A Sequential Algorithm for Fast Fitting of Dirichlet Process Mixture Models

Alec Greaves-Tunnell

Department of Statistics, University of Washington Seattle, WA, 98195, USA

Abstract

This paper discusses the motivation, implementation, and analysis of a sequential algorithm for approximate Bayesian inference in Dirichlet process mixture models introduced by Zhang et al. [2014]. Bayesian Dirichlet process mixture models are a popular tool for clustering due to their flexibility, but standard sampling methods for approximating the posterior distribution do not scale well to large data sets. As data continues to grow in many applications, there is increasing interest in computationally efficient methods for fast approximation of the posterior distribution of model parameters. The authors propose one such method, V-SUGS, which extends a simple sequential algorithm known as SUGS ("sequential update and greedy search") by improving the representation of uncertainty over cluster assignments in the posterior. Comparative analysis of the two algorithms shows that V-SUGS performs better on data whose clusterings are close or overlapping while preserving the attractive computational structure of the original algorithm. Further experiments demonstrate the speed and relative accuracy with which the model can be fit to two large biological data sets.

1 Introduction

Finite mixture modeling offers a useful method for the identification of latent classes or clusters in a given set of observations, but the requirement that the number of mixture components be specified ahead of time comes as a major limitation, particularly in applications involving large or complex data. The Bayesian nonparametric framework generalizes the class of finite mixture models such that this requirement is eliminated, as the number of mixture components and the parameters of each individual mixture component can be estimated simultaneously. The present paper [Zhang et al., 2014] proposes a fast approximate method for fitting a class of Bayesian nonparametric models known as Dirichlet process mixture models (DPMMs). The authors are particularly interested

in applications of these models to large data sets, which raises significant computational challenges for the usual sampling approaches to inference. Their proposed alternative is a sequential algorithm whose recursive update structure means that an approximation to the posterior distribution can be computed in a single pass over the data.

1.1 Dirichlet Process Mixture Models

The nonparametric Bayesian approach to mixture modeling requires a prior over an infinite dimensional mixing distribution. The most common choice of prior is a Dirichlet process, which specifies a distribution over probability measures. Formally, a Dirichlet process parameterized by a concentration parameter $\alpha > 0$ and base measure F on a measurable space $(\mathcal{X}, \mathcal{A})$ is a prior over probability measures G such that for any finite partition $\{A\}_{i=1}^{m}$ of $\mathcal{A}$,

$$(G(A_1), ..., G(A_m)) \sim Dir(\alpha F(A_1), ..., \alpha F(A_m)).$$

Proof of existence and early characterization of Dirichlet processes is due to Ferguson [1973], who demonstrated a conjugacy property for the Dirichlet process analogous to the conjugacy of the finite-dimensional Dirichlet distribution for the multinomial distribution. Importantly, this property can be used to show that draws from a Dirichlet process will be discrete almost surely [Sudderth, 2006]. The Dirichlet process can thus serve as a prior over mixing distributions in a hierarchically defined mixture model. In particular, suppose we have observations y_i, $i \in \{1, ..., n\}$ drawn from a countably infinite mixture of densities $P(\cdot)$, each parameterized by θ. Rather than having to specify a mixing distribution F over the possible values of θ, we can place a Dirichlet process prior on F with concentration parameter α and base measure F_0.

The resulting model is called a Dirichlet process mixture model (DPMM), and can be summarized as

$$\begin{aligned} F &\sim DP(\alpha, F_0), \\ \theta_i &\overset{iid}{\sim} F, \\ y_i|\theta_i &\overset{ind}{\sim} P(\cdot|\theta_i), \end{aligned}$$

Let $\tilde{\theta} \in \{\tilde{\theta}_l\}_{l=1}^{\infty}$ denote the set of distinct values in the sequence $\theta_{1:n}$, $n = 1, 2, ...$, and write $\delta_i = j$ if $\theta_i = \tilde{\theta}_j$. A Polya urn characterization of the Dirichlet process due to Blackwell and

MacQueen [1973] shows that the predictive distribution of θ_{i+1} given $\theta_1, ..., \theta_i$ is

$$p(\theta_{i+1} = \theta | \theta_1, ..., \theta_i, \alpha, F_0) = \frac{1}{\alpha + i - 1} \left(\alpha f_0(\theta) + \sum_{j=1}^{N_i} n_j^{(i)} \delta_{\tilde{\theta}_j} \right), \quad (1)$$

where $n_j^{(i)}$ is the number of elements in $\theta_{1:i}$ that take the value $\tilde{\theta}_j$, N_i is the total number of distinct values in the sequence $\theta_{1:i}$, and f_0 is the density corresponding to F_0. Thus we have a description for how an observation y_i is generated under the model: first, we draw θ_i according to Eq. 1, and then we draw y_i from the distribution $P(y|\theta_i) = P(y|\tilde{\theta}_{\delta_i})$.

Eq. 1 implies that there is a positive probability that θ_i takes a value in $\theta_{1:i-1}$, or equivalently, that $\delta_i = j$ for some $j \in \{1, ..., N_{i-1}\}$. The model thus implicitly partitions or clusters the observations $y_{1:n}$ according to the value of their "assignments" $\delta_{1:n}$; the corresponding distribution over partitions is the well-known Chinese restaurant process (CRP) [Pitman, 2002]. Lo [1984] demonstrated that after marginalizing over the random measure F, the posterior distribution in a Dirichlet process mixture model can be written as the posterior for this partition of the observations $\delta_{1:n}$ multiplied by the product of independent posteriors over the mixture component parameters θ_l, $l \in \{1, ..., N_i\}$. This is important in practice, as it eliminates the need to work directly with the infinite-dimensional parameter F.

1.2 Inference in DPMMs

Dirichlet process mixture models are valued for their capacity to grow the number of mixture components with the complexity of the data, and interest in their application to clustering and density estimation problems has resulted in an extensive literature on methods for posterior computation. The bulk of this work has relied on Markov chain sampling methods, which approximate the exact posterior via the ergodic theorem. Initial approaches focused on Gibbs sampling [Bush and MacEachern, 1996, West and Escobar, 1993], which is generally feasible when F_0 is conjugate to the likelihood given by $P(\cdot|\theta)$. In the case of non-conjugate priors, modifications to the Gibbs sampling algorithm are required to deal with intractable integrals in the conditional probabilities [MacEachern and Müller, 1998]; a review of the basic issues and approaches to sampling for DPMMs is given by Neal [2000]. More recent approaches have focused on "split-merge" algorithms, which boost efficiency by generating proposals that allow for more rapid exploration over the space

of partitions [Jain and Neal, 2007], with further gains coming from parallelization [Bouchard-Côté et al., 2015, Chang and Fisher III, 2013].

While advances in sampling methodology have increased the computational efficiency of this approach, the authors of the current paper express concern over the traditionally poor scaling of these methods to very large data sets. Increasing interest in the application of DPMMs to such data sets has driven interest in methods that can quickly compute some approximation to the posterior when sampling is inadvisable or downright impossible. There exist a variety of of alternatives to MCMC for DPMMs, most notably variational inference [Blei and Jordan, 2006], but Zhang et al. [2014] choose to focus on sequential algorithms, which recursively update their state after each next observation. Here, the interest in sequential algorithms mostly derives from their computational efficiency with respect to the size of the data, as opposed to their potential application to settings in which the data truly arrive in streaming fashion. Existing recursive methods for approximating the posterior in DPMMs include a MAP search algorithm due to Daumé III [2007] and an adaptation of expectation propagation [Minka and Ghahramani, 2003]. The present authors' proposed V-SUGS method directly extends the "sequential update and greedy search" (SUGS) algorithm proposed by Wang and Dunson [2011], opting for probabilistic assignment of observations to clusters instead of a greedy hard assignment. Comparative analysis of the two algorithms shows that V-SUGS offers superior performance when clusters of data are close or overlapping, while both offer dramatic reductions in computational cost when compared to basic Gibbs sampling.

2 Methods

The V-SUGS algorithm modifies the SUGS implementation of a recursive greedy update scheme by allowing for uncertain cluster assignments and subsequently making a variational update to the mixture component parameters. Before we discuss the details of this method, we will first give an overview of SUGS and the variational Bayes approach to posterior approximation.

2.1 The SUGS algorithm

Suppose that observations y_i are obtained for subjects $i = 1, ..., n$. Then if we let δ_i be the cluster assignment of y_i and θ_j be the mixture component parameters for cluster j, the result of Lo [1984]

implies that a Dirichlet process mixture model for the data with concentration parameter $\alpha > 0$ and base measure F_0 can be written as

$$p(\delta, \theta) = p(\delta)p(\theta)$$
$$y_i|\delta, \theta \overset{ind}{\sim} P(\cdot|\theta_{\delta_i}),$$

where $p(\theta) = \prod_{j=1}^{\infty} f_0(\theta_j)$, with f_0 the density associated with F_0. Furthermore, the predictive distribution in Eq. 1 implies that $p(\delta)$ follows

$$p(\delta_i|\delta_{1:i-1}) = \begin{cases} \frac{n_j^{(i)}}{\alpha+i-1} & j \in \{1, ..., N_i\}, \\ \frac{\alpha}{\alpha+i-1} & j = N_i + 1, \end{cases}$$

for $i = 2, ..., n$, with $p(\delta_1 = 1) = 1$, $n_j^{(i)}$ the number of observations in $y_{1:i-1}$ assigned to cluster j, and N_i the number of clusters that have been assigned an observation (sometimes called "active" clusters) up to time i.

The objective of SUGS is to find an approximation $\pi(\theta|\hat{\delta}_{1:i}, y_{1:i})$ to the conditional posterior $p(\theta|\delta_{1:i}, y_{1:i})$; instead of attempting to maintain a joint posterior over δ and θ, the algorithm specifies a procedure for making a specific choice of cluster assignments $\hat{\delta}_{1:i}$ and approximating the posterior over θ given these assignments. This is a choice made for simplicity and speed in the algorithm, and it can be thought of as approximating the true posterior over δ by a point mass at $\hat{\delta}_{1:i}$. For each i, SUGS makes a greedy choice for $\hat{\delta}_i$ as the argument that maximizes the conditional allocation probability

$$\hat{\pi}_{i,\hat{\delta}_{1:i-1}}(\delta_i|y_{1:i}) \propto p(\delta_i|\hat{\delta}_{1:i-1}) \int p(y_i|\theta_{\delta_i}) \pi_{i-1,\hat{\delta}_{1:i-1}}(\theta|y_{1:i-1}) d\theta.$$

This quantity approximates the true conditional probability of cluster membership $p(\delta_i|y_{i:i})$; for every i, it defines a discrete distribution over $N_{i-1} + 1$ mixture components - the N_{i-1} mixture components active at time $i - 1$, plus a new mixture component given by the prior $f_0(\theta)$. Since SUGS is designed to be a sequential algorithm, no revision of the assignment $\hat{\delta}_i$ is made after time i.

After assigning observation y_i to cluster $\hat{\delta}_i$, we must update the posterior over the mixture component parameters θ. Conveniently, the marginal of θ_l associated with the conditional density

$p(\theta|\delta_{1:i}, y_{1:i})$ can be written as

$$p(\theta_l) \propto \left[\prod_{j=1}^{i} \mathbb{1}_{(\delta_j=l)} p(y_j|\theta_{\delta_j})\right] f_0(\theta_l),$$

which shows that the mixture component parameters θ_l, $l = 1, ..., N_i$, are conditionally independent. Therefore, the assignment of an observation to a cluster l only requires an update of the parameters θ_l associated with that cluster. In general the posterior is obtained by the application of Bayes' rule, which may involve a difficult integral to obtain the normalizing constant. However, if the mixture components in the DPMM belong to the exponential family and conjugate priors are used, then the approximate posterior densities $\pi(\theta_l|\hat{\delta}_{1:i}, y_{1:i})$ are available in closed form and sufficient statistics can be updated recursively upon the assignment of a new observation to a cluster.

The complete SUGS algorithm is thus given by:

Algorithm 1: Sequential update and greedy search (SUGS)

set $\hat{\delta}_1 = 1$;

calculate $\pi(\theta_1|\hat{\delta}_1, y_1)$;

for $i \in \{2, ..., n\}$ **do**

  choose $\hat{\delta}_i = \text{argmax}_{\delta_i \in \{1,...,N_i+1\}} \hat{\pi}_{i,\hat{\delta}_{1:i-1}}(\delta_i|y_{1:i})$;

  update $\pi(\theta_{\hat{\delta}_i}|\hat{\delta}_{1:i-1}, y_{1:i-1})$ using the observation y_i;

return cluster assignments $\hat{\delta}_{1:n}$, approx posterior $\pi(\boldsymbol{\theta}|\hat{\delta}_{1:n}, y_{1:n}) = \prod_{l=1}^{N_n} \pi(\theta_l|\hat{\delta}_{1:n}, y_{1:n})$

The computational efficiency of this algorithm is readily apparent: SUGS requires only a single pass through the data, making a small number of deterministic calculations at each observation. Wang and Dunson [2011] consider extensions to SUGS intended to account for uncertainty in the DP hyperparameter α and the sensitivity of the results to data ordering; these considerations are also relevant for V-SUGS, so they will be discussed at the end of this section.

2.2 Variational Bayes

The authors of the current paper propose to extend SUGS by using probabilistic rather than deterministic cluster assignments within the same recursive assignment and update scheme. While this improves the posterior representation of uncertainty with regard to cluster assignments, it is no

longer guaranteed that the full posterior over cluster assignments and component parameters will be conjugate to the likelihood. The proposed method, V-SUGS, gives a formula for how to choose the posterior distribution over cluster assignments given a new observation, but we are then faced with the challenge of finding the updated posterior over the component parameters. The authors suggest a variational approach to approximating this posterior; here we give a brief review of this framework.

Variational Bayes methods are a set of techniques for approximating the complex distributions that often arise in Bayesian inference [Jordan et al., 1999]. Consider a generic setting for Bayesian inference, in which we have model parameters $\xi \in \Xi \subset \mathbb{R}^d$ and observations y, and we wish to find $p(\xi|y)$ given a prior density $p(\xi)$ and likelihood $p(y|\xi)$. Broadly, the idea of variational Bayes is to approximate this potentially complex posterior density by a density $\pi(\xi)$ restricted to belong to some simpler, more structured class. In particular, if we divide ξ into blocks $\xi_1, ..., \xi_k$, we require that we can write $\pi(\xi) = \prod_{j=1}^{k} q(\xi_j)$.

Given a known form for the approximating densities $q(\xi_j)$, a particular $\pi(\xi)$ with the above form is obtained by minimization of the Kullback-Leibler divergence $KL(q||p) = \int \log\left(\frac{\pi(\xi)}{p(\xi|y)}\right)\pi(\xi)d\xi$, where minimization of this quantity is equivalent to maximizing a lower bound on the log-marginal likelihood (sometimes called the evidence lower bound, or ELBO)

$$L(q) = \int \log\left(\frac{p(\xi)p(y|\xi)}{\pi(\xi)}\right)\pi(\xi)d\xi. \tag{2}$$

The choice of $KL(q||p)$ as a distance criterion yields a simple form for the optimal $q(\xi_j)$:

$$q(\xi_j) \propto \exp\{\mathbb{E}_{-q(\xi_j)} \log p(\xi)p(y|\xi)\}, \tag{3}$$

where $\mathbb{E}_{-q(\xi_j)}$ denotes expectation with respect to $\prod_{i\neq j} q(\xi_i)$. In practice this result can be used to define a simple gradient descent algorithm for minimizing $KL(q||p)$, in which initial values are chosen for the factors of $\pi(\xi)$ and then iteratively updated according to the above equation.

2.3 The V-SUGS algorithm

In the current paper, Zhang et al. [2014] extend SUGS by incorporating uncertainty in the assignment of observations to clusters; the idea is to make a better choice than the greedy hard assignment of SUGS but to retain the recursive structure that makes it so computationally efficient. The authors

propose to replace the hard cluster assignments in SUGS with an uncertain or "soft" assignment over the available clusters, subsequently taking a variational approach to the problem of recursively updating an approximation to the posterior. The goal of their algorithm, V-SUGS, is to approximate the joint posterior distribution over the cluster assignments δ and component parameters θ. Taking a variational view of the same recursive updating scheme as SUGS, they suppose that at time $i-1$ (or equivalently, after seeing $i-1$ observations) we are given an approximate posterior of the form

$$\pi_{i-1}(\delta_{1:i-1}, \theta_{1:i-1}|y_{1:i-1}) = \prod_{j=1}^{i-1} q_{i-1}(\delta_j) \prod_{l=1}^{T} q_{i-1}(\theta_l).$$

The objective is then to define a procedure that will update this approximate posterior to $\pi_i(\delta_{1:i}, \theta_{1:i}|y_{1:i})$ given a new observation y_i, where this approximation takes the form $\prod_{j=1}^{i} q_i(\delta_j) \prod_{l=1}^{T} q_i(\theta_l)$.

It should be noted that the product over component parameters is capped at T terms, which conflicts with the infinite capacity property of the DPMM. Restriction of the maximum number of mixture components yields a *truncated Dirichlet process mixture model*. Such a model is defined analogously to the original DPMM, but the number of distinct values of θ is limited by some integer $T > 1$. The truncated Dirichlet process has a generalized Polya urn representation as $p(\delta, \theta)$, where now

$$p(\theta) = \prod_{l=1}^{T} f_0(\theta_l)$$

and $p(\delta)$ is recursively defined with $p(\delta_1 = 1) = 1$ and

$$p(\delta_i|\delta_{1:i-1}) = \begin{cases} \frac{n_j^{(i)} + \alpha/T}{\alpha + i - 1} & j \in \{1, ..., N_i\}, \\ \frac{\alpha(1 - N_j/T)}{\alpha + i - 1} & j = N_i + 1. \end{cases}$$

Hence the probability of assigning an observation to a new cluster is 0 if there already exist T active clusters (ie, $N_i = T$). The authors note that a variety of existing inference methods make use of truncated DPMMs; nevertheless we observe that truncation is very important in this particular method, as it is responsible for controlling the time and storage complexity of the V-SUGS algorithm.

As with SUGS, the update to V-SUGS can be decomposed into two steps: first, make some choice for $q_i(\delta_i)$, the approximate posterior cluster assignment of the i^{th} observation given $y_{1:i}$, and second, compute $q_i(\theta_l)$ for each cluster given $q_i(\delta_i)$.

The recursion is started by setting $q_1(\delta_1 = 1) = 1$ and updating $q_1(\theta_1)$ as the posterior over θ_1 given prior $f_0(\theta)$ and likelihood $p(y_1|\theta)$. Thus after the first step we have

$$\pi_1(\theta, \delta_1|y_1) = \mathbb{1}_{\{\delta_1=1\}} p(\theta_1|y_1, \delta_1 = 1) \prod_{l=2}^{T} f_0(\theta_l).$$

In keeping with the requirement of sequential structure, the choice at time i for $q_i(\delta_i)$ is not revisited at future times. Thus at time i, V-SUGS sets $q_i(\delta_j) = q_{i-1}(\delta_j)$ for $j \in \{1, ..., i-1\}$, and for $q_i(\delta_i)$ the authors take

$$q_i(\delta_i = l) \propto q_{il} \int p(y_i|\theta_{\delta_i}) q_{i-1}(\theta_{\delta_i}) d\theta_{\delta_i}, \tag{4}$$

where $l \in \{1, ..., \min(i, T)\}$ and we define

$$q_{il} = \begin{cases} \frac{\sum_{j=1}^{i-1} q_{i-1}(\delta_j = l) + \alpha/T}{\alpha + i - 1} & j \in \{1, ..., \min(i, T)\}, \\ \frac{\alpha(1 - \min(i-1, T)/T)}{\alpha + i - 1} & j = \min(i, T) + 1. \end{cases} \tag{5}$$

The idea here is that $q_i(\delta_i)$ approximates the true posterior

$$\begin{aligned} p(\delta_i|y_{1:i}) &\propto p(\delta_i|y_{1:i-1}) p(y_i|\delta_i, y_{1:i-1}) \\ &= p(\delta_i|y_{1:i-1}) \int p(y_i|\theta_{\delta_i}) p(\theta_{\delta_i}|y_{1:i-1}) d\theta_{\delta_i}, \end{aligned}$$

with q_{il} an approximation of $p(\delta_i|y_{1:i-1})$ and $\int p(y_i|\theta_{\delta_i}) q_{i-1}(\theta_{\delta_i}) d\theta_{\delta_i}$ approximating $\int p(y_i|\theta_{\delta_i}) p(\theta_{\delta_i}|y_{1:i-1}) d\theta_{\delta_i}$.

Given this choice of $q_i(\delta_i)$ the $q_i(\theta_l)$ are computed via the variational update

$$\begin{aligned} q_i(\theta_l) &\propto q_{i-1}(\theta_l) \exp\left(\mathbb{E}_{-q_i(\theta_l)} \left(\sum_{k=1}^{\min(i,T)} \mathbb{1}_{\{\delta_i = k\}} \log(p(y_i|\theta_k)) \right) \right) \\ &\propto q_{i-1}(\theta_l) p(y_i|\theta_l)^{q_i(\delta_i = l)}, \end{aligned} \tag{6}$$

which can be simply interpreted as splitting the likelihood contribution of the i^{th} observation across the active clusters according to the assignment weights $q_i(\delta_i)$. The full algorithm for V-SUGS can thus be concisely summarized as follows:

Algorithm 2: V-SUGS

set $q_1(\delta_1 = 1) = 1$;

calculate $\pi_1(\theta, \delta_1 | y_1) = \mathbb{1}_{\{\delta_1=1\}} p(\theta_1 | y_1, \delta_1 = 1) \prod_{l=2}^{T} f_0(\theta_l)$;

for $i \in \{2, ..., n\}$ **do**

take $q_i(\delta_j) = q_{i-1}(\delta_j)$ for $j \in \{1, ..., i-1\}$;

choose $q_i(\delta_i = l) \propto q_{il} \int p(y_i | \theta_{\delta_i}) q_{i-1}(\theta_{\delta_i}) d\theta_{\delta_i}$;

for $l \in \{1, ..., T\}$ **do**

update $q_i(\theta_l) \propto q_{i-1}(\theta_l) p(y_i | \theta_l)^{q_i(\delta_i = l)}$

return soft cluster assignments $q_n(\delta_{1:n})$, truncated posterior $\prod_{l=1}^{T} q_n(\theta_l)$

It is clear from the definition of the two algorithms that SUGS and V-SUGS are closely related. In particular, it is possible to recover SUGS from the definition of V-SUGS simply by taking $q_i(\delta_i) = \mathbb{1}_{\{\delta_i = \hat{\delta}_i\}}$, where $\hat{\delta}_i = \text{argmax}_{\delta_i \in \{1,...,N_i+1\}} \hat{\pi}_{i, \hat{\delta}_{1:i-1}}(\delta_i | y_{1:i})$, as in Algorithm 1.

2.3.1 Derivation for DP mixture of Gaussians

In this section we give a derivation for the the V-SUGS algorithm applied to a Dirichlet process mixture of univariate Gaussians; this is the model implemented for the simulation study in the results section, and its multivariate extension (which simply uses a Wishart instead of Gamma prior for the mixture component variance) is used for the applications to biological data.

In this case, we have that the mixture component parameters can be written as $\theta_l = (\mu_l, \tau_l)$, where μ_l is the mean and τ_l the precision of the l^{th} component. The observations are distributed as $y_i | \delta, \theta \overset{ind}{\sim} \mathcal{N}(\mu_{\delta_i}, \tau_{\delta_i}^{-1})$, and for the sake of conjugacy we choose a normal gamma prior for θ_l, so that $f_0(\theta_l) = \mathcal{N}(\mu_l; \rho, \nu\tau_l^{-1}) Ga(\tau_l; a, b)$. This choice of prior guarantees that the approximate posterior $q_i(\theta_l)$ over θ_l will also be normal gamma, so at time i each mixture component can be represented simply by the four parameters $\rho_l^{(i)}, \nu_l^{(i)}, a_l^{(i)}, b_l^{(i)}$.

At time i, computing the soft assignment for observation i consists of computing the values of a categorical distribution over $\min(i, T)$ mixture components. From Eq. 4 we have that these values are obtained as the product of the truncated DP predictive distribution and an approximation to the likelihood of observation y_i under each mixture component given the previous observations. The truncated DP predictive distribution is straightforward to calculate from Eq. 5. The likelihood term can be thought of as the posterior predictive probability for y_i given the posterior

over θ_{δ_i} at time $i-1$; when $p(y_i|\theta_{\delta_i})$ is normal and $q_{i-1}(\theta_{\delta_i})$ is normal gamma with parameters $\rho_{\delta_i}^{(i-1)}, \nu_{\delta_i}^{(i-1)}, a_{\delta_i}^{(i-1)}, b_{\delta_i}^{(i-1)}$, this evaluates to a t density evaluated at y_i:

$$\int p(y_i|\theta_{\delta_i}) q_{i-1}(\theta_{\delta_i}) d\theta_{\delta_i} = t_{2a_{\delta_i}^{(i-1)}}(y_i; \rho_{\delta_i}^{(i-1)}, (b_{\delta_i}^{(i-1)}/a_{\delta_i}^{(i-1)})(\nu_{\delta_i}^{(i-1)}+1)), \tag{7}$$

where $t_d(y; l, s^2)$ denotes a t density with d degrees of freedom, location l and scale parameter s.

After calculating the soft assignments $q_i(\delta_i)$, we must compute the update to the component parameters. The solution to the variational update given in Eq. 6 shows that we should choose $q_i(\theta_l) \propto q_{i-1}(\theta_l) p(y_i|\theta_l)^{q_i(\delta_i=l)}$. Without the exponent $q_i(\delta_i = l)$ on the likelihood term, this is simply the posterior distribution given observation y_i and prior $q_{i-1}(\theta_l)$; given our choice of a conjugate normal gamma prior, this reduces to a simple and commonly derived set of updates (see for example Bishop [2006]) to the component parameters $\rho_l^{(i-1)}, \nu_l^{(i-1)}, a_l^{(i-1)}, b_l^{(i-1)}$. Conveniently, the exponent can be absorbed into the Gaussian likelihood term as a weight. The parameter updates can thus be derived in the same way as usual, but in this case the posterior for component l incorporates an observation of fractional sample size $q_i(\delta_i = l)$. The resulting updates to the component parameters are given by

$$\nu_l^{(i)} = ((\nu_l^{(i-1)})^{-1} + q_i(\delta_i = l))^{-1}, \tag{8}$$

$$\rho_l^{(i)} = \nu_l^{(i)}((\nu_l^{(i-1)})^{-1}\rho_l^{(i-1)} + q_i(\delta_i = l) y_i), \tag{9}$$

$$a_l^{(i)} = a_l^{(i-1)} + \frac{q_i(\delta_i = l)}{2}, \tag{10}$$

$$b_l^{(i)} = b_l^{(i-1)} + \frac{1}{2}\left(q_i(\delta_i = l) y_i^2 + \frac{\rho_l^{(i)^2}}{\nu_l^{(i)}} - \frac{\rho_l^{(i-1)^2}}{\nu_l^{(i-1)}} \right). \tag{11}$$

2.4 Extensions to the algorithm

After their introduction of SUGS, Wang and Dunson [2011] discuss two extensions to mitigate undesirable sources of instability in the results, namely the choice of the Dirichlet process parameter α and the fundamental sensitivity of sequential algorithms to the ordering of the data. These concerns are echoed and likewise addressed by Zhang et al. [2014], as V-SUGS can be expected to suffer from the same issues. Here we give an outline of the authors' approach to addressing these issues, focusing on the particular solutions proposed for V-SUGS.

2.4.1 Choosing the DPMM hyperparameter α

As is evident from the predictive distribution in (1) derived by Blackwell and MacQueen [1973], the Dirichlet process parameter α determines the probability with which the next observation from a DP-distributed random measure takes a new value instead of some previously observed value. Correspondingly, as a hyperparameter in a Dirichlet process mixture model α controls the propensity of the model to assign data to a new cluster over one of the existing clusters. Clearly, the choice of α will thus have an impact on the clusterings obtained by a DPMM, but in most contexts it seems unlikely that a user will be able to offer any convincing justification for this choice.

To deal with this issue, Wang and Dunson [2011] discuss a modification to SUGS that allows for some uncertainty in the choice of α while preserving the form of the algorithm. Here, a discrete prior over α is specified over a grid of values $\{\alpha_k^*\}_{k=1}^K$ with weights $w_k = Pr(\alpha = \alpha_k^*)$. Then letting $\phi_k^{i-1} = Pr(\alpha = \alpha_k^* | y_{1:i-1})$ and writing $\tilde{q}_i(\delta_i)$ to denote the cluster assignment probabilities obtained after marginalizing out α, the update for $\tilde{q}_i(\delta_i)$ is obtained by integrating over the conditional posterior for α given observations $y_{1:i-1}$:

$$\tilde{q}_i(\delta_i = l) = \frac{\sum_{k=1}^K \phi_k^{i-1} q_i(\delta_i = l)}{\sum_{k=1}^K \phi_k^{i-1} \sum_{l=1}^T q_i(\delta_i = l)},$$

and the posterior over α at time i, denoted $\phi_k^i = Pr(\alpha = \alpha_k^* | y_{1:i})$ for $k \in \{1, ..., K\}$, is obtained as

$$\phi_k^i = \frac{\sum_{l=1}^T \phi_k^{i-1} q_i(\delta_i = l)}{\sum_{s=1}^K \sum_{l=1}^T \phi_s^{i-1} q_i(\delta_i = l)}.$$

2.4.2 Addressing order dependence

The sequential treatment of observations in both SUGS and V-SUGS results in a dependence of the final approximate posterior on the order in which the observations are processed. This is an unappealing property, as the Dirichlet process mixture model itself treats the data as exchangeable. For both SUGS and V-SUGS, the proposed solution is to run the algorithm on multiple permutations of the same data, keeping the results that scored highest according to some approximation of the marginal likelihood. While this is clearly slower than a single pass over the data, the authors argue that the algorithm is fast enough that it remains comparatively inexpensive from a computational standpoint to make multiple runs. However, neither paper suggests any heuristics for choosing the number of permutations or evaluating when a given number of permutations might be satisfactory.

The SUGS algorithm uses a pseudo-marginal likelihood criterion to score results on a given permutation of the data; Wang and Dunson [2011] give empirical evidence that this quantity works better in practice than marginal likelihood, whose use can lead to overfitting and poor predictive performance. From the variational perspective of V-SUGS, the natural choice for scoring results is $L(q)$, the evidence lower bound for the log-marginal likelihood (see Eq. 2). This quantity can be computed recursively over a pass through the data, so it fits easily within the structure of the existing V-SUGS algorithm. In particular, we can think of the posterior at time $i-1$ as the prior to be combined with the likelihood for the i^{th} observation, since $p(\delta_i, \theta|y_{1:i}) \propto p(\delta_i, \theta|y_{1:i-1})p(y_i|\theta_{\delta_i})$. For the Dirichlet process mixture of univariate Gaussians, we approximate $p(\delta_i|y_{1:i-1})$ by q_{il} as in Eq. 5 and $p(\theta|y_{1:i-1})$ by the product of normal gamma terms $\prod_{l=1}^{T} q_{i-1}(\theta_l)$. The contribution of the observation y_i to the evidence lower bound is thus calculated from Eq. 2 as

$$
\begin{aligned}
L(q_i) = \sum_{l=1}^{T} \Bigg\{ & (a_l^{(i)} - a_l^{(i-1)})\psi(a_l^{(i)}) - \log(\Gamma(a_l^{(i)})) + \log(\Gamma(a_l^{(i-1)})) \\
& + a_l^{(i-1)}(\log(b_l^{(i)}) - \log(b_l^{(i-1)})) + a_l^{(i)} \frac{b_l^{(i-1)} - b_l^{(i)}}{b_l^{(i)}} \\
& + \frac{(\rho_l^{(i)} - \rho_l^{(i-1)})^2}{2\nu_l^{(i-1)}} \frac{a_l^{(i)}}{b_l^{(i)}} + \frac{1}{2}\left(\frac{\nu_l^{(i)}}{\nu_l^{(i-1)}} - 1 - \log\left(\frac{\nu_l^{(i)}}{\nu_l^{(i-1)}} \right) \right) \Bigg\} + \sum_{l=1}^{T} q_i(\delta_i = l) \\
& \times \left\{ \frac{1}{2}\psi(a_l^{(i)}) - \frac{1}{2}\log(b_l^{(i)}) - \frac{1}{2}\log(2\pi) - \frac{1}{2}\left\{ \nu_l^{(i)} + (y_i - \rho_l^{(i)})^2 \frac{b_l^{(i)}}{a_l^{(i)}} \right\} \right\} \\
& - \sum_{l=1}^{\min(i,T)} q_i(\delta_i = l)\log(q_i(\delta_i = l)) + \sum_{l=1}^{\min(i,T)} q_i(\delta_i = l)\log(q_{il}),
\end{aligned}
$$

where $\psi(\cdot)$ is the digamma function. We observe that the first sum is obtained as the expectation of the log ratio of the approximate distributions over component parameters $q_i(\theta)$ and $q_{i-i}(\theta)$, the second sum as the expectation of the likelihood, and the third and fourth sums as the expectation of the log ratio of the predictive distribution q_{il} and the approximate posterior over assignments $q_i(\delta_i)$, where all expectations are taken with respect to $q_i(\delta, \theta) = \prod_{l=1}^{\min(i,T)} q_i(\delta_l) \prod_{j=1}^{T} q_i(\theta_j)$.

3 Results

In their results section, Zhang et al. [2014] compare their V-SUGS method against only SUGS and a standard Gibbs sampler for DPMMs. The authors claim that it suffices to compare V-SUGS

to SUGS, as the original SUGS paper showed that SUGS performed competitively against other sequential methods for approximation of the posterior in a DPMM; they treat Gibbs sampling as a computationally expensive but reliable means of generating a "ground truth" in cases where the true distribution of the data is not known. Given the relevance of SUGS to the present paper and its results, we implement both SUGS and V-SUGS and apply them to both simulated and real biological data.

3.1 Simulation Results

The simulation study aims to compare the V-SUGS, SUGS, and Gibbs sampling in terms of computational cost and accuracy on a relatively small data set drawn from a known mixture of univariate Gaussians. Following the simulation study of Wang and Dunson [2011], the authors of the present paper generate observations from the three component mixture model

$$y_i \overset{iid}{\sim} \frac{2}{5}\mathcal{N}(-du, 0.25) + \frac{3}{10}\mathcal{N}(0, 0.5) + \frac{3}{10}\mathcal{N}(du, 2),$$

where $0 \leq du \leq 5$ controls the degree to which the mixture components are separated.

SUGS and V-SUGS are first compared in terms of the computational cost of a single run on a simulated data set of size $N = 500$; this cost is studied as a function of the DP hyperparameter α and (in the case of V-SUGS) the truncation point T (see Table 1). Throughout the simulation study, we follow the authors in using the hyperparameter settings $\rho = 0$, $\nu = 10$, $a = 1$, $b = 1/10$, with a single "run" of SUGS or V-SUGS defined to use 50 different permutations of the data.

	α	0.1	1	10	50
SUGS		12.28	25.85	1458.23	2280.36
V-SUGS	$T = 10$	38.32	40.73	40.05	42.88
	$T = 500$	185.26	171.71	168.76	182.44
	$T = 100$	317.41	318.70	312.10	311.07

Table 1: Computational cost (seconds) of one data set with $N = 500$ for SUGS and V-SUGS.

Our implementations of SUGS and V-SUGS appear to to be slower than that of the authors by a factor of 4-5 for SUGS and 3-4 for V-SUGS. Importantly, however, our implementations exhibit

the same scaling properties as observed by the authors; specifically, SUGS grows rapidly in computational cost as α increases, while the computational cost of V-SUGS is constant in α but grows linearly with T. As expected, both algorithms scale linearly in the length of the data. The constant factor difference in computation cost between our implementations and those of the authors can be traced to two differences in computing setup. First, the authors give hardware specifications that indicate they used a slightly faster machine than the one used for our implementation. Second, and more importantly, the authors implemented their algorithms in Matlab, while we implemented SUGS and V-SUGS in Python. Matlab uses dynamic compilation routines to accelerate looped code chunks where possible, whereas Python lacks any such feature; for sequential algorithms such as SUGS and V-SUGS, this can be expected to yield noticeable differences in speed of execution, even for identically implemented code.

We proceed to study the relative accuracy of V-SUGS and SUGS for density estimation on the simulated data. The authors quantify accuracy by computing

$$e = \frac{\sum_{i=1}^{N}(\hat{f}(y_i) - f(y_i))^2}{var(\hat{f})},$$

where $\hat{f}(y_i)$ and $f(y_i)$ are the predictive and true densities, respectively, evaluated at the observed data points y_i. The paper contains no discussion of how $\hat{f}$ is computed for either SUGS or V-SUGS. Our derivation for these quantities follows a related calculation in West and Escobar [1993]. Given the (soft or hard) cluster assignments obtained from SUGS or V-SUGS, we obtain the conditional probability of cluster assignment for a new observation from Eq. 1. For each of these possible clusters, the predictive density is a t-density, as in Eq. 7. The full predictive density is thus given by a mixture of t-densities with parameters that depend on the active model components and the prior, and whose weights correspond to the predictive distribution over cluster assignment given the observed data.

[INSERT DENSITY ESTIMATION FIGURE]

Finally, the authors compare clustering results for SUGS, V-SUGS, and a standard Gibbs sampler applied to the simulated data, again across a range of values for α and $d\mu$. Gibbs sampling was conducted in R using DPpackage [Jara et al., 2011], an optimized general-purpose package for posterior simulation in a broad class of Bayesian nonparametric models.

[INSERT CLUSTERING RESULTS TABLE]

3.2 Flow Cytometry Data

The first application to real data uses a flow cytometry data set from Manolopoulou et al. [2010]. Flow cytometry is a method by which specific protein structures are detected on individual cell surfaces by fluorescence; a typical run can measure multiple fluorescent markers for millions of cells in a matter of minutes. In this data set, we have 50,000 observations of human peripheral blood cells, each containing measurements for six different markers: forward scatter (which measures cell size), side scatter (which measures cell granularity), CD4 (a marker for helper T cells), IFNg+IL-2 (a marker for effector cytokines), CD8 (a marker for cytotoxic T cells), and CD3 (a marker for all T cells). The data are modeled as arising from a mixture of six-dimensional multivariate Gaussian components. For the current paper, the objective is to compare the performance of SUGS and V-SUGS in terms of density estimation and clustering under this model.

Adaptation of SUGS and V-SUGS to the multivariate setting is straightforward for a Dirichlet process mixture of Gaussians: we replace the normal gamma prior with a normal Wishart prior and derive posterior updates analogous to those in Eqs. 8-11.

[INSERT FLOW CYTO FIGS]

4 Conclusion

The V-SUGS algorithm proposed by Zhang et al. [2014] offers fast approximate inference in Dirichlet process mixture models by sequentially making soft allocations of observations to clusters and computing a variational update to the component parameters. As an extension of SUGS, it enhances the posterior representation of uncertainty over cluster assignments while retaining the recursive structure that is the source of its computational efficiency. This yields particularly meaningful improvements in density estimation and clustering accuracy when the true distribution of the data features clusters that are close together. The authors do not directly compare V-SUGS to any methods apart from SUGS and standard Gibbs sampling, relying instead on indirect comparisons via the experiments of Wang and Dunson [2011], who show that SUGS outperforms other sequential methods for inference in DPMMs. However, continued developments in both sampling methodology and approximate inference techniques for DPMMs mean that these comparisons would have been somewhat dated in 2014, when V-SUGS was published. Even so, the

authors' contribution of a sequential variational approach to posterior approximation has provided a useful foundation for further research; Lin [2013] and Campbell et al. [2015] have subsequently extended this idea such that the truncation limit no longer needs to be fixed and updates can be performed in a distributed and asynchronous fashion. It remains for future work to address more thoroughly some of the deeper challenges associated with the sequential variational approach, including sensitivity to data ordering and the lack of theoretical controls on error via risk or regret bounds.

References

Christopher Bishop. Pattern Recognition and Machine Learning. Springer, 2006.

David Blackwell and James B MacQueen. Ferguson distributions via pólya urn schemes. The annals of statistics, pages 353–355, 1973.

David M Blei and Michael I Jordan. Variational inference for dirichlet process mixtures. Bayesian analysis, 1(1):121–143, 2006.

Alexandre Bouchard-Côté, Arnaud Doucet, and Andrew Roth. Particle gibbs split-merge sampling for bayesian inference in mixture models. arXiv preprint arXiv:1508.02663, 2015.

Christopher A Bush and Steven N MacEachern. A semiparametric bayesian model for randomised block designs. Biometrika, 83(2):275–285, 1996.

Trevor Campbell, Julian Straub, John W Fisher III, and Jonathan P How. Streaming, distributed variational inference for bayesian nonparametrics. In Advances in Neural Information Processing Systems, pages 280–288, 2015.

Jason Chang and John W Fisher III. Parallel sampling of dp mixture models using sub-cluster splits. In Advances in Neural Information Processing Systems, pages 620–628, 2013.

Hal Daumé III. Fast search for dirichlet process mixture models. In Conference on Artificial Intelligence and Statistics, 2007.

Thomas S Ferguson. A bayesian analysis of some nonparametric problems. The annals of statistics, pages 209–230, 1973.

Sonia Jain and Radford M Neal. Splitting and merging components of a nonconjugate dirichlet process mixture model. Bayesian Analysis, 2(3):445–472, 2007.

Alejandro Jara, Timothy E Hanson, Fernando A Quintana, Peter Müller, and Gary L Rosner. Dppackage: Bayesian semi-and nonparametric modeling in r. Journal of Statistical Software, 40 (5):1, 2011.

Michael I Jordan, Zoubin Ghahramani, Tommi S Jaakkola, and Lawrence K Saul. An introduction to variational methods for graphical models. Machine learning, 37(2):183–233, 1999.

Dahua Lin. Online learning of nonparametric mixture models via sequential variational approximation. In Advances in Neural Information Processing Systems, pages 395–403, 2013.

Albert Y Lo. On a class of bayesian nonparametric estimates: I. density estimates. The Annals of Statistics, 12(1):351–357, 1984.

Steven N MacEachern and Peter Müller. Estimating mixture of dirichlet process models. Journal of Computational and Graphical Statistics, 7(2):223–238, 1998.

Ioanna Manolopoulou, Cliburn Chan, and Mike West. Selection sampling from large data sets for targeted inference in mixture modeling. Bayesian Analysis, 5(3):1, 2010.

Thomas Minka and Zoubin Ghahramani. Expectation propagation for infinite mixtures. In NIPS Workshop on Nonparametric Bayesian Methods and Infinite Models, volume 19, 2003.

Radford M Neal. Markov chain sampling methods for dirichlet process mixture models. Journal of Computational and Graphical Statistics, 9(2):249–265, 2000.

Jim Pitman. Combinatorial stochastic processes. Technical report, Technical Report 621, Dept. Statistics, UC Berkeley, 2002.

Erik B Sudderth. Graphical models for visual object recognition and tracking. PhD thesis, MIT, 2006.

Lianming Wang and David B Dunson. Fast bayesian inference in dirichlet process mixture models. Journal of Computational and Graphical Statistics, 20(1):196–216, 2011.

Mike West and Michael D Escobar. Hierarchical priors and mixture models, with application in regression and density estimation. Institute of Statistics and Decision Sciences, Duke University, 1993.

Xiaole Zhang, David J Nott, Christopher Yau, and Ajay Jasra. A sequential algorithm for fast fitting of dirichlet process mixture models. Journal of Computational and Graphical Statistics, 23(4):1143–1162, 2014.