b})^T(\log(\boldsymbol{\phi}_j^*) - \tilde{\boldsymbol{\mu}}_j\boldsymbol{b}),$$

where

$$\log(\boldsymbol{\phi}_j^*) = \begin{pmatrix}\log(\phi_{j,1}^*)\\ \vdots \\ \log(\phi_{j,G}^*)\end{pmatrix}, \quad \tilde{\boldsymbol{\mu}}_j = \begin{pmatrix}1 & \log(\mu_{j,1}^*)\\ \vdots & \vdots \\ 1 & \log(\mu_{j,G}^*)\end{pmatrix}, \quad \boldsymbol{b} = \begin{pmatrix}b_0\\ b_1\end{pmatrix}.$$

Therefore,

$$\begin{aligned}\pi(\boldsymbol{b}|\boldsymbol{\mu}^*, \boldsymbol{\phi}^*, \alpha_\phi^2) \propto \exp\Bigg(-\frac{1}{2\alpha_\phi^2}\Bigg[&\boldsymbol{b}^T\left(\sum_{j=1}^{J}\tilde{\boldsymbol{\mu}}_j^T\tilde{\boldsymbol{\mu}}_j + \mathbf{I}\right)\boldsymbol{b} - 2\boldsymbol{b}^T\left(\sum_{j=1}^{J}\tilde{\boldsymbol{\mu}}_j^T\log(\boldsymbol{\phi}_j^*) + \boldsymbol{m}_b\right)\\ &+ \sum_{j=1}^{J}\log(\boldsymbol{\phi}_j^*)^T\log(\boldsymbol{\phi}_j^*)\Bigg]\Bigg),\end{aligned}$$

i.e.

$$\boldsymbol{b}|\ldots \sim \mathrm{N}(\tilde{\boldsymbol{m}}_b, \alpha_\phi^2\tilde{\mathbf{V}}_b),$$

where

$$\begin{aligned}\tilde{\boldsymbol{m}}_b &= \left(\sum_{j=1}^{J}\tilde{\boldsymbol{\mu}}_j^T\tilde{\boldsymbol{\mu}}_j + \mathbf{I}\right)^{-1}\left(\sum_{j=1}^{J}\tilde{\boldsymbol{\mu}}_j^T\log(\boldsymbol{\phi}_j^*) + \boldsymbol{m}_b\right),\\ \tilde{\mathbf{V}}_b &= \left(\sum_{j=1}^{J}\tilde{\boldsymbol{\mu}}_j^T\tilde{\boldsymbol{\mu}}_j + \mathbf{I}\right)^{-1}.\end{aligned}$$

As for α_ϕ^2, the full condition is

$$\begin{aligned}\pi(\alpha_\phi^2|\boldsymbol{\mu}^*, \boldsymbol{\phi}^*) &= \int \pi(\boldsymbol{b}, \alpha_\phi^2|\boldsymbol{\mu}^*, \boldsymbol{\phi}^*)\, d\boldsymbol{b}\\ &\propto \int \left(\frac{1}{\alpha_\phi^2}\right)^{\nu_1+1}\exp\left(-\frac{\nu_2}{\alpha_\phi^2}\right)\left(\frac{1}{\alpha_\phi^2}\right)^{JG/2}\left(\frac{1}{\alpha_\phi^2}\right)\\ &\quad\times\exp\Bigg(-\frac{1}{2\alpha_\phi^2}\Bigg[(\boldsymbol{b}-\tilde{\boldsymbol{m}}_b)^T\tilde{\mathbf{V}}_b^{-1}(\boldsymbol{b}-\tilde{\boldsymbol{m}}_b) - \tilde{\boldsymbol{m}}_b^T\tilde{\mathbf{V}}_b^{-1}\tilde{\boldsymbol{m}}_b\\ &\quad + \sum_{j=1}^{J}\log(\boldsymbol{\phi}_j^*)^T\log(\boldsymbol{\phi}_j^*) + \boldsymbol{m}_b^T\boldsymbol{m}_b\Bigg]\Bigg)\, d\boldsymbol{b}.\end{aligned}$$

Conditional on $\boldsymbol{\mu}^*, \boldsymbol{\phi}^*$,

$$\int \frac{1}{\alpha_\phi^2} \exp\left(-\frac{1}{2\alpha_\phi^2}(\boldsymbol{b}-\tilde{\boldsymbol{m}}_b)^T\tilde{\mathbf{V}}_b^{-1}(\boldsymbol{b}-\tilde{\boldsymbol{m}}_b)\right) d\boldsymbol{b} = \text{const.},$$

it follows that

$$\begin{aligned}
\pi(\alpha_\phi^2|\boldsymbol{\mu}^*,\boldsymbol{\phi}^*) &\propto \left(\frac{1}{\alpha_\phi^2}\right)^{\nu_1+1} \exp\left(-\frac{\nu_2}{\alpha_\phi^2}\right)\left(\frac{1}{\alpha_\phi^2}\right)^{JG/2} \\
&\quad\times \exp\left(-\frac{1}{2\alpha_\phi^2}\left[-\tilde{\boldsymbol{m}}_b^T\tilde{\mathbf{V}}_b^{-1}\tilde{\boldsymbol{m}}_b + \sum_{j=1}^J \log\left(\boldsymbol{\phi}_j^*\right)^T\log\left(\boldsymbol{\phi}_j^*\right) + \boldsymbol{m}_b^T\boldsymbol{m}_b\right]\right) \\
&\propto \left(\frac{1}{\alpha_\phi^2}\right)^{\nu_1+1+JG/2} \\
&\quad\times \exp\left(-\frac{1}{\alpha_\phi^2}\left[\nu_2 + \frac{1}{2}\left(\sum_{j=1}^J \log\left(\boldsymbol{\phi}_j^*\right)^T\log\left(\boldsymbol{\phi}_j^*\right) - \tilde{\boldsymbol{m}}_b^T\tilde{\mathbf{V}}_b^{-1}\tilde{\boldsymbol{m}}_b + \boldsymbol{m}_b^T\boldsymbol{m}_b\right)\right]\right).
\end{aligned}$$

Therefore,

$$\alpha_\phi^2|\boldsymbol{\mu}^*,\boldsymbol{\phi}^* \sim \text{IG}(\tilde{\nu}_1,\tilde{\nu}_2),$$

where

$$\tilde{\nu}_1 = \nu_1 + JG/2, \quad \tilde{\nu}_2 = \nu_2 + \frac{1}{2}\left(\sum_{j=1}^J \log\left(\boldsymbol{\phi}_j^*\right)^T\log\left(\boldsymbol{\phi}_j^*\right) - \tilde{\boldsymbol{m}}_b^T\tilde{\mathbf{V}}_b^{-1}\tilde{\boldsymbol{m}}_b + \boldsymbol{m}_b^T\boldsymbol{m}_b\right).$$

A.9 Component-specific Parameters $\mu_{j,g}^*$ and $\phi_{j,g}^*$

The full condition distribution is

$$\begin{aligned}
\pi(\mu_{j,g}^*,\phi_{j,g}^*|\mathbf{Z},\boldsymbol{b},\alpha_\phi^2,\mathbf{Y},\boldsymbol{\beta}) \propto& \text{log-N}\left(\mu_{j,g}^*|0,\alpha_\mu^2\right)\times\text{log-N}\left(\phi_{j,g}^*|b_0+b_1\log\left(\mu_{j,g}^*\right),\alpha_\phi^2\right) \\
&\times \prod_{(c,d):z_{c,d}=j}\text{NB}(y_{c,d,g}|\mu_{j,g}^*\beta_{c,d},\phi_{j,g}^*) \\
\propto& \left(\frac{1}{\mu_{j,g}^*\phi_{j,g}^*}\right)\exp\left(-\frac{1}{2\alpha_\mu^2}(\log\mu_{j,g}^*)^2 - \frac{1}{2\alpha_\phi^2}(\log\phi_{j,g}^* - (b_0+b_1\log\mu_{j,g}^*))^2\right) \\
&\times \prod_{(c,d):z_{c,d}=j}\binom{y_{c,g,d}+\phi_{j,g}^*-1}{\phi_{j,g}^*-1}\left(\frac{\phi_{j,g}^*}{\mu_{j,g}^*\beta_{c,d}+\phi_{j,g}^*}\right)^{\phi_{j,g}^*}\left(\frac{\mu_{j,g}^*}{\mu_{j,g}^*\beta_{c,d}+\phi_{j,g}^*}\right)^{y_{c,g,d}}.
\end{aligned}$$

This is not a standard distribution and hence we will apply AMH to sample for $\left(\mu_{j,g}^*,\phi_{j,g}^*\right)$ (Section A.4.3). For clarity, we will drop the subscript j and g here. We apply the following transformation

$$\boldsymbol{x} = (x_1,x_2) = \left(\log\left(\mu^*\right),\log\left(\phi^*\right)\right) \in \mathbb{R}^2,$$

with inverse transformation

$$\mu^* = \exp(x_1), \quad \phi^* = \exp(x_2).$$

The Jacobian term is

$$J_{\boldsymbol{x}} = \begin{pmatrix} \frac{dx_1}{d\mu^*} & \frac{dx_1}{d\phi^*} \\ \frac{dx_2}{d\mu^*} & \frac{dx_2}{d\phi^*} \end{pmatrix} = \begin{pmatrix} \frac{1}{\mu^*} & 0 \\ 0 & \frac{1}{\phi^*} \end{pmatrix},$$

giving $|J_{\boldsymbol{x}}| = 1/(\mu^*\phi^*)$.

The logarithm of the full conditional distribution is

$$\begin{aligned} \log \pi(\mu^*_{j,g}, \phi^*_{j,g}|\ldots) = &- \log\big(\mu^*_{j,g}\phi^*_{j,g}\big) - \frac{1}{2\alpha_\mu^2}(\log \mu^*_{j,g})^2 - \frac{1}{2\alpha_\phi^2}(\log \phi^*_{j,g} - (b_0 + b_1 \log \mu^*_{j,g}))^2 \\ &+ \sum_{(c,d):z_{c,d}=j} \log \binom{y_{c,g,d} + \phi^*_{j,g} - 1}{\phi^*_{j,g} - 1} + \phi^*_{j,g} \log\left(\frac{\phi^*_{j,g}}{\mu^*_{j,g}\beta_{c,d} + \phi^*_{j,g}}\right) \\ &+ y_{c,g,d} \log\left(\frac{\mu^*_{j,g}}{\mu^*_{j,g}\beta_{c,d} + \phi^*_{j,g}}\right) + \text{const.} \end{aligned}$$

Combining all terms together, the acceptance probability is

$$\begin{aligned} \alpha\left((\mu^*, \phi^*)_{new}, (\mu^*, \phi^*)_{old}\right) = \min\Bigg(1, \exp\Bigg[&\log \pi((\mu^*, \phi^*)_{new}) - \log \pi((\mu^*, \phi^*)_{old}) \\ &- \log(\mu^*_{old}) - \log(\phi^*_{old}) + \log(\mu^*_{new}) + \log(\phi^*_{new})\Bigg]\Bigg). \end{aligned}$$

Then the covariance matrix is updated following the multivariate case (d = 2) in step 4 of Section A.4.3.

Note that due to label switching, the covariance matrix may have very large values. Therefore, to mitigate the multiplicative effect of the scale parameter s_{d} on the covariance, we fix $s_{\mathrm{d}} = 1$ instead of $s_{\mathrm{d}} = 2.4^2/2$. In addition, the adaptive Metropolis-Hastings algorithm above is applied for all occupied components. For empty components, new samples are drawn from the prior directly and transformed to $\boldsymbol{x}$ to update the covariance matrix.

The above step is repeated for every j and g.

A.10 Capture Efficiencies $\beta_{c,d}$

The full conditional distribution is

$$\begin{aligned} \pi(\beta_{c,d}|\left\{y_{c,g,d}\right\}_{g=1}^{G}, z_{c,d} = j, \boldsymbol{\mu}^*_{1:J,1:G}, \boldsymbol{\phi}^*_{1:J,1:G}) \propto &\text{Beta}(\beta_{c,d}|a_d^\beta, b_d^\beta) \times \prod_{g=1}^{G} \text{NB}(y_{c,g,d}|\mu^*_{j,g}\beta_{c,d}, \phi^*_{j,g}) \\ \propto &(\beta_{c,d})^{a_d^\beta - 1}(1 - \beta_{c,d})^{b_d^\beta - 1} \\ &\times \left[\prod_{g=1}^{G} \left(\frac{1}{\phi^*_{j,g} + \mu^*_{j,g}\beta_{c,d}}\right)^{\phi^*_{j,g} + y_{c,g,d}} (\beta_{c,d})^{y_{c,g,d}}\right]. \end{aligned}$$

This does not have a closed form and we will apply AMH (Section A.4.3) with the following variable transformation

$$x = \log\left(\frac{\beta_{c,d}}{1-\beta_{c,d}}\right) \in \mathbb{R},$$

with Jacobian equal to

$$J_x = \frac{dx}{d\beta_{c,d}} = \frac{d}{d\beta_{c,d}}(\log(\beta_{c,d}) - \log(1-\beta_{c,d})) = \frac{1}{\beta_{c,d}(1-\beta_{c,d})}.$$

The inverse transformation is given by

$$\beta_{c,d} = \frac{1}{1+\exp(-x)}.$$

Next, the logarithm of the full conditional distribution is

$$\begin{aligned}\log \pi(\beta_{c,d}|\ldots) =& (a_d^\beta - 1)\log(\beta_{c,d}) + (b_d^\beta - 1)\log(1-\beta_{c,d}) \\ &- \sum_{g=1}^{G}\left[(\phi_{j,g}^* + y_{c,g,d})\log\left(\phi_{j,g}^* + \mu_{j,g}^*\beta_{c,d}\right) - y_{c,g,d}\log(\beta_{c,d})\right] + \text{const.}\end{aligned}$$

Therefore, the acceptance probability is given by

$$\begin{aligned}\alpha(\beta_{new}, \beta_{old}) =& \min\Bigg(1, \exp\Bigg[\log\pi(\beta_{new}) - \log\pi(\beta_{old}) \\ &+ \log(\beta_{new}) + \log(1-\beta_{new}) - \log(\beta_{old}) - \log(1-\beta_{old})\Bigg]\Bigg),\end{aligned}$$

and we update the variance of the transformed variable x following step 4 of Section A.4.3.

This step is repeated for every c and d.

A.11 Hyperparameters r_j, s^2, h_j and m^2

This section details the sampling steps for hyperparameters related to kernel parameters.

A.11.1 Prior Means r_j

For each j, we have

$$\begin{aligned}\pi(r_j|\left\{t_{j,d}^*\right\}_{d=1}^D, \mu_r, \sigma_r^2, s^2) &\propto \prod_{d=1}^{D} \mathrm{N}(t_{j,d}^*|r_j, s^2) \times \mathrm{N}(r_j|\mu_r, \sigma_r^2) \\ &\propto \exp\left[-\frac{1}{2s^2}\sum_{d=1}^{D}\left(r_j - t_{j,d}^*\right)^2\right] \times \exp\left[-\frac{1}{2\sigma_r^2}\left(r_j - \mu_r\right)^2\right].\end{aligned}$$

Recall our calculation for $t_{j,d}^*$ in Section A.4.2. It can be noticed that the full conditional distribution for r_j is a normal distribution

$$r_j|\ldots \sim \mathrm{N}(\hat{\mu}_r, \hat{\sigma}_r^2),$$

where

$$\hat{\sigma}_r^2 = \left(\frac{1}{\sigma_r^2} + \frac{D}{s^2}\right)^{-1}, \quad \hat{\mu}_r = \frac{\mu_r/\sigma_r^2 + \sum_{d=1}^{D} t_{j,d}^*/s^2}{1/\sigma_r^2 + D/s^2}.$$

A.11.2 Prior Variance s^2

$$\begin{aligned}\pi(s^2|\left\{t^*_{j,d}\right\}_{j=1,d=1}^{J,D},\eta_1,\eta_2,\boldsymbol{r}) &\propto \prod_{j=1}^{J}\prod_{d=1}^{D}\mathrm{N}(t^*_{j,d}|r_j,s^2)\times \mathrm{IG}(s^2|\eta_1,\eta_2)\\ &\propto (s^2)^{-\frac{JD}{2}}\exp\left[-\frac{1}{s^2}\times\frac{1}{2}\sum_{j=1}^{J}\sum_{d=1}^{D}\left(t^*_{j,d}-r_j\right)^2\right]\\ &\quad\times(s^2)^{-\eta_1-1}\exp\left[-\frac{\eta_2}{s^2}\right],\end{aligned}$$

i.e.

$$s^2|\ldots\sim \mathrm{IG}\left(\frac{JD}{2}+\eta_1,\eta_2+\frac{1}{2}\sum_{j=1}^{J}\sum_{d=1}^{D}\left(t^*_{j,d}-r_j\right)^2\right).$$

A.11.3 Prior Means h_j

For each j,

$$\begin{aligned}\pi(h_j|\left\{{\sigma^*_{j,d}}^2\right\}_{d=1}^{D},\mu_h,\sigma_h^2,m^2) &\propto \prod_{d=1}^{D}\text{log-N}({\sigma^*_{j,d}}^2|h_j,m^2)\times\mathrm{N}(h_j|\mu_h,\sigma_h^2)\\ &\propto \exp\left[-\frac{1}{2m^2}\sum_{d=1}^{D}\left(h_j-\log\left({\sigma^*_{j,d}}^2\right)\right)^2\right]\times\exp\left[-\frac{1}{2\sigma_h^2}\left(h_j-\mu_h\right)^2\right].\end{aligned}$$

Similar to r_j, the full conditional is a normal distribution

$$h_j|\ldots\sim\mathrm{N}(\hat{\mu}_h,\hat{\sigma}_h^2),$$

where

$$\hat{\sigma}_h^2=\left(\frac{1}{\sigma_h^2}+\frac{D}{m^2}\right)^{-1},\quad \hat{\mu}_r=\frac{\mu_h/\sigma_h^2+\sum_{d=1}^{D}\log\left({\sigma^*_{j,d}}^2\right)/m^2}{1/\sigma_h^2+D/m^2}.$$

A.11.4 Prior Variance m^2

$$\begin{aligned}\pi(m^2|\left\{{\sigma^*_{j,d}}^2\right\}_{j=1,d=1}^{J,D},\kappa_1,\kappa_2,\boldsymbol{h}) &\propto \prod_{j=1}^{J}\prod_{d=1}^{D}\text{log-N}({\sigma^*_{j,d}}^2|h_j,m^2)\times\mathrm{IG}(m^2|\kappa_1,\kappa_2)\\ &\propto (m^2)^{-\frac{JD}{2}}\exp\left[-\frac{1}{m^2}\times\frac{1}{2}\sum_{j=1}^{J}\sum_{d=1}^{D}\left(\log\left({\sigma^*_{j,d}}^2\right)-h_j\right)^2\right]\\ &\quad\times(m^2)^{-\kappa_1-1}\exp\left[-\frac{\kappa_2}{m^2}\right],\end{aligned}$$

i.e.

$$m^2|\ldots\sim\mathrm{IG}\left(\frac{JD}{2}+\kappa_1,\kappa_2+\frac{1}{2}\sum_{j=1}^{J}\sum_{d=1}^{D}\left(\log\left({\sigma^*_{j,d}}^2\right)-h_j\right)^2\right).$$

A.12 bayNorm Estimates of Capture Efficiencies, Mean and Dispersion Parameters

In the bayNorm approach, capture efficiencies for cells from group d are estimated by

$$\hat{\beta}_{c,d}^{\text{bay}} = \frac{\sum_{g=1}^{G} y_{c,g,d}}{\frac{1}{C_d}\sum_{c=1}^{C_d}\sum_{g=1}^{G} y_{c,g,d}} \times \lambda,$$

where λ is the an estimate of global mean capture efficiency across cells. Under the default setting of bayNorm, $\lambda = 0.06$. The estimates are used to construct empirical priors for the capture efficiencies in our C-HDP model.

As for mean and dispersion parameters, Tang et al. (2020) propose two approaches to estimate them. The first one is based on maximizing the marginal likelihood for each gene, assuming independence between cells. The log-marginal likelihood for gene g is

$$\log M_g = \sum_{d=1}^{D}\sum_{c=1}^{C_d} \log \text{NB}(y_{c,g,d}|\mu_g\beta_{c,g}, \phi_g).$$

Alternatively, μ_g and ϕ_g can be estimated through the method of moments estimation (MME) by matching the empirical and theoretical moments of the normalized count $y_{c,g,d}^s = y_{c,g,d}/\beta_{c,d}$. In addition, two approaches can be combined to produce more robust and efficient estimation (Tang et al., 2020), where μ_g is obtained through MME, and ϕ_g from MME is adjusted by a factor obtained from fitting a linear regression between ϕ_g from MME and ϕ_g from maximizing the marginal likelihood.

Appendix B. Posterior Inference for Calcium Imaging Data

We use a vector autoregression model with lag one to account for the temporal nature of the data:

$$\boldsymbol{y}_{i,d}|\boldsymbol{y}_{i-1,d}, z_{i,d} = j, \boldsymbol{a}_j^*, \mathbf{B}_j^*, \boldsymbol{\Sigma}_j^* \sim \text{N}(\boldsymbol{a}_j^* + \mathbf{B}_j^* \boldsymbol{y}_{i-1,d}, \boldsymbol{\Sigma}_j^*),$$

where $\boldsymbol{y}_{i,d} = (y_{i,1,d}, \ldots, y_{i,G,d})^T \in \mathbb{R}^G$, $\boldsymbol{a}_j^*$ denotes the intercept, $\mathbf{B}_j^*$ is a matrix of coefficients in VAR, and $\boldsymbol{\Sigma}_j^*$ is the covariance matrix. Define $\mathbf{L}_j^* = (\boldsymbol{a}_j^* \quad \mathbf{B}_j^*)^T$ and $\boldsymbol{x}_{i,d} = (1, y_{i-1,1,d}, \ldots, y_{i-1,G,d})^T$.

B.1 Prior Specification

Below we specify the priors for all parameters.

Base Measure. The component-specific parameters $\mathbf{L}_j^*$ and $\boldsymbol{\Sigma}_j^*$ are given conjugate priors:

$$\mathbf{L}_j^*|\boldsymbol{\Sigma}_j^* \overset{ind}{\sim} \text{MN}(\mathbf{L}_0, \mathbf{V}_0, \boldsymbol{\Sigma}_j^*), \quad \boldsymbol{\Sigma}_j^* \overset{i.i.d}{\sim} \text{IW}(\omega_0, \boldsymbol{\Phi}_0).$$

For hyperparameters, we obtain empirical estimates for the coefficient matrix and covariance matrix, denoted by $\hat{\mathbf{L}}_0$ and $\hat{\boldsymbol{\Phi}}_0$, respectively, by regressing $\boldsymbol{y}_{i,d}$ on $\boldsymbol{y}_{i-1,d}$ across all groups. Then we set $\mathbf{L}_0 = \hat{\mathbf{L}}_0$ and $\boldsymbol{\Phi}_0 = \hat{\boldsymbol{\Phi}}_0 / J_0^{2/G}$ (Fraley and Raftery, 2007), where J_0 is a guess of the number of clusters. In addition, we use $\mathbf{V}_0 = 100\mathbf{I}$ and $\omega_0 = G + 2$.

Kernel Parameters. For a periodic kernel with parameters $\boldsymbol{\psi}_{j,d}^* = (\mu_{j,d}^*, {\sigma_{j,d}^*}^2, \lambda_{j,d}^*)$, the following hierarchical priors are used:

$$\mu_{j,d}^* \overset{ind}{\sim} \text{Unif}\left(-\frac{\pi\lambda_{j,d}^*}{2}, \frac{\pi\lambda_{j,d}^*}{2}\right), \quad \lambda_{j,d}^* \overset{ind}{\sim} \text{log-N}(r_j, s^2), \quad r_j \overset{i.i.d}{\sim} \text{N}(\mu_r, \sigma_r^2), \quad s^2 \sim \text{IG}(\eta_1, \eta_2),$$

$${\sigma_{j,d}^*}^2 \overset{ind}{\sim} \text{IG}(a_j, b_j), \quad a_j = 2 + \frac{h_j^2}{m^2}, \quad b_j = h_j^2 + \frac{h_j^3}{m^2}, \quad h_j \overset{i.i.d}{\sim} \text{log-N}(\mu_h, \sigma_h^2),, \quad m^2 \sim \text{IG}(\kappa_1, \kappa_2).$$

Here shape a_j and scale b_j are modelled as functions of the mean h_j and variance m^2 of the inverse-gamma prior. Note that $\mu_{j,d}^*$ is restricted within one period $(\pi\lambda_{j,d}^*)$ for identifiability. For calcium imaging data, we set $\mu_r = -2, \sigma_r = 0.5, \eta_1 = 5, \eta_2 = 1, \mu_h = -1, \sigma_h = 0.5, \kappa_1 = 26, \kappa_2 = 1$.

For concentration parameters, the priors are the same as those for Pax6 data (Section A). With a finite-dimensional truncation at J, the complete model is

$$
\begin{aligned}
\boldsymbol{y}_{i,d}|z_{i,d}=j,\boldsymbol{y}_{i-1,d},\mathbf{L}_j^*,\boldsymbol{\Sigma}_j^* &\overset{ind}{\sim} \mathrm{N}((\mathbf{L}_j^*)^T\boldsymbol{x}_{i,d},\boldsymbol{\Sigma}_j^*),\\
z_{i,d}|p_{1,d}^J(t_{i,d}),\ldots,p_{J,d}^J(t_{i,d}) &\overset{ind}{\sim} \mathrm{Cat}(p_{1,d}^J(t_{i,d}),\ldots,p_{J,d}^J(t_{i,d})),\\
p_{j,d}^J(t_{i,d}) &= \frac{q_{j,d}^J K(t_{i,d}|\boldsymbol{\psi}_{j,d}^*)}{\sum_{k=1}^J q_{k,d}K(t_{i,d}|\boldsymbol{\psi}_{k,d}^*)},\\
q_{j,d}^J &\overset{ind}{\sim} \mathrm{Gamma}(\alpha p_j^J,1),\\
p_1^J,\ldots,p_J^J &\sim \mathrm{Dir}\left(\frac{\alpha_0}{J},\ldots,\frac{\alpha_0}{J}\right),\\
\mu_{j,d}^*|\lambda_{j,d}^* &\overset{ind}{\sim} \mathrm{Unif}\left(-\frac{\pi\lambda_{j,d}^*}{2},\frac{\pi\lambda_{j,d}^*}{2}\right),\\
\lambda_{j,d}^* &\overset{ind}{\sim} \text{log-N}(r_j,s^2),\\
r_j &\overset{i.i.d}{\sim} \mathrm{N}(\mu_r,\sigma_r^2),\\
s^2 &\sim \mathrm{IG}(\eta_1,\eta_2),\\
{\sigma_{j,d}^*}^2 &\overset{ind}{\sim} \mathrm{IG}(a_j,b_j),\\
a_j = 2+\frac{h_j^2}{m^2},\quad b_j &= h_j^2+\frac{h_j^3}{m^2},\\
h_j &\overset{i.i.d}{\sim} \text{log-N}(\mu_h,\sigma_h^2),\\
m^2 &\sim \mathrm{IG}(\kappa_1,\kappa_2),\\
\mathbf{L}_j^*|\boldsymbol{\Sigma}_j^* &\overset{ind}{\sim} \mathrm{MN}(\mathbf{L}_0,\mathbf{V}_0,\boldsymbol{\Sigma}_j^*),\\
\boldsymbol{\Sigma}_j^* &\sim \mathrm{IW}(\omega_0,\boldsymbol{\Phi}_0),\\
\alpha &\sim \mathrm{Gamma}(1,1),\\
\alpha_0 &\sim \mathrm{Gamma}(1,1).
\end{aligned}
$$

Define $\mathbf{Z} = \{z_{i,d}\}_{i=1,d=1}^{n_d,D}, \mathbf{Y} = \{\boldsymbol{y}_{i,d}\}_{i=1,d=1}^{n_d,D}, \boldsymbol{t} = \{t_{i,d}\}_{i=1,d=1}^{n_d,D}, \boldsymbol{q}^J = \left\{q_{j,d}^J\right\}_{j=1,d=1}^{J,D}, \boldsymbol{p}^J = (p_1^J, \ldots, p_J^J), \mathbf{L}^* = (\mathbf{L}_1^*, \ldots, \mathbf{L}_J^*), \boldsymbol{\Sigma}^* = (\boldsymbol{\Sigma}_1^*, \ldots, \boldsymbol{\Sigma}_J^*), \boldsymbol{\mu}^* = \left\{\mu_{j,d}^*\right\}_{j=1,d=1}^{J,D}, \boldsymbol{\lambda}^* = \left\{\lambda_{j,d}^*\right\}_{j=1,d=1}^{J,D}, \boldsymbol{\sigma}^{*2} = \left\{{\sigma_{j,d}^*}^2\right\}_{j=1,d=1}^{J,D}, \boldsymbol{r} = (r_1, \ldots, r_J), \boldsymbol{h} = (h_1, \ldots, h_J)$. The posterior distribution is

$$
\begin{aligned}
&\pi(\mathbf{Z}, \boldsymbol{q}^J, \boldsymbol{p}^J, \mathbf{L}^*, \boldsymbol{\Sigma}^*, \alpha, \alpha_0, \boldsymbol{\mu}^*, \boldsymbol{\lambda}^*, \boldsymbol{\sigma}^{*2}, \boldsymbol{r}, s^2, \boldsymbol{h}, m^2 | \mathbf{Y}, \boldsymbol{t}) \\
\propto &\prod_{j=1}^{J} \prod_{(i,d):z_{i,d}=j} \mathrm{N}(\boldsymbol{y}_{i,d} | (\mathbf{L}_j^*)^T \boldsymbol{x}_{i,d}, \boldsymbol{\Sigma}_j^*) \\
&\times \prod_{j=1}^{J} \prod_{d=1}^{D} \prod_{i:z_{i,d}=j} K(t_{i,d} | \boldsymbol{\psi}_{j,d}^*) \times \prod_{j=1}^{J} \prod_{d=1}^{D} \left(q_{j,d}^J\right)^{N_{j,d}} \\
&\times \prod_{j=1}^{J} \prod_{d=1}^{D} \prod_{i=1}^{n_d} \exp\left(-\xi_{i,d} q_{j,d}^J K(t_{i,d} | \boldsymbol{\psi}_{j,d}^*)\right) \\
&\times \prod_{j=1}^{J} \prod_{d=1}^{D} \mathrm{Gamma}(q_{j,d}^J | \alpha p_j^J, 1) \times \mathrm{Dir}\left(\boldsymbol{p}^J | \frac{\alpha_0}{J}, \ldots, \frac{\alpha_0}{J}\right) \\
&\times \prod_{j=1}^{J} \left[\mathrm{MN}(\mathbf{L}_j^* | \mathbf{L}_0, \mathbf{V}_0, \boldsymbol{\Sigma}_j^*) \times \mathrm{IW}(\boldsymbol{\Sigma}_j^* | \omega_0, \boldsymbol{\Phi}_0)\right] \\
&\times \mathrm{Gamma}(\alpha | 1, 1) \times \mathrm{Gamma}(\alpha_0 | 1, 1) \\
&\times \prod_{j=1}^{J} \prod_{d=1}^{D} \left[\text{log-N}(\lambda_{j,d}^* | r_j, s^2) \times \mathrm{Unif}\left(\mu_{j,d}^* | -\frac{\pi \lambda_{j,d}^*}{2}, \frac{\pi \lambda_{j,d}^*}{2}\right) \times \mathrm{IG}\left({\sigma_{j,d}^*}^2 | a_j, b_j\right)\right] \\
&\times \prod_{j=1}^{J} \left[\mathrm{N}(r_j | \mu_r, \sigma_r^2) \times \text{log-N}(h_j | \mu_h, \sigma_h^2)\right] \times \mathrm{IG}(s^2 | \eta_1, \eta_2) \times \mathrm{IG}(m^2 | \kappa_1, \kappa_2),
\end{aligned}
$$

where $N_{j,d} = \sum_{i=1}^{n_d} \mathbb{I}(z_{i,d} = j)$ is the number of observations in component j in the d-th experiment. The Gibbs sampling steps for updating the allocation variables $\mathbf{Z}$ and parameters $\boldsymbol{q}^J$ are similar to those in the clustering model for Pax6 data, except that the likelihood and the kernel have changed. As for concentration parameters α, α_0 and component probabilities $\boldsymbol{p}^J$, the steps are the same as illustrated in Section A. Therefore, below we only illustrate the sampling step for component-specific parameters, kernel parameters and hyperparameters.

B.2 Kernel Parameters

The details of sampling kernel parameters are as follows.

B.2.1 Location $\mu^*_{j,d}$

For each j and d, the full conditional distribution for $\mu^*_{j,d}$ is

$$\begin{aligned}&\pi(\mu^*_{j,d}|\left\{z_{i,d}\right\}_{i=1}^{n_d}, \left\{\xi_{i,d}\right\}_{i=1}^{n_d}, \left\{t_{i,d}\right\}_{i=1}^{n_d}, q^J_{j,d}, {\sigma^*_{j,d}}^2, \lambda^*_{j,d})\\ &\propto \prod_{i:z_{i,d}=j} K(t_{i,d}|\psi^*_{j,d}) \times \prod_{i=1}^{n_d} \exp\left(-\xi_{i,d} q^J_{j,d} K(t_{i,d}|\psi^*_{j,d})\right) \times \mathbb{I}\left(\mu_{j,d*} \in \left(-\frac{\pi\lambda^*_{j,d}}{2}, \frac{\pi\lambda^*_{j,d}}{2}\right)\right).\end{aligned}$$

The distribution does not have a closed form since $\mu^*_{j,d}$ is inside $\sin(\cdot)$ function. The AMH scheme described in Algorithm 5 from Griffin and Stephens (2013) is applied to achieve a targeted average acceptance probability for univariate parameters.

Adaptive Metropolis-Hastings for $\mu^*_{j,d}$

1. Apply the following transformation to μ^*, dropping subscript for simplicity:

$$x = g(\mu^*) = \log\left(\frac{\mu^* - \mu^-}{\mu^+ - \mu^*}\right) \in \mathbb{R},$$

   where $\mu^- = -\pi\lambda^*_{j,d}/2$ and $\mu^+ = \pi\lambda^*_{j,d}/2$ are the lower bound and upper bound. The Jacobian term is

$$J_x = \frac{dx}{d\mu^*} = \frac{\mu^+ - \mu^-}{(\mu^* - \mu^-)(\mu^+ - \mu^*)}.$$

   The inverse transformation is

$$\mu^* = \mu^+ + \frac{\mu^- - \mu^+}{1 + \exp(x)}.$$

2. At iteration n, letting $x_{old} = g(\mu^*_{old})$, we propose $x_{new} \sim N(x_{old}, \zeta^n)$ where $\zeta^{(n)}$ is the adaptive variance with an initial value $\zeta^{(1)} = 0.01$, which will be updated at each iteration (see step 4 below). Then μ^*_{new} is obtained through the inverse transformation.

3. The logarithm of the full conditional distribution is

$$\begin{aligned}\log\pi(\mu^*_{j,d}|\ldots) = &-\frac{2}{{\sigma^*_{j,d}}^2} \times \sum_{I_{j,d}} \sin^2\left(\frac{t_{i,d} - \mu^*_{j,d}}{\lambda^*_{j,d}}\right)\\ &- \sum_{i=1}^{n_d} \xi_{i,d} q^J_{j,d} \exp\left(-\frac{2}{{\sigma^*_{j,d}}^2} \times \sin^2\left(\frac{t_{i,d} - \mu^*_{j,d}}{\lambda^*_{j,d}}\right)\right) + \text{const.},\end{aligned}$$

   where $I_{j,d} = \{i : z_{i,d} = j\}$. Let Q_n denote the proposal distribution at step n. The acceptance probability of this proposal is given by

$$\begin{aligned}\alpha(\mu^*_{new}, \mu^*_{old}) &= \min\left(1, \frac{\pi(\mu^*_{new}) Q_n(\mu^*_{old}|\mu^*_{new})}{\pi(\mu^*_{old}) Q_n(\mu^*_{new}|\mu^*_{old})}\right)\\ &= \min\left(1, \frac{\pi(\mu^*_{new})(\mu^*_{new} - \mu^-)(\mu^+ - \mu^*_{new})}{\pi(\mu^*_{old})(\mu^*_{old} - \mu^-)(\mu^+ - \mu^*_{old})}\right)\\ &= \min\Bigg(1, \exp\Big[\log\pi(\mu^*_{new}) - \log\pi(\mu^*_{old}) + \log\big(\mu^*_{new} - \mu^-\big) + \log\big(\mu^+ - \mu^*_{new}\big)\\ &\quad - \log\big(\mu^*_{old} - \mu^-\big) - \log\big(\mu^+ - \mu^*_{old}\big)\Big]\Bigg).\end{aligned}$$

4. After making the decision to accept the proposed value or not, we now update the adaptive variance. Define

$$\omega^{(n)} = \exp\left(\log\left(\zeta^{(n)}\right) + n^{-0.7} \times (\alpha(\mu^*_{new}, \mu^*_{old}) - \bar{\alpha})\right),$$

where $\bar{\alpha}$ is desired average acceptance probability (0.234 or 0.44). The updated variance is

$$\zeta^{(n+1)} = \begin{cases} \omega^-, & \text{if } \omega^{(n)} < \omega^-, \\ \omega^{(n)}, & \text{if } \omega^{(n)} \in [\omega^-, \omega^+], \\ \omega^+, & \text{if } \omega^{(n)} > \omega^+, \end{cases}$$

where $\omega^- = \exp(-50)$ and $\omega^+ = \exp(50)$.

B.2.2 Period $\lambda^*_{j,d}$

As for $\lambda^*_{j,d}$, the full conditional distribution is

$$\begin{aligned}
&\pi(\lambda^*_{j,d}|r_j, s^2, \{z_{i,d}\}_{i=1}^{n_d}, \{\xi_{i,d}\}_{i=1}^{n_d}, \{t_{i,d}\}_{i=1}^{n_d}, q^J_{j,d}, {\sigma^*_{j,d}}^2, \mu^*_{j,d}) \\
\propto &\prod_{i:z_{i,d}=j} K(t_{i,d}|\boldsymbol{\psi}^*_{j,d}) \times \prod_{i=1}^{n_d} \exp\left(-\xi_{i,d} q^J_{j,d} K(t_{i,d}|\boldsymbol{\psi}^*_{j,d})\right) \\
&\times \text{log-N}(\lambda^*_{j,d}|r_j, s^2) \times \text{Unif}\left(\mu^*_{j,d}| - \frac{\pi\lambda^*_{j,d}}{2}, \frac{\pi\lambda^*_{j,d}}{2}\right) \\
\propto &\prod_{i:z_{i,d}=j} K(t_{i,d}|\boldsymbol{\psi}^*_{j,d}) \times \prod_{i=1}^{n_d} \exp\left(-\xi_{i,d} q^J_{j,d} K(t_{i,d}|\boldsymbol{\psi}^*_{j,d})\right) \\
&\times \frac{1}{\lambda^*_{j,d}} \exp\left(-\frac{1}{2s^2}\left(\log(\lambda^*_{j,d}) - r_j\right)^2\right) \times \frac{1}{\lambda^*_{j,d}} \times \mathbb{I}\left(\lambda^*_{j,d} > \frac{2|\mu^*_{j,d}|}{\pi}\right).
\end{aligned}$$

Similar to $\mu^*_{j,d}$, the distribution is non-standard and the AMH scheme described in Section B.2.1 is applied. We use the following transformation to transform $\lambda^*_{j,d}$ on the real line, dropping subscript for notation simplicity:

$$x = g(\lambda^*) = \log(\lambda^* - \lambda^-) \in \mathbb{R},$$

where $\lambda^- = 2|\mu^*|/\pi$ is the lower bound. The Jacobian term is

$$J_x = \frac{dx}{d\lambda^*} = \frac{1}{\lambda^* - \lambda^-},$$

and the inverse transformation is

$$\lambda^* = \exp(x) + \lambda^-.$$

The logarithm of the conditional distribution is

$$\begin{aligned}
\log \pi(\lambda^*_{j,d}| \ldots) = &- \frac{2}{{\sigma^*_{j,d}}^2} \times \sum_{I_{j,d}} \sin^2\left(\frac{t_{i,d} - \mu^*_{j,d}}{\lambda^*_{j,d}}\right) - \sum_{i=1}^{n_d} \xi_{i,d} q^J_{j,d} \exp\left[-\frac{2}{{\sigma^*_{j,d}}^2} \times \sin^2\left(\frac{t_{i,d} - \mu^*_{j,d}}{\lambda^*_{j,d}}\right)\right] \\
&- 2\log(\lambda^*_{j,d}) - \frac{1}{2s^2}\left(\log(\lambda^*_{j,d}) - r_j\right)^2 + \text{const.},
\end{aligned}$$

where $I_{j,d} = \{i : z_{i,d} = j\}$. The acceptance probability is

$$\begin{aligned}\alpha(\lambda^*_{new}, \lambda^*_{old}) &= \min\left(1, \frac{\pi(\lambda^*_{new})Q_n(\lambda^*_{old}|\lambda^*_{new})}{\pi(\lambda^*_{old})Q_n(\lambda^*_{new}|\lambda^*_{old})}\right) \\ &= \min\left(1, \frac{\pi(\lambda^*_{new})(\lambda^*_{new} - \lambda^-)}{\pi(\lambda^*_{old})(\lambda^*_{old} - \lambda^-)}\right) \\ &= \min\left(1, \exp\left[\log\pi(\lambda^*_{new}) - \log\pi(\lambda^*_{old}) + \log\big(\lambda^*_{new} - \lambda^-\big) - \log\big(\lambda^*_{old} - \lambda^-\big)\right]\right).\end{aligned}$$

The adaptive variance is updated as described in Section B.2.1.

B.2.3 Bandwidth ${\sigma^*_{j,d}}^2$

As for ${\sigma^*_{j,d}}^2$, the full conditional distribution is

$$\begin{aligned}&\pi({\sigma^*_{j,d}}^2|h_j, m^2, \{z_{i,d}\}_{i=1}^{n_d}, \{\xi_{i,d}\}_{i=1}^{n_d}, \{t_{i,d}\}_{i=1}^{n_d}, q^J_{j,d}, \mu^*_{j,d}, \lambda^*_{j,d}) \\ &\propto \prod_{i:z_{i,d}=j} K(t_{i,d}|\boldsymbol{\psi}^*_{j,d}) \times \prod_{i=1}^{n_d} \exp\left(-\xi_{i,d} q^J_{j,d} K(t_{i,d}|\boldsymbol{\psi}^*_{j,d})\right) \times \mathrm{IG}\left({\sigma^*_{j,d}}^2|a_j, b_j\right),\end{aligned}$$

where $a_j = 2 + \frac{h_j^2}{m^2}, b_j = h_j^2 + \frac{h_j^3}{m^2}$. Here we introduce the latent variable $u_{i,j,d}$, same as the step in Section A.4. Then the full conditional distribution becomes

$$\begin{aligned}&\pi({\sigma^*_{j,d}}^2, \{u_{i,j,d}\}_{i=1}^{n_d}|h_j, m^2, \{z_{i,d}\}_{i=1}^{n_d}, \{\xi_{i,d}\}_{i=1}^{n_d}, \{t_{i,d}\}_{i=1}^{n_d}, q^J_{j,d}, \mu^*_{j,d}, \lambda^*_{j,d}) \\ &\propto \prod_{i:z_{i,d}=j} K(t_{i,d}|\boldsymbol{\psi}^*_{j,d}) \times \prod_{i=1}^{n_d} \mathbb{I}\left(u_{i,j,d} < M_{i,j,d}\right) \times \mathrm{IG}\left({\sigma^*_{j,d}}^2|a_j, b_j\right),\end{aligned}$$

where $M_{i,j,d} = \exp\left(-\xi_{i,d} q^J_{j,d} K(t_{i,d}|\boldsymbol{\psi}^*_{j,d})\right)$.

We sample $u_{i,j,d} \sim \mathrm{Unif}\left(0, \exp\left(-\xi_{i,d} q^J_{j,d} K(t_{i,d}|\boldsymbol{\psi}^*_{j,d})\right)\right)$. For ${\sigma^*_{j,d}}^2$, the conditional distribution is

$$\begin{aligned}\pi({\sigma^*_{j,d}}^2|\ldots) &\propto \prod_{i:z_{i,d}=j} K(t_{i,d}|\boldsymbol{\psi}^*_{j,d}) \times \mathrm{IG}\left({\sigma^*_{j,d}}^2|a_j, b_j\right) \times \mathbb{I}({\sigma^*_{j,d}}^2 \in E_{j,d}) \\ &\propto \exp\left[-\frac{2}{{\sigma^*_{j,d}}^2} \times \sum_{I_{j,d}} \sin^2\left(\frac{t_{i,d} - \mu^*_{j,d}}{\lambda^*_{j,d}}\right)\right] \times \left({\sigma^*_{j,d}}^2\right)^{-a_j - 1} \exp\left(-\frac{b_j}{{\sigma^*_{j,d}}^2}\right) \times \mathbb{I}({\sigma^*_{j,d}}^2 \in E_{j,d}) \\ &\propto \left({\sigma^*_{j,d}}^2\right)^{-a_j - 1} \times \exp\left[-\frac{2}{{\sigma^*_{j,d}}^2} \times \sum_{I_{j,d}} \sin^2\left(\frac{t_{i,d} - \mu^*_{j,d}}{\lambda^*_{j,d}}\right) - \frac{b_j}{{\sigma^*_{j,d}}^2}\right] \times \mathbb{I}({\sigma^*_{j,d}}^2 \in E_{j,d}).\end{aligned}$$

The truncation region can be derived in the same fashion as described in Section A.4.3, and is given by

$$E_{j,d} = \left(0, \sigma^+_{j,d}\right), \quad \sigma^+_{j,d} = \min_{i:-\log u_{i,j,d} < \xi_{i,d} q^J_{j,d}} -\frac{2\sin^2\left(\frac{t_{i,d} - \mu^*_{j,d}}{\lambda^*_{j,d}}\right)}{\log\left[-\frac{\log u_{i,j,d}}{\xi_{i,d} q^J_{j,d}}\right]}.$$

Thus the full conditional is a truncated inverse-gamma distribution with truncation region $E_{j,d}$

$$\sigma^{*}_{j,d}{}^2|\ldots \sim \text{IG}\left(a_j, b_j + 2\sum_{I_{j,d}} \sin^2\left(\frac{t_{i,d}-\mu^*_{j,d}}{\lambda^*_{j,d}}\right)\right).$$

The latent variable is not introduced for $\mu^*_{j,d}$ and $\lambda^*_{j,d}$ as the full conditional distributions are still not of a standard form.

B.3 Component-specific Parameters $\mathbf{L}^*_j$ and $\mathbf{\Sigma}^*_j$

Recall the density of an inverse-Wishart distribution is

$$f(\mathbf{\Sigma}|\omega, \mathbf{\Phi}) = \frac{|\mathbf{\Phi}|^{\omega/2}}{\frac{\omega G}{2}\Gamma_G\left(\frac{\omega}{2}\right)}|\mathbf{\Sigma}|^{-\frac{\omega+G+1}{2}}\exp\left(-\frac{1}{2}Tr(\mathbf{\Phi}\mathbf{\Sigma}^{-1})\right),$$

where $\mathbf{\Sigma}$ is of dimension $G\times G$, $\Gamma_G(\cdot)$ is the multivariate gamma function, and Tr denotes the trace. The density of a matrix normal distribution is

$$f(\mathbf{Y}|\mathbf{M},\mathbf{U},\mathbf{V}) = \frac{\exp\left(-\frac{1}{2}Tr\left(\mathbf{V}^{-1}(\mathbf{Y}-\mathbf{M})^T\mathbf{U}^{-1}(\mathbf{Y}-\mathbf{M})\right)\right)}{(2\pi)^{NG/2}|\mathbf{V}|^{N/2}|\mathbf{U}|^{G/2}},$$

where $\mathbf{Y}$ and $\mathbf{M}$ are of dimension $N\times G$, $\mathbf{U}$ is $N\times N$ for variance among the rows of $\mathbf{Y}$, $\mathbf{V}$ is $G\times G$ for variance among columns,

For empty components, new samples for $\mathbf{L}^*_j, \mathbf{\Sigma}^*_j$ are drawn from the prior directly. For occupied components, let N_j denote the size of component j, $\mathbf{Y}_j$ denote the data matrix with dimension $N_j\times G$, stacking all observations belonging to component j, and $\mathbf{X}_j$ the corresponding design matrix of dimension $N_j\times(G+1)$. The first column of $\mathbf{X}_j$ is one. The likelihood for component j can be expressed as

$$\mathbf{Y}_j \sim \text{MN}(\mathbf{X}_j\mathbf{L}^*_j, \mathbf{I}_{N_j}, \mathbf{\Sigma}^*_j).$$

The full conditional distribution for $\mathbf{L}^*_j$ and $\mathbf{\Sigma}^*_j$ is

$$\begin{aligned}
&\pi(\mathbf{L}^*_j, \mathbf{\Sigma}^*_j|\mathbf{Y},\mathbf{Z},\mathbf{L}_0,\mathbf{V}_0,\mathbf{\Phi}_0,\omega_0)\\
\propto&\text{MN}(\mathbf{Y}_j|\mathbf{X}_jL^*_j,\mathbf{I}_{N_j},\mathbf{\Sigma}^*_j)\times\text{MN}(\mathbf{L}^*_j|\mathbf{L}_0,\mathbf{V}_0,\mathbf{\Sigma}^*_j)\times\text{IW}(\mathbf{\Sigma}^*_j|\omega_0,\mathbf{\Phi}_0)\\
\propto&|\mathbf{\Sigma}^*_j|^{-N_j/2}\exp\left(-\frac{1}{2}Tr\left(\mathbf{\Sigma}^{*-1}_j(\mathbf{Y}_j-\mathbf{X}_j\mathbf{L}^*_j)^T(\mathbf{Y}_j-\mathbf{X}_j\mathbf{L}^*_j)\right)\right)\\
&\times|\mathbf{\Sigma}^*_j|^{-(G+1)/2}\exp\left(-\frac{1}{2}Tr\left(\mathbf{\Sigma}^{*-1}_j(\mathbf{L}^*_j-\mathbf{L}_0)^T\mathbf{V}_0^{-1}(\mathbf{L}^*_j-\mathbf{L}_0)\right)\right)\\
&\times|\mathbf{\Sigma}^*_j|^{-(\omega_0+G+1)/2}\exp\left\{-\frac{1}{2}Tr\left(\mathbf{\Phi}_0\mathbf{\Sigma}^{*-1}_j\right)\right\}.
\end{aligned}$$

Consider the first two exponential terms. It can be shown that

$$\begin{aligned}
A&=(\mathbf{Y}_j-\mathbf{X}_j\mathbf{L}^*_j)^T(\mathbf{Y}_j-\mathbf{X}_j\mathbf{L}^*_j)+(\mathbf{L}^*_j-\mathbf{L}_0)^T\mathbf{V}_0^{-1}(\mathbf{L}^*_j-\mathbf{L}_0)\\
&=\mathbf{Y}_j^T\mathbf{Y}_j+\mathbf{L}_0^T\mathbf{V}_0^{-1}\mathbf{L}_0-2\mathbf{Y}_j^T\mathbf{X}_j\mathbf{L}^*_j+(\mathbf{L}^*_j)^T\mathbf{X}_j^T\mathbf{X}_j\mathbf{L}^*_j+(\mathbf{L}^*_j)^T\mathbf{V}_0^{-1}\mathbf{L}^*_j-2(\mathbf{L}^*_j)^T\mathbf{V}_0^{-1}\mathbf{L}_0\\
&=\mathbf{Y}_j^T\mathbf{Y}_j+\mathbf{L}_0^T\mathbf{V}_0^{-1}\mathbf{L}_0+(\mathbf{L}^*_j)^T(\mathbf{X}_j^T\mathbf{X}_j+\mathbf{V}_0^{-1})\mathbf{L}^*_j-2(\mathbf{L}^*_j)^T(\mathbf{X}_j^T\mathbf{Y}_j+\mathbf{V}_0^{-1}\mathbf{L}_0).
\end{aligned}$$

Let $\mathbf{V}_n = (\mathbf{X}_j^T\mathbf{X}_j + \mathbf{V}_0^{-1})^{-1}$. Completing the square for $\mathbf{L}_j^*$ yields

$$\begin{aligned} A &= \mathbf{Y}_j^T\mathbf{Y}_j + \mathbf{L}_0^T\mathbf{V}_0^{-1}\mathbf{L}_0 + (\mathbf{L}_j^*)^T\mathbf{V}_n^{-1}\mathbf{L}_j^* - 2(\mathbf{L}_j^*)^T\mathbf{V}_n^{-1}\underbrace{\mathbf{V}_n(\mathbf{X}_j^T\mathbf{Y}_j + \mathbf{V}_0^{-1}\mathbf{L}_0)}_{\mathbf{L}_n} \\ &= \mathbf{Y}_j^T\mathbf{Y}_j + \mathbf{L}_0^T\mathbf{V}_0^{-1}\mathbf{L}_0 + (\mathbf{L}_j^* - \mathbf{L}_n)^T\mathbf{V}_n^{-1}(\mathbf{L}_j^* - \mathbf{L}_n) - \mathbf{L}_n^T\mathbf{V}_n^{-1}\mathbf{L}_n. \end{aligned}$$

Since $Tr(A+B) = Tr(A) + Tr(B)$ and $Tr(AB) = Tr(BA)$, the joint full conditional distribution can be written as

$$\begin{aligned} \pi(\mathbf{L}_j^*, \mathbf{\Sigma}_j^* | \ldots) \propto & |\mathbf{\Sigma}_j^*|^{-(G+1)/2} \exp\left(-\frac{1}{2} Tr\left(\mathbf{\Sigma}_j^{*-1}(\mathbf{L}_j^* - \mathbf{L}_n)^T\mathbf{V}_n^{-1}(\mathbf{L}_j^* - \mathbf{L}_n)\right)\right) \\ & \times |\mathbf{\Sigma}_j^*|^{-(N_j+\omega_0+G+1)/2} \exp\left(-\frac{1}{2} Tr\left(\mathbf{\Sigma}_j^{*-1}\left(\mathbf{\Phi}_0 + \mathbf{Y}_j^T\mathbf{Y}_j + \mathbf{L}_0^T\mathbf{V}_0^{-1}\mathbf{L}_0 - \mathbf{L}_n^T\mathbf{V}_n^{-1}\mathbf{L}_n\right)\right)\right), \end{aligned}$$

which corresponds to the following full conditional distributions:

$$\mathbf{L}_j^* | \mathbf{\Sigma}_j^*, \ldots \sim \mathrm{MN}(\mathbf{L}_n, \mathbf{V}_n, \mathbf{\Sigma}_j^*), \quad \mathbf{\Sigma}_j^* | \ldots \sim \mathrm{IW}(\omega_n, \mathbf{\Phi}_n),$$

where

$$\begin{aligned} &\mathbf{L}_n = \mathbf{V}_n(\mathbf{X}_j^T\mathbf{Y}_j + \mathbf{V}_0^{-1}\mathbf{L}_0), \quad \mathbf{V}_n = (\mathbf{X}_j^T\mathbf{X}_j + \mathbf{V}_0^{-1})^{-1}, \\ &\mathbf{\Phi}_n = \mathbf{\Phi}_0 + \mathbf{Y}_j^T\mathbf{Y}_j + \mathbf{L}_0^T\mathbf{V}_0^{-1}\mathbf{L}_0 - \mathbf{L}_n^T\mathbf{V}_n^{-1}\mathbf{L}_n, \quad \omega_n = N_j + \omega_0. \end{aligned}$$

B.4 Hyperparameters r_j, s^2, h_j and m^2

This section details the sampling steps for hyperparameters related to kernel parameters.

B.4.1 Prior Means r_j

For each j, we have

$$\begin{aligned} \pi(r_j | \left\{\lambda_{j,d}^*\right\}_{d=1}^D, \mu_r, \sigma_r^2, s^2) &\propto \prod_{d=1}^D \text{log-N}(\lambda_{j,d}^* | r_j, s^2) \times \mathrm{N}(r_j | \mu_r, \sigma_r^2) \\ &\propto \exp\left[-\frac{1}{2s^2}\sum_{d=1}^D \left(r_j - \log\left(\lambda_{j,d}^*\right)\right)^2\right] \times \exp\left[-\frac{1}{2\sigma_r^2}\left(r_j - \mu_r\right)^2\right]. \end{aligned}$$

The full conditional distribution for r_j is a normal distribution

$$r_j | \ldots \sim \mathrm{N}(\hat{\mu}_r, \hat{\sigma}_r^2),$$

where

$$\hat{\sigma}_r^2 = \left(\frac{1}{\sigma_r^2} + \frac{D}{s^2}\right)^{-1}, \quad \hat{\mu}_r = \frac{\mu_r/\sigma_r^2 + \sum_{d=1}^D \log\left(\lambda_{j,d}^*\right)/s^2}{1/\sigma_r^2 + D/s^2}.$$

B.4.2 Prior Variance s^2

$$\begin{aligned}\pi(s^2|\left\{\lambda^*_{j,d}\right\}_{j=1,d=1}^{J,D},\eta_1,\eta_2,\boldsymbol{r}) &\propto \prod_{j=1}^{J}\prod_{d=1}^{D}\text{log-N}(\lambda^*_{j,d}|r_j,s^2)\times \text{IG}(s^2|\eta_1,\eta_2)\\ &\propto (s^2)^{-\frac{JD}{2}}\exp\left[-\frac{1}{s^2}\times\frac{1}{2}\sum_{j=1}^{J}\sum_{d=1}^{D}\left(\log\left(\lambda^*_{j,d}\right)-r_j\right)^2\right]\\ &\quad\times(s^2)^{-\eta_1-1}\exp\left[-\frac{\eta_2}{s^2}\right],\end{aligned}$$

i.e.

$$s^2|\ldots\sim\text{IG}\left(\frac{JD}{2}+\eta_1,\eta_2+\frac{1}{2}\sum_{j=1}^{J}\sum_{d=1}^{D}\left(\log\left(\lambda^*_{j,d}\right)-r_j\right)^2\right).$$

B.4.3 Prior Means h_j

For each j,

$$\begin{aligned}\pi(h_j|\left\{{\sigma^*_{j,d}}^2\right\}_{d=1}^{D},\mu_h,\sigma_h^2,m^2) &\propto \prod_{d=1}^{D}\text{IG}\left({\sigma^*_{j,d}}^2|a_j,b_j\right)\times\text{log-N}(h_j|\mu_h,\sigma_h^2),\\ &\propto \prod_{d=1}^{D}\frac{b_j^{a_j}}{\Gamma(a_j)}({\sigma^*_{j,d}}^2)^{-a_j-1}\exp\left(-\frac{b_j}{{\sigma^*_{j,d}}^2}\right)\\ &\quad\times\frac{1}{h_j}\exp\left(-\frac{1}{2\sigma_h^2}(\log(h_j)-\mu_h)^2\right),\end{aligned}$$

where $a_j=2+\frac{h_j^2}{m^2}, b_j=h_j^2+\frac{h_j^3}{m^2}$. The distribution has no closed form and hence the AMH scheme described in Section B.2.1 is applied. The log transformation is applied with $x=g(h_j)=\log(h_j)\in\mathbb{R}$. The Jacobian term is $J_x=dx/dh_j=1/h_j$, and the inverse transformation is $h_j=\exp(x)$.

The logarithm of the full conditional density is

$$\begin{aligned}\log\pi(h_j|\ldots) =& Da_j\log(b_j)-D\log(\Gamma(a_j))-(a_j+1)\sum_{d=1}^{D}\log\left({\sigma^*_{j,d}}^2\right)\\ &-b_j\sum_{d=1}^{D}\frac{1}{{\sigma^*_{j,d}}^2}-\log(h_j)-\frac{1}{2\sigma_h^2}(\log(h_j)-\mu_h)^2+\text{const.}\end{aligned}$$

Hence the acceptance probability of the new sample is

$$\begin{aligned}\alpha(h_{new},h_{old}) &= \min\left(1,\frac{\pi(h_{new})Q_n(h_{old}|h_{new})}{\pi(h_{old})Q_n(h_{new}|h_{old})}\right)\\ &= \min\left(1,\frac{\pi(h_{new})h_{new}}{\pi(h_{old})h_{old}}\right)\\ &= \min\left(1,\exp\left[\log\pi(h_{new})-\log\pi(h_{old})+\log(h_{new})-\log(h_{old})\right]\right).\end{aligned}$$

B.4.4 Prior Variance m^2

The full conditional distribution is

$$\begin{aligned}\pi(m^2|\left\{{\sigma^*_{j,d}}^2\right\}_{j=1,d=1}^{J,D}, \kappa_1, \kappa_2, \boldsymbol{h}) &\propto \prod_{j=1}^{J}\prod_{d=1}^{D} \mathrm{IG}\left({\sigma^*_{j,d}}^2|a_j, b_j\right) \times \mathrm{IG}(m^2|\kappa_1, \kappa_2),\\ &\propto \prod_{j=1}^{J}\prod_{d=1}^{D} \frac{b_j^{a_j}}{\Gamma(a_j)}({\sigma^*_{j,d}}^2)^{-a_j-1}\exp\left(-\frac{b_j}{{\sigma^*_{j,d}}^2}\right)\\ &\quad\times (m^2)^{-\kappa_1-1}\exp\left(-\frac{\kappa_2}{m^2}\right),\end{aligned}$$

which is not of a standard form and we apply adaptive Metropolis-Hastings (Section B.2.1). Similar to h_j, a log transformation is applied. The logarithm of the full conditional density is

$$\begin{aligned}\log\pi(m^2|\ldots) = &\sum_{j=1}^{J}\left(Da_j\log(b_j) - D\log(\Gamma(a_j)) - (a_j+1)\sum_{d=1}^{D}\log\left({\sigma^*_{j,d}}^2\right) - b_j\sum_{d=1}^{D}\frac{1}{{\sigma^*_{j,d}}^2}\right)\\ &-(\kappa_1+1)\log\left(m^2\right) - \frac{\kappa_2}{m^2} + \text{const.}\end{aligned}$$

Hence the acceptance probability is

$$\begin{aligned}\alpha(m^2_{new}, m^2_{old}) &= \min\left(1, \frac{\pi(m^2_{new})Q_n(m^2_{old}|m^2_{new})}{\pi(m^2_{old})Q_n(m^2_{new}|m^2_{old})}\right)\\ &= \min\left(1, \frac{\pi(m^2_{new})m^2_{new}}{\pi(m^2_{old})m^2_{old}}\right)\\ &= \min\left(1, \exp\left[\log\pi(m^2_{new}) - \log\pi(m^2_{old}) + \log\left(m^2_{new}\right) - \log\left(m^2_{old}\right)\right]\right).\end{aligned}$$

Appendix C. Consensus Clustering

In practice, it is common to run multiple MCMC chains with a large number of iterations to account for sensitivity to different initial values and to ensure convergence. Nevertheless, for high-dimensional data, chains can easily get trapped into local posterior modes even after sufficiently long time. To overcome such problems and reduce computational costs, Coleman et al. (2022) develop a general method to exploit the posterior distribution of data partitions through an ensemble of Bayesian clustering results. The method does not require the chain to reach convergence and hence is expected to relieve computational burden. In addition, it can be readily integrated into existing Bayesian clustering frameworks without requiring substantial redevelopment of the original method.

The core idea behind consensus clustering is to run a large number of chains, denoted as chain width W, each for a small number of iterations, denoted as chain depth D. Then the D-th sample in each chain is combined to produce a posterior similarity matrix (PSM), based on which the optimal clustering can be obtained, e.g. by minimizing VI.

Coleman et al. (2022) propose a heuristic way to choose appropriate values for W and D. The rational is that increasing W and D may improve the performance substantially in the beginning, but the improvement will gradually diminish with further increases. This is similar to PCA where more variance will always be captured for more principal components, but the gain in variance will be smaller and smaller, and eventually we will have few returns.

To choose D and W, one begins with candidate sets $D' = (d_1, \ldots, d_I)$ and $W' = (w_1, \ldots, w_J)$ arranged in increasing order. For a given number of chains w_j, the PSM is computed based on the samples at the d_i-th iteration across w_j chains, and compared with the PSM obtained from the $d_{(i-1)}$-th iteration across w_j chains. The mean absolute difference (MAD) between the two matrices is a measurement of how stable the clustering is. Plotting these values as a function of D, it is expected to see an elbow-shaped curve, and a suitable D can be selected at which the curve plateaus. Similarly, to choose W, we can fix D and compute MAD between $w_{(j-1)}$ and w_j.

Appendix D. Simulation Study

In the first part of this section, we provide additional results for the simulation study on the Gaussian mixtures discussed in the manuscript. Recall that the true relationship between the weights and the covariate follows a softmax function, while a Gaussian kernel is applied in the C-HDP and DDP. Nevertheless, Figure 18 shows that the covariate-dependent probabilities can be accurately estimated in the C-HDP, with the truth mostly covered by the samples. On the other hand, although the group indicator is included in the DDP as an additional categorical covariate, the posterior samples may still fail to cover the true relationship, e.g. in clusters 2 and 3 of Dataset 5 (Figure 19). When it comes to the DDP without using the group indicator (Figure 20), the estimated relationship is the same across groups, which appears as an average across data sets and exhibits much smaller uncertainty in the posterior samples. Finally, as the HDP does not account for the covariate, the probabilities are only constant with x, showing large uncertainty (Figure 21).

Further, to better understand how the quality of the approximation depends on the truncation level J, we compare the performance of the C-HDP across different truncation levels. The results demonstrate that inference is stable (Figure 22 left for clustering and middle for density estimation) as long as we use a large enough J, which in this case is three as the true number of components is three. The computational run time is approximately linear with J.

Next, we conduct more simulation studies to assess the covariate-dependent HDP model. Two settings are considered for two different types of kernels: a Gaussian kernel and a periodic kernel, and two types of likelihoods: negative-binomial and vector autoregression. Posterior inference is performed based on the MCMC algorithms detailed in Section A and Section B.

We are interested in the posterior inference of the clustering, the covariate-dependent probabilities, and component-specific parameters. For the clustering, we compare the optimal clustering with the true clustering based on adjusted Rand index (ARI) and variation of information (VI; normalized on $[0, 1]$). A large value for ARI and a small value for VI suggest good performance. For covariate-dependent probabilities, we check if the true relationship is covered by the posterior samples. Credible intervals are computed to verify if parameters can be correctly estimated.

D.1 Simulation Setting 1: A Gaussian Kernel

Below we consider two simulation scenarios for the single-cell clustering model. The first simulated data is generated from the proposed model to test the ability to recover true clustering and parameters, especially the time-dependent probabilities. For the second simulated data, we investigate the robustness of the model under misspecification of the relationship between the probability and covariate.

D.1.1 Simulation 1 and 2

In Simulation 1, the relationship between the probability and covariate is based on the proposed Gaussian kernel, while in Simulation 2, the relationship follows from a logistic regression. In each simulation setting, two generated data sets have the same number of

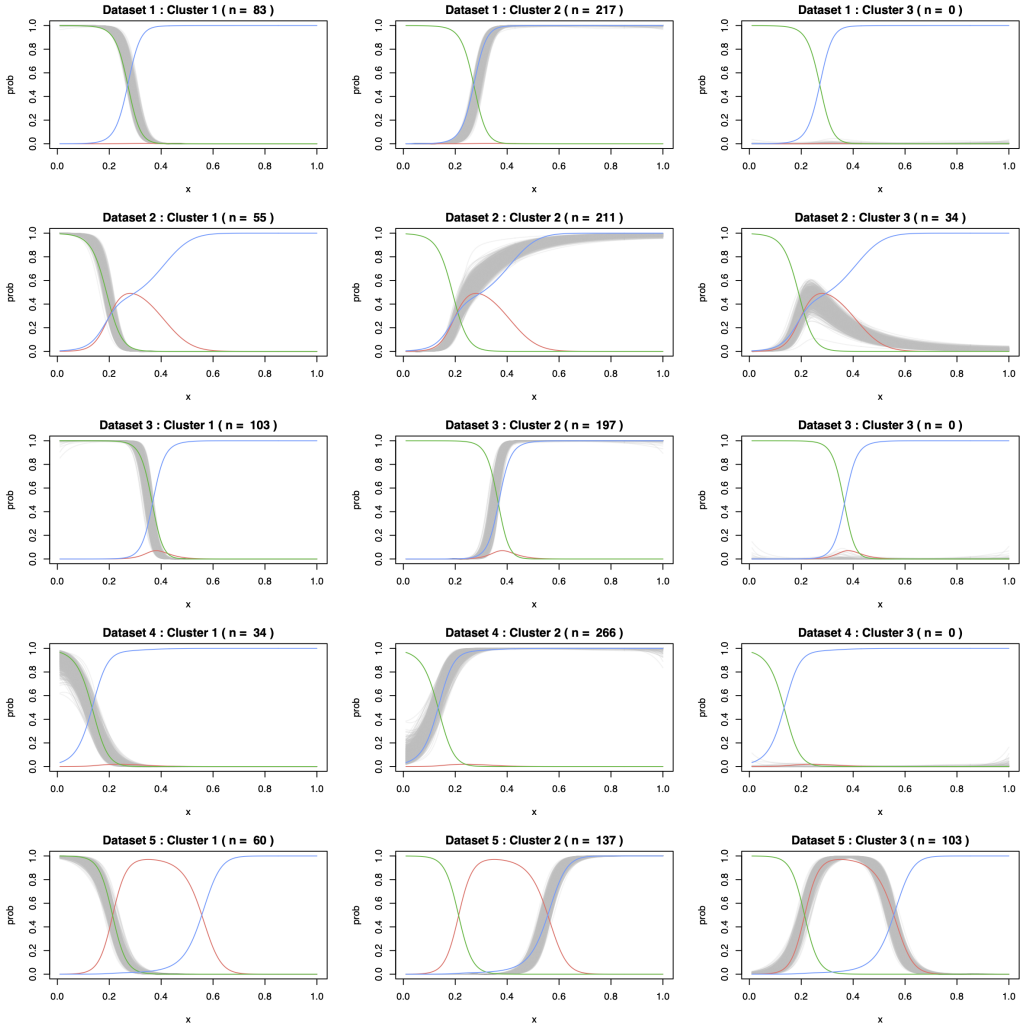


Figure 18: C-HDP: Posterior samples of the covariate-dependent probabilities (grey) for each cluster in each data set in a selected replicate. In each panel, the true relationships for three clusters are shown in colored lines. The size of each estimated cluster is indicated in the title.

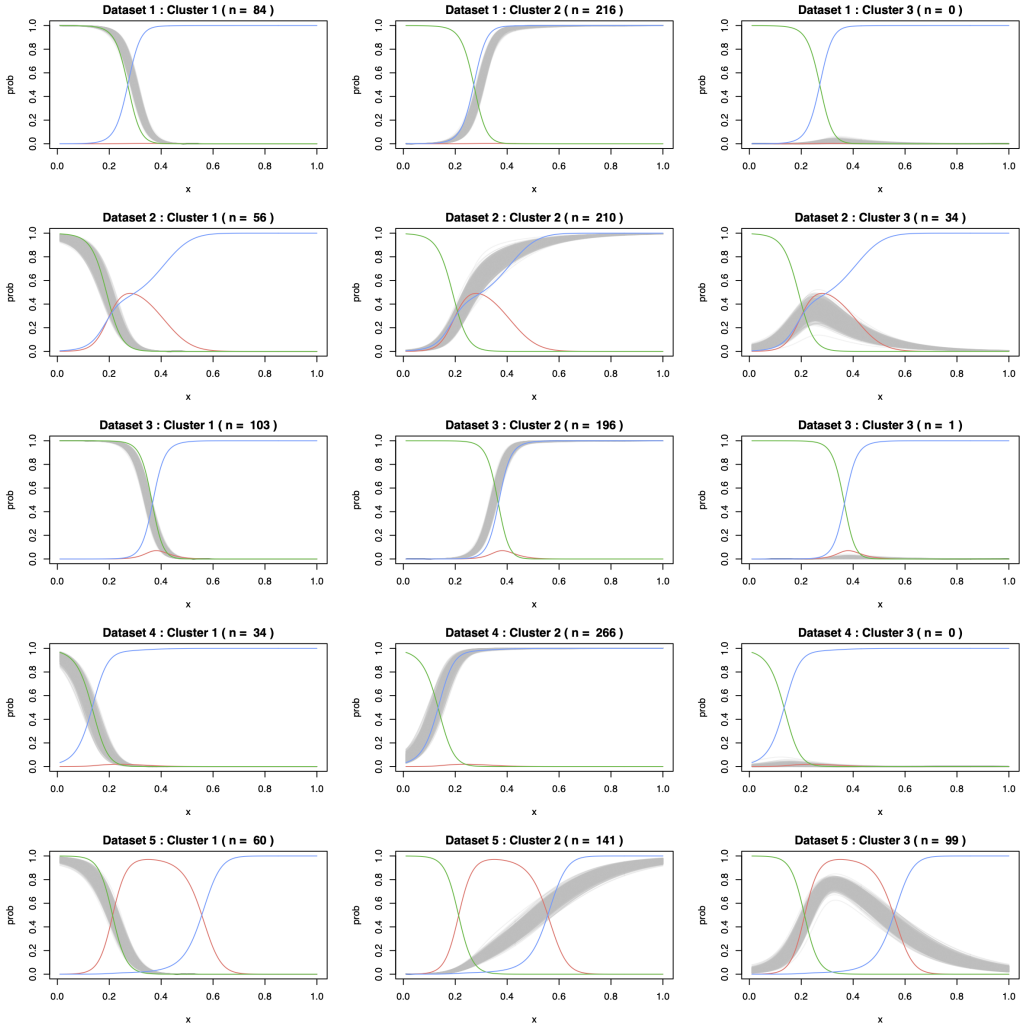


Figure 19: DDP (with group): Posterior samples of the covariate-dependent probabilities (grey) for each cluster in each data set in a selected replicate. In each panel, the true relationships for three clusters are shown in colored lines. The size of each estimated cluster is indicated in the title.

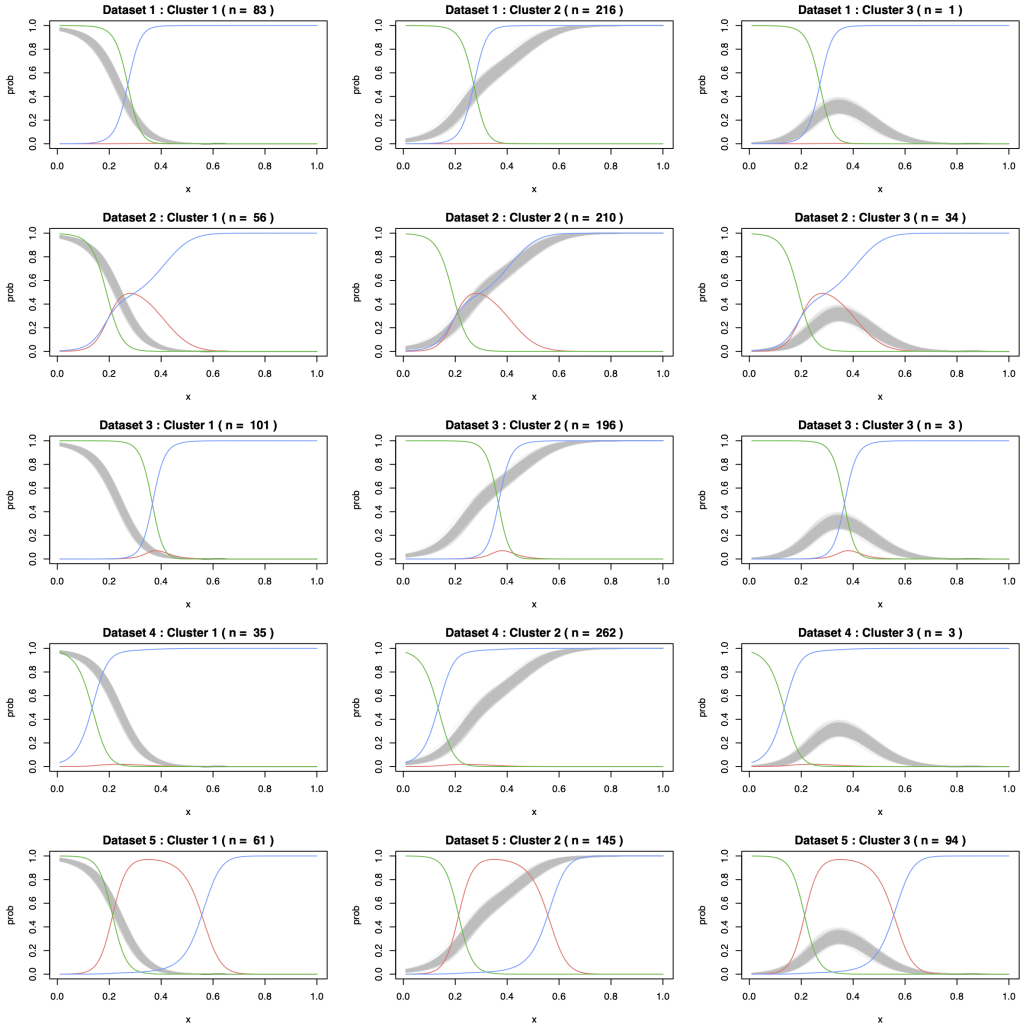


Figure 20: DDP (without group): Posterior samples of the covariate-dependent probabilities (grey) for each cluster in each data set in a selected replicate. In each panel, the true relationships for three clusters are shown in colored lines. The size of each estimated cluster is indicated in the title.

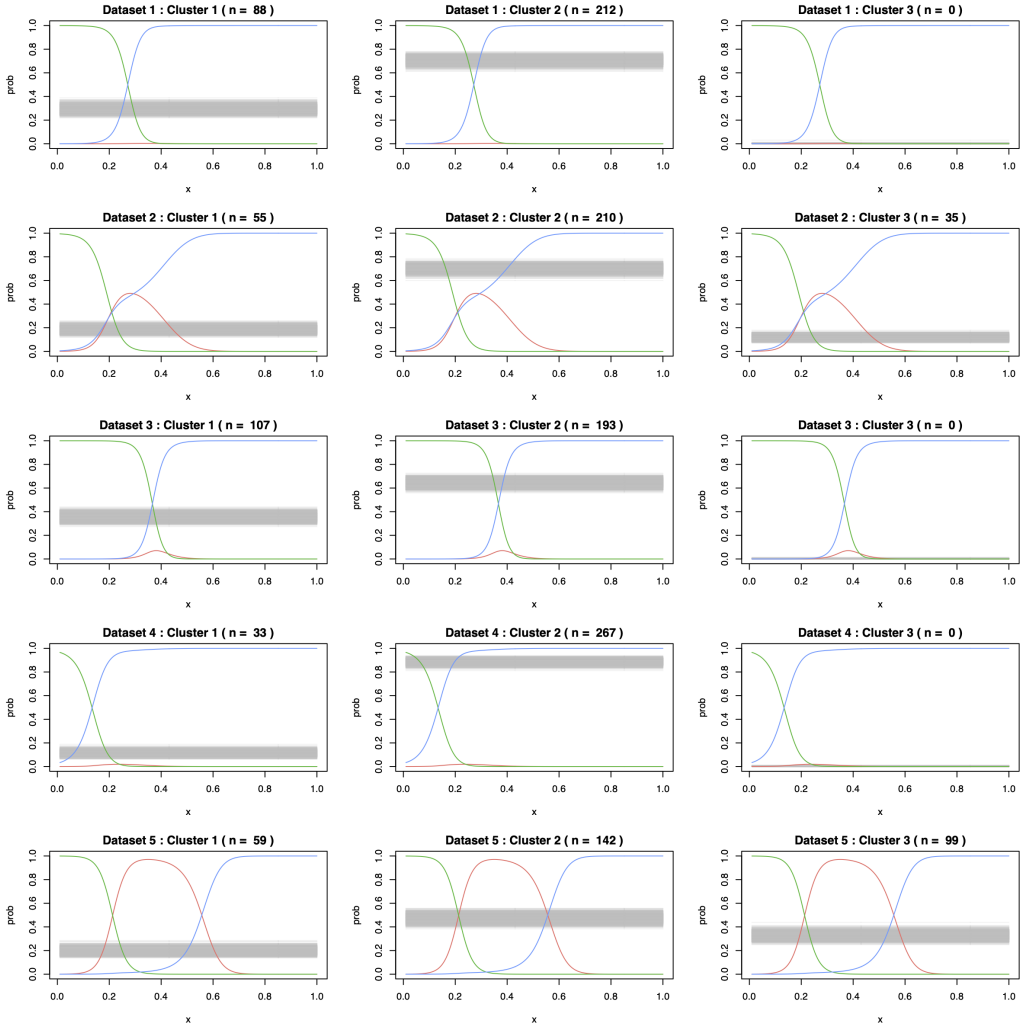


Figure 21: HDP: Posterior samples of the probabilities (grey) for each cluster in each data set in a selected replicate. In each panel, the true relationships for three clusters are shown in colored lines. The size of each estimated cluster is indicated in the title.

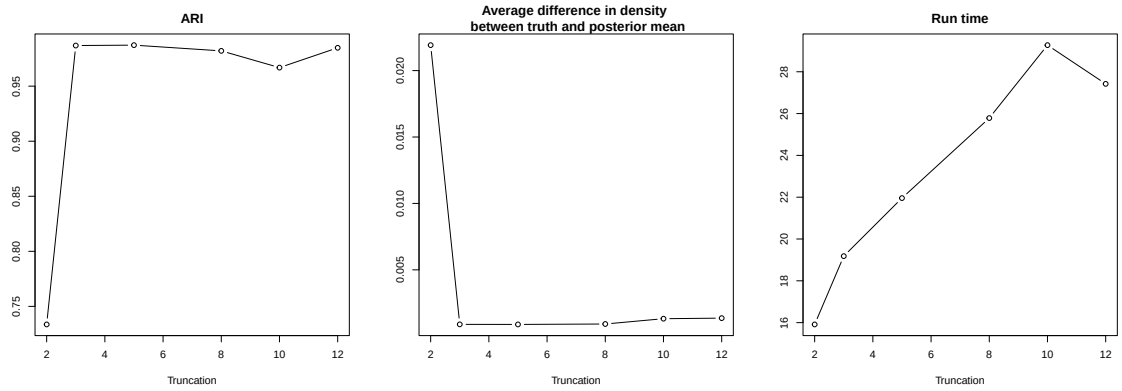


Figure 22: For a data set simulated from Gaussian mixtures with three clusters, adjusted rand index (ARI) comparing the truth to the estimated clustering (left), average difference in the density between the truth and posterior mean across observations (middle), and computational run time in minutes versus the truncation level used in the C-HDP (right).

cells $C_1 = C_2 = 100$ and genes $G = 10$, and two clusters are generated. To avoid excessive zero counts in the data, we generate capture efficiencies with a mean of 0.6. The data-generating process is detailed below.

Data-generating Process. In Simulation 1 and 2, the data is generated from the following:

$$\begin{aligned} y_{c,g,d}|y^0_{c,g,d}, \beta_{c,d} &\overset{ind}{\sim} \text{Bin}(y^0_{c,g,d}, \beta_{c,d}), \\ y^0_{c,g,d}|z_{c,d} = j, \mu^*_{j,g}, \phi^*_{j,g} &\overset{ind}{\sim} \text{NB}(\mu^*_{j,g}, \phi^*_{j,g}), \\ z_{c,d}|p^J_{1,d}(t_{c,d}), \ldots, p^J_{J,d}(t_{c,d}) &\overset{ind}{\sim} \text{Cat}(p^J_{1,d}(t_{c,d}), \ldots, p^J_{J,d}(t_{c,d})), \\ t_{c,d} &\overset{i.i.d}{\sim} \text{Unif}(0,1). \end{aligned}$$

The component-specific parameters and capture efficiencies are simulated from

$$\begin{aligned} \mu^*_{j,g}|j = 1 \overset{i.i.d}{\sim} \text{log-N}(1, \alpha^2_\mu), \quad \mu^*_{j,g}|j = 2 \overset{i.i.d}{\sim} \text{log-N}(3, \alpha^2_\mu), \\ \phi^*_{j,g}|\mu^*_{j,g} \overset{ind}{\sim} \text{log-N}(b_0 + b_1 \log(\mu^*_{j,g}), \alpha^2_\phi), \\ \beta_{c,d} \overset{ind}{\sim} \text{Beta}(a^\beta_d, b^\beta_d). \end{aligned}$$

where $b_0 = 0.25, b_1 = 0.5, \alpha^2_\mu = \alpha^2_\phi = 0.1, a^\beta_d = 3$ and $b^\beta_d = 2$.

For Simulation 1, the time-dependent probabilities are based on the proposed Gaussian kernel with the following parameters:

$$\begin{aligned} \text{Group 1}: \quad & (t^*_{1,1}, t^*_{2,1}) = (0.4, 0.9), \quad (\sigma^*_{1,1}, \sigma^*_{2,1}) = (0.08, 0.15), \quad (q_{1,1}, q_{2,1}) = (0.5, 0.5), \\ \text{Group 2}: \quad & (t^*_{1,2}, t^*_{2,2}) = (0.8, 0.3), \quad (\sigma^*_{1,2}, \sigma^*_{2,2}) = (0.1, 0.1), \quad (q_{1,2}, q_{2,2}) = (0.3, 0.7). \end{aligned}$$

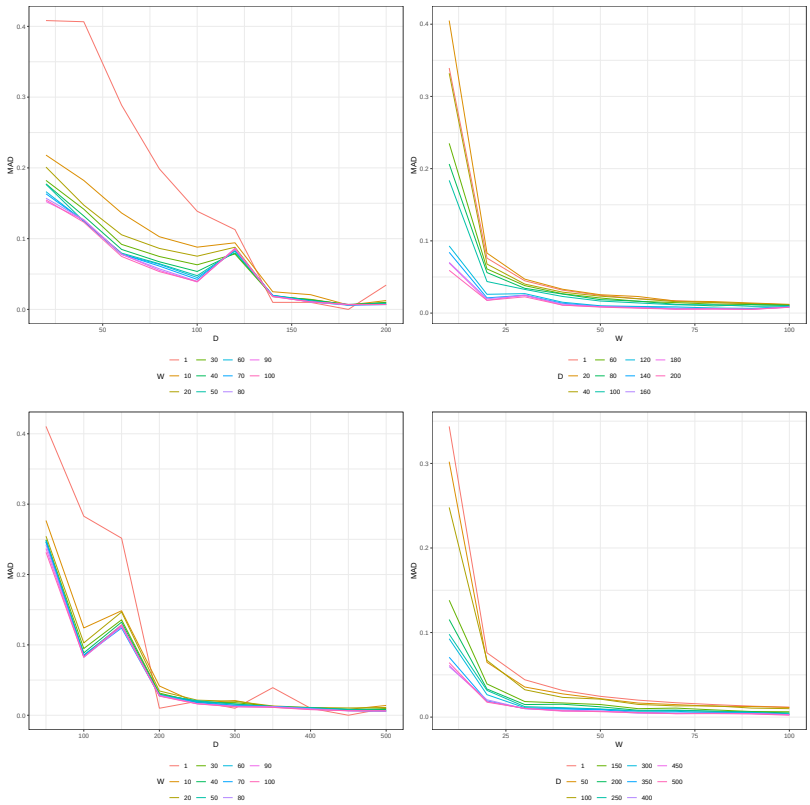


Figure 23: Choice of W and D in consensus clustering for Simulation 1 (top) and Simulation 2 (bottom).

For Simulation 2, the time-dependent probabilities $p^J_{j,d}(t)$ are based on a logistic regression where the probability of belonging to the first cluster ($j = 1$) in each group is given by

$$p^J_{1,1}(t) = \frac{1}{1 + \exp(-4 + 20t^2)}, \quad p^J_{1,2}(t) = \frac{1}{1 + \exp(4 - 10t)},$$

and $p^J_{2,d}(t) = 1 - p^J_{1,d}(t)$.

To fit the proposed model, we use a truncation level at $J = 4$ in both settings. Consensus clustering is performed with 100 parallel chains for both settings, and 200 iterations and 500 iterations, respectively (see Figure 23 for the choice of tuning parameters). The MCMC setup for the post-processing step with a fixed clustering is the same in both simulation scenarios. One chain of length 10000 is run, and the first 8000 iterations are thrown away, followed by a thinning of 2.

Results. The posterior similarity matrix from consensus clustering is shown in Figure 24, suggesting some uncertainty in cell allocations. Further, we compare our method with four popular methods for clustering scRNA-seq data: 1) Seurat (Satija et al., 2015), 2) CIDR (Lin et al., 2017), 3) TSCAN (Ji and Ji, 2016), and 4) SC3 (Kiselev et al., 2017). Table

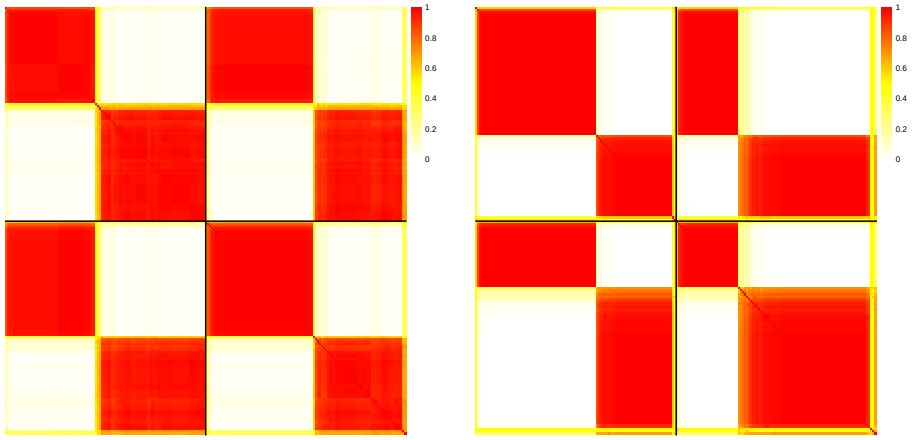


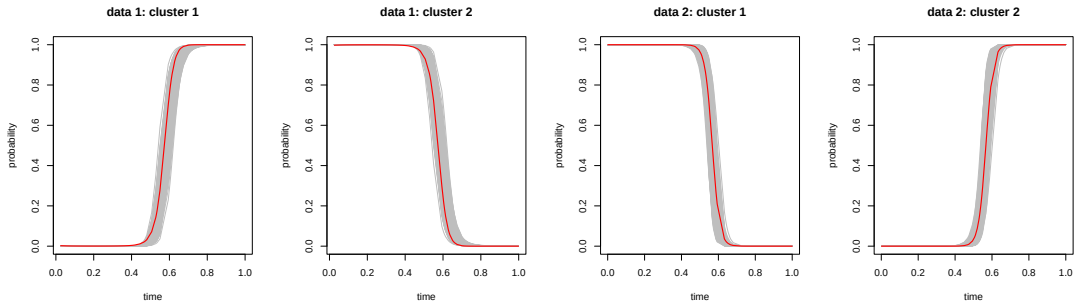
Figure 24: Posterior similarity matrix for Simulation 1 (left) and Simulation 2 (right). Diagonal blocks correspond to within-group PSM. The black solid line separates two groups.

1 shows that our C-HDP method performs the best in both simulation settings, with ARI close to 1 and VI close to 0, followed by SC3, TSCAN, Seruat and CIDR.

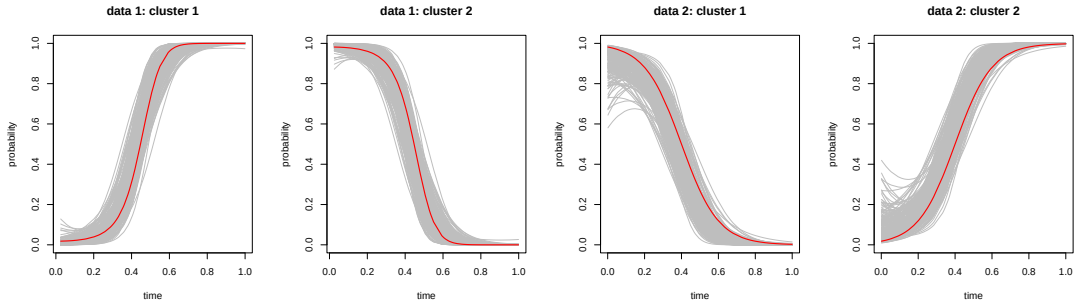
Table 1: Simulation 1 and 2: Comparison of different estimated clusterings from the proposed C-HDP model and other four competing methods, based on ARI and VI. The best result is highlighted in bold.

	C-HDP	Seurat	CIDR	TSCAN	SC3
Simulation 1 - ARI	**0.9602**	0.5908	0.0712	0.8456	0.9020
Simulation 2 - ARI	**0.9020**	0.4874	0.0826	0.7911	0.8272
Simulation 1 - VI	**0.0185**	0.1344	0.3419	0.0532	0.0377
Simulation 2 - VI	**0.0423**	0.1589	0.4080	0.0664	0.0578

As for time-dependent probabilities, Figure 25 shows that the true relationship is covered by the posterior samples in both simulation scenarios, implying the robustness of the proposed kernel-based constructions for covariate-dependent probabilities. The posterior variability is comparatively smaller in the first simulation setting, which is probably because the true probabilities change more abruptly with latent time in Simulation 1 (probability close to 0 or 1), showing more decisive cluster classifications. In addition, it is worth noticing that the individual kernel parameters $t^*_{j,d}, {\sigma^*_{j,d}}^2$ and $q_{j,d}$ may only be weakly identifiable, given that true values are not covered by the posterior samples while the relationship between the probability and weight can still be accurately recovered.



(a) Simulation 1



(b) Simulation 2

Figure 25: Posterior samples for time-dependent probabilities for Simulation 1 (top) and Simulation 2 (bottom). The red solid line denotes the truth.

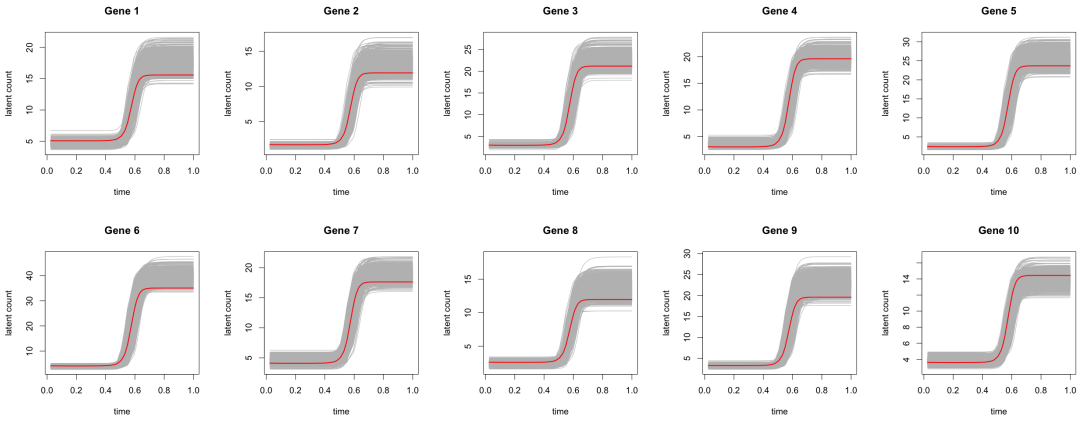
Next, we compute the expected count for a new cell c from the d-th group as a function of latent time:

$$\mathbb{E}(y^0_{c,g,d}|t_{c,d} = t, \mathbf{X}, \mathbf{Y}) = \int \sum_{j=1}^{J} p^J_{j,d}(t)\mu^*_{j,g} d\pi(\boldsymbol{q}^J_{1:J,d}, \boldsymbol{\mu}^*_{1:J,g}, \boldsymbol{\psi}^*_{1:J,d}|\mathbf{X}, \mathbf{Y}),$$

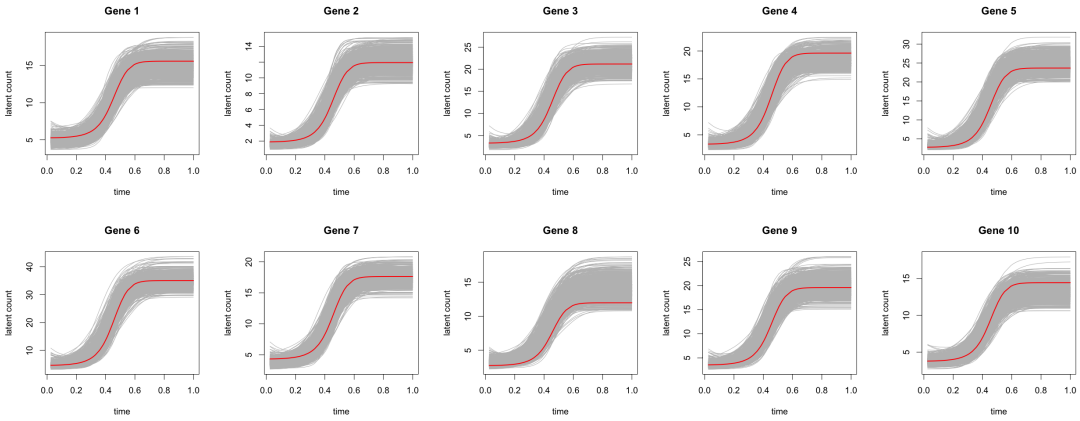
which is approximated from the MCMC samples. Figure 26 shows that the truth is well covered by the posterior samples. The large uncertainty in time-dependent probabilities for Simulation 2 is propagated into the expected count.

As for component-specific parameters, Figure 27 demonstrates that mean expressions, dispersion parameters and their relationship can be accurately inferred, with larger uncertainty in dispersion than mean expression.

Following the approach in Liu et al. (2024), we perform posterior predictive checks to examine model fit by comparing the kernel density estimation (KDE) of three statistics between the true data and replicated data, the latter generated from posterior predictive distributions. As shown in Figure 28 and Figure 29, the KDEs of the observed data is contained within the replicated data, indicating a reasonable fit of the model.

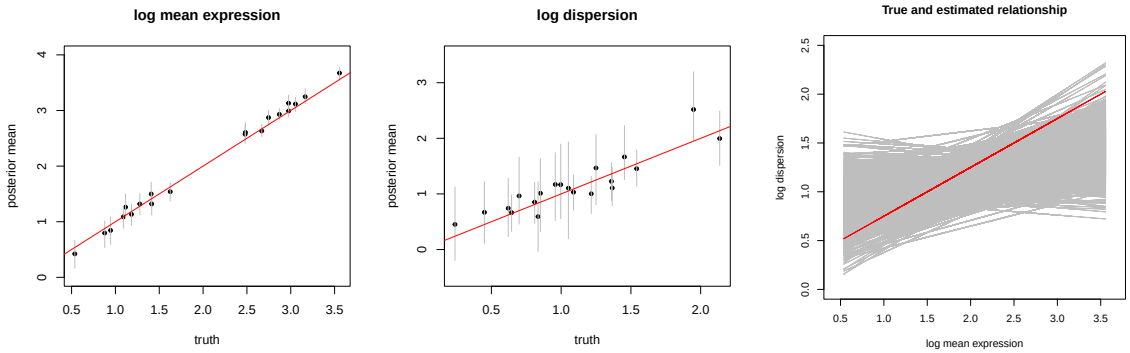


(a) Simulation 1

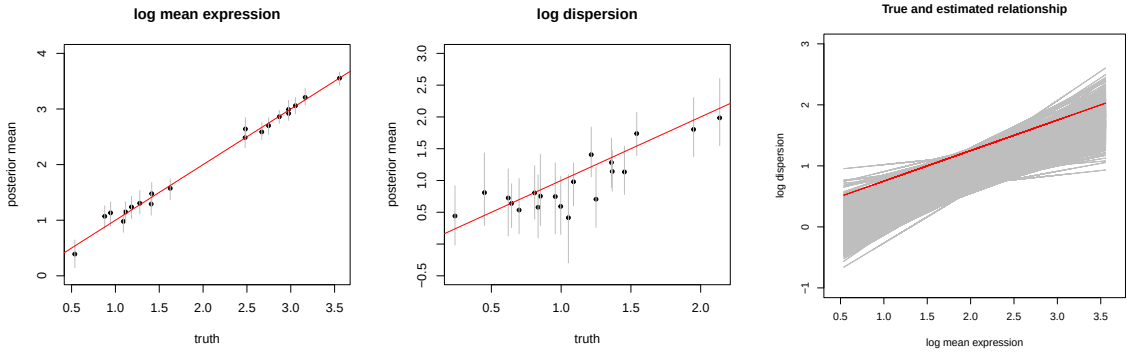


(b) Simulation 2

Figure 26: Estimated latent counts against time for the first data set in Simulation 1 (top 2 rows) and Simulation 2 (bottom 2 rows). The red solid line denotes the truth.



(a) Simulation 1



(b) Simulation 2

Figure 27: Left: Posterior mean of log mean expression against truth, with 95% HPD CIs shown in grey. Middle: Posterior mean of log dispersion against truth, with 95% HPD CIs shown in grey. The red line denotes $y = x$. Right: Posterior samples for mean-dispersion relationships. The red line denotes truth.

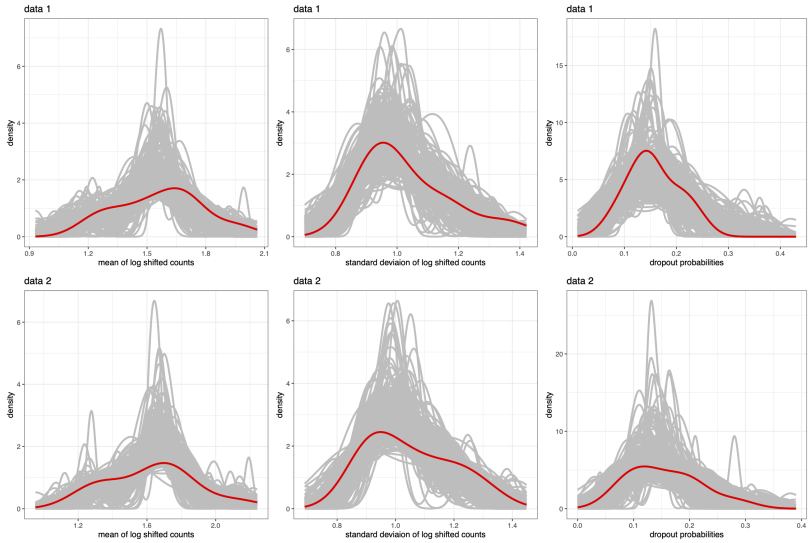


Figure 28: Posterior predictive checks for Simulation 1. Each panel shows the kernel density estimation of one statistic, with replicated and true data sets in grey and red, respectively. Left to right: mean of log shifted counts, standard deviation of log shifted counts and dropout probabilities.

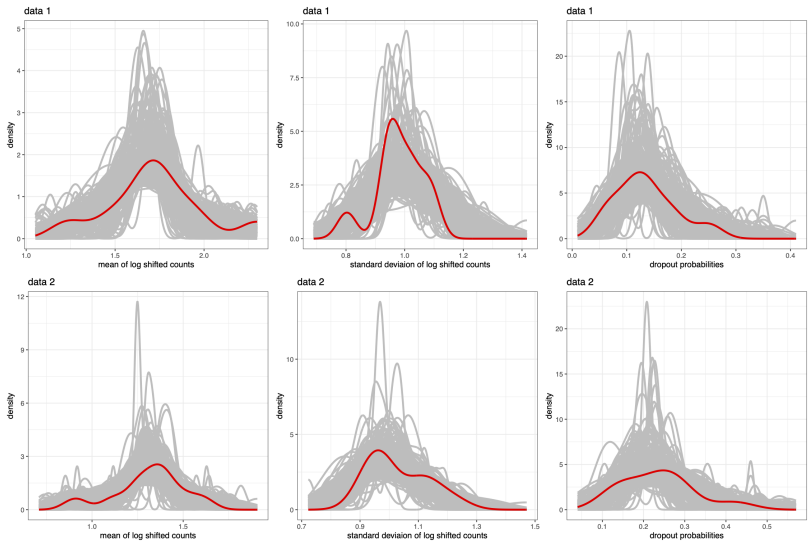


Figure 29: Posterior predictive checks for Simulation 2. Each panel shows the kernel density estimation of one statistic, with replicated and true data sets in grey and red, respectively. Left to right: mean of log shifted counts, standard deviation of log shifted counts and dropout probabilities.

D.2 Simulation Setting 2: A Periodic Kernel

For a VAR model with a periodic kernel, two data sets consisting of 3 clusters are generated from the proposed model in Section B, each consisting of $n_1 = n_2 = 151$ observations with dimension $G = 2$. The time associated with each data set is equally spaced on $[0, 1]$. The data is generated from

$$\begin{aligned}
&\boldsymbol{y}_{i,d}|\boldsymbol{y}_{i-1,d}, z_{i,d} = j, \mathbf{L}_j^*, \boldsymbol{\Sigma}_j^* \overset{ind}{\sim} \mathrm{N}((\mathbf{L}_j^*)^T \boldsymbol{x}_{i,d}, \boldsymbol{\Sigma}_j^*), \\
&z_{i,d}|p_{1,d}^J(t_{i,d}), \ldots, p_{J,d}^J(t_{i,d}) \overset{ind}{\sim} \mathrm{Cat}(p_{1,d}^J(t_{i,d}), \ldots, p_{J,d}^J(t_{i,d})),
\end{aligned}$$

where $\boldsymbol{x}_{i,d} = (1, y_{i-1,1,d}, y_{i-1,2,d})^T$. The component-specific parameters are given by

$$\begin{aligned}
&\mathbf{L}_1^* = \begin{pmatrix} 0 & 0.9 & -0.1 \\ 0 & 0.1 & 0.8 \end{pmatrix}^T, \quad \mathbf{L}_2^* = \begin{pmatrix} 1 & 0.5 & 0.1 \\ 1 & -0.1 & 0.5 \end{pmatrix}^T, \quad \mathbf{L}_3^* = \begin{pmatrix} -1 & 0.9 & -0.2 \\ -1 & 0 & 0.9 \end{pmatrix}^T, \\
&\boldsymbol{\Sigma}_1^* = \begin{pmatrix} 0.001 & 0.002 \\ 0.002 & 0.004 \end{pmatrix}, \quad \boldsymbol{\Sigma}_2^* = \begin{pmatrix} 0.005 & 0 \\ 0 & 0.005 \end{pmatrix}, \quad \boldsymbol{\Sigma}_3^* = \begin{pmatrix} 0.05 & -0.02 \\ 0.02 & 0.05 \end{pmatrix}.
\end{aligned}$$

where $\mathbf{L}_j^* = (\boldsymbol{a}_j^* \quad \mathbf{B}_j^*)^T$. For time-dependent probabilities, we set $\boldsymbol{q}_{1:3,1} = (0.5, 0.5, 0)$ and $\boldsymbol{q}_{1:3,2} = (0.6, 0, 0.4)$, leading to one shared cluster across groups and one unique cluster in each group. The kernel parameters are

$$\begin{aligned}
&\text{Group 1}: \quad \mu_{1:3,1}^* = (0, 0.2, -0.1), \quad \lambda_{1:3,1}^* = \left(\frac{0.3}{\pi}, \frac{0.6}{\pi}, \frac{0.2}{\pi}\right), \quad {\sigma_{1:3,1}^*}^2 = (0.2, 0.1, 0.05), \\
&\text{Group 2}: \quad \mu_{1:3,2}^* = (0.1, -0.1, 0), \quad \lambda_{1:3,2}^* = \left(\frac{0.6}{\pi}, \frac{0.4}{\pi}, \frac{0.3}{\pi}\right), \quad {\sigma_{1:3,2}^*}^2 = (0.05, 0.2, 0.1).
\end{aligned}$$

For each data set, the first observation is not clustered as its previous time point is not observed. To infer the clustering, we run the MCMC algorithm described in Section B with $J = 6$ for 5000 iterations, followed by a burnin of 3000. For the post-processing step with fixed clustering, we run one chain with 16000 iterations, and apply a burnin of 12000, followed by a thinning of 2, giving 2000 MCMC samples.

D.2.1 Results

Based on the VI criterion, the C-HDP model correctly finds the true clustering (ARI = 1), with the posterior similarity matrix shown in Figure 30, demonstrating small uncertainty.

In addition, to interpret the clusters, we show the time-series plot for each dimension (Figure 30 right), the identified clusters appear to share homogeneous dependence on the previous time. For instance, cluster 3 mainly shows an increasing trend in the first dimension, which can be further confirmed in Figure 31 where cluster 3 is mostly above the equivalent line $y = x$ in dimension 1. On the contrary, cluster 1 generally decreases in the first dimension. In addition, by computing the posterior estimated relationship between consecutive time points:

$$\boldsymbol{y}_{i,d} = \hat{\boldsymbol{a}}_j^* + \hat{\mathbf{B}}_j^* \boldsymbol{y}_{i-1,d},$$

where $\hat{\boldsymbol{a}}_j^*$ and $\hat{\mathbf{B}}_j^*$ denote the posterior mean of the coefficients for cluster j that $\boldsymbol{y}_{i,d}$ belongs to, we notice that the estimated relationship for cluster 3 in the second dimension is almost

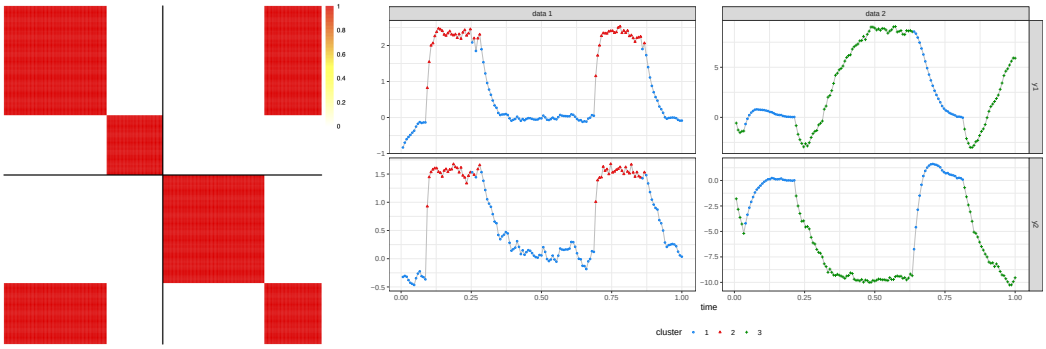


Figure 30: Left: Posterior similarity matrix. Diagonal blocks correspond to within-group PSM. Right: Plot of each dimension against time t for each data set, with observations colored by cluster labels.

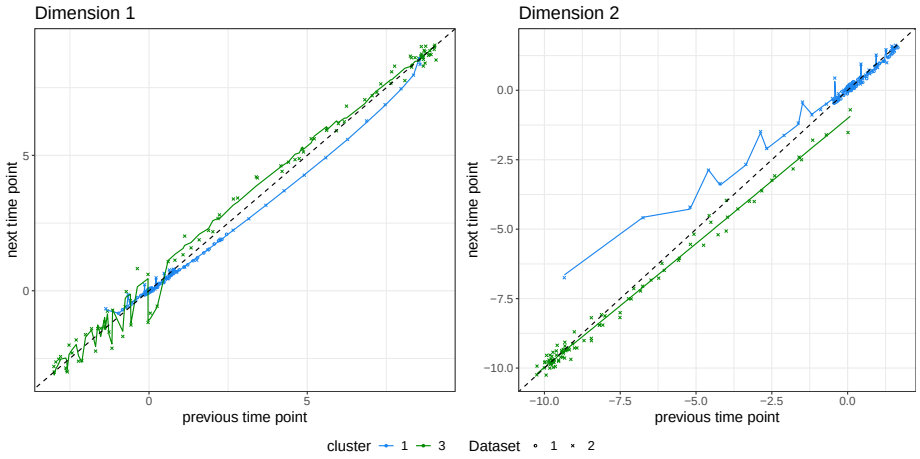


Figure 31: Plot of current time point against the previous time point for each dimension g, for clusters 1 and 3 in both data sets. The black dashed line denotes $y = x$. The solid colored line denotes the posterior estimated relationship between successive time frames for each cluster.

linear, as shown in the green solid line in Figure 31. This implies a dominant influence from the past observation in the same reduced dimension, which corresponds to the zero coefficient in $\mathbf{L}_3^*$ representing the impact of the first dimension on the second dimension.

For a comparison, we also fit a simple Gaussian mixture model (GMM) with an unconstrained covariance matrix[2]. However, GMM finds 11 clusters between 2 to 15 based on BIC, yielding a poor ARI of 0.3507 with the true clustering. Figure 32 shows that GMM groups observations into small blocks with similar observed values, rather than considering their relative dependence.

2. GMM is implemented using R package `mclust` (Scrucca et al., 2016).

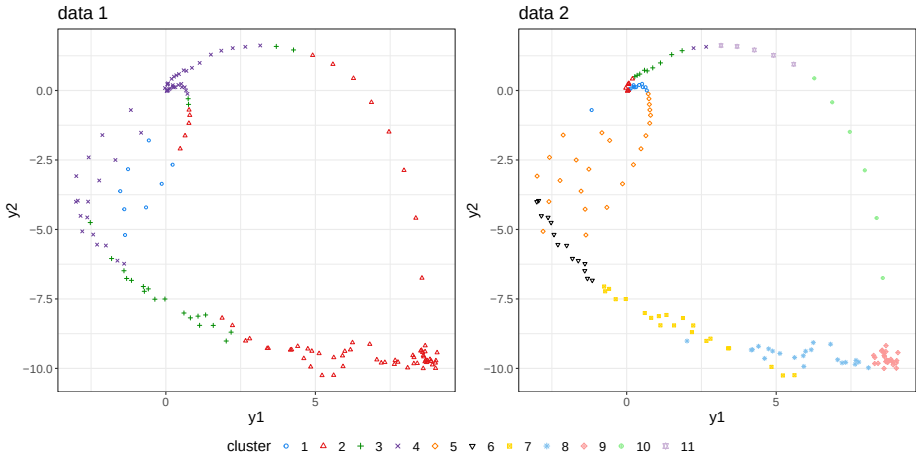


Figure 32: Pairwise scatterplots for two data sets, with observations colored by cluster membership from a simple Gaussian mixture model.

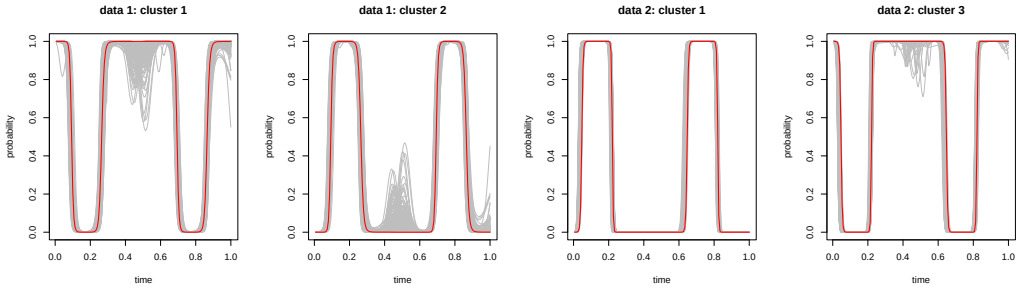


Figure 33: Posterior samples for time-dependent probabilities. The red solid line denotes the truth.

The true time-dependent probabilities are contained within the MCMC samples (Figure 33), with large uncertainty at time point around 0.5. In addition, similar to the Gaussian kernel, individual kernel parameters and $q_{j,d}^J$ still cannot be contained in the MCMC samples, suggesting weak identifiability.

For component-specific parameters, each element in the coefficient matrix $\mathbf{L}_j^*$ and covariance matrix $\mathbf{\Sigma}_j^*$ is investigated and all values are well estimated falling within the 99% HPD CIs (Figure 34 and Figure 35).

For posterior predictive checks to examine model fit, one replicate for observation i in the data set d is generated from

$$\boldsymbol{y}_{i,d}^{rep,(l)} \sim \mathrm{N}\left(\left(\mathbf{L}_j^{*(l)}\right)^T \boldsymbol{x}_{i,d}, \mathbf{\Sigma}_j^{*(l)}\right),$$

where $\boldsymbol{x}_{i,d} = (1, y_{i-1,1,d}, y_{i-1,2,d})^T$, and $\mathbf{L}_j^{*(l)}$ and $\mathbf{\Sigma}_j^{*(l)}$ denote the l-th posterior MCMC draw.

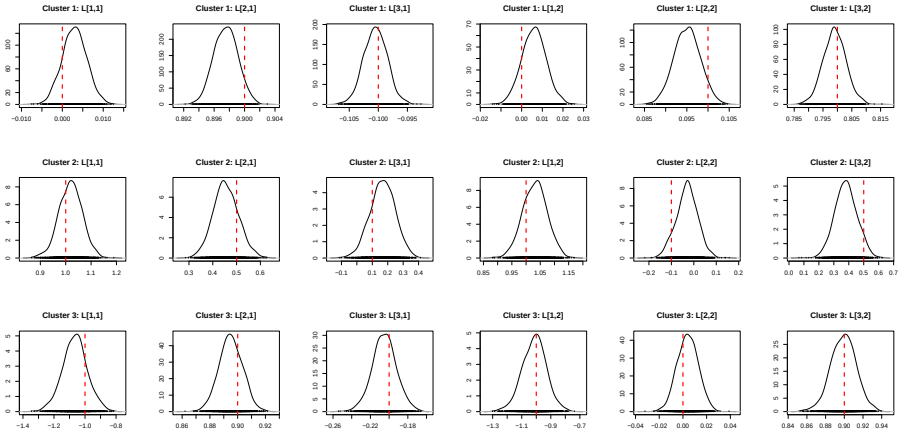


Figure 34: Density plots for each element in the coefficient matrix $\mathbf{L}_j^*$. The red dashed line denotes the truth.

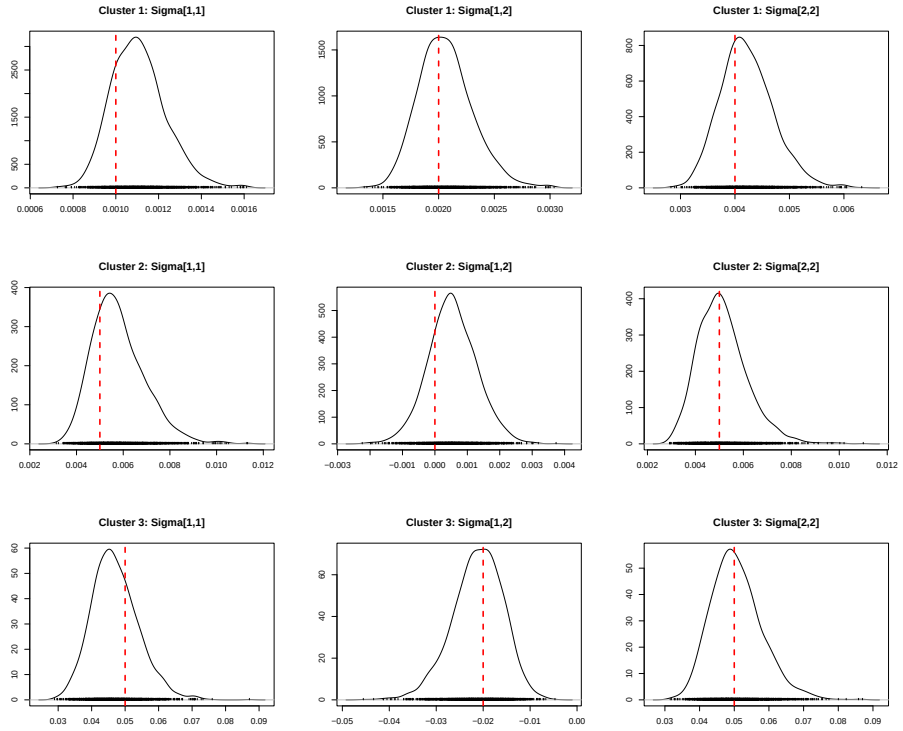


Figure 35: Density plots for each element from the upper triangular part of the covariance matrix $\mathbf{\Sigma}_j^*$. The red dashed line denotes the truth.

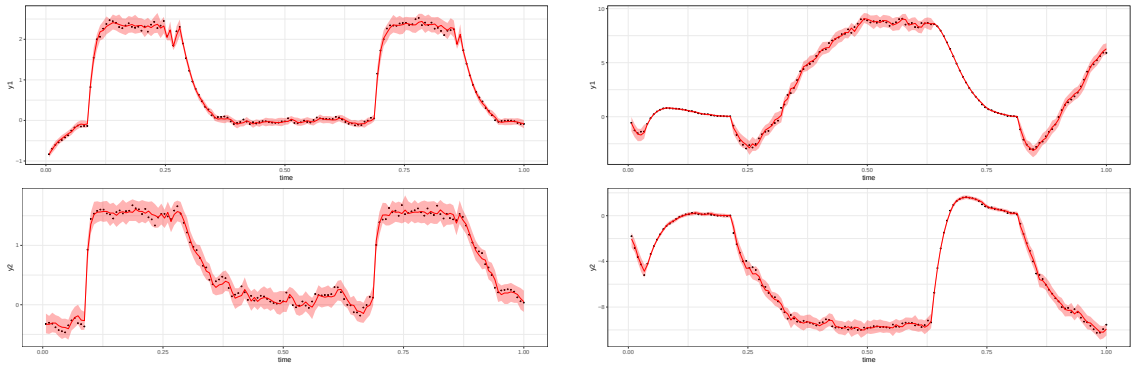


Figure 36: Posterior predictive checks based on 200 replicates. Red lines denote the posterior mean of the replicates, with 99% HPD CIs shown in the red area. Black points denote the observed data. Rows correspond to dimensions. Left: Group 1. Right: Group 2.

Conditional on the optimal clustering and observed data, we generate 200 replicated data sets from posterior predictive distributions, based on samples from the post-processing MCMC. Figure 36 shows that the replicated data closely resembles the actual data, thus supporting the model fit.

Appendix E. Additional Results for Pax6 Data

In this section, we present additional findings for clustering the Pax6 data using our C-HDP model. To infer the clustering, we perform consensus clustering (Coleman et al., 2022) (see Section C for a review), which runs large numbers of chains (100) with a small number of iterations (500) to better exploit the posterior distribution of the clustering. Figure 37 shows the decision for choosing tuning parameters in consensus clustering. The truncation level is $J = 30$. For the post-processing step with a fixed optimal clustering, the total number of MCMC iterations is 48000. A burn-in of 43000 iterations is then applied and with a thinning of 5 there are 1000 samples in total for posterior inference.

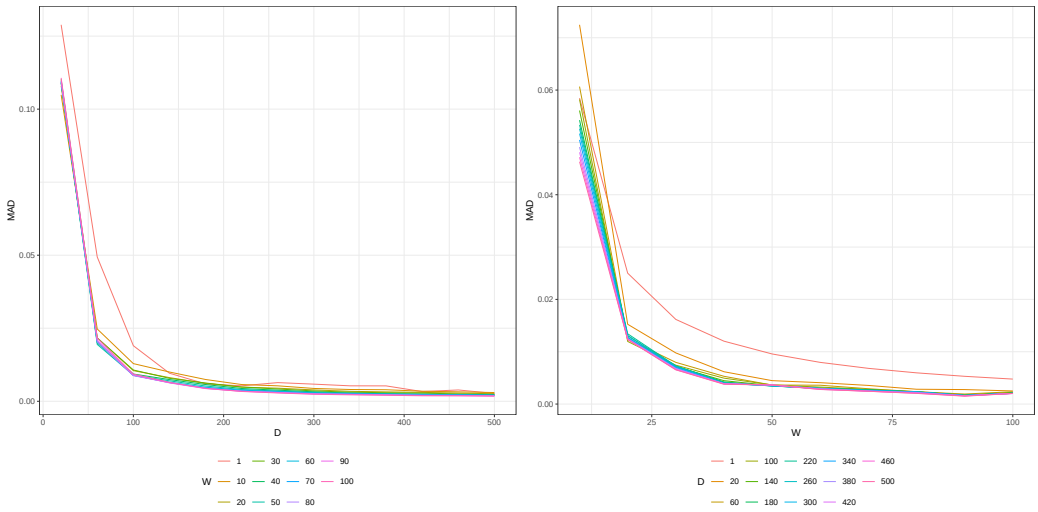


Figure 37: Choice of W and D in consensus clustering for Pax6.

E.1 General Results

By examining the posterior allocation probability for each cell, Figure 38 suggests there is some uncertainty in cell allocations at the boundary between clusters 3 and 9, with moderate allocation probabilities (between 0.25 and 0.75 (dark blue)).

Figure 39 shows the time-dependent probabilities for all 14 clusters from two groups. All the three under-represented clusters in HOM (3, 7, 9) are associated with larger latent time and high probabilities that are close to 1, while for the control group the probabilities are relatively lower (< 0.6).

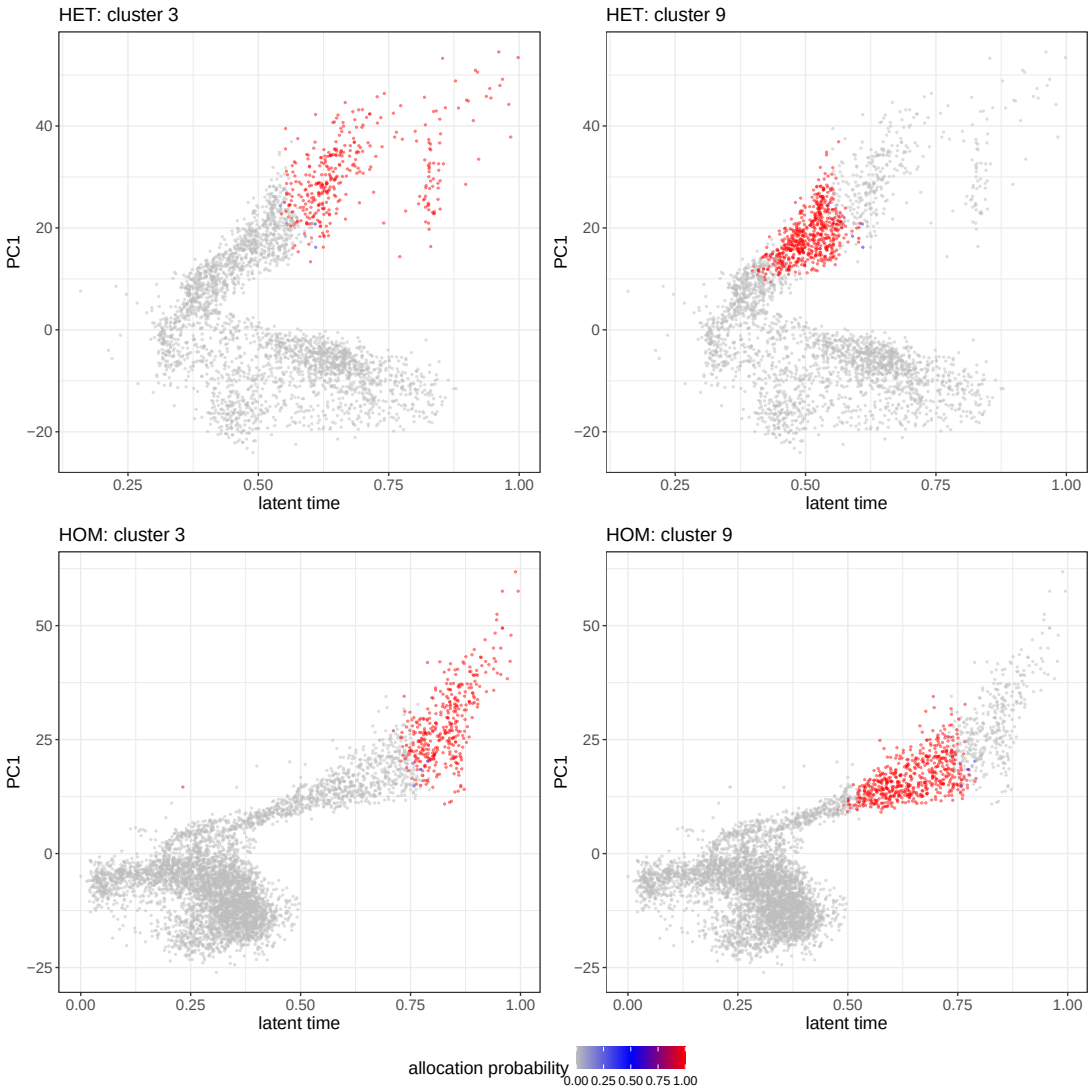


Figure 38: Plot of the first principal component against latent time for two example clusters in two experimental conditions. In each panel, cells are colored by the posterior allocation probability of belonging to the specific cluster indicated in the title.

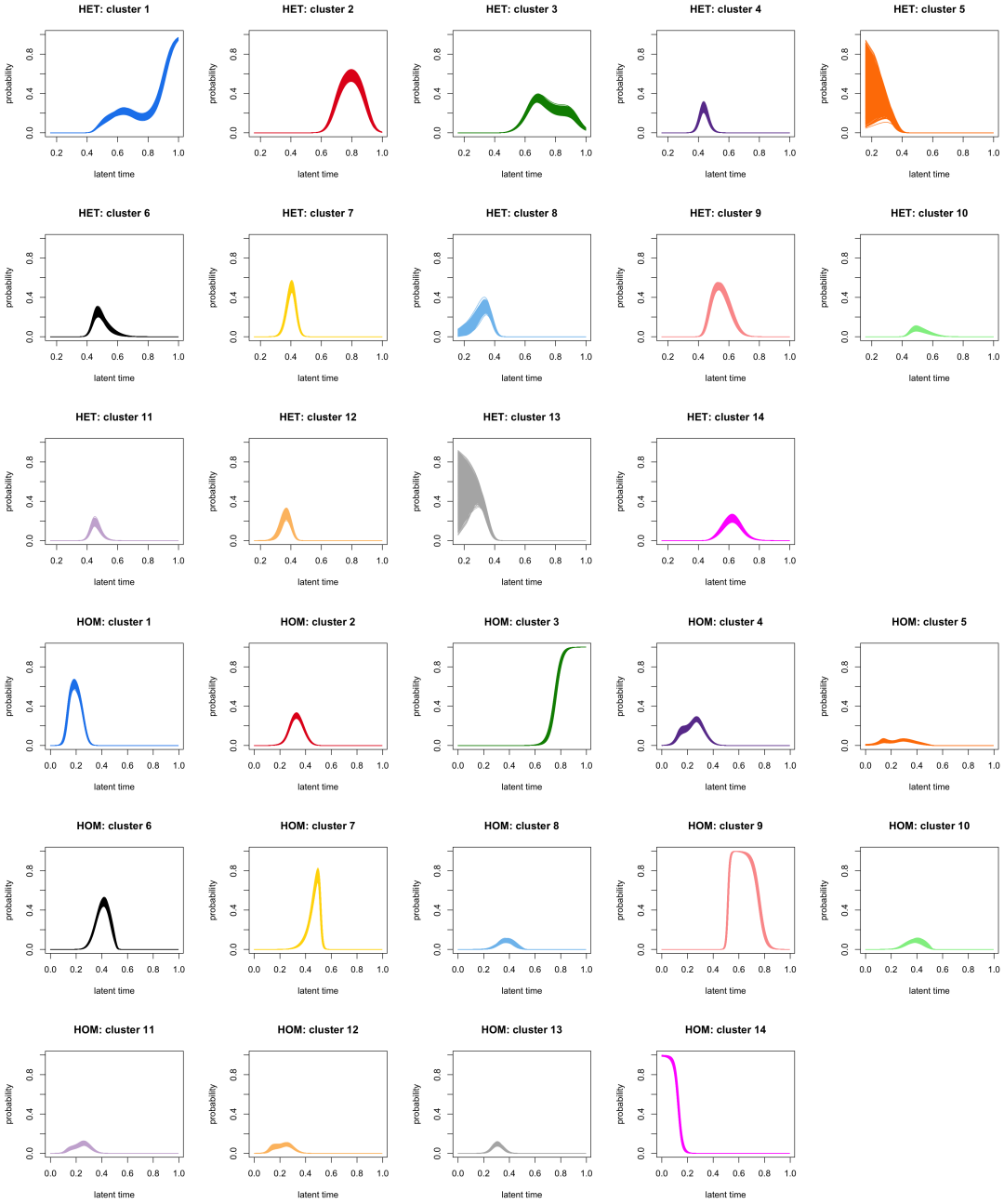


Figure 39: Time-dependent probabilities for each cluster in each group for Pax6. Top 3 rows show the results for HET, with bottom 3 rows for HOM. Clusters 3, 7, 9 are under-represented in HOM.

E.2 Latent Counts

Tang et al. (2020) and Liu et al. (2024) provide posterior mean of the latent counts given the allocation variables, capture efficiencies and unique parameters

$$\mathbb{E}(y^0_{c,g,d}|y_{c,g,d}, z_{c,d} = j, \beta_{c,d}, \mu^*_{j,g}, \phi^*_{j,g}) = y_{c,g,d}\frac{\mu^*_{j,g} + \phi^*_{j,g}}{\mu^*_{j,g}\beta_{c,d} + \phi^*_{j,g}} + \mu^*_{j,g}\frac{\phi^*_{j,g}(1-\beta_{c,d})}{\mu^*_{j,g}\beta_{c,d} + \phi^*_{j,g}},$$

which can be used to approximate the posterior mean of latent counts as

$$\mathbb{E}(y^0_{c,g,d}|\mathbf{Y}) \approx \frac{1}{L}\sum_{l=1}^{L}\mathbb{E}(y^0_{c,g,d}|y_{c,g,d}, z^{(l)}_{c,d} = j, \beta^{(l)}_{c,d}, \mu^{*(l)}_{j,g}, \phi^{*(l)}_{j,g}). \tag{16}$$

Figure 40 shows the t-SNE (Van der Maaten and Hinton, 2008) plot for the observed and estimated latent counts from Equation (16) using all 1000 samples from the post-processing step. From the observed counts, some clusters are already quite separated, such as the purple and dark green clusters. The separation is much more apparent in the latent counts.

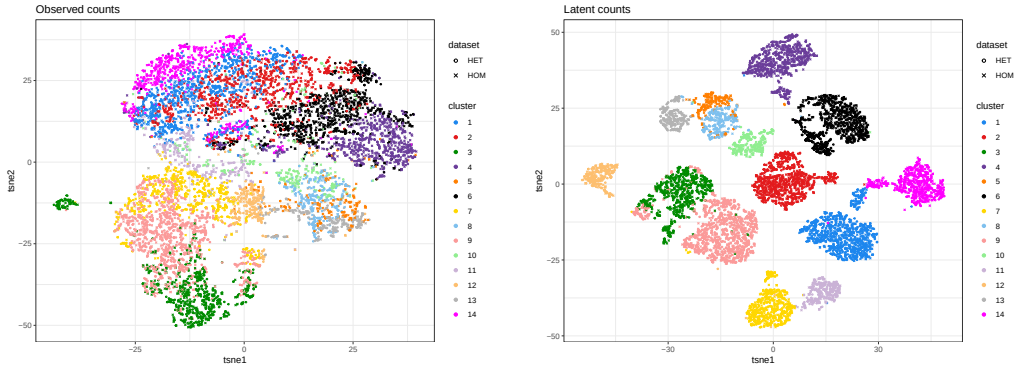


Figure 40: t-SNE plot for observed counts (left) and posterior mean of latent counts (right). Cells belonging to different clusters are shown in different colors. Different symbols indicate different experimental conditions.

Figure 41 and Figure 42 show the estimated latent counts and observed counts for each cell on the log scale after adding a pseudo-count of 1. Global differentially expressed genes that distinguish between different clusters (see Liu et al., 2024 for details) show different patterns across clusters, whilst within each cluster, the pattern is similar across groups.

E.3 Posterior Predictive Checks

The posterior predictive checks are conducted following Liu et al. (2024) by comparing replicated data generated from posterior predictive distributions to true data. Figure 43 and Figure 44 demonstrate that a single replicated data set exhibits similar relationships between pairwise statistics to the true data. The pointwise differences in statistics are nearly negligible, implying that the replicated data is consistent with the observed data.

For multiple replicates, Figure 45 shows that the estimated kernel of key statistics is similar between the simulated 200 data sets and the true observed data. Therefore, there is no strong disagreement between the model and the data.

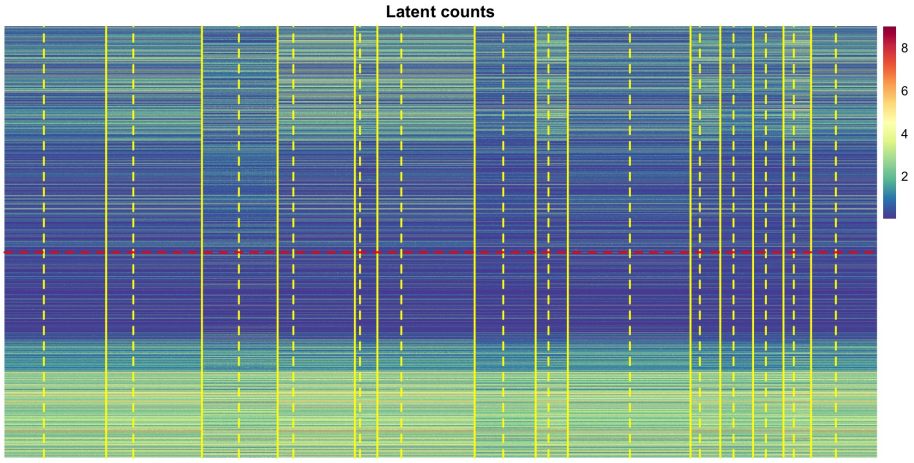


Figure 41: Heatmap for the latent counts on the log scale after adding a pseudo-count of 1. Each row represents a gene and each column represents a cell. Genes above the red dashed lines are global differentially expressed genes (Liu et al., 2024). Yellow solid lines separate clusters. Yellow dashed lines separate cells from different groups within each cluster.

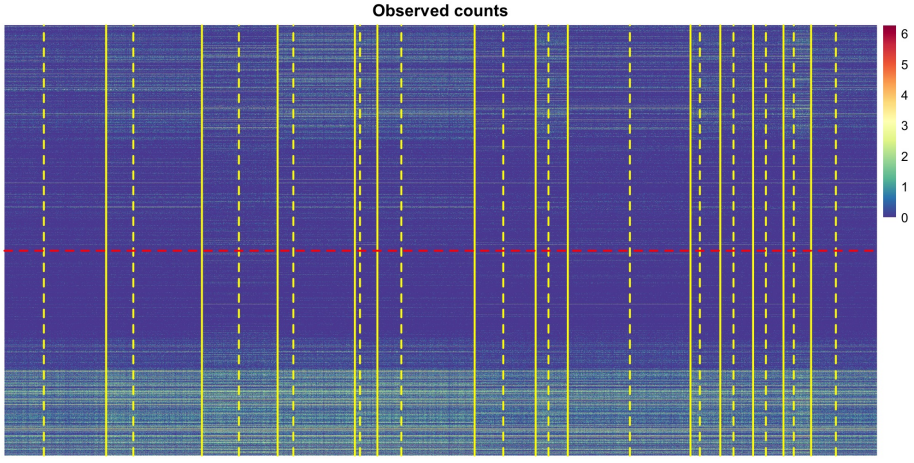


Figure 42: Heatmap for the observed counts on the log scale after adding a pseudo-count of 1. Each row represents a gene and each column represents a cell. Genes above the red dashed lines are global differentially expressed genes (Liu et al., 2024). Yellow solid lines separate clusters. Yellow dashed lines separate cells from different groups within each cluster.

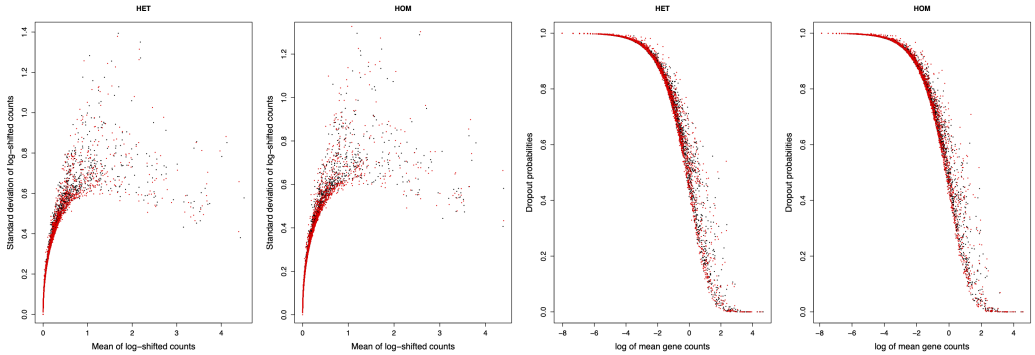


Figure 43: Posterior predictive checks with one single replicated data set. Left two plots show the relationship between mean and standard deviation of log-shifted counts in true (red) and replicated data (black) for HET and HOM. Right two plots show the relationship between log of mean counts and dropout probabilities.

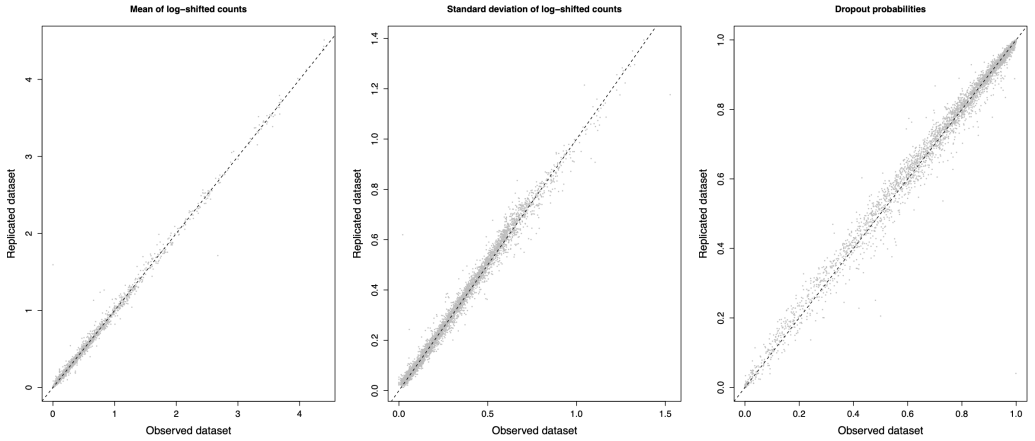


Figure 44: Posterior predictive checks with one single replicated data set. Each panel shows pointwise differences in a statistic between true and replicated data. The black dashed line corresponds to $y = x$.

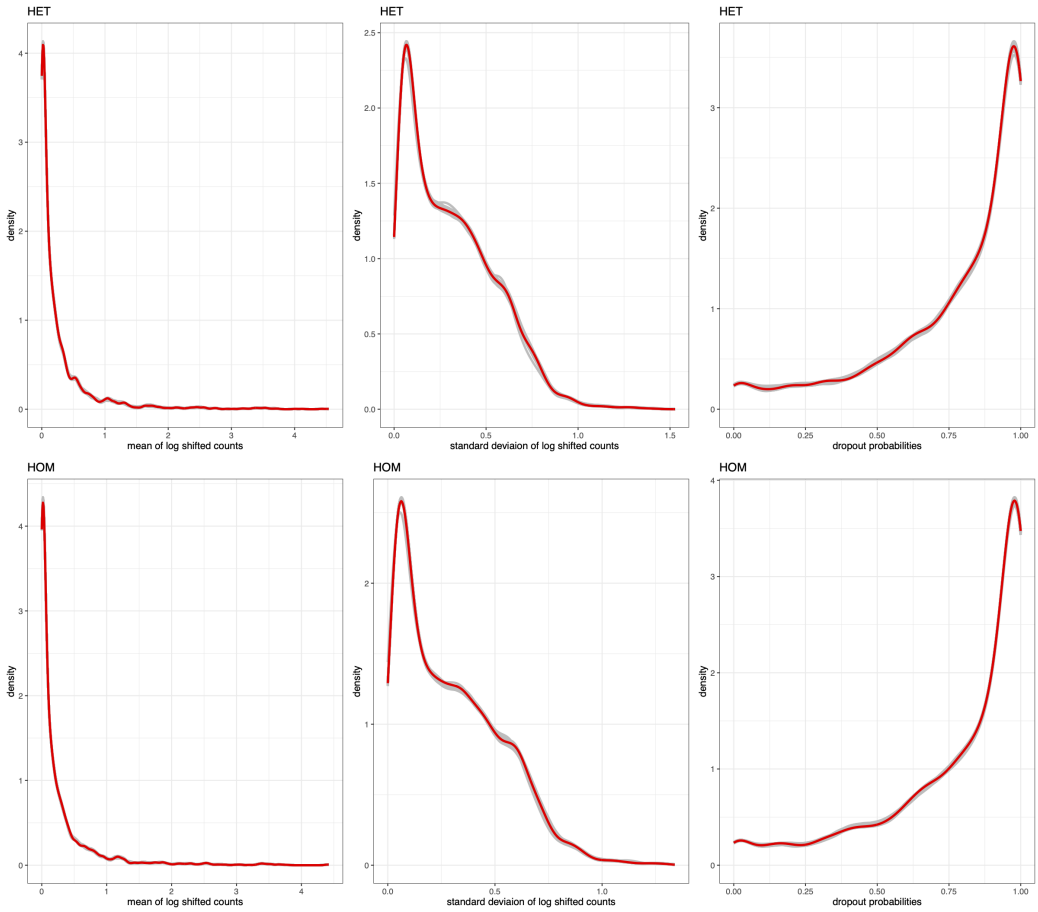


Figure 45: Posterior predictive checks with multiple replicates for HET (top) and HOM (bottom). Each panel shows the kernel density estimation of one statistic, with replicated and true data in grey and red, respectively. Left to right: mean of log shifted counts, standard deviation of log shifted counts and dropout probabilities.

Appendix F. Additional Results for Calcium Imaging Data

In this section we provide additional results for clustering the calcium imaging data. To infer the clustering, two chains of length 10000 have been run with a truncation level of $J = 25$. With a burnin of 6000 and a thinning of 2 in each chain, 4000 posterior samples from both chains are used to estimate an optimal clustering based on VI. For the post-processing step, we run one chain of length 8000 and then apply a burnin of 4000 and thinning of 2, leading to 2000 samples.

F.1 General Results

Figure 46 displays the time-series plot for all clusters. It is noticed that observations from the same cluster do share similar dependence on the past observed time point.

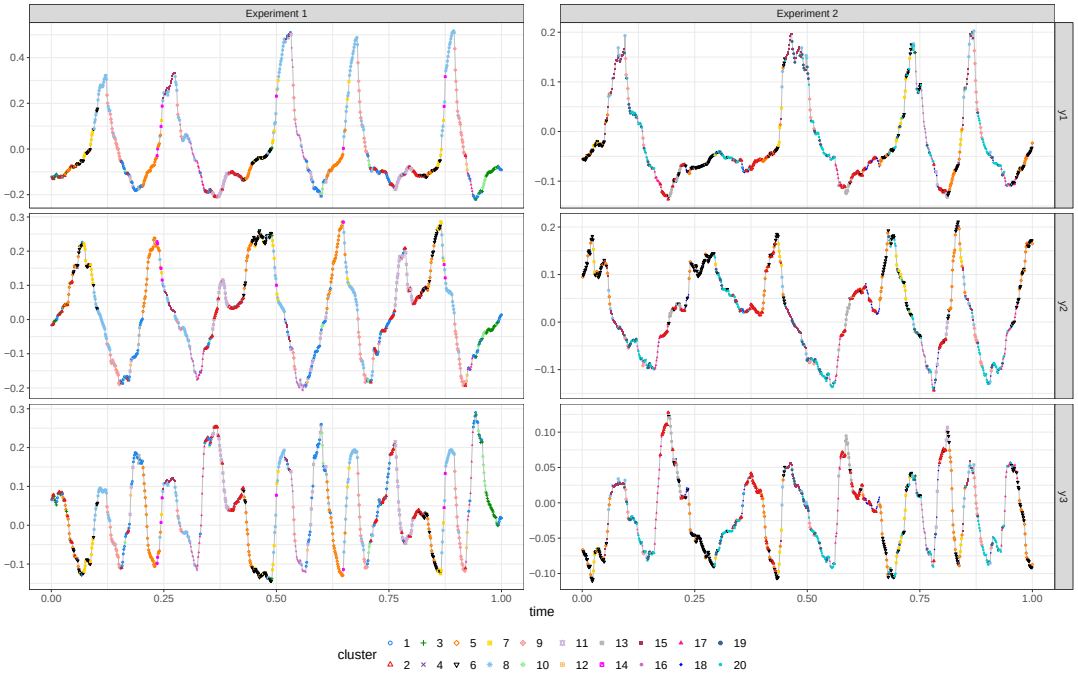
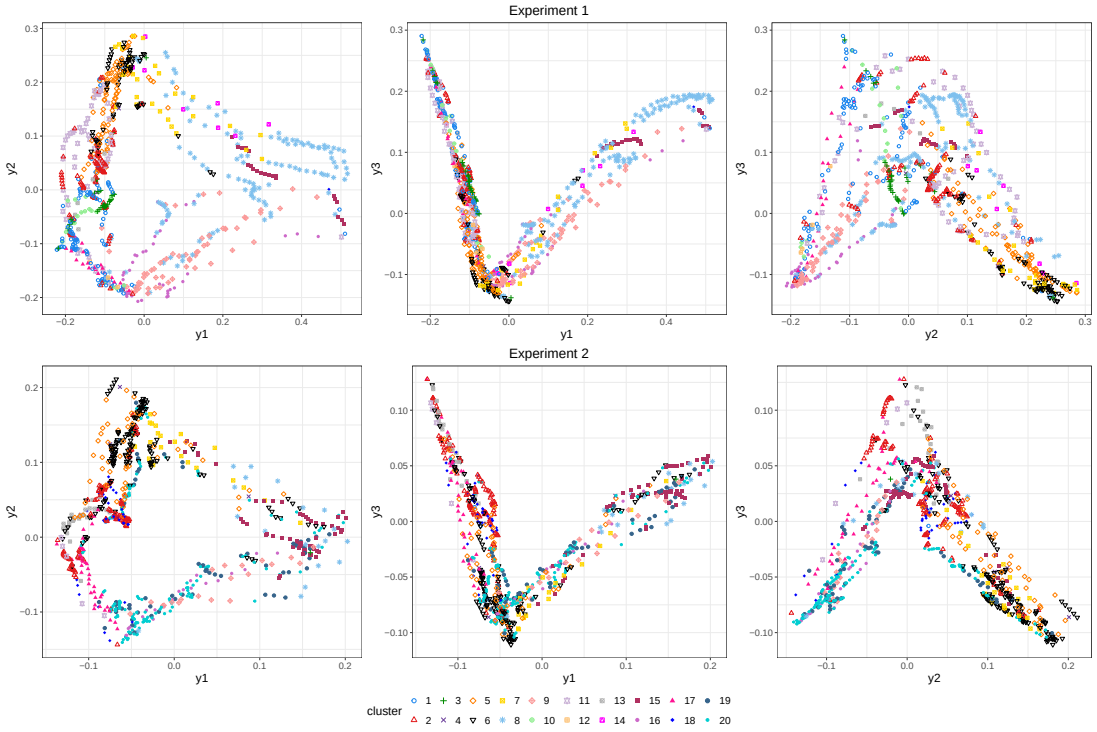


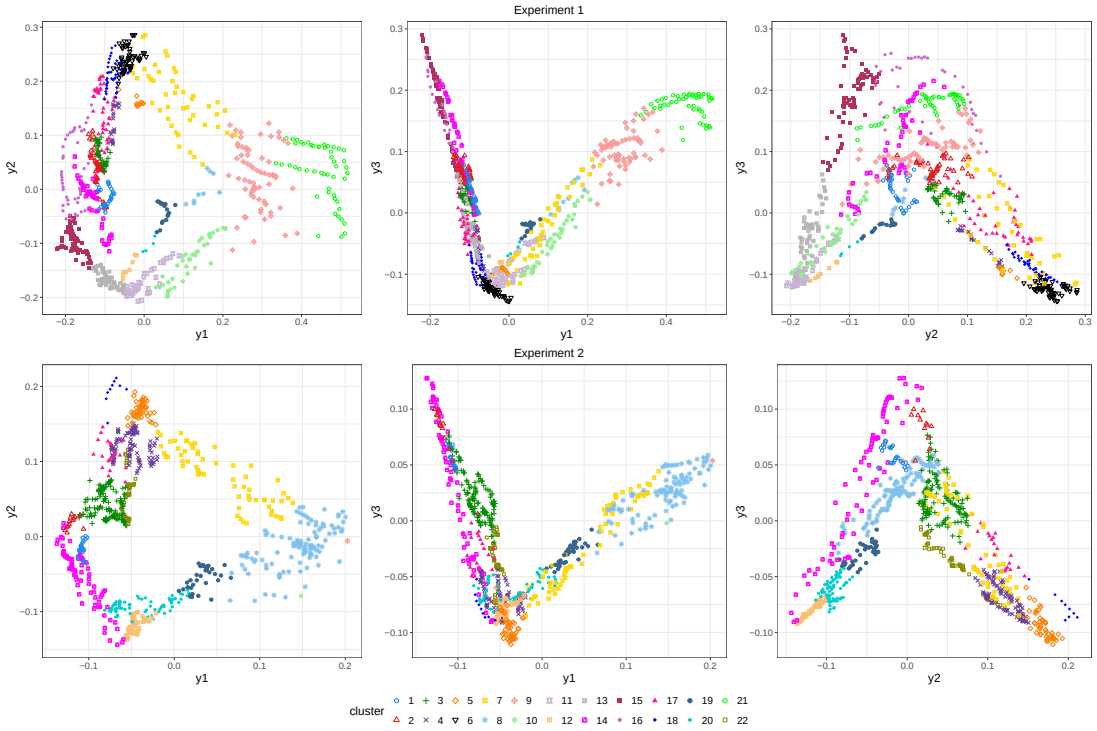
Figure 46: Plot of each dimension against time t, with points colored by cluster labels. Left: Experiment 1. Right: Experiment 2.

Figure 47 shows the complete pairwise scatterplots from C-HDP and simple Gaussian mixture model. In C-HDP, the identified clusters shared between experiments tend to accumulate at similar positions in the lower-dimensional embeddings, whereas GMM identifies fewer shared clusters and just cuts time frames into groups of similar values.

To investigate the differences between neural activity patterns, we visualize the posterior mean of the coefficient matrix $\mathbf{B}_j^*$ for each cluster (Figure 48). Within each cluster, the contribution to each dimension is dominated by the corresponding dimension from the past time frame (red diagonal grids), with the strongest effect observed in cluster 14 (first dimension). In contrast, interactions between different lower-dimensional embeddings are



(a) C-HDP



(b) GMM

Figure 47: Pairwise scatterplots for two experiments, with observations colored by cluster membership from C-HDP (a) and GMM (b).

generally weaker and tend to show more negative contributions. Specifically, cluster 13 exhibits the strongest negative between-dimension influence, followed by cluster 5.

Figure 49 shows posterior samples for time-dependent probabilities across all clusters, with probabilities close to zero for small-size clusters. In addition, clusters may have different periodicities, e.g. clusters 2, 6 and 9, suggesting varying frequencies associated with pattern across experiments. Although cluster 5 seems to have a similar periodicity, the probabilities are higher in the first experiment.

F.2 Predictions

Forecasting is one of the primary goals in time-series modelling, which allows us to understand the encoded neural activity in the near future. We predict the neural activity for 20 future times points, using the same time increment as the observed data. In particular, time-dependent probabilities are first used to generate allocation variables $z_{i,d}$, based on which future observed values are simulated. The true future neural activity are found to be generally covered by the samples of the predictions (Figure 50), and the uncertainty increases fast as predictions are made further away.

In addition to neural activity, we can also estimate the covariate-dependent probability for future time points. Unlike Gaussian kernels where probabilities at future time points are almost around zero due to absence of data, for periodic kernel, the repetitive pattern is similar for either interpolation and extrapolation, with similar uncertainty (Figure 51).

F.3 Posterior Predictive Checks

Conditional on the optimal clustering and observed data, we generate 200 replicated data sets using MCMC samples from the post-processing step. In particular, one replicate for observation i in experiment d is given by

$$\boldsymbol{y}_{i,d}^{rep,(l)} \sim \mathrm{N}\left(\left(\mathbf{L}_j^{*(l)}\right)^T \boldsymbol{x}_{i,d}, \boldsymbol{\Sigma}_j^{*(l)}\right),$$

where $\boldsymbol{x}_{i,d} = (1, y_{i-1,1,d}, y_{i-1,2,d}, y_{i-1,3,d})^T$, $\mathbf{L}_j^{*(l)}$ and $\boldsymbol{\Sigma}_j^{*(l)}$ denote the l-th posterior draw.

From Figure 52, the replicated data closely resembles the true data sets. The neural activity in the true data is similar to the posterior mean of the replicates with short 99% CIs, indicating no strong disagreement between the data and the model.

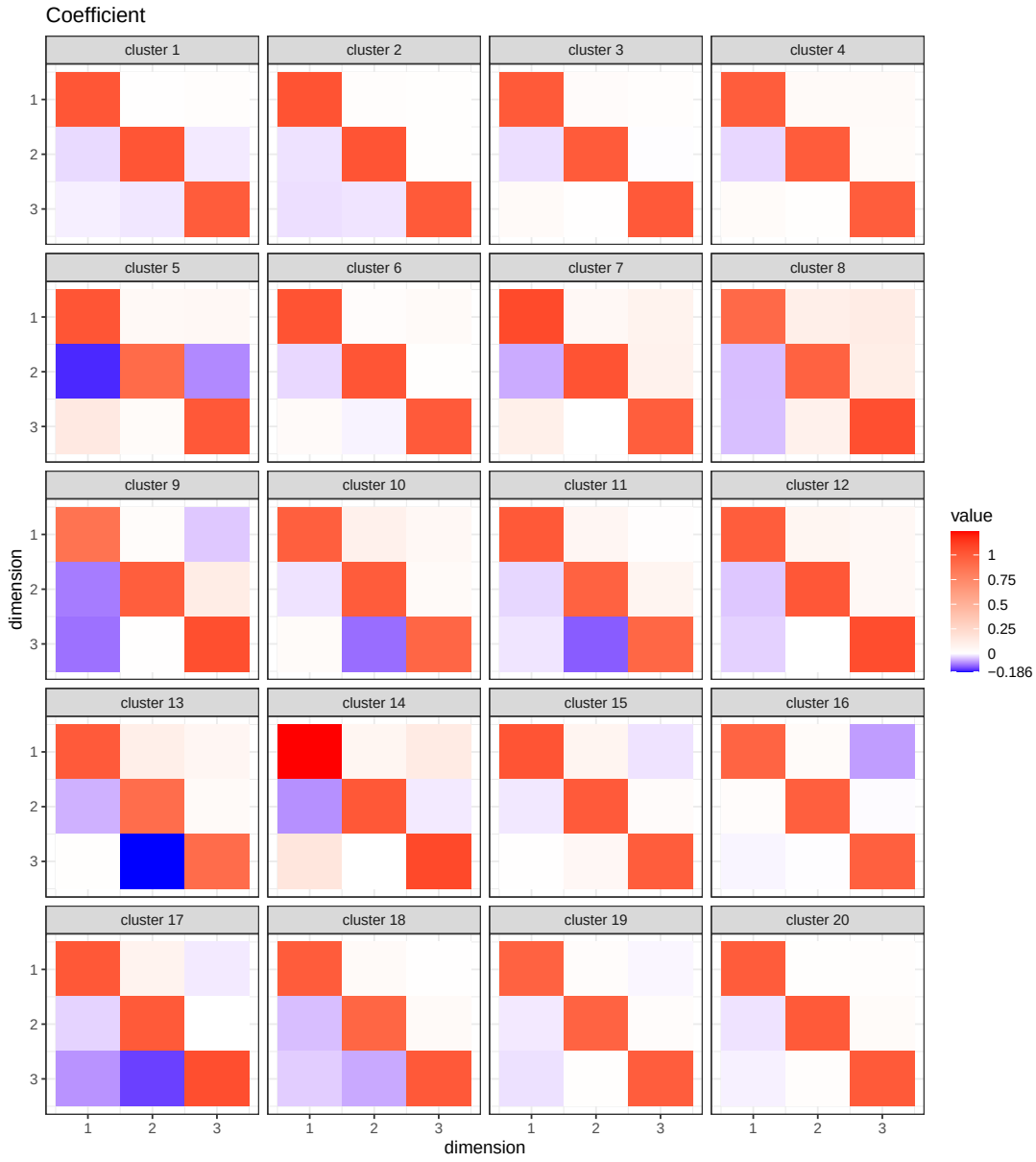
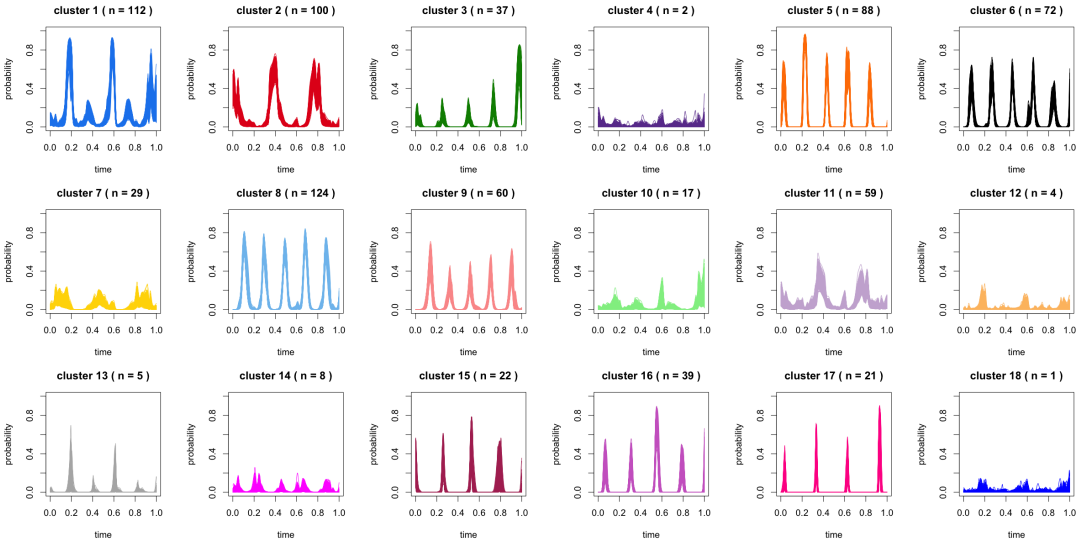
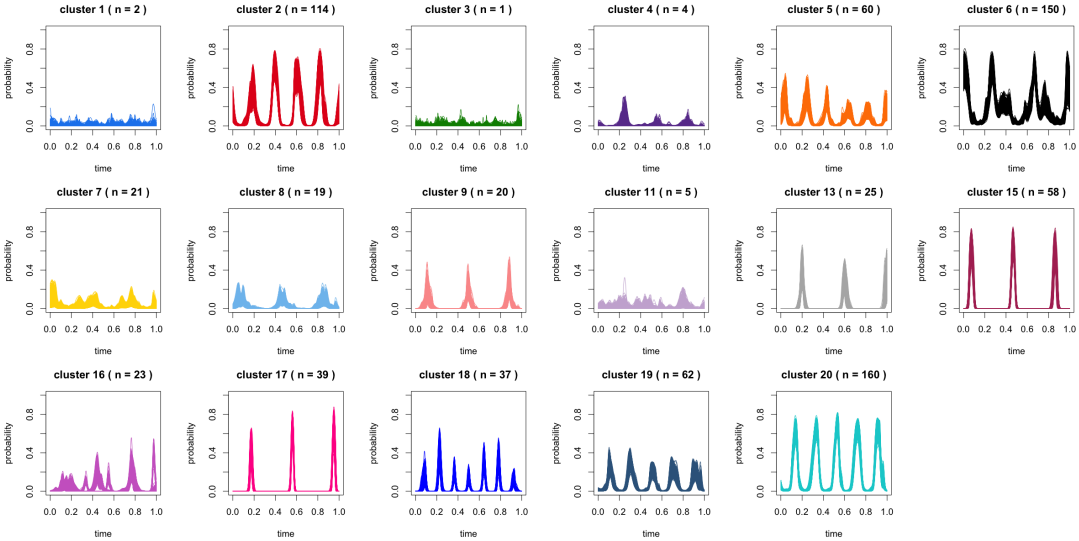


Figure 48: Posterior mean of the coefficient matrix $\mathbf{B}_j^*$ for each cluster (excluding the intercept). Values close to 0 are shown in white, with red for positive coefficients and blue for negative coefficients.



(a) Experiment 1



(b) Experiment 2

Figure 49: Time-dependent probabilities for each cluster in each experiment in the calcium imaging data. Only non-empty components are plotted. Cluster size is indicated in the title.

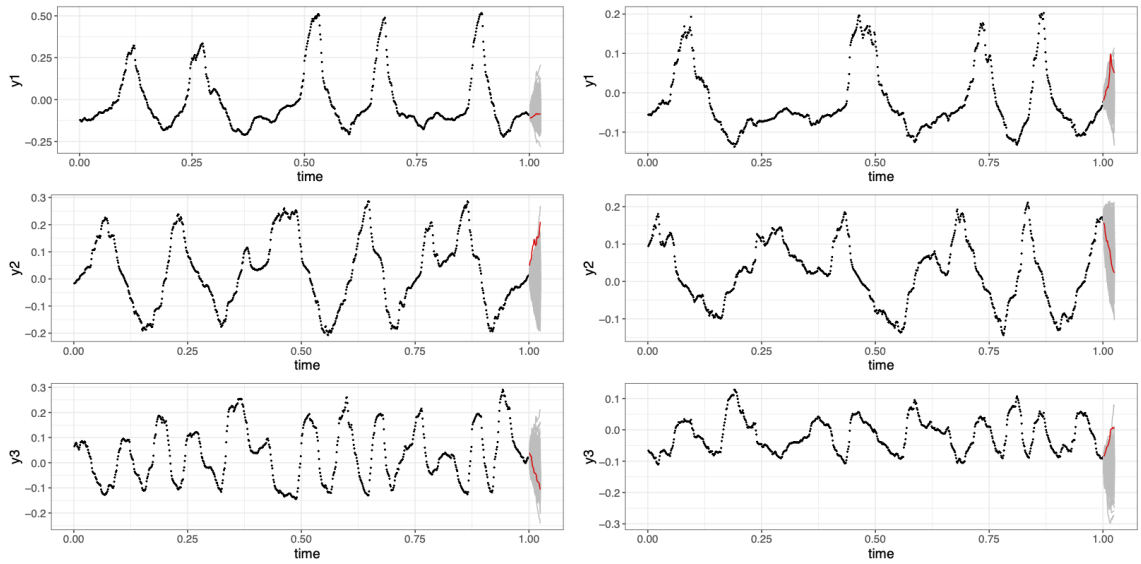


Figure 50: Predicting future trends. Red solid line denotes true future observations, with posterior predictive samples shown in grey. Black points denote the observed data used for model fitting. Left: Experiment 1. Right: Experiment 2.

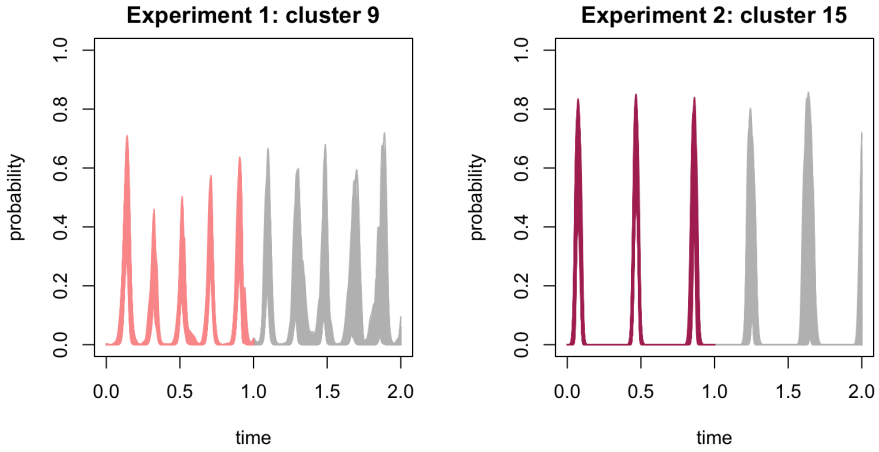


Figure 51: Time-dependent probabilities for future time. Grey area shows predictions.

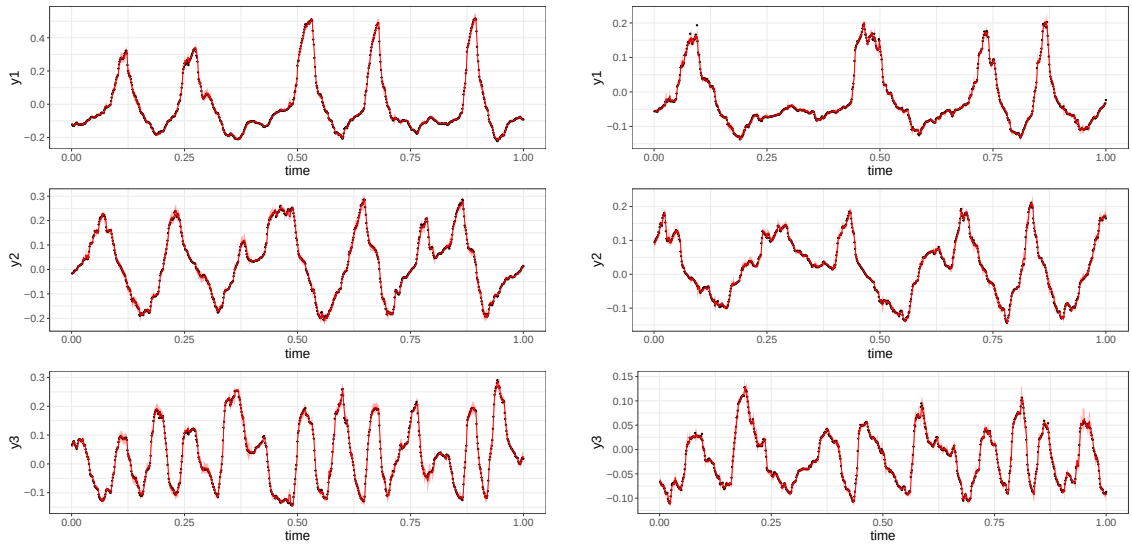


Figure 52: Posterior predictive checks based on 200 replicates. The red line denotes the posterior mean of the replicates, with 99% HPD CIs shown in the red area. Black points denote the observed data. Left: Experiment 1. Right: Experiment 2.

References

I. Antoniano-Villalobos, S. Wade, and S. G. Walker. A Bayesian nonparametric regression model with normalized weights: A study of hippocampal atrophy in Alzheimer's disease. *Journal of the American Statistical Association*, 109(506):477–490, 2014.

R. Argiento, A. Cremaschi, and M. Vannucci. Hierarchical normalized completely random measures to cluster grouped data. *Journal of the American Statistical Association*, 2020.

M. Beraha, A. Guglielmi, and F. A. Quintana. The semi-hierarchical Dirichlet process and its application to clustering homogeneous distributions. *Bayesian Analysis*, 16(4): 1187–1219, 2021.

V. Bergen, M. Lange, S. Peidli, F. A. Wolf, and F. J. Theis. Generalizing RNA velocity to transient cell states through dynamical modeling. *Nature Biotechnology*, 38(12):1408–1414, 2020.

C. M. Bishop. *Pattern Recognition and Machine Learning*. Information Science and Statistics. Springer, New York, 2006.

D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

P. Brennecke, S. Anders, J. K. Kim, A. A. Kołodziejczyk, X. Zhang, V. Proserpio, B. Baying, V. Benes, S. A. Teichmann, J. C. Marioni, et al. Accounting for technical noise in single-cell RNA-seq experiments. *Nature Methods*, 10(11):1093–1095, 2013.

I. M. Caballero, M. N. Manuel, M. Molinek, I. Quintana-Urzainqui, D. Mi, T. Shimogori, and D. J. Price. Cell-autonomous repression of Shh by transcription factor Pax6 regulates diencephalic patterning by controlling the central diencephalic organizer. *Cell Reports*, 8 (5):1405–1418, 2014.

F. Camerlenghi, D. B. Dunson, A. Lijoi, I. Prünster, and A. Rodríguez. Latent nested nonparametric priors (with discussion). *Bayesian Analysis*, 14(4):1303, 2019.

M. Catalano and H. Lavenant. Hierarchical integral probability metrics: A distance on random probability measures with low sample complexity. In *International Conference on Machine Learning*, pages 5841–5861. PMLR, 2024.

N. K. Chandra, A. Canale, and D. B. Dunson. Escaping the curse of dimensionality in Bayesian model-based clustering. *Journal of Machine Learning Research*, 24(144):1–42, 2023.

S. Coleman, P. D. Kirk, and C. Wallace. Consensus clustering for Bayesian mixture models. *BMC Bioinformatics*, 23(1):290, 2022.

A. M. Dai and A. J. Storkey. The supervised hierarchical Dirichlet process. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(2):243–255, 2014.

L. D'Angelo, A. Canale, Z. Yu, and M. Guindani. Bayesian nonparametric analysis for the detection of spikes in noisy calcium imaging data. *Biometrics*, 79(2):1370–1382, 2023.

L. K. Davis, K. Meyer, D. Rudd, A. Librant, E. Epping, V. Sheffield, and T. Wassink. Pax6 3' deletion results in aniridia, autism and mental retardation. *Human Genetics*, 123:371–378, 2008.

P. De Blasi, S. Favaro, A. Lijoi, R. H. Mena, I. Prünster, and M. Ruggiero. Are Gibbs-type priors the most natural generalization of the Dirichlet process? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(2):212–229, 2013.

M. De Iorio, P. Müller, G. L. Rosner, and S. N. MacEachern. An ANOVA model for dependent random measures. *Journal of the American Statistical Association*, 99(465): 205–215, 2004.

M. De Iorio, W. Johnson, P. Müller, and G. Rosner. Bayesian nonparametric non-proportional hazards survival modelling. *Biometrics*, 65:762–771, 2009.

F. Denti, F. Camerlenghi, M. Guindani, and A. Mira. A common atoms model for the Bayesian nonparametric analysis of nested data. *Journal of the American Statistical Association*, 118(541):405–416, 2023.

A. Diana, E. Matechou, J. Griffin, and A. Johnston. A hierarchical dependent Dirichlet process prior for modelling bird migration patterns in the UK. *The Annals of Applied Statistics*, 14(1):473–493, 2020.

D. Dunson and J. Park. Kernel stick-breaking processes. *Biometrika*, 95:307–323, 2008.

N. Eling, A. C. Richard, S. Richardson, J. C. Marioni, and C. A. Vallejos. Correcting the mean-variance dependency for differential variability testing using single-cell RNA sequencing data. *Cell Systems*, 7(3):284–294, 2018.

G. Estivill-Torrus, H. Pearson, V. van Heyningen, D. J. Price, and P. Rashbass. Pax6 is required to regulate the cell cycle and the rate of progression from symmetrical to asymmetrical division in mammalian cortical progenitors. *Development*, 129(2):455–466, 01 2002.

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

E. Fong, S. Lyddon, and C. Holmes. Scalable nonparametric sampling from multimodal posteriors with the posterior bootstrap. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pages 1952–1962. PMLR, 2019.

N. Foti and S. Williamson. Slice sampling normalized kernel-weighted completely random measure mixture models. In *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.

C. Fraley and A. E. Raftery. Bayesian regularization for normal mixture estimation and model-based clustering. *Journal of Classification*, 24(2):155–181, 2007.

S. Frühwirth-Schnatter and S. Pyne. Bayesian inference for finite mixtures of univariate and multivariate skew-normal and skew-t distributions. *Biostatistics*, 11(2):317–336, 2010.

S. Fruhwirth-Schnatter, G. Celeux, and C. P. Robert. *Handbook of Mixture Analysis*. CRC Press, 2019.

A. Gelman, X.-L. Meng, and H. Stern. Posterior predictive assessment of model fitness via realized discrepancies. *Statistica Sinica*, pages 733–760, 1996.

S. Ghosal and A. van der Vaart. Entropies and rates of convergence for maximum likelihood and Bayes estimation for mixtures of normal densities. *The Annals of Statistics*, 29:1233–1263, 2001.

S. Ghosal and A. van der Vaart. *Fundamentals of Nonparametric Bayesian Inference*, volume 44. Cambridge University Press, 2017.

S. Ghosal, J. Ghosh, and R. Ramamoorthi. Posterior consistency of Dirichlet mixtures in density estimation. *The Annals of Statistics*, 27:143–158, 1999.

J. E. Griffin and M. F. Steel. Stick-breaking autoregressive processes. *Journal of Econometrics*, 162(2):383–396, 2011.

J. E. Griffin and D. A. Stephens. Advances in Markov chain Monte Carlo. In *Bayesian Theory and Applications*. Oxford University Press, Oxford, 2013.

M. Götz, A. Stoykova, and P. Gruss. Pax6 controls radial glia differentiation in the cerebral cortex. *Neuron*, 21(5):1031–1044, 1998. ISSN 0896-6273.

N. L. Hjort. Nonparametric Bayes estimators based on beta processes in models for life history data. *The Annals of Statistics*, pages 1259–1294, 1990.

P. Hoffman. Seurat - guided clustering tutorial, 2023. URL `https://satijalab.org/seurat/articles/pbmc3k_tutorial.html`.

H. Ishwaran and L. F. James. Gibbs sampling methods for stick-breaking priors. *Journal of the American Statistical Association*, 96(453):161–173, 2001.

H. Ishwaran and M. Zarepour. Exact and approximate sum representations for the Dirichlet process. *Canadian Journal of Statistics*, 30(2):269–283, 2002.

Z. Ji and H. Ji. TSCAN: Pseudo-time reconstruction and evaluation in single-cell RNA-seq analysis. *Nucleic Acids Research*, 44(13):e117–e117, 2016.

T. Jordan, I. Hanson, D. Zaletayev, S. Hodgson, J. Prosser, A. Seawright, N. Hastie, and V. van Heyningen. The human PAX6 gene is mutated in two patients with aniridia. *Nature Genetics*, 1(5):328–332, 1992.

D. Karlis and E. Xekalaki. Mixed Poisson distributions. *International Statistical Review*, 73:35–58, 2005.

T. Kikkawa, C. R. Casingal, S. H. Chun, H. Shinohara, K. Hiraoka, and N. Osumi. The role of Pax6 in brain development and its impact on pathogenesis of autism spectrum disorder. *Brain Research*, 1705:95–103, 2019.

D. Kim and A. Oh. Hierarchical Dirichlet scaling process. In *Proceedings of the 31st International Conference on Machine Learning*, pages 973–981. PMLR, 2014.

J. F. Kingman. Random discrete distributions. *Journal of the Royal Statistical Society: Series B (Methodological)*, 37(1):1–15, 1975.

V. Y. Kiselev, K. Kirschner, M. T. Schaub, T. Andrews, A. Yiu, T. Chandra, K. N. Natarajan, W. Reik, M. Barahona, A. R. Green, et al. SC3: consensus clustering of single-cell RNA-seq data. *Nature Methods*, 14(5):483–486, 2017.

M. Krnjajić, A. Kottas, and D. Draper. Parametric and nonparametric Bayesian model specification: A case study involving models for count data. *Computational Statistics & Data Analysis*, 52(4):2110–2128, 2008.

G. La Manno et al. RNA velocity of single cells. *Nature*, 560:494–498, 2018.

D. Lähnemann, J. Köster, E. Szczurek, D. J. McCarthy, S. C. Hicks, M. D. Robinson, C. A. Vallejos, K. R. Campbell, N. Beerenwinkel, A. Mahfouz, et al. Eleven grand challenges in single-cell data science. *Genome Biology*, 21:1–35, 2020.

S. Lee and G. J. McLachlan. Finite mixtures of multivariate skew t-distributions: some recent and new results. *Statistics and Computing*, 24(2):181–202, 2014.

A. Lijoi and I. Prünster. Models beyond the Dirichlet process. In *Bayesian Nonparametrics*, pages 80–136, Cambridge, UK, 2011. Cambridge University Press.

A. Lijoi, I. Prünster, and G. Rebaudo. Flexible clustering via hidden hierarchical Dirichlet priors. *Scandinavian Journal of Statistics*, feb 2022. doi: 10.1111/sjos.12578. URL `https://doi.org/10.1111%2Fsjos.12578`.

P. Lin, M. Troup, and J. W. Ho. CIDR: Ultrafast and accurate clustering through imputation for single-cell RNA-seq data. *Genome Biology*, 18:1–11, 2017.

J. Liu, S. Wade, and N. Bochkina. Shared Differential Clustering across Single-cell RNA Sequencing Datasets with the Hierarchical Dirichlet Process. *Econometrics and Statistics*, 2024. ISSN 2452-3062.

S. Liverani, D. I. Hastie, L. Azizi, M. Papathomas, and S. Richardson. PReMiuM: An R package for profile regression mixture models using Dirichlet processes. *Journal of Statistical Software*, 64:1–30, 2015.

A. Y. Lo. On a class of Bayesian nonparametric estimates: I. Density estimates. *The Annals of Statistics*, pages 351–357, 1984.

S. N. MacEachern. Dependent nonparametric processes. In *ASA Proceedings of the Section on Bayesian Statistical Science*, pages 50–55, 1999.

M. Manuel, K. B. Tan, Z. Kozic, M. Molinek, T. S. Marcos, M. F. A. Razak, D. Dobolyi, R. Dobie, B. E. P. Henderson, N. C. Henderson, W. K. Chan, M. I. Daw, J. O. Mason, and D. J. Price. Pax6 limits the competence of developing cerebral cortical cells to respond to inductive intercellular signals. *PLoS Biology*, 20(9):1–54, 09 2022.

E. A. Mukamel, A. Nimmerjahn, and M. J. Schnitzer. Automated analysis of cellular signals from large-scale calcium imaging data. *Neuron*, 63(6):747–760, 2009.

J. O'Keefe and J. Dostrovsky. The hippocampus as a spatial map: Preliminary evidence from unit activity in the freely-moving rat. *Brain Research*, 34:171–175, 1971.

T. Pan, W. Shen, C. P. Davis-Stober, and G. Hu. A Bayesian nonparametric approach for handling item and examinee heterogeneity in assessment data. *British Journal of Mathematical and Statistical Psychology*, 77(1):196–211, 2024.

F. A. Quintana, P. Müller, A. Jara, and S. N. MacEachern. The dependent Dirichlet process and related models. *Statistical Science*, 37(1):24–41, 2022.

V. Rao and Y. Teh. Spatial normalized gamma processes. In *Advances in Neural Information Processing Systems*, volume 22. Curran Associates, Inc., 2009.

L. Ren, D. B. Dunson, and L. Carin. The dynamic hierarchical Dirichlet process. In *Proceedings of the 25th International Conference on Machine Learning*, pages 824–831. Association for Computing Machinery, 2008.

L. Ren, L. Du, L. Carin, and D. B. Dunson. Logistic stick-breaking process. *Journal of Machine Learning Research*, 12(1), 2011.

T. Rigon and D. Durante. Tractable Bayesian density regression via logit stick-breaking priors. *Journal of Statistical Planning and Inference*, 211:131–142, 2021.

G. O. Roberts, A. Gelman, and W. R. Gilks. Weak convergence and optimal ccaling of random walk Metropolis algorithms. *The Annals of Applied Probability*, 7(1):110–120, 1997. ISSN 10505164. URL `http://www.jstor.org/stable/2245134`.

A. Rodriguez and D. Dunson. Nonparametric Bayesian models through probit stick-breaking processes. *Bayesian Analysis*, 6:145–178, 2011.

A. Rodriguez, D. B. Dunson, and A. E. Gelfand. The nested Dirichlet process. *Journal of the American statistical Association*, 103(483):1131–1154, 2008.

A. Rubin, N. Geva, L. Sheintuch, and Y. Ziv. Hippocampal ensemble dynamics timestamp events in long-term memory. *eLife*, 4:e12247, 2015. ISSN 2050-084X.

A. Rubin, L. Sheintuch, N. Brande-Eilat, O. Pinchasof, Y. Rechavi, N. Geva, and Y. Ziv. Revealing neural correlates of behavior without behavioral measurements. *Nature Communications*, 10(1):4745, 2019.

R. Satija, J. A. Farrell, D. Gennert, A. F. Schier, and A. Regev. Spatial reconstruction of single-cell gene expression data. *Nature Biotechnology*, 33(5):495–502, 2015.

L. Scrucca, M. Fop, T. B. Murphy, and A. E. Raftery. mclust 5: clustering, classification and density estimation using gaussian finite mixture models. *The R Journal*, 8(1):289, 2016.

J. Sethuraman. A constructive definition of Dirichlet priors. *Statistica Sinica*, pages 639–650, 1994.

B. Shahbaba and R. Neal. Nonlinear models using Dirichlet process mixtures. *Journal of Machine Learning Research*, 10(8), 2009.

W. Shen, S. T. Tokdar, and S. Ghosal. Adaptive Bayesian multivariate density estimation with Dirichlet mixtures. *Biometrika*, 100(3):623–640, 2013.

W. Tang, F. Bertaux, P. Thomas, C. Stefanelli, M. Saint, S. Marguerat, and V. Shahrezaei. bayNorm: Bayesian gene expression recovery, imputation and normalization for single-cell RNA-sequencing data. *Bioinformatics*, 36(4):1174–1181, 2020.

Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei. Hierarchical Dirichlet processes. *Journal of the American Statistical Association*, 101(476):1566–1581, 2006.

C. A. Vallejos, J. C. Marioni, and S. Richardson. BASiCS: Bayesian analysis of single-cell sequencing data. *PLoS Computational Biology*, 11(6):e1004333, 2015.

L. Van der Maaten and G. Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(11), 2008.

S. Wade and Z. Ghahramani. Bayesian cluster analysis: Point estimation and credible balls (with discussion). *Bayesian Analysis*, 13(2):559–626, 2018.

S. Wade and V. Inácio. Bayesian dependent mixture models: A predictive comparison and survey. *Statistical Science*, 40(1):81 – 108, 2025.

Q. Wu and X. Luo. Nonparametric Bayesian two-level clustering for subject-level single-cell expression data. *Statistica Sinica*, 32(4):1835–1856, 2022.

Y. Wu and S. Ghosal. The L_1-consistency of Dirichlet mixtures in multivariate density estimation. *Journal of Multivariate Analysis*, 101:2411–2419, 2010.