Bayesian Analysis (2019) **14**, Number 1, pp. 161–180

Mixture Modeling on Related Samples by ψ-Stick Breaking and Kernel Perturbation

Jacopo Soriano[*‡] and Li Ma[†§]

Abstract. There has been great interest recently in applying nonparametric kernel mixtures in a hierarchical manner to model multiple related data samples jointly. In such settings several data features are commonly present: (i) the related samples often share some, if not all, of the mixture components but with differing weights, (ii) only some, not all, of the mixture components vary across the samples, and (iii) often the shared mixture components across samples are not aligned perfectly in terms of their kernel parameters such as the location and spread in Gaussian kernels, but rather display small misalignments either due to systematic cross-sample difference or more often due to uncontrolled, extraneous causes. Properly incorporating these features in mixture modeling will enhance the efficiency of inference, whereas ignoring them not only reduces efficiency but can jeopardize the validity of the inference due to issues such as confounding. We propose to use two techniques for incorporating these features in modeling related data samples using kernel mixtures. The first technique, called ψ-stick breaking, is a joint generative process for the mixing weights through the breaking of both a stick shared by all the samples for the components that do not vary in size across samples and an idiosyncratic stick for each sample for those components that do vary in size. The second technique is to imbue random perturbation into the kernels, thereby accounting for cross-sample misalignment. These techniques can be used either separately or together in both parametric and nonparametric kernel mixtures. We derive efficient Bayesian inference recipes based on Markov Chain Monte Carlo (MCMC) sampling for models featuring these techniques, and illustrate their work through both simulated data and a real flow cytometry data set in prediction/estimation and testing multi-sample differences.

MSC 2010 subject classifications: Primary 62F15, 62G99; secondary 62G07.

Keywords: Bayesian nonparametrics, hierarchical models, clustering, flow cytometry.

1 Introduction

Kernel mixtures are a powerful tool for modeling a variety of data sets, especially in the presence of a natural clustering structure (Escobar and West, 1995; MacEachern and Müller, 1998). A good portion of the rapidly expanding literature on Bayesian nonparametrics is aimed at building effective mixture models. A recent focus of the literature is on how to jointly model in a hierarchical manner data samples that are similar or otherwise related, the main objective being effective borrowing of strength

[*]Google Inc., Mountain View, CA 94043, USA, jsoriano.stat@gmail.com
[†]Department of Statistical Science, Duke University, Durham, NC 27708, USA, li.ma@duke.edu
[‡]Part of the research was completed while JS was a PhD student at Duke University.
[§]Supported by NSF grants DMS-1309057 and DMS-1612889, and a Google Faculty Research Award.

 https://doi.org/10.1214/18-BA1106

across samples, thereby substantially enhancing inference on the underlying data generative mechanisms as well as prediction. This is particularly important for complex data sets, for which each individual sample may only contain very limited information regarding the underlying probability distribution. Among many notable efforts in this direction, Lopes et al. (2003) proposed a hierarchical model for multiple finite mixtures. Müller et al. (2004) proposed a nonparametric extension of Lopes et al. (2003)'s model by replacing finite mixtures with Dirichlet process (DP) mixtures. In a different vein, Cron et al. (2013) proposed to use the hierarchical DP (HDP) model (Teh et al., 2006) as the mixing distribution to characterize variation across multiple mixture distributions. Rodríguez et al. (2008) proposed the nested DP (NDP) mixture, which is an infinite mixture of DP mixtures that induces an additional level of clustering among multiple mixture distributions themselves (to be distinguished from the clustering within each mixture distribution).

While applicable to a variety of mixture modeling contexts, our work is motivated during our attempt to apply existing hierarchical mixture models to the analysis of data collected from flow cytometry experiments. Flow cytometry is a laser-based technology that measures biomarkers on a large number of cells, so each cell is an observation from a distribution in $\mathbb{R}^p$, where p is the number of biomarkers measured. The cell population typically comes from a blood sample in immunological studies, and it consists of cells of various subtypes—e.g., T cells, B cells, etc.—with each subtype forming a "cluster" in the sample space. Because each cell subtype has a specific function in the immune system, inference on the abundance of the various subtypes across blood samples of a patient under different stimulating conditions, for instance, is of interest. Mixture models are natural tools for characterizing such data as the data is indeed a mixture of various cell types (Chan et al., 2008), and because a typical flow cytometry study will involve multiple samples collected under different conditions, the need for joint modeling to achieve effective borrowing of strength also naturally arises (Cron et al., 2013).

During the analysis of flow cytometry experiments using mixtures, we encountered a number of important challenges that we believe are present in numerous (if not most of) other applications involving mixture modeling of related samples (not only with location-scale kernels but beyond). Below we summarize the three main data features/challenges that motivate the current work:

I. *Samples often share clusters but with differing weights.* Related samples tend to share some (even most) of their clusters, and these common clusters vary across related samples in their weights. In flow cytometry, for instance, data samples often share a vast majority of the cell subtypes, and the most common type of variation across samples is the differences in the relative sizes of the subtypes.

II. *Only some, not all, clusters vary.* Often, only a fraction, not all, of the clusters vary across samples. In flow cytometry, not all cell subtypes are affected by the experimental conditions of interest. Very often only one or two cell types are affected and thus vary across the samples while the rest do not.

III. *Misalignment across samples in shared clusters.* Even the same cluster shared among samples is often not perfectly aligned across samples, either due to actual

systematic difference across the samples, or very often due to the presence of extraneous, uncontrolled additional sources of variation, i.e., some "random" effect. This is easily seen in mixtures of location-scale families, where the location and spread of some shared clusters differ to various extent across samples. Such misalignment is ubiquitous in flow cytometry data, with numerous potential causes. For example even tiny differences in the chemical concentrations applied in the experimental protocol across experiments can cause noticeable "perturbations" in the cell subtypes.

As far as we know, none of the existing hierarchical approaches satisfactorily address all of these issues in a single coherent framework. Table 1 provides a summary of these data features and the extent to which some of the state-of-the-art methods (along with the method we propose herein) address each of them.

	Shared clusters with varying weights	Only a subset of clusters differ	Misalignment in kernels
Lopes et al. (2003); Müller et al. (2004)	Not allowed	Allowed	Not allowed
Teh et al. (2006); Cron et al. (2013)	Allowed	Not allowed	Not allowed
Rodríguez et al. (2008)	Not allowed	Not allowed	Not allowed
This work	Allowed	Allowed	Allowed

Table 1: Comparison of hierarchical mixture models in terms of how they cope with the three common data features/challenges in modeling multiple related data samples.

Specifically, the existing approaches exploit some aspects of these features but do not fully take them into account. By introducing a cluster-specific hierarchical relationship among the samples, Lopes et al. (2003) and Müller et al. (2004) allow some clusters to be shared among the samples. However, their models require that the kernel parameters and the mixture weight for each cluster be either both shared across samples or both different, without the option to decouple these two different types of variations. In particular, no clusters are allowed to have only one type of variation—e.g., mixing weights—under these models. In the context of flow cytometry, for instance, this would mean that cell subtypes cannot change just in abundance across the samples but not in their location and spread, clearly an unrealistic assumption. On the other hand, by using the hierarchical DP (Teh et al., 2006) as the mixing distribution, Cron et al. (2013) does allow variations to exist in weights alone, but enforces the constraint that all clusters must all vary across samples, excluding the common situation in applications such as flow cytometry that only some clusters (e.g., subtypes) vary while others remain unchanged across conditions. Finally, under the nested DP mixture (Rodríguez et al., 2008; Rodriguez and Dunson, 2014), the clusters in each sample must either be completely identical as those in another sample if they fall into the same model level cluster or all be completely different, in both weights and kernel parameters, if they belong to different model level clusters.

New hierarchical modeling strategies are needed to address these limitations. To meet this need, we propose to adopt two modeling devices that can be embedded into a single hierarchical mixture modeling framework—the first for the mixing weights and

the other for the kernel parameters. For the weights, we adopt a stick breaking process that induces shared weights on some clusters (those that do not change in abundance) through breaking a "shared" stick across all samples while inducing different weights on the other clusters through breaking an "idiosyncratic" stick for each sample. This technique will allow us to address challenges I and II. For the mixture kernels, we utilize a *hierarchical* kernel to induce local perturbations in the kernel parameters across samples, which mimics the effect on the kernels due to uncontrolled random effects. Similar ideas for allowing variation in kernel parameters across samples within clusters have appeared in earlier works. See for example MacEachern (2008); Dunson (2009); Lock and Dunson (2013).

It is worth noting that our stick-breaking prior for the weights is in essence the marginal prior on the cluster weights for each individual sample induced under the model of Müller et al. (2004). A key distinction of our model lies in how it handles the weights and the kernel parameters jointly. The decoupling of weights from cluster centroids allows mixture components to share centroids with differing weights. Formally, this is achieved through making the additional constraint that all samples share the same clusters, just that some clusters have the same abundance across samples while other clusters can be more abundant or scarce in some samples compared to others. (Note that this does not exclude the possibility that some clusters are present in some samples and absent in others.) An additional benefit of this constraint is that it makes our model more structured and hence improves model identifiability.

The rest of the paper is organized as follows. We start in Section 2.1 with a brief review of the relevant background regarding nonparametric mixture modeling and stick breaking. Then in Section 2.2 we introduce the two techniques in turn. In Section 2.3 we provide a recipe for posterior inference based on Markov chain Monte Carlo (MCMC) sampling. In Section 3 we compare our method to current methods through simulation studies that cover prediction/estimation and testing multi-sample differences. In addition, we apply our method to analyze two flow cytometry data sets. We conclude in Section 4 with a few remarks.

2 Method

2.1 Background: Dirichlet process mixtures and stick breaking

While our techniques can be embedded into mixture models with various weight generating mechanisms and kernel families, we shall introduce and illustrate them in the context of DP mixtures of Gaussians, which is the most widely adopted nonparametric mixture model.

Suppose n observations $\boldsymbol{y} = (y_1, y_2, \ldots, y_n)$ are from a mixture model:

$$y_i \overset{\text{iid}}{\sim} F, \quad i = 1, \ldots, n, \quad \text{and} \quad f(\cdot) = \sum_{k \in \mathcal{K}} \pi_k \, g(\cdot|\lambda_k),$$

where f denotes the probability density function of F, $g(\cdot|\lambda)$ is a kernel distribution parametrized by λ, $\mathcal{K}$ the countable (possibly infinite) index set of the mixture com-

ponents (or clusters), and π_k the associated weight for the kth mixture component. Location-scale families are commonly adopted as the kernel distribution, in which case λ_k specifies the location and spread of the kth cluster. By definition the weights satisfy $\pi_k \geq 0$ and $\sum_k \pi_k = 1$. An alternative and computationally attractive formulation utilizes a latent cluster membership label $Z_i \in \mathcal{K}$ for each observation, such that

$$y_i \,|\, Z_i = k \sim g(\cdot|\lambda_k) \quad \text{and} \quad \Pr(Z_i = k) = \pi_k \quad \text{for } i = 1, 2, \ldots, N \text{ and } k \in \mathcal{K}.$$

Bayesian inference under mixture models can proceed after specifying prior distributions on the weights and the kernel parameters $\{(\pi_k, \lambda_k) : k \in \mathcal{K}\}$ (Marin et al., 2005). A flexible and convenient choice on the prior for the mixing weights is a generative procedure called the stick breaking process (SBP) (Sethuraman, 1994; Ishwaran and James, 2001). The general scheme of SBP starts with the drawing of a sequence of independent random variables $v_1, v_2, \ldots$ supported on $(0, 1)$. Then the weight for the kth cluster is given as

$$\pi_k = v_k \prod_{l=1}^{k-1} (1 - v_l).$$

A popular two-parameter specification is the Poisson-Dirichlet process (Kingman, 1975; Pitman and Yor, 1997), corresponding to $v_i \sim \text{Beta}(1-\gamma, \alpha+\gamma)$ for some parameters α and γ. In particular, when $\gamma = 0$, this boils down to the weight generative mechanism from a Dirichlet process (Ferguson, 1973; Sethuraman, 1994), which we shall refer to as the SBP(α) process.

When adopting the SBP(α) prior on the weights, along with a prior H on the kernel parameters, we obtain a Dirichlet process mixture (DPM) model:

$$\boldsymbol{\pi} = (\pi_k : k \in \mathcal{K}) \sim \text{SBP}(\alpha) \quad \text{and} \quad \lambda_k \overset{\text{iid}}{\sim} H, \quad k \in \mathcal{K}.$$

The most commonly adopted kernel distributions are location-scale families such as the (multivariate) Gaussian family, i.e., $g(\cdot|\lambda_k) = N(\cdot|\mu_k, \Sigma_k)$. In this case, H is often chosen to be the corresponding conjugate prior such as a normal-inverse-Wishart (NIW) prior on (μ_k, Σ_k).

2.2 Two techniques for hierarchically modeling related samples

Now assume J samples of observations $\boldsymbol{y}_j = (y_{1,j}, \ldots, y_{n_j,j})$ for $j = 1, \ldots, J$ have been collected, and the observations in each sample are modeled by a mixture:

$$\begin{aligned} y_{i,j} &\overset{\text{iid}}{\sim} F_j, \quad i = 1, \ldots n_j \quad \text{and} \quad j = 1, \ldots, J \\ f_j(\cdot) &= \sum_{k \in \mathcal{K}} \pi_{j,k}\, g(\cdot|\lambda_{j,k}), \quad j = 1, \ldots, J, \end{aligned}$$

where f_j is the probability density function of F_j, and $\lambda_{j,k}$ represent the kernel parameter for the kth cluster in the jth sample. To characterize potential relationship across the samples, let us assume that the kth component under each sample represent the same cluster (e.g., cell subtype). Note that this does not exclude the possibility of having

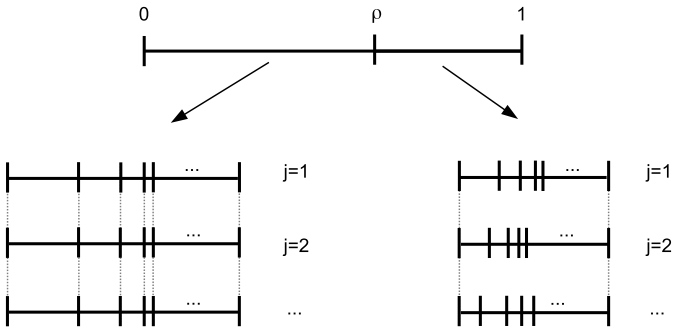

Figure 1: Illustration of the ψ-stick breaking procedure with the s-stick (left) and the i-sticks (right).

novel clusters that appear in only one or some of the samples, in which case the weights $\pi_{j,k} = 0$ if cluster k is absent in the jth sample. Again we let $\mathcal{K}$ be the collection of all cluster indices over all the samples. Let $Z_{i,j}$ be a latent variable indicating that the data point $y_{i,j}$ belongs to the kth cluster with $k \in \mathcal{K}$. Then the model with Gaussian kernels, for example, can be equivalently written as

$$[y_{i,j}|Z_{i,j} = k, \mu_{j,k}, \Sigma_k] \overset{\text{ind}}{\sim} N(y_{i,j}|\mu_{j,k}, \Sigma_k) \quad \text{and} \quad \Pr(Z_{i,j} = k) = \pi_{j,k} \text{ for } k \in \mathcal{K}.$$

We next introduce techniques for prior choices on the weights and on the kernel parameters by extending the stick breaking prior and the kernel respectively, which will address the three data features and challenges described in the Introduction.

ψ-stick breaking for weights We consider a generative stick breaking procedure called "ψ-stick breaking" (for reasons to be explained below), which breaks J sticks of unit length—one for each sample—in a dependent manner to generate the mixing weights $\{\pi_{j,k} : k = 1, 2, \ldots\}$ for $j = 1, 2, \ldots, J$. We start by observing that each cluster falls into one of two categories $\mathcal{K}_0$ and $\mathcal{K}_1$, that is $\mathcal{K} = \mathcal{K}_0 \cup \mathcal{K}_1$ with $\mathcal{K}_0 \cap \mathcal{K}_1 = \emptyset$: those in $\mathcal{K}_0$ have weights that do not vary across the J samples (e.g., cell types whose abundance is constant across experimental conditions), i.e., $\pi_{j,k} = \pi_{j',k}$ for $j, j' = 1, \ldots, J$ for $k \in \mathcal{K}_0$, whereas those in $\mathcal{K}_1$ have varying weights across samples. One can think of $\mathcal{K}$ as the set of natural numbers, which label the clusters, and $\mathcal{K}_0$ and $\mathcal{K}_1$ form a partition of $\mathcal{K}$. In the context of flow cytometry, one may think of $\mathcal{K}_0$ as those "house-keeping" cells with stable abundance over samples, where as $\mathcal{K}_1$ are those that are sensitive to the experimental conditions under investigation.

The generative process proceeds in two steps and is illustrated in Figure 1. In the first step, we break the J sticks at exactly the same spot into two pieces of length ρ and $1-\rho$ respectively, where $\rho \in (0, 1)$ is drawn as a Beta random variable. Then in the second step, we use the J pieces of length ρ to generate the weights for the components in $\mathcal{K}_0$, and the J pieces of length $1-\rho$ for the subtypes in $\mathcal{K}_1$. Hence the parameter ρ is interpreted as the overall proportion of the clusters with constant weights across samples.

Specifically, one can imagine that we *tie* the J sticks of length ρ together and break them using a single SBP as if they were a single stick—always at the same locations. For this reason, we shall refer to the common stick formed by tying the J sticks of length ρ as the "shared" stick, or the s-stick. Let $\{w_{0,k} : k \in \mathcal{K}_0\}$ with $\sum_{k\in\mathcal{K}_0} w_{0,k} = 1$ be the randomly generated *relative* sizes of the components in $\mathcal{K}_0$ in terms of the proportions of the s-stick. So the absolute size of each cluster that does not change across samples is given by $\pi_{j,k} = \rho w_{0,k}$ for all $j = 1, 2, \ldots, J$ and $k \in \mathcal{K}_0$.

On the other hand, we break the J sticks of length $1-\rho$ *independently* using separate independent SBPs, each generating the weights for one of the J samples, corresponding to the sizes of clusters that vary across samples. For this reason, we shall refer to the J sticks of length $1-\rho$ as the "idiosyncratic" sticks, or the i-sticks. We let $\{w_{j,k} : k \in \mathcal{K}_1\}$ for $j = 1, 2, \ldots, J$ with $\sum_{k\in\mathcal{K}_1} w_{j,k} = 1$ be the randomly generated lengths of the components as proportions of the corresponding i-stick. So for the kth cluster, its weight in the jth sample is given by $\pi_{j,k} = (1-\rho)w_{j,k}$.

Using SBP(α) processes for breaking each of the s- and i-sticks, we arrive at a joint generative model for the weights in all of the J samples, which we call "shared/idiosyncratic" (si or ψ) stick breaking. Specifically, with a Beta prior on the length of the shared stick, we arrive at the following hierarchical model for weights

$$\begin{aligned} \pi_{j,k} &= \begin{cases} \rho w_{0,k} & j = 1, \ldots, J \text{ and } k \in \mathcal{K}_0, \\ (1-\rho) w_{j,k} & j = 1, \ldots, J \text{ and } k \in \mathcal{K}_1, \end{cases} \\ \rho &\sim \text{Beta}(a_\rho, b_\rho), \\ (w_{0,k} : k \in \mathcal{K}_0) &\sim \text{SBP}(\alpha_0), \\ (w_{j,k} : k \in \mathcal{K}_1) &\overset{\text{iid}}{\sim} \text{SBP}(\alpha_1), \quad j = 1, \ldots, J. \end{aligned} \tag{1}$$

See Figure 1 for a visualization of the hierarchical prior on the mixture weights.

The hyperparameters α_0 and α_1 specify the size of the clusters as well as the number of clusters (in $\mathcal{K}_0$ and $\mathcal{K}_1$ respectively), with smaller values corresponding to a small number of large clusters and larger values corresponding to a large number of small clusters. We infer on α_0 and α_1 in a hierarchical Bayesian paradigm by placing a Gamma hyperprior on them: $\alpha_0, \alpha_1 \overset{\text{ind}}{\sim} \text{Gamma}(\tau_{\alpha,1}, \tau_{\alpha,2})$.

Local kernel perturbation We utilize a hierarchical setup to incorporate local perturbation in the kernel parameters, thereby adjusting for the misalignment and allowing more effective borrowing of information across the samples on each cluster. Specifically, we model the kernel parameters $\{\lambda_{j,k}\}$ as follows

$$\begin{aligned} \lambda_{0,k} &\overset{\text{iid}}{\sim} H_0(\cdot \,|\, \phi_0) \quad \text{for } k \in \mathcal{K}, \\ \lambda_{j,k} &\overset{\text{iid}}{\sim} H(\cdot \,|\, \lambda_{0,k}, \epsilon) \quad \text{for } j = 1, 2, \ldots, J, \end{aligned}$$

where $\lambda_{0,k}$ represent the cross-sample "centroid" kernel parameters for the kth cluster, with a hyperprior H_0 specified by hyperparameter ϕ_0. Given $\lambda_{0,k}$, the sample-specific

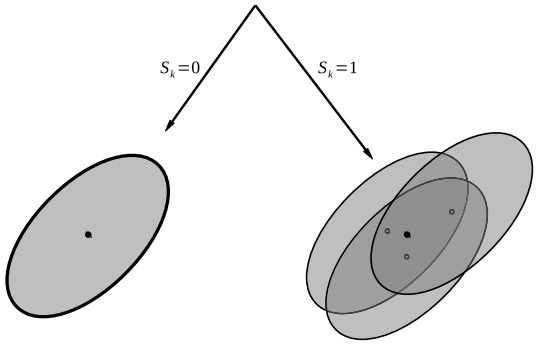

Figure 2: A locally perturbed Gaussian kernel with a spike-and-slab setup. When $S_k = 0$, all kernels for the kth cluster are identical across samples. When $S_k = 1$, the kernels are centered around a common mean but are not identical.

kernel parameters for the kth cluster $\lambda_{j,k}$ is drawn from H with additional hyperparameter ϵ, which specifies the dispersion of cluster k among the samples around the "centroid". Note that here ϵ is not a contamination parameter. Rather it quantifies the prior variance of the $\lambda_{j,k}$'s and controls the shrinkage of the samples toward each cluster mean.

The above specification enforces that each cluster k will have misalignment. More generally, in some problems misalignment may exist in only a subset of the clusters. To allow for such cases, again appeal to a "spike-and-slab" setup by introducing an additional Bernoulli latent indicator S_k for each cluster, such that $S_k = 1$ if there is misalignment in cluster k whereas $S_k = 0$ if otherwise. That is,

$$\lambda_{j,k} \overset{\text{ind}}{\sim} \begin{cases} \delta_{\lambda_0,k} & \text{if } S_k = 0 \\ H(\cdot|\lambda_{0,k}, \epsilon) & \text{if } S_k = 1 \end{cases} \qquad \text{and} \qquad S_k \overset{\text{iid}}{\sim} \text{Bernoulli}(\varphi),$$

where $\delta_{\cdot}$ represents a point mass.

Putting the pieces together in the context of Gaussian kernels, we arrive at the following spike-and-slab version of the locally perturbed kernel model:

$$\begin{aligned} \Sigma_k^{-1} &\overset{\text{iid}}{\sim} \text{Wishart}(\Psi_1, \nu_1), \\ [\mu_{j,k}|\mu_{0,k}, \Sigma_k, S_k] &\overset{\text{ind}}{\sim} \delta_{\mu_{0,k}} 1_{\{S_k=0\}} + \text{Normal}(\mu_{0,k}, \epsilon\Sigma_k) 1_{\{S_k=1\}}, \\ [\mu_{0,k}|\Sigma_k] &\overset{\text{ind}}{\sim} \text{Normal}(m_1, \Sigma_k/k_0), \\ S_k &\overset{\text{iid}}{\sim} \text{Bernoulli}(\varphi). \end{aligned}$$

This model is illustrated in Figure 2. The hyperparameter ϵ specifies the total amount of local variation between the means of each group $\mu_{j,k}$ and the grand mean $\mu_{0,k}$, and φ specifies the proportion of clusters that have misalignment. The hyperparameters m_1, Ψ_1, k_0, ϵ, and φ are all characterizing "global" features of the data that pertain to all of the clusters and samples. We can reliably infer them by pooling information through

hierarchical Bayes. In particular, in our numerical examples we adopt the following hyperpriors: $\epsilon \sim \text{Uniform}(a_\epsilon, b_\epsilon)$, $m_1 \sim \text{Normal}(m_0, V_0)$, $\Psi_1 \sim \text{Inverse-Wishart}(\Psi_2, \nu_2)$, $k_0 \sim \text{Gamma}(\tau_1/2, \tau_2/2)$, and $\varphi \sim \text{Beta}(a_\varphi, b_\varphi)$.

Model identifiability. A common issue that arises in hierarchical mixture models is model identifiability. In comparison to other models such as Müller et al. (2004), our model imposes the additional constraint that all samples share the cluster centroids. For this reason, our model can suffer less identifiability issue as it does not use $J + 1$ sets of independent centroids to characterize J distributions. Our model uses $J + 1$ sets of weights along with a common set of centroids, thereby effectively pooling information across the samples to identify the clusters.

On the other hand, our model does incorporate local perturbations in the kernel through embedding a hierarchical structure into the mixture kernels, and in this regard, new identifiability issue could arise. Intuitively, when the scale of perturbation is similar to (or even larger than) that of cross-cluster differences, then the data is simply too noisy and one cannot hope to identify either through our model or others (or sometimes even human judgment) whether the shifts in some clusters are due to misalignment/perturbation or that they are a different cluster. In fact, only when the extent of cross-sample misalignment for the same clusters are substantially smaller than that of the cross-cluster difference, such as in typical flow cytometry studies, can our model be identifiable, and such identifiability can be enforced through the prior specification that incorporates such constraints—e.g., in the choice of a_ϵ and b_ϵ.

In each specific application of our model, it is recommended to check whether the analysis is prone to suffer from identifiability issues. Some strategies for such diagnosis include checking the contrasting plots between the prior and the posteriors, as well as the joint posterior distribution of pairs of parameters, as well as completing a sensitivity analysis. We will illustrate all these strategies in our real data analysis example involving two flow cytometry data sets.

2.3 Posterior inference based on MCMC sampling

Posterior inference can be carried out through Markov Chain Monte Carlo (MCMC). One option is to use Müller et al. (2004)'s standard Pólya urn scheme. A benefit of this sampling scheme is that all the random weights are integrated out. However it can be computationally inefficient for large datasets such as in flow cytometry experiments. Alternatively, one can approximate the nonparametric model with a finite model and use a blocked Gibbs sampler (Ishwaran and James, 2001), which is more efficient in terms of mixing and computational speed, and hence is what we recommend.

To this end, two different finite approximation strategies are commonly adopted for DPMs and other stick breaking mixtures: (i) truncating the stick breaking at some maximum number of components and (ii) using finite-dimensional symmetric Dirichlet distribution. These two approximations might look very different at first, but the main difference between the two is in the induced stochastic ordering of the weights, which is irrelevant in mixture models. In fact, as Kurihara et al. (2007) points out, one can apply a size-biased permutation to the order of the weights of a finite symmetric Dirich-

let distribution and obtain a distribution which is practically identical to the truncated SBP. However, the two strategies are not computationally equivalent for mixture models. The weights under the symmetric finite-Dirichlet approximation are exchangeable, which results in substantially improved mixing over truncating the SBP. Therefore we opt for the symmetric finite Dirichlet approximation in our implementation. This approximation has been studied and used by many authors in a variety of contexts. See Neal (2000), Green and Richardson (2001) and Ishwaran and Zarepour (2002), among others. Specifically, under this approximation, the infinite sequences of mixture weights in (1) are replaced by:

$$(w_{0,k} : k \in \mathcal{K}_0) \sim \text{Dirichlet}(\alpha_0/K_0, \alpha_0/K_0, \ldots, \alpha_0/K_0),$$
$$(w_{j,k} : k \in \mathcal{K}_1) \overset{\text{iid}}{\sim} \text{Dirichlet}(\alpha_1/K_1, \alpha_1/K_1, \ldots, \alpha_1/K_1), \quad \text{for } j = 1, \ldots, J,$$

where K_0 and K_1 represent the numbers of mixture components that are shared and differential across the groups, respectively. In the nonparametric case, both $\mathcal{K}_0$ and $\mathcal{K}_1$ are infinite, while in the finite approximation we need to choose K_0 and K_1. A simple choice is to set $K_0 = K_1 = K$ for some large K which represents an upperbound to the a priori expected number of mixture components. Earlier works that investigated the truncation level for finite approximation of Dirichlet processes—such as Muliere and Tardella (1998) as well as Ishwaran and James (2001)—provide generic guidelines on how to select the truncation level. In the particular context of flow cytometry analysis, because the number of clusters itself is not a quantity of direct inferential interest, a very simple and practically sufficient strategy that we have usually applied is that one can experiment with an initially small number of clusters of e.g., 20, and gradually increase the number of clusters in increments of 5 until one consistently have a small number of empty clusters from the MCMC run. In our numerical examples, we have simply set K at a very large number, in particular 100.

We give the details on the MCMC sampler for the joint posterior in terms of the full conditionals in Supplementary Materials S1 (Soriano and Ma, 2019).

Label switching is a concern for mixture models in general. For this reason in inference we do not try to identify the meaning of parameters for individual subtypes or clusters, but instead focus our attention on parameters that are not cluster-specific. Specifically, as will be illustrated in the numerical examples, we focus our attention on the posterior distribution of the global hyperparameters φ, ϵ, and ρ, which are not cluster-specific.

3 Numerical examples

In this section we provide three numerical examples. In the first example data are simulated under different mixture distributions, and we compare the goodness-of-fit of our method with respect to competing approaches. In the second example we compare the performance of our model to other competing methods in testing and identifying differences across distributions. In the last example we analyze two real flow cytometry datasets. In all of the examples, we shall refer to our Dirichlet process mixtures of

Gaussians with ψ-stick breaking and kernel perturbation as CREMID, as it models Closely RElated MIxture Distributions.

In all of the numerical examples, we adopt the following choices of the hyperparameters: $K = 100$, $a_\rho = b_\rho = 0.5$, $\tau_{\alpha,1} = \tau_{\alpha,2} = 1$, $a_{\varphi,1} = b_{\varphi,2} = 0.5$, $a_\epsilon = 10^{-10}$, $b_\epsilon = 1$, $\tau_1 = \tau_2 = 4$, m_0 and V_0 are set to the empirical mean and covariance of the observations, $\Psi_2 = 100V_0$, and $\nu_1 = \nu_2 = p + 2$ where p is the dimension of the data. We carry out a sensitivity analysis on how the choice of the key hyperparameters for ρ, φ, and ϵ influence inference in the context of the real data example given in Section 3.3 in Supplementary Materials S3.

3.1 Example 1: Estimation and predictive performance

In this first example, we investigate how CREMID helps achieve more effective borrowing of information across samples thereby enhancing predictive performance. To this end, we consider four representative simulation scenarios. We use the sum of L_1 distances of the estimated univariate predictive densities from the true densities as measure of goodness of fit. (Note that we used this metric instead of the more natural log predictive score or the L_1 distance between the multivariate predictive density from the true density, because at the time of writing, the available software for a competitor, Müller et al. (2004)'s model, provides the marginal predictive densities but not the other two metrics.)

We consider the following multi-sample scenarios in $\mathbb{R}^4$. In each scenario, there are three data samples ($j = 1, 2, 3$) and the sample size for each is 100. Below we outline the four different scenarios. Some of the parameters for the simulation scenarios are omitted here, but are provided in the Supplementary Materials S2.

1. Local shift:

$$y_{i,j}|\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\pi} \sim \pi_1 N(y_{i,j}|\mu_1 + \delta_j, \Sigma_1) + \sum_{k=2}^{4} \pi_k N(y_{i,j}|\mu_k, \Sigma_k),$$

   where $\delta_j = (j/2, 0, 0, 0)$ and $\mu_k \sim U(0, 10)$ for $k = 1, \ldots, 4$.

2. Global shifts:

$$y_{i,j}|\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\pi} \sim \sum_{k=1}^{4} \pi_k N(y_{i,j}|\mu_k + \frac{j}{10}\mathbb{1}_4, \Sigma_k),$$

   where $\mu_k \sim U(0, 10)$ for $k = 1, \ldots, 4$.

3. Local weight difference:

$$\begin{aligned} y_{i,j}|\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\pi} \sim\ & (\pi_1 - 0.04(j-1))N(y_{i,j}|\mu_1, \Sigma_1) \\ & + (\pi_2 + 0.04(j-1))N(y_{i,j}|\mu_2, \Sigma_2) + \sum_{k=3}^{4} \pi_k N(y_{i,j}|\mu_k, \Sigma_k), \quad (2) \end{aligned}$$

   where $\boldsymbol{\pi} = (0.09, 0.01, 0.8, 0.1)$ and $\mu_k \sim U(0, 10)$ for $k = 1, \ldots, 4$.

4. Global weight differences:

$$y_{i,j}|\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\pi} \sim \sum_{k=1}^{8} \pi_{j,k} N(y_{i,j}|\mu_k, \Sigma_k),$$
$$\pi_j \propto \exp(m_j),$$
$$m_j \sim N(0, S),$$

where $\mu_k \sim U(0, 10)$ for $k = 1, \ldots, 8$.

We compare our method to Müller et al. (2004)'s model. We use the function `HDPMdensity` in the R package `DPpackage` (Jara et al., 2011) for fitting this model. (It is worth noting that Müller et al. (2004)'s model differs from the standard hierarchical Dirichlet process mixture model, which uses the hierarchical DP as a mixture distribution. We shall nevertheless use HDPM to refer to Müller et al. (2004)'s model throughout the numerical examples as this is the name given to this model by `DPpackage` at the time of this writing.) In addition, we also compare these to methods to independent finite mixture of Gaussians for each of the three samples, using Mclust (Fraley and Raftery, 2002), available in the R package `mclust`.

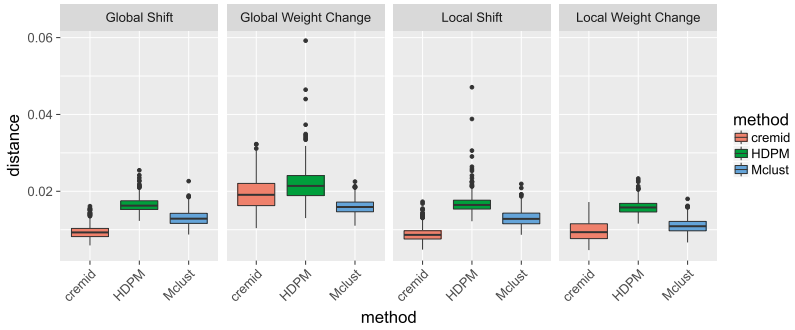

Figure 3: Box-plots of the sum of L_1 distances of the estimated univariate predictive densities from the true densities for three methods.

In Figure 3 we show the sum of L_1 distances of the estimated univariate predictive densities from the true densities for the three methods under 500 simulations. Our approaches outperform HDPM and Mclust in the two shift scenarios. CREMID is the most accurate method in the two location shift scenarios as well as in the local weight change scenario. In the global weight change scenario, both our method and HDPM underperform Mclust. Because the samples are different in all cluster weights, we pay a price for assuming that some cluster weights are shared.

3.2 Example 2: Testing cross-sample differences in cluster weights

We consider the same multi-sample scenarios in $\mathbb{R}^4$ used in Example 1. For each dataset we define a corresponding *null* data set by permuting the labels of the three samples.

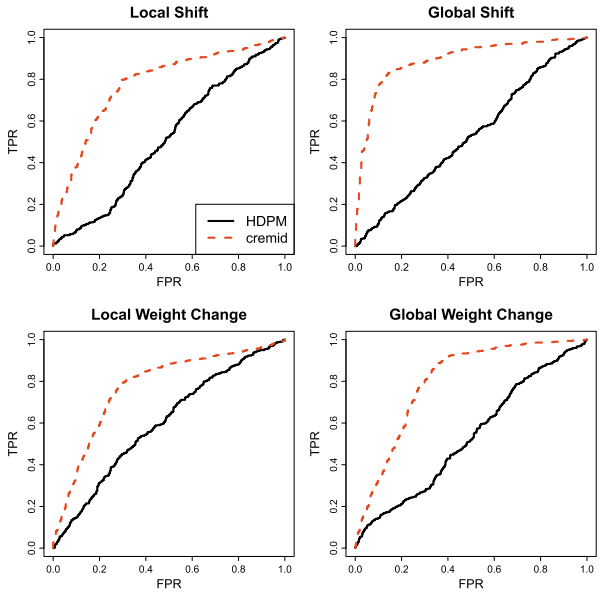

Figure 4: ROC curves for two methods in Example 3.2 based on 500 simulations: HDPM (Müller et al., 2004) in black solid, our method in red dashed.

In Figure 4 we compare the ROC curves of our method and HDPM for testing the hypothesis that the three distributions are identical. Our method is substantially more powerful than HDPM in all four scenarios.

In these simulations, for our method we use $\mathrm{E}(\rho(1-\varphi)|\boldsymbol{y})$ as the test statistic. This quantity goes to zero when there are differences in the mixture weights or in the mixture kernels across samples, and it goes to one when the distributions are identical across samples. One can adopt different test statistics under our method depending on the inference objective. For instance, if one is interested in testing just the presence of differences in weights then a suitable test statistic is $\mathrm{E}(\rho|\boldsymbol{y})$, and if one is interested in just kernel perturbations, then $\mathrm{E}(1-\varphi|\boldsymbol{y})$ can be a suitable choice.

We compare our method only to HDPM since Mclust does not provide a way to test for differences across samples. In HDPM each F_j is defined as a mixture of two components: $F_j = uH_0 + (1-u)H_j$ for $j = 1, \ldots, J$. The distribution H_0 represents the common part, and H_j represents the idiosyncratic part. The hyperparameter u is a contamination parameter controlling the "degree of similarity" across the F_j's has a Beta hyperprior. We use $\mathrm{E}(u|\boldsymbol{y})$ as the test statistic.

To see that these different test statistics under the two models are actually comparable, we note that when there is only weight difference and no kernel perturbations, both u under HDPM and ρ ($\approx \rho(1-\varphi)$ since the true $\varphi = 0$) under our model captures the proportion of clusters with no weight differences. Similarly, when there is only kernel perturbation and no weight differences, both u under HDPM and $1-\varphi$ ($\approx \rho(1-\varphi)$ as

the true $\rho = 1$) under our model characterize the proportion of clusters with no kernel perturbation. In this sense, the test statistic $\mathrm{E}(u \mid \boldsymbol{y})$ for HDPM is a counterpart of the test statistic $\mathrm{E}(\rho(1-\varphi) \mid \boldsymbol{y})$ under our model, and so that the difference in the performance of the two methods can be attributed to the different abilities to characterize relationships across multiple samples rather than the choice in the test statistic.

3.3 Application: flow cytometry

In flow cytometry experiments, biomarkers are measured on a large number of blood cells. Different cell subtypes, i.e., groups of cells sharing similar biomarker's levels, have distinct functions in human immune system. Identifying variations in the abundance of subtypes across multiple samples is an important immunological question. Additionally, the location of a given subtype across samples can slightly change due to both experimental variability and other uncontrolled "random effects".

We analyze two datasets where each one contains three samples of 5,000 blood cells, and for each cell six biomarkers have been measured. A sensitivity analysis for evaluating how the choice of the hyperpriors influence the posterior inference is given in Supplementary Materials S3. In practical applications of our model, we recommend users to carry out such a sensitivity analysis to judge the robustness of the resulting inference in each specific context.

A control study

The blood from a given patient was split in three samples, and each sample went through a separate experimental procedure to generate the data. Since the three samples are essentially biologically identical, one expects no variations in the abundance of the different subtypes or large location shifts of the cell types. Small perturbations of the cell types are likely due to additional variations in the experimental procedures.

In Figure 5 we plot the posterior distributions of ρ and ϵ for this data set under our proposed model. The parameter ρ reflects the total mass assigned to mixture components whose weights are identical across groups. For this dataset *a posteriori* this parameter concentrates around one, indicating that there is no evidence of a difference in the mixture weights across the three replicates. The parameter ϵ controls the expected amount of shift in the location of each kernel across samples. Its posterior does not concentrate around zero, indicating the presence of small misalignment among the replicate samples due to uncontrolled sources of variation. It is the decoupling of these two sources of variations that allows us to correctly infer the absence of variations in the mixture weights across the distributions of the three samples.

Samples under different stimulation conditions

In another data set, three blood samples from an individual underwent different stimulation treatments. One sample was left unstimulated, while the two remaining samples were stimulated with CEF (cytomegalovirus, Epstein-Barr virus, and influenza virus)

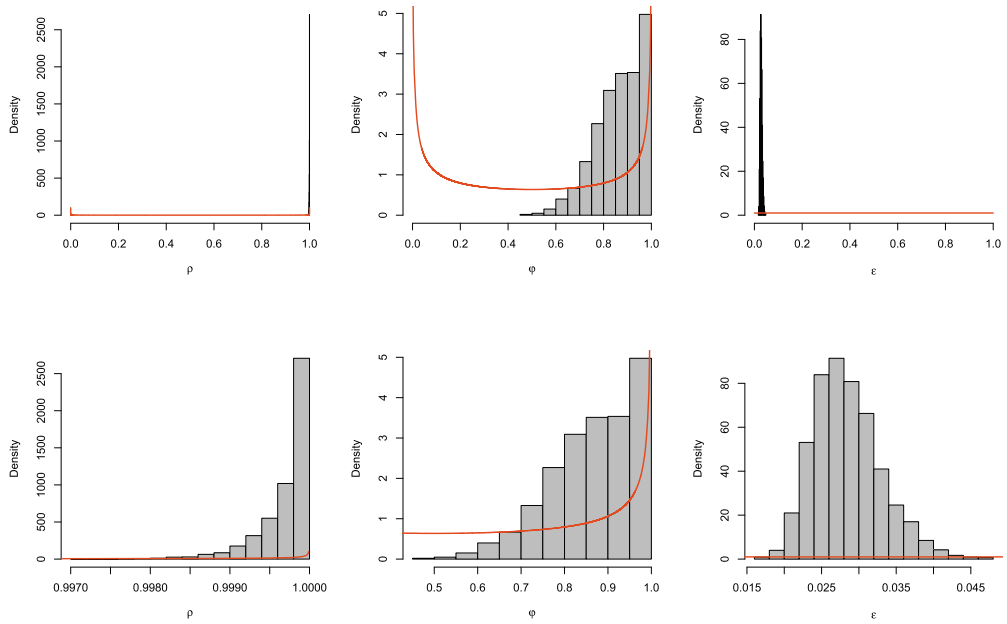

Figure 5: Histograms of the posterior draws of ρ, φ, and ϵ for the flow cytometry control study. The red lines indicate the prior distributions. The top row gives the histograms on the entire parameter space. The bottom row provides a zoom-in view of the corresponding histograms.

and CMV (cytomegalovirus) pp65 peptides, respectively. The samples underwent separate experimental procedures in data generation. In Figure 6 we plot the posterior distributions of ρ and ϵ. The parameter ρ concentrates around 0.6, indicating that there are differences in some of the mixture weights across the three samples. The parameter ϵ concentrates around 0.2, either due to effects of the experiment conditions on the locations of the kernels, which is also a systematic cross-sample difference, or substantial additional variations in the experimental procedures in comparison to the control study.

To ensure that the analysis is not suffering from serious identifiability issues, we carry out a sensitivity analysis in Supplementary Materials S3 and present the pair-wise joint posterior samples for ρ, φ, and ϵ in Supplementary Materials S4.

To judge the goodness-of-fit, we also compare the predictive performance of our model with Mclust, evaluated by the log predictive likelihood of the a "test" sample. We randomly select 1,000 data points from the whole data set as a "test" sample, while using 5,000 observations as the "training sample". We repeat this random training/test data split 100 times, and report the mean and standard deviation of the log-p predictive score on the testing set in Table 2. We had hoped to compare our method to other methods such as Müller et al. (2004) but at the time of writing, the existing software in `R` (the `HDPMdensity` function in `DPpackage`) crashes for the data sets, most probably due to the large sample sizes, and it does not output predictive scores.

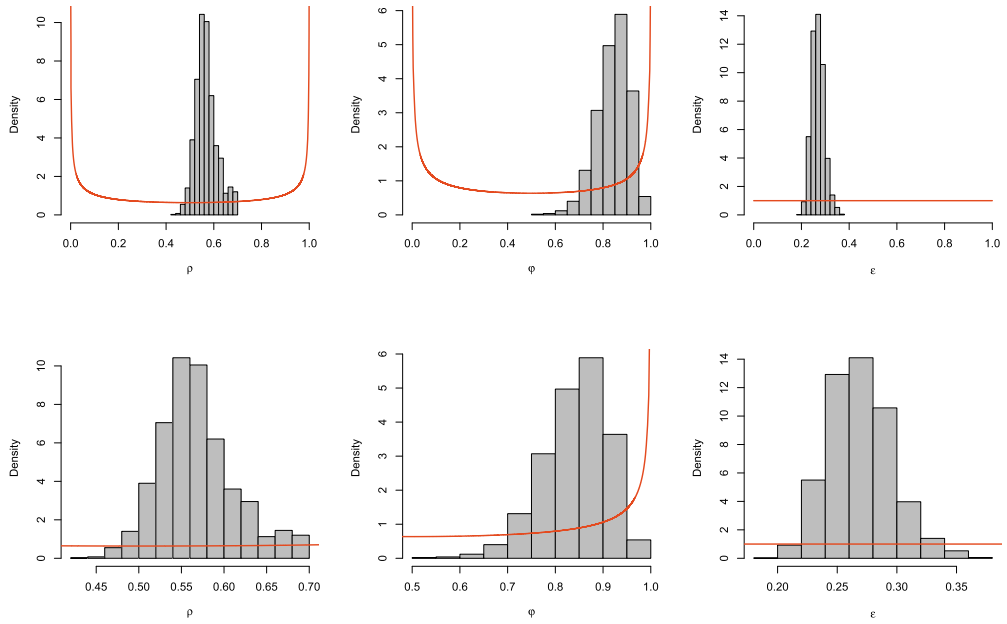

Figure 6: Histograms of the posterior draws of ρ, φ, and ϵ for the flow cytometry samples under different conditions. The red lines indicate the prior distributions. The top row gives the histograms on the entire parameter space. The bottom row provides a zoom-in view of the corresponding histograms.

	Method	
Data set	CREMID	MClust
Control study	-15488 (103)	-16248 (123)
Different stimulation conditions	-14490 (111)	-15297 (121)

Table 2: Average log-p predictive scores and their standard deviations (in parentheses) over 100 random training/testing split for CREMID versus MClust. Larger average values (or smaller absolute values for negative scores) indicate better fit to the data.

4 Conclusion

In this work we have illustrated two useful techniques in modeling related data sets using mixture models—the shared-idiosyncratic stick breaking and the locally perturbed kernel. When used together, they incorporate three common data features observed in real applications—(i) samples often share the same clusters with different weights; (ii) only some clusters vary across samples; (iii) misalignment in the clusters due to extraneous causes. We have derived Bayesian inference recipe through MCMC sampling and carried out an extensive numerical studies to illustrate the gain in inferential efficiency in both estimation, prediction, and hypothesis testing.

It is worth noting that in the special case with no kernel perturbation, i.e., $\epsilon = 0$ or $\varphi = 0$, our model is very similar to a variant of Müller et al. (2004)'s model with the idiosyncratic components arising from a hierarchical Dirichlet process (Teh et al., 2006). The distinction of our model from that model lies in the independent generation of the idiosyncratic weights, which could be considered a special case of the hierarchical DP in the limit.

The model we have considered for illustration involves misalignment in the cluster locations. It is conceptually straightforward to extend the model to involve perturbations in the covariance structure of the clusters as well. The current model results in a particularly simple and efficient sampling scheme—given the grand mean μ_k and the perturbation parameter ϵ, we can update the hidden states S_k marginally with respect to the individual cluster means $\mu_{j,k}$. Updating the individual variance-covariance matrix of the kernels would require an additional Metropolis step, which can be inefficient for moderately large p.

While the two techniques—ψ-stick breaking and kernel perturbation—are demonstrated in the context of mixtures of location-scale families, they are generally applicable to modeling related mixtures of other forms of kernels as well, such as mixtures of generalized linear models and mixtures of factor models. The computational details will vary but the general ideas remain the same.

Finally, we note that a recent manuscript by Camerlenghi et al. (2018) presents new strategies for generalizing Müller et al. (2004)'s model on the weights using nested processes. Their development could potentially be adopted to further enrich our framework.

Software

R code for the proposed MCMC sampler and code for the numerical examples are available at https://github.com/MaStatLab/cremid/ and https://github.com/MaStatLab/MPG-examples/, respectively.

Supplementary Material

Supplementary Materials for "Mixture modeling on related samples by ψ-stick breaking and kernel perturbation" (DOI: 10.1214/18-BA1106SUPP; .pdf).

References

Camerlenghi, F., Dunson, D. B., Lijoi, A., Prünster, I., and Rodríguez, A. (2018). "Latent nested nonparametric priors." *ArXiv e-prints:1801.05048*. 177

Chan, C., Feng, F., Ottinger, J., Foster, D., West, M., and Kepler, T. B. (2008). "Statistical mixture modeling for cell subtype identification in flow cytometry." *Cytometry Part A*, 73(8): 693–701. 162

Cron, A., Gouttefangeas, C., Frelinger, J., Lin, L., Singh, S. K., Britten, C. M., Welters, M. J., van der Burg, S. H., West, M., and Chan, C. (2013). "Hierarchical modeling for rare event detection and cell subset alignment across flow cytometry samples." *PLoS Computational Biology*, 9(7): e1003130. 162, 163

Dunson, D. B. (2009). "Nonparametric Bayes local partition models for random effects." *Biometrika*, 96(2): 249–262. URL http://www.jstor.org/stable/27798822 MR2507141. doi: https://doi.org/10.1093/biomet/asp021. 164

Escobar, M. D. and West, M. (1995). "Bayesian Density Estimation and Inference Using Mixtures." *Journal of the American Statistical Association*, 90(430): 577–588. MR1340510. 161

Ferguson, T. S. (1973). "A Bayesian Analysis of Some Nonparametric Problems." *Annals of Statistics*, 1(2): 209–230. URL http://dx.doi.org/10.1214/aos/1176342360 MR0350949. 165

Fraley, C. and Raftery, A. E. (2002). "Model-Based Clustering, Discriminant Analysis, and Density Estimation." *Journal of the American Statistical Association*, 97(458): 611–631. URL http://www.jstor.org/stable/3085676 MR1951635. doi: https://doi.org/10.1198/016214502760047131. 172

Green, P. J. and Richardson, S. (2001). "Modelling heterogeneity with and without the Dirichlet process." *Scandinavian Journal of Statistics*, 28(2): 355–375. MR1842255. doi: https://doi.org/10.1111/1467-9469.00242. 170

Ishwaran, H. and James, L. F. (2001). "Gibbs sampling methods for stick-breaking priors." *Journal of the American Statistical Association*, 96(453). MR1952729. doi: https://doi.org/10.1198/016214501750332758. 165, 169, 170

Ishwaran, H. and Zarepour, M. (2002). "Exact and approximate sum representations for the Dirichlet process." *Canadian Journal of Statistics*, 30(2): 269–283. MR1926065. doi: https://doi.org/10.2307/3315951. 170

Jara, A., Hanson, T. E., Quintana, F. A., Müller, P., and Rosner, G. L. (2011). "DPpackage: Bayesian semi-and nonparametric modeling in R." *Journal of Statistical Software*, 40(5): 1. MR3309338. doi: https://doi.org/10.1007/978-3-319-18968-0. 172

Kingman, J. F. (1975). "Random discrete distributions." *Journal of the Royal Statistical Society. Series B (Methodological)*, 1–22. MR0368264. 165

Kurihara, K., Welling, M., and Teh, Y. W. (2007). "Collapsed Variational Dirichlet Process Mixture Models." In *IJCAI*, volume 7, 2796–2801. 169

Lock, E. F. and Dunson, D. B. (2013). "Bayesian consensus clustering." *Bioinformatics*, 29(20): 2610–2616. URL http://dx.doi.org/10.1093/bioinformatics/btt425 164

Lopes, H. F., Müller, P., and Rosner, G. L. (2003). "Bayesian Meta-analysis for Longitudinal Data Models Using Multivariate Mixture Priors." *Biometrics*, 59(1): 66–75. MR1978473. doi: https://doi.org/10.1111/1541-0420.00008. 162, 163

MacEachern, S. N. (2008). "Discussion of "The nested Dirichlet process" by A.E. Gelfand, D.B. Dunson and A. Rodriguez." *Journal of the American Statistical Association*, 103: 1149–1151. MR2528831. doi: https://doi.org/10.1198/016214508000000553. 164

MacEachern, S. N. and Müller, P. (1998). "Estimating Mixture of Dirichlet Process Models." *Journal of Computational and Graphical Statistics*, 7(2): 223–238. 161

Marin, J.-M., Mengersen, K., and Robert, C. P. (2005). "Bayesian Modelling and Inference on Mixtures of Distributions." In Dey, D. and Rao, C. (eds.), *Bayesian Thinking: Modeling and Computation*, volume 25 of *Handbook of Statistics*, 459–507. Elsevier. MR2530974. 165

Muliere, P. and Tardella, L. (1998). "Approximating distributions of random functionals of Ferguson-Dirichlet priors." *Canadian Journal of Statistics*, 26(2): 283–297. 170

Müller, P., Quintana, F., and Rosner, G. (2004). "A method for combining inference across related nonparametric Bayesian models." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 66(3): 735–749. MR2088779. doi: https://doi.org/10.1111/j.1467-9868.2004.05564.x. 162, 163, 164, 169, 171, 172, 173, 175, 177

Neal, R. M. (2000). "Markov chain sampling methods for Dirichlet process mixture models." *Journal of Computational and Graphical Statistics*, 9(2): 249–265. MR1823804. doi: https://doi.org/10.2307/1390653. 170

Pitman, J. and Yor, M. (1997). "The two-parameter Poisson-Dirichlet distribution derived from a stable subordinator." *The Annals of Probability*, 855–900. MR1434129. doi: https://doi.org/10.1214/aop/1024404422. 165

Rodriguez, A. and Dunson, D. B. (2014). "Functional clustering in nested designs: Modeling variability in reproductive epidemiology studies." *Annals of Applied Statistics*, 8(3): 1416–1442. MR3271338. doi: https://doi.org/10.1214/14-AOAS751. 163

Rodríguez, A., Dunson, D. B., and Gelfand, A. E. (2008). "The Nested Dirichlet Process." *Journal of the American Statistical Association*, 103(483): 1131–1154. MR2528831. doi: https://doi.org/10.1198/016214508000000553. 162, 163

Sethuraman, J. (1994). "A Constructive Definition of Dirichlet Priors." *Statistica Sinica*, 4: 639–650. MR1309433. 165

Soriano, J. and Ma, L. (2019). "Supplementary Materials for "Mixture modeling on related samples by ψ-stick breaking and kernel perturbation"." *Bayesian Analysis*. doi: https://doi.org/10.1214/18-BA1106SUPP. 170

Teh, Y. W., Jordan, M. I., Beal, M. J., and Blei, D. M. (2006). "Hierarchical Dirichlet processes." *Journal of the American Statistical Association*, 101(476). MR2279480. doi: https://doi.org/10.1198/016214506000000302. 162, 163, 177

Acknowledgments

The authors thank Shai Gorsky, the referees, and the AE for very helpful comments. The authors are grateful to Cliburn Chan for helpful discussions. The flow cytometry data set was provided by EQAPOL (HHSN272201000045C), an NIH/NIAID/DAIDS-sponsored, international resource that supports the development, implementation, and oversight of quality assurance programs (Sanchez PMC4138253).