JMLR: Workshop and Conference Proceedings 45:237–252, 2015 ACML 2015

Streaming Variational Inference for Dirichlet Process Mixtures

Viet Huynh HVHUYNH@DEAKIN.EDU.AU
Dinh Phung DINH.PHUNG@DEAKIN.EDU.AU
Svetha Venkatesh SVETHA.VENKATESH@DEAKIN.EDU.AU
Pattern Recognition and Data Analytics Centre, Deakin University, Australia

Editor: Geoffrey Holmes and Tie-Yan Liu

Abstract

Bayesian nonparametric models are theoretically suitable to learn streaming data due to their complexity relaxation to the volume of observed data. However, most of the existing variational inference algorithms are not applicable to streaming applications since they require truncation on variational distributions. In this paper, we present two truncation-free variational algorithms, one for mix-membership inference called TFVB (truncation-free variational Bayes), and the other for hard clustering inference called TFME (truncation-free maximization expectation). With these algorithms, we further developed a streaming learning framework for the popular Dirichlet process mixture (DPM) models. Our experiments demonstrate the usefulness of our framework in both synthetic and real-world data.

1. Introduction

We are at the dawn of a new revolution in the Information Age: *data*. "Every animate and inanimate object on Earth will soon be generating data" (Smolan and Erwitt, 2013). While we collectively are tweeting 8000 messages around the world every second, our homes, cars, cities and even our bodies are also constantly generating terabytes of signals. A common setting is that these data are collected sequentially in time and our modern machine learning tools need algorithms to learn from data stream without the need to revisit past data. More importantly, not only data are getting bigger in size, but also are their *growing* complexity, structure, and geometry. Hence, dealing with streaming data require flexible models that can expand with data size and complexity. Bayesian nonparametric (BNP) models naturally fit this purpose since their complexity, e.g., the number of mixture components, can grow as new data appear. One challenge, however, for Bayesian models in general and Bayesian nonparametric models in particular is that it lacks efficient inference methods to deal with large scale and streaming data.

Two main inference approaches for BNP models are simulation methods such as Markov Chain Mote Carlo (MCMC) and deterministic variational methods. To deal with streaming data, sequential MCMC and particle MCMC were developed. However, MCMC algorithms are often unable to cope with large-scale data sets due to its slow convergence and unpredictable convergence diagnosis. On the other pillar, deterministic variational inference is preferred in large-scale settings. Significant efforts on scalable variational learning with

nonparametric Bayesian models have been made recent years (Hoffman et al., 2013; Blei and Jordan, 2006; Kurihara et al., 2006, 2007; Tank et al., 2015). Following this trend, in this paper we seek a variational method which can handle streaming data.

The first variational inference for a fundamental building block in Bayesian nonparametric models, the Dirichlet process mixture (DPM), was developed by Blei et al. (Blei and Jordan, 2006). Later works by Kurihara et al. (Kurihara et al., 2007) attempted to develop collapsed variational version for the DPM models. However, these works employed variational distribution with truncation technique which limited the number of clusters to be fixed . By using truncation, these techniques possess a technical limitation in growing model capacity with data – a key feature of nonparametric modelling – and hence can not be applied to a streaming setting.

To circumvent the problem of truncation, Kurihara et al. (Kurihara et al., 2006) suggested to compute the evidence lower bound (ELBO)[1] of variational approximation as criteria to increase the number of clusters. This strategy usually induces excessive computational burden. More recent works tried to avoid truncation by using simulation in each variational iteration (Wang and Blei, 2012) and heuristic (Lin, 2013). In this paper, we used a different strategy to circumvent truncation with lightweight computational cost. Furthermore, by using maximization expectation scheme proposed by Welling et al. (Welling and Kurihara, 2006), the truncation problem can effectively solved.

In terms of streaming algorithms for nonparametric models, a recent work by Tank at al. (Tank et al., 2015) is based on expectation propagation (EP) approximation in which instead of minimizing KL divergence from variational distribution q to posterior distribution p, $KL\left(q \mid p\right)$, it optimizes the reserve KL divergence $KL\left(p \mid q\right)$. However, it is noticed by Broderick et al. (Broderick et al., 2013) that EP-based optimization is extremely computational demand, hence much less efficient. Works by Sudderth et al. (Bryant and Sudderth, 2012; Hughes and Sudderth, 2013) introduced an online learning algorithm for Bayesian nonparametric models based on stochastic variational inference framework which inherits limitation in terms of defining number of data points in advance. In this paper, for streaming algorithm, we adapt the recent framework from (Broderick et al., 2013) and use proposed truncation-free variational inference to introduce a new streaming algorithm for Dirichlet process mixture models.

Our main contributions in the paper include: (1) truncation-free variational framework for learning with Bayesian nonparametric models, particularly Dirichlet process mixture models with exponential family derivation solutions; (2) streaming learning algorithms which can leverage the "expanding complexity with data" nature of for Bayesian nonparametric models; (3) an application of image analysis which can be learned on the fly with streaming data.

2. Background

Now we review exponential family distributions, Dirichlet process mixture models (DPM) and variational Bayes inference scheme which are background for our algorithms in the section 3.

1. Some authors, including Kurihara, call this term as free energy.

2.1. Exponential family

Let x be a random variable taking value in the domain $\mathcal{X}$. Let T be a vector valued function $T : \mathcal{X} \to \mathbb{R}^d$ so that $T(x)$ is d-dimension vector. Let θ, also in $\mathbb{R}^d$, denote the parameter and $\langle x, y \rangle$ be the inner product between two vectors x, y in $\mathbb{R}^d$. The random variable x follows exponential family if the probability density function has the following form:

$$p(x \mid \theta) = \exp\left\{\langle \theta, T(x) \rangle - A(\theta) + \ln h(x)\right\}, \tag{1}$$

where $A(\theta)$, called log-partition function, is simply a normalization term to make $p(x \mid \theta)$ sum up to one and $h(x)$ is called base measure. The first order partial derivative of log-partition function is the expectation of sufficient statistic $T(x)$, i.e.

$$\frac{\partial A(\theta)}{\partial \theta} = \mathbb{E}\left[T(x)\right]_{p(x|\theta)} \tag{2}$$

This property is useful when working with variational Bayes and we use it to derive our results in subsequent development.

The conjugate prior $p(\theta \mid \eta)$ can be expressed in an exponential family form with the sufficient statistics $\{\theta, -A(\theta)\}$ as follows

$$p(\theta \mid \eta) = \exp\left\{\langle \eta_c, \theta \rangle - \eta_\sigma A(\theta) - B(\eta)\right\}, \tag{3}$$

where $\eta = \{\eta_c, \eta_\sigma\}$ is the hyperparameters. Note that $\eta_c \in \mathbb{R}^d$, η_σ is a scalar, and thus $\eta \in \mathbb{R}^{d+1}$.

2.2. Dirichlet process mixture

A Dirichlet process $\mathrm{DP}(\alpha, H)$ is a distribution of a random probability measure G over the measurable space $(\Theta, \mathcal{B})$ where H is a *base* probability measure and $\alpha > 0$ is the *concentration* parameter. It is defined such that, for any finite measurable partition $(A_k : k = 1, \ldots, K)$ of Θ, the resultant random vector $(G(A_1), \ldots, G(A_k))$ is distributed according to a Dirichlet distribution with parameters $(H(A_1), \ldots, H(A_k))$. In 1994, Sethuraman (Sethuraman, 1994) provided an alternative constructive definition which makes the discreteness property of a Dirichlet process explicitly via a stick breaking construction. This is useful while dealing with infinite parameter space and defined as

$$\begin{aligned} G &= \sum_{k=1}^{\infty} \beta_k \delta_{\theta_k} \text{ where } \theta_k \overset{\text{iid}}{\sim} H,\ k = 1, \ldots, \infty \\ \boldsymbol{\beta} &= (\beta_k)_{k=1}^{\infty},\ \beta_k = v_k \prod_{s<k} (1 - v_s) \text{ with } v_k \overset{\text{iid}}{\sim} \mathrm{Beta}(1, \alpha) \end{aligned} \tag{4}$$

It can be shown that $\sum_{k=1}^{\infty} \beta_k = 1$ with probability one, and as a convention in (Pitman, 2002), we hereafter write $\boldsymbol{\beta} \sim \mathrm{GEM}(\alpha)$.

Due to its discreteness, the Dirichlet process is often not applied directly to model data (e.g., it is unable to model continuous data) instead it can be effectively used as a nonparametric prior on the mixture components θ, which in turn serves as the parameters within another likelihood function F to generate data - a model which is known as Dirichlet

process mixture model (DPM) (Antoniak, 1974; Escobar and West, 1995). To be precise, under a DPM formalism an observation x_n is generated from a two-step process: $x_n \sim F(x_n \mid \theta_n)$ where $\theta_n \sim G$. Using the stick-breaking representation in Equation (4), it is not hard to see that DPM yields an infinite mixture model representation:

$$p(x \mid \alpha, H) = \sum_{k=1}^{\infty} \beta_k f(x \mid \theta_k), \tag{5}$$

where f denotes the density function for F. Dirichlet process mixture models have been embraced with a great success and enthusiasm recently (Gelfand et al., 2005; Neal, 2000). The crucial advantage is its ability to naturally address the problem of model selection - a major obstacle encountered in several parametric mixture modeling, such as the Gaussian mixture models of which number of mixtures cannot be specified a priori in a principal way.

Given the DPM model as described in Equation (4) and (5) and there exist K clusters with n_k elements in k-th cluster after observing n data point $x_1, \ldots, x_n$ generated from the model. The posterior distribution of stick breaking $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_K, \beta_{new})$ is

$$(\beta_1, \ldots, \beta_K, \beta_{new}) \sim \text{Dir}(n_1, \ldots, n_K, \alpha) \tag{6}$$

There is a nice property of Dirichlet distribution called aggregation which justifies our algorithm in the next section. If $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_K, \beta_{new})$ follows Equation (6) then

$$\left(\beta_1, \ldots, \beta_L, \sum_{l=L+1}^{K} \beta_l + \beta_{new}\right) \sim \text{Dir}\left(n_1, \ldots, n_L, \sum_{l=L+1}^{K} n_l + \alpha\right) \text{ for any } 1 \leq L \leq K \tag{7}$$

2.3. Variational inference

Let consider generative model with n observed i.d.d variables $x_1, \ldots, x_n$ which are supposed to be generated from n latent variables $z_1, \ldots, z_n$ and parameters θ. Latent variables z_i's are assumed to be continuous (when they are discrete, the integration will be replaced by the summation). In Bayesian settings, we suppose that parameters θ are distributed with prior distributions $p(\theta \mid \eta)$. Our objective in Bayesian inference is to compute the posterior $p(z, \theta \mid x, \eta)$. In variational inference scheme, instead of computing $p(z, \theta \mid x, \eta)$ directly, we use approximation distribution $q(z, \theta)$, usually called variational distribution. Let consider KL divergence between approximation distribution $q(z, \theta)$ and target distribution $p(z, \theta \mid x, \eta)$, denoted $KL(q \mid p)$, now our goal is to find $q(z, \theta)$ that minimize

$$\begin{aligned} KL(q \mid p) &= -\int q(z, \theta) \ln \frac{p(z, \theta \mid x, \eta)}{q(z, \theta)} dz d\theta \\ &= -\int q(z, \theta) \ln \frac{p(x, z, \theta \mid \eta)}{q(z, \theta) p(x \mid \eta)} dz d\theta \\ &= -\mathcal{F}(q(z, \theta), p(x, z, \theta \mid \eta)) + p(x \mid \eta). \end{aligned}$$

In above equation, the term $p(x \mid \eta)$ is constant with respect to q. Therefore, maximization $\mathcal{F}(q(z, \theta), p(x, z, \theta \mid \eta))$ means $KL(q \mid p)$ minimization.

Mean-field approximation for mixture models

If our model are mixture models with K components $\theta_{1:K}$ and we assume the mean-field independence

$$q(z,\theta) = \prod_{i=1}^{n} q_{z_i}(z) \prod_{k=1}^{K} q_{\theta_k}(\theta),$$

then

$$\begin{aligned}\mathcal{F}(q(z,\theta), p(x,z,\theta \mid \eta)) &= \int q(z,\theta) \ln \frac{p(x,z,\theta \mid \eta)}{q(z,\theta)} dz d\theta \\ &= -\mathrm{KL}(q_{z_i}(z), \tilde{p}(x, z_i \mid \eta))\end{aligned}$$

where $\ln \tilde{p}(x, z_i) = \mathbb{E}[\ln p(x,z,\theta \mid \eta)]_{q_{\theta_{1:K}} q_{z_{-i}}}$.

Here, in order to reduce the clutter, we denote

$$q_{\theta_{1:K}} \triangleq q_{\theta_1}(\theta) \cdots q_{\theta_K}(\theta) \qquad \text{and} \qquad q_{-z_i} \triangleq q_{z_1}(z) \cdots q_{z_{i-1}}(z)\, q_{z_{i+1}}(z) \cdots q_{z_n}(z)$$

This term is maximized with respect to $q_{z_i}(z)$ when

$$q_{z_i}(z) = \tilde{p}(x, z_i \mid \eta) = \exp\left(\mathbb{E}[\ln p(x,z,\theta \mid \eta)]_{q_{\theta_{1:K}} q_{-z_i}}\right) \tag{8}$$

Similarly, keeping other variables fixed, except $q_{\theta_k}(\theta)$, $\mathcal{F}(q(z), q(\theta))$ is maximized when

$$q_{\theta_k}(\theta) \propto p(\theta \mid \eta) \exp\left(\mathbb{E}[\ln p(x_i \mid \theta_k)]_{q_{z_{1:n}}}\right)$$

Maximization expectation approximation

Using the maximization expectation framework in (Welling and Kurihara, 2006), let's consider a special case when $q_{z_i}(z) = \delta\left(z_i - z^{MAP}\right)$, it means that we choose the point estimation for z_i. The Equation (8) becomes

$$z_i = \underset{z}{\mathrm{argmax}} \mathbb{E}[\ln p(x, z_{-i}, \theta \mid \eta)]_{q_{\theta_{1:K}}},$$

while approximation for θ's is remained the same.

3. Proposed inference framework

In this section, we describe two variational inference algorithms for Dirichlet process mixture models in which no truncation is needed. One algorithm is developed by using mean field approximation while the other follows maximization expectation approximation. In order to develop streaming algorithm for DPM, we follow the framework in (Broderick et al., 2013) which was developed for parametric models. Since our derivation for algorithms is for mixture of exponential family, we also describe results for case studies of mixture of Multinomial and multivariate isotrophic Gaussian distributions.

Algorithm 1: Truncation-free Variational Bayes for DPM

Input: $x_1, \ldots, x_n$
Output: $\{q_{z_i}(z)\}_{i=1}^{n}, \{q_{\theta_k}(\theta)\}_{k=1}^{K}$
$K \leftarrow 1$ and initialize $\{q(z_i)\}_{i=1}^{n}$ with dimension $K+1$;
repeat
  for $k \leftarrow 1$ **to** $K+1$ **do**
    Estimate $q_{\theta_k}(\theta)$ using Equation (9) ;
  end
  for $i \leftarrow 1$ **to** n **do**
    for $k \leftarrow 1$ **to** $K+1$ **do**
      Compute $\mathbb{E}[n_k]$ using Equation (14) ;
      Compute $q_{z_i}(z)$ using Equation (11);
    end
  end
  if $\mathbb{E}[n_{K+1}] > 1$ **then**
    $K = K + 1$;
    Increase dimension for $q_{z_i}(z)$ and set $q_{z_i}(K+1) = 0$;
  end
until *Convergence*;
Normalize $\{q_{z_i}(z)\}_{i=1}^{n}$ with K dimensions

3.1. Truncation-free variational inference

3.1.1. Mean field approximation

Let consider DPM model as in Equation (5). Suppose we have n observations $x_1, \ldots, x_n$ from exponential family $f(x \mid \theta)$, each x_i is associated with a latent indicator z_i which gets value from 1 to K where K is some value between 1 and n. The variables to be inferred are z_i's, θ_k's. The mean field approximate distribution is assumed to take the form:

$$q(z_{1:n}, \theta_{1:K}) = \prod_{k=1}^{K} q_{\theta_k}(\theta \mid \eta) \prod_{i=1}^{n} q_{z_i}(z),$$

where $q_{\theta_k}(\theta \mid \eta)$ is also an exponential family which is a conjugate prior for the likelihood $f(x \mid \theta)$ and K is the optimized number of clusters.

Following a standard procedure for variational inference we have the expectation for parameter variables

$$q_{\theta_k}(\theta) \propto p(\theta \mid \eta) \exp\left(\sum_{i=1}^{n} q_{z_i}(k) \ln p(x_i \mid \theta_k)\right), \tag{9}$$

which has closed form if we use a conjugate distribution in the exponential family:

$$q_{\theta_k}(\theta) \propto \exp\left(\left\langle \theta, \tilde{\eta}_c^k \right\rangle - \tilde{\eta}_\sigma^k A(\theta) - B(\eta)\right),$$

where $\tilde{\eta}_c^k = \eta_c + \sum_{i=1}^{n} q_{z_i}(k) T(x_i)$ and $\tilde{\eta}_\sigma^k = \eta_\sigma + \sum_{i=1}^{n} q_{z_i}(k)$.

For $q_{z_i}(z)$, we compute as follows

$$q_{z_i}(z) \propto \exp\left(\mathbb{E}\left[\ln p\left(z \mid z_{-i}, x_{1:n}, \theta_{1:K+1}\right)\right]_{q_{\theta_{1:K+1}} q_{z^{-i}}}\right) \tag{10}$$

Here $q_{z_i}(z)$ has its support from 1 to $K+1$. The value $q_{z_i}(z = K+1)$ means probability that data point x_i belong to some cluster(s) which might express more about current data point. We note that we do not actually know the value of K but we can estimate it during optimization process (e.g., we start with K=1 and let it grow). In each iteration we consider to increase the value of K by comparing $\mathbb{E}(n_{K+1}) > 1$, i.e. the number of data points in new cluster are greater than one. The justification for choosing $\mathbb{E}(n_{K+1})$ as a criteria to change the value of K is given in the supplementary document.

The expectation for hidden variables, $q_{z_i}(z)$, can be computed as follows. First, we have

$$\begin{aligned}\ln p\left(z \mid z_{-i}, x_{1:n}, \theta_{1:K}\right) &= \ln p\left(x_i \mid z, z_{-i}, x_{-i}, \theta_{1:K}\right) + \ln p\left(z \mid z_{-i}, x_{-i}, \theta_{1:K}\right) + \text{const} \\ &= \ln f\left(x_i \mid \theta_z\right) + \ln p\left(z \mid z_{-i}\right) + \text{const}\end{aligned}$$

Therefore,

$$\begin{aligned}&\exp\left(\mathbb{E}\left[\ln p\left(z \mid z_{-i}, x_{1:n}, \theta_{1:K}\right)\right]_{q_{\theta_{1:K}} q_{-z_i}}\right) \\ &= \exp\left(\mathbb{E}\left[\ln f\left(x_i \mid \theta_z\right) + \ln p\left(z \mid z_{-i}\right) + \text{const}\right]_{q_{\theta_{1:K}} q_{-z_i}}\right) \\ &\propto \exp\left(\mathbb{E}\left[\ln f\left(x_i \mid \theta_z\right)\right]_{q_{\theta_z}} + \mathbb{E}\left[\ln p\left(z \mid z_{-i}\right)\right]_{q_{-z_i}}\right)\end{aligned} \tag{11}$$

The first term in Equation (11) can be computed as follows (for $k = 1, \ldots, K$, we use q_{θ_k} estimated in previous steps, otherwise we use prior $q_{\theta_k}(\theta) = q(\theta)$)

$$\begin{aligned}\mathbb{E}_{q_{\theta_k}}\left\{\ln f\left(x_i \mid \theta\right)\right\} &= \mathbb{E}_{q_{\theta_k}}\left\{\left\langle \theta, T\left(x_i\right)\right\rangle - A\left(\theta\right)\right\} \\ &= \left\langle T\left(x_i\right), \mathbb{E}_{q_{\theta_k}}\left\{\theta\right\}\right\rangle - \mathbb{E}_{q_{\theta_k}}\left\{A\left(\theta\right)\right\} \\ &= \left\langle T\left(x_i\right), \frac{\partial B\left(\eta\right)}{\partial \eta_c}\right\rangle - \frac{\partial B\left(\eta\right)}{\partial \eta_\sigma}.\end{aligned} \tag{12}$$

The last equation is obtained by using property of exponential family in Equation 2 with distribution $q_{\theta_k}(\theta \mid \eta)$.

To compute the second term in Equation (11), we first estimate $p(z_i = k \mid z_{-i})$. Using Chinese restaurant process, i.e., π in the stick-breaking representation in Equation (6) is marginalized out, we have

$$p\left(z \mid z_{-i}\right) = \begin{cases} \frac{n_k^{-i}}{n-1+\alpha} & \text{if } z = 1, \ldots, K \\ \frac{\alpha}{n-1+\alpha} & z > K. \end{cases} \tag{13}$$

The second term inside Equation (11) can be represented as

$$\mathbb{E}\left[\ln p\left(z \mid z_{-i}\right)\right]_{q_{-z_i}} = \sum_{z_{-i}} \prod_{j \neq i} q\left(z_j\right) \ln p\left(z_i = k \mid z_{-i}\right).$$

Since this term could be too slow to compute due to exponentially complication of z_i's, using approximation technique in (Kurihara et al., 2007), we can compute the above equation as follows. Let consider n_k as a random variable which sums over Bernoulli variables $n_k = \sum_{i=1}^{n} \mathbf{1}(z_i = k)$. Under the central limit theorem, this sum is expected to be closely approximated by a Gaussian distribution with mean and variance given by

$$\begin{aligned} \mathbb{E}[n_k] &= \sum_{i=1}^{n} q_{z_i}(z = k) \\ \mathrm{Var}[n_k] &= \sum_{i=1}^{n} q_{z_i}(z = k)(1 - q_{z_i}(z = k)) \quad k = 1, \ldots, K+1 \end{aligned} \tag{14}$$

Using the following second order Taylor expansion for the moments of functions of random variables with $f(n_k) = \ln p(z_{-i})$ at the value $\mathbb{E}\left[n_k^{-i}\right]$, i.e.

$$f''(n_k) = \begin{cases} -\frac{1}{\left(n_k^{-i}\right)^2} & \text{if } k \leq K \\ -\frac{1}{\left(n_{K+1}^{-i}+\alpha\right)^2} & k = K+1, \end{cases}$$

Finally, the second term (11) can be approximated as follows:

$$\mathbb{E}[\ln p(z \mid z_{-i})] = \begin{cases} \ln \frac{\mathbb{E}\left[n_k^{-i}\right]}{n-1+\alpha} - \frac{1}{2}\frac{\mathrm{Var}\left[n_k^{-i}\right]}{\left(\mathbb{E}\left[n_k^{-i}\right]\right)^2} & \text{if } k \leq K \\ \ln \frac{\alpha}{n-1+\alpha} & k = K+1. \end{cases} \tag{15}$$

We can summarize inference routine as in algorithm 1.

3.1.2. Maximization expectation approximation

Now we derive an inference algorithm with Maximization Expectation framework for DPM inference. The procedure is similar to VB case, except that Equation (10) now becomes

$$\begin{aligned} z_i =& \operatorname*{argmax}_k \mathbb{E}[\ln p(z = k \mid z_{-i}, x_{1:n}, \theta_{1:K})]_{q_{\theta_{1:K}} q_{-z_i}} \\ =& \operatorname*{argmax}_k \left(\mathbb{E}[\ln f(x_i \mid \theta_k)]_{q_{\theta_k}} + \mathbb{E}[\ln p(z = k \mid z_{-i})]_{q_{-z_i}}\right) \end{aligned} \tag{16}$$

where $\mathbb{E}[\ln f(x_i \mid \theta_k)]_{q_{\theta_k}}$ is computed in Equation (12) and

$$\mathbb{E}[\ln p(z = k \mid z_{-i})]_{q_{-z_i}} \propto \begin{cases} \ln n_k & \text{if } k \leq K \\ \ln \alpha & k = K+1, \end{cases}$$

while the Equation (9) can be manipulated to:

$$q_{\theta_k}(\theta) \propto p(\theta \mid \eta)\, p\left(\{x\}^k \mid \theta\right), \tag{17}$$

where $\{x\}^k = \{x_i \mid z_i = k\}$.

Algorithm 2 summarizes our Maximization Expectation inference algorithm for DPM.

Algorithm 2: Truncation-free Maximization Expectation for DPM

Input: $x_1, \ldots, x_n$
Output: $\{q_{z_i}(z)\}_{i=1}^n, \{q_{\theta_k}(\theta)\}_{k=1}^K$
$K \leftarrow 1$ and initialize $z_i = 1$ for all $i = 1, \ldots, n$;
repeat
  for $k \leftarrow 1$ **to** K **do**
    Estimate $q_{\theta_k}(\theta)$ using Equation (17) ;
  end
  for $i \leftarrow 1$ **to** n **do**
    Compute z_i using quation (16);
    if $z_i = K + 1$ **then**
      $K = K + 1$;
      Intialize a new $q_{\theta_{K+1}}(\theta) = p(\theta)$;
    end
  end
until *Convergence*;
Normalize $\{q_{z_i}(z)\}_{i=1}^n$ with K dimensions

3.2. Streaming learning with DPM

We use the framework proposed by Broderick (Broderick et al., 2013) to develop streaming variational inference and maximization expectation algorithms for DPM.

Consider i.i.d. data stream $x_1, x_2, \ldots$ generated from (infinite) mixture models $p(x \mid z, \theta)$ with prior $p(z, \theta)$ and $C_1 \triangleq \{x_1, x_2, \ldots, x_{S_1}\}$ be the first batch of data. Suppose that we have seen and processed $b-1$ batches of data from which we obtained posterior distribution $p(z, \theta \mid C_1, \ldots, C_{b-1})$, denoted as $p^{(b-1)}(z, \theta)$. We can compute the posterior after the b-th batch as

$$\begin{aligned} p(z, \theta \mid C_1, \ldots, C_b) &\propto p(C_b \mid z, \theta)\, p(z, \theta \mid C_1, \ldots, C_{b-1}) \\ &= p(C_b \mid z, \theta)\, p^{(b-1)}(z, \theta). \end{aligned}$$

This means that we treat the posterior after observing $b-1$ batches as a new prior for incoming b-th batch of data. In our model, we approximate $p(z, \theta \mid C)$ by an approximation $q(z, \theta)$ using variational Bayes in or ME. Therefore the posterior $p^{(b)}(z, \theta)$ can be recursively calculated as follows [2]

$$p^{(b)}(z, \theta) \approx q^{(b)}(z, \theta) = \mathcal{A}\left(C_b, q^{(b-1)}(z, \theta)\right),$$

where $\mathcal{A}(C, q^{prior})$ is the algorithm to approximate the posterior q^{post} with prior q^{prior} which can be Algorithm 1 or 2 and $q^{(1)}(z, \theta) = \mathcal{A}(C_1, p(z, \theta))$. We summarize this inference algorithm in the Algorithm 3.

2. Here we re-use the notation in (Broderick et al., 2013).

Algorithm 3: Streaming inference for DPM

Input: $C_1, \ldots, C_B, \ldots$ where C_b: one batch (S data points) in stream data $\{x_i\}_{i=1}^{S}$
Output: $\{z_i\}_{i=1}^{n}$ or $\{q_{z_i}(z)\}_{i=1}^{n}, \{q_{\theta_k}(\theta)\}_{k=1}^{K}$
$b = 1$;
Approximate $q^{(b)}(z, \theta)$ using Algorithm 1 or 2 with prior $p(z, \theta) = q^{(b-1)}(z, \theta)$;
while *there is more data* **do**
  $b = b + 1$ and collect new data batch C_b;
  Approximate $q^{(b)}(z, \theta)$ using Algorithm 1 or 2 with prior $p(z, \theta) = q^{(b-1)}(z, \theta)$;
end

3.3. Case study: mixture model of Multinomial and isotropic Gaussian

Mixture model of Multinomial

Suppose that $f(x_i \mid \theta)$ follows a d-dimension Multinomial distribution with parameter $\theta = (\ln\theta_1, \ldots, \ln\theta_d)$ in exponential family, $T(x_i) = (x_{i1}, \ldots, x_{id})$ and $A(\theta) = 0$. The parameter θ follows a conjugate prior Dirichlet distribution which is also exponential family with hyperparameter $\eta_c = [\eta_1 - 1, \ldots, \eta_d - 1]$ and $B(\eta) = \sum_{i=1}^{d} \Gamma(\eta_i) - \Gamma\left(\sum_{i=1}^{d} \eta_i\right)$[3].

$$p(\theta \mid \eta) = \exp\left(\langle \eta_c, \theta \rangle - \left(\sum_{i=1}^{d} \ln\Gamma(\eta_i) - \ln\Gamma\left(\sum_{i=1}^{d} \eta_i\right)\right)\right)$$

The posterior estimation for η^k of component k is

$$\tilde{\eta}_c^k = \left[\eta_1 - 1 + \sum_{i=1}^{n} q_{ik} x_{i1}, \ldots, \eta_d - 1 + \sum_{i=1}^{n} q_{ik} x_{id}\right]$$
$$B\left(\tilde{\eta}^k\right) = \sum_{i=1}^{d} \ln\Gamma(\tilde{\eta}_{ci}) - \ln\Gamma\left(\sum_{i=1}^{d} \tilde{\eta}_{ci}\right)$$

where $q_{ik} = q_{z_i}(k)$, i.e.

$$q_{\theta_k}(\theta) \propto \exp\left(\left\langle \tilde{\eta}_c^k, \theta \right\rangle - B\left(\tilde{\eta_c}^k\right)\right).$$

Therefore,

$$\begin{aligned}
\mathbb{E}_{q_{\theta_k}}\{\ln f(x_i \mid \theta)\} =& \mathbb{E}_{q_{\theta_k}}\{\langle T(x_i), \theta \rangle - A(\theta)\} + \ln h(x_i) \\
=& \left\langle T(x_i), \frac{\partial B(\tilde{\eta})}{\partial \tilde{\eta}} \right\rangle + \ln h(x_i) \\
=& \langle T(x_i), m(\tilde{\eta}) \rangle + \ln h(x_i) \qquad (18)
\end{aligned}$$

where

$$m(\tilde{\eta}) = [m(\tilde{\eta}_1), \ldots, m(\tilde{\eta}_d)] \text{ s.t. } m(\tilde{\eta}_i) = \psi(\tilde{\eta}_i) - \psi(\tilde{\eta})$$

3. η_σ can be any value since $A(\theta) = 0$

and

$$\ln h\left(x_i\right) = \ln \Gamma\left(\sum_{i=1}^{d} x_i + 1\right) - \sum_{i=1}^{d} \ln \Gamma\left(x_i + 1\right).$$

Here, $\psi\left(\cdot\right)$ is digamma function.

Mixture model of multivariate isotropic Gaussian

Now we consider the case that $f\left(x_i \mid \theta\right)$ is multivariate isotropic Gaussian, which can be represented in exponential family as follows (here $\tau = \sigma^{-1}$ is unknown precision parameter and I is identity matrix of size d).

$$\begin{aligned} p\left(x \mid \mu, \frac{1}{\tau} I\right) &= \frac{1}{(2\pi)^{d/2} \left|\sigma I\right|^{d/2}} \exp\left\{-\frac{1}{2}(x-\mu)^{\mathsf{T}} (\sigma I)^{-1} (x-\mu)\right\} \\ &= \exp\left\{\tau \mu^{\mathsf{T}} x - \frac{\tau}{2} x^{\mathsf{T}} x - \left(\frac{\tau}{2}\mu^{\mathsf{T}}\mu - \frac{d}{2}\ln\tau + \frac{d}{2}\ln 2\pi\right)\right\} \\ &= \exp\left(\left\langle \theta, T\left(x\right)\right\rangle - A\left(\theta\right)\right) \end{aligned}$$

where $\theta = \left[\tau\mu^{\mathsf{T}}, -\frac{\tau}{2}\right]^{\mathsf{T}}$, $A\left(\theta\right) = \frac{\tau}{2}\mu^{\mathsf{T}}\mu - \frac{d}{2}\ln\tau + \frac{d}{2}\ln 2\pi$ and $T\left(x\right) = \left[x, x^{\mathsf{T}}x\right]^{\mathsf{T}}$. The parameter θ follows a conjugate prior multivariate Gaussian-Gamma distribution as follows

$$q\left(\theta \mid \alpha, \beta, \mu_0, \lambda\right) = \mathcal{N}\left(\mu \mid \mu_0, \left(\lambda\tau\right)^{-1} I\right).\mathrm{Gamma}\left(\tau \mid \alpha, \beta\right),$$

which can be represent in exponential form with natural parameters

$$q\left(\theta \mid \eta\right) = \exp\left\{\left\langle \eta, \left[\ln\tau, \tau, \tau\mu, \tau\mu^{\mathsf{T}}\mu\right]\right\rangle - B\left(\eta\right)\right\}$$

where $\eta = \left[\eta_1, \eta_2, \eta_3, \eta_4\right]^{\mathsf{T}}$ is a vector with $d+3$ dimensions with

$$\eta_1 = \alpha - 1 + \frac{d}{2} \qquad \eta_2 = -\beta - \frac{\lambda\mu_0^{\mathsf{T}}\mu_0}{2} \qquad \eta_3 = \lambda\mu_0 \qquad \eta_4 = -\frac{\lambda}{2},$$

and

$$B\left(\eta\right) = \ln\Gamma\left(\eta_1 + 1 - \frac{d}{2}\right) - \left(\eta_1 + 1 - \frac{d}{2}\right)\ln\left(-\eta_2 + \frac{\eta_3^{\mathsf{T}}\eta_3}{4\eta_4}\right) - \frac{d}{2}\ln\left(-2\eta_4\right) + \frac{d}{2}\ln 2\pi$$

Now we have

$$q\left(\theta \mid \alpha, \beta, \mu_0, \lambda\right) = \exp\left\{\left\langle \eta, \left[\ln\tau, \tau, \tau\mu, \tau\mu^{\mathsf{T}}\mu\right]\right\rangle - B\left(\eta\right)\right\}$$

Therefore, using property from Equation 2, we have

$$\mathbb{E}_{q_{\theta_k}}\left[\ln\tau\right] = \frac{\partial B\left(\eta\right)}{\partial\eta_1} = \psi\left(\alpha\right) - \ln\beta \qquad \mathbb{E}_{q_{\theta_k}}\left[\tau\right] = \frac{\partial B\left(\eta\right)}{\partial\eta_2} = \frac{\alpha}{\beta}$$

$$\mathbb{E}_{q_{\theta_k}}\left[\tau\mu\right] = \frac{\partial B\left(\eta\right)}{\partial\eta_3} = \frac{\alpha}{\beta}\mu_0 \qquad \mathbb{E}_{q_{\theta_k}}\left[\tau\mu^{\mathsf{T}}\mu\right] = \frac{\partial B\left(\eta\right)}{\partial\eta_4} = \frac{\alpha\mu_0^{\mathsf{T}}\mu_0}{\beta} + \frac{d}{\lambda}$$

and

$$\mathbb{E}_{q_{\theta_k}}\left\{A\left(\theta\right)\right\} = \frac{1}{2}\left(\frac{\alpha\mu_0^{\mathsf{T}}\mu_0}{\beta} + \frac{d}{\lambda} - \left(\psi\left(\alpha\right) - \ln\beta\right)d + d\ln 2\pi\right)$$

Therefore,

$$\begin{aligned}\mathbb{E}_{q_{\theta_k}}\left\{\ln f\left(x_i \mid \theta\right)\right\} &= \mathbb{E}_{q_{\theta_k}}\left\{\left\langle T\left(x_i\right), \theta\right\rangle - A\left(\theta\right)\right\} \\ &= \left\langle T\left(x_i\right), m\left(\theta\right)\right\rangle - \mathbb{E}_{q_{\theta_k}}\left\{A\left(\theta\right)\right\} \end{aligned} \tag{19}$$

where $m\left(\theta\right) = \left[\mathbb{E}_{q_{\theta_k}}\left[\tau\mu\right], -\frac{\mathbb{E}_{q_{\theta_k}}[\tau]}{2}\right] = \left[\frac{\alpha}{\beta}\mu_0, -\frac{\alpha}{2\beta}\right]$, $T\left(x_i\right) = \left[x_i, x_i^{\mathsf{T}}x_i\right]$ and

Now we can compute $q_{z_i}\left(z\right)$ as follows

$$q_{z_i}\left(k\right) = \frac{\exp\left(\mathbb{E}\left\{\ln f\left(x_i \mid \theta\right)\right\} + \mathbb{E}\left[\ln p\left(z_i = k \mid z_{-i}\right)\right]\right)}{\sum_{k'=1}^{K+1}\exp\left(\mathbb{E}\left\{\ln f\left(x_i \mid \theta\right)\right\} + \mathbb{E}\left[\ln p\left(z_i = k' \mid z_{-i}\right)\right]\right)}$$

where $\mathbb{E}_{q_{\theta_k}}\left\{\ln f\left(x_i \mid \theta\right)\right\}$ computed using Equation (18) and $\mathbb{E}\left[\ln p\left(z_i = k \mid z_{-i}\right)\right]$ computed using Equation (15) .

4. Experiments

In this experiment we demonstrate our algorithms in two settings: dataset in batch mode and streaming (mini-batch). The datasets include synthetic bar topics and real-world data set MNIST[4]. For the base-line methods, we choose two variational methods developed for DPM: non-collapsed in (Blei and Jordan, 2006) and collapsed symmetric Dirichlet priors in (Kurihara et al., 2007)[5], denoted as BJV and FSD, respectively. Two our methods in Algorithms 1 and 2 are denoted as TFVB and TFME.

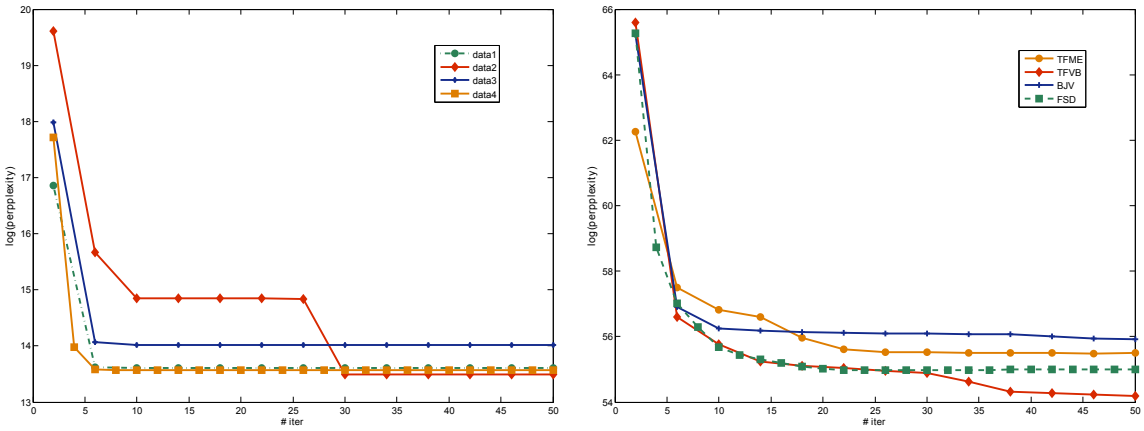
In the batch setting, we will compare inference performance of our algorithms with base-line methods validation. In streaming setting, we will illustrate the capability of our algorithms for learning on the fly with incremental data in which the complexity of models will be learning automatically. All models are implemented using Matlab and ran on Intel i7-3.4GHz machine with installed Windows 7..

4.1. Data sets and experimental settings

Our inference scheme is developed for class of exponential which includes many popular probability distributions. We will demonstrate experiments with two most popular exponential distributions, Multinomial and multivariate Gaussian.

Synthetic bar topics. We sampled 1000 data points from ten 25-dimension bar topics which is shown in top row of Figure 3. Each data point is a sample from 25-dimension Multinomial distribution. We chose 80% of these as training set, the rest was considered as held-out data for testing. We ran each algorithm for 100 iterations. For BJV and FSD

4. http://yann.lecun.com/exdb/mnist
5. In this paper, authors described two methods called TSB and FSD, however, they showed that there is very little difference TSB and FSD.



(*a*) Bar topic data (*b*) MNIST

Figure 1: Perplexity with different algorithms.

methods, we set different truncation levels at $T = 10,\ 12,\ 15,\ 20,\ 25,\ 30$ and choose the best truncation level.

MNIST. We used a subset of 25.000 digits images of this data set. Similar to synthetic data, we also split 80/20% of training-testing data. Each image with 764 (28×28) dimensions was reduced to 50 using PCA. We applied all four algorithms on this data set. We fitted models in which the mixture component is isotropic Gaussian with mean μ and covariance matrix τI. The base measure H and variational distribution $q(\mu, \tau)$ were conjugate priors $\mathcal{N}\left(\mathbf{0}, (5\tau)^{-1} I\right) \times \text{Gamma}(4, 2)$.

For two data sets, we ran in two different settings: batch mode setting in which we ran 4 algorithms and compute perplexity on testing data[6]; streaming setting in which we divided into multiple batches

4.2. Experimental results

Dataset in batch mode

Figure 1(*a*) shows the results on *synthetic bar topics*. The perplexity (lower is better) of TFVB is lowest among four methods however, it is converge slower. At the beginning, we usually set the number of cluster as a small number and let it be increased which will take more time to reach the true number of components in compared with truncation methods. In TFME case, we choose point estimation(hard clustering) for each data point which can easier lead to local minimum. The predictive performance of base-line methods depends on truncation level which is difficult to set in advance with "never ending" data from streaming applications.

The mean digit images of the discovered clusters from *MNIST data* are shown in Figure 2. Figure 1(*b*) depicts perplexity (in log scale) running on MNIST data with 4 methods. Similar to bar topic data, TFME and TFVB also have competitive performance to base-line methods.

6. We explain our methods for computing perplexity with variational inference in supplementary document.

Figure 2: Digit clustering results with MNIST data of TFME algorithm.

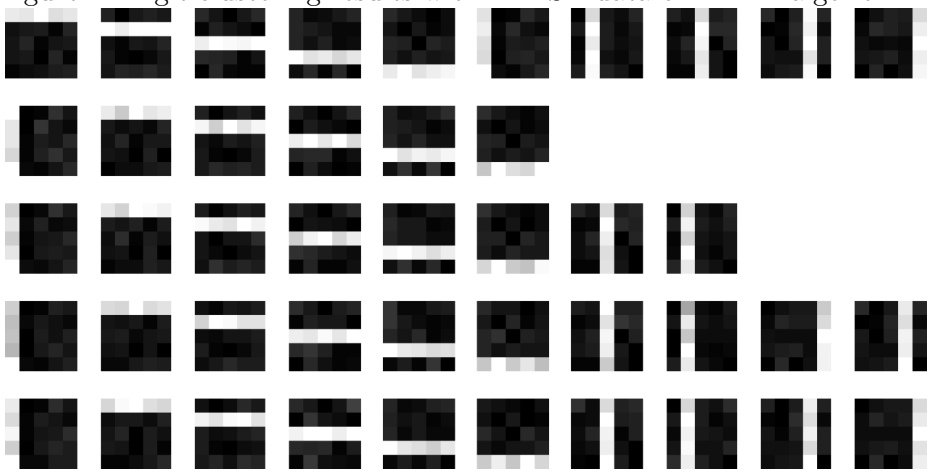


Figure 3: Topics discovered by streaming algorithms with bar topics. The top row is ground truth topics. The second, third and fourth rows are topics discovered when new data comes with new topics in each following batch. The last row depicts topics discovered when observing new data but from old topics.

Streaming setting

For bar topics, we generated data as follows. In the first batch, 1000 data points were generated from only six out of ten topics. In next 3 following batches, data were generated from 8, 10 and 10 topics correspondingly. Our streaming algorithm 3 using TFME and TFVB could discover correctly the number of topics as shown in Figure 3 without defining number of clusters.

For MNIST data, using the same experimental setting with synthetic data, we divided our digit data into 8 batches: 4000 digit images from 0 to 2, further each mini-batches of 3000 images with digit ground truth from 0 to 4,... 0 to 9 (7 more mini-batches). All these batches were run with TFVB and TFME to check clustering results. As shown in Figure 4, with data from the first batch, our algorithm found 7 clusters of six digits. With new data

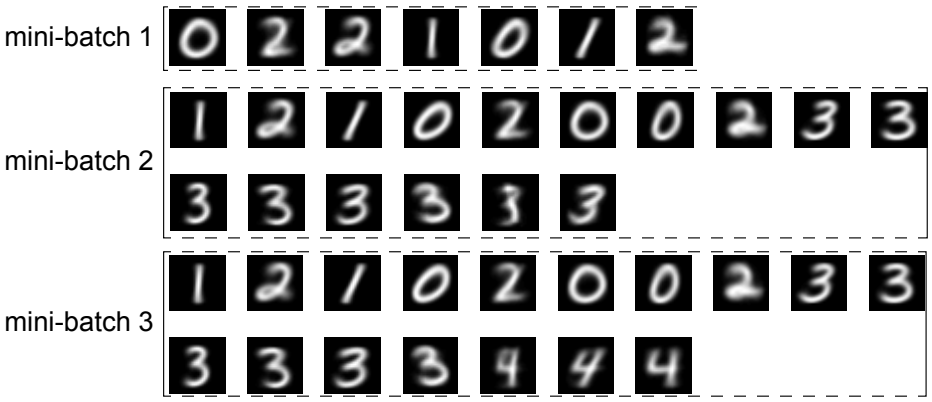


Figure 4: Topics discovered by streaming algorithms with MNIST data set. Mini-batch 1 with 4000 digit images from 0 to 2 with 7 clusters discovered while mini-batch 2 with more 3000 images of 0 to 3 digits and mini-batch 3 with more 3000 images of 5 digits includes more clusters found by our algorithms.

coming from 8 then 10 digits, the algorithms incrementally discovered more clusters without revisiting old data from previous batches. We demonstrate other mini-batches results in supplementary material.

5. Conclusion

We have developed truncation-free variational inference algorithms for the Dirichlet process mixture models and demonstrated its applicability to multi-dimensional data from exponential family distributions. Based on these truncation-free algorithms, we have introduced a streaming framework which can learn on the fly in the true streaming setting where data are never-ending collected. We have demonstrated the advantages of our algorithm in comparison with other methods including collapsed (FSD) and non-collapsed variational (BJV) in terms of automatically learning number of clusters. Incremental learning with our streaming algorithms which leverage the natural property of nonparametric models makes these models more practical. Though our methods are developed for Dirichlet mixture models, extensions to other BNP models, such as Pitman-Yor process (Pitman and Yor, 1997), Hierarchical Dirichlet process (Teh et al., 2006) are straightforward by using Chinese restaurant process representation (Aldous, 1985).

References

David J Aldous. *Exchangeability and related topics*. Springer, 1985.

C.E. Antoniak. Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems. *The Annals of Statistics*, 2(6):1152–1174, 1974.

D.M. Blei and M.I. Jordan. Variational inference for dirichlet process mixtures. *Bayesian Analysis*, 1(1):121–143, 2006.

Tamara Broderick, Nicholas Boyd, Andre Wibisono, Ashia C Wilson, and Michael I Jordan. Streaming variational bayes. In *Advances in Neural Information Processing Systems*, pages 1727–1735, 2013.

Michael Bryant and Erik B Sudderth. Truly nonparametric online variational inference for hierarchical dirichlet processes. In *Advances in Neural Information Processing Systems*, pages 2699–2707, 2012.

M.D. Escobar and M. West. Bayesian density estimation and inference using mixtures. *Journal of the american statistical association*, 90(430):577–588, 1995.

A.E. Gelfand, A. Kottas, and S.N. MacEachern. Bayesian nonparametric spatial modeling with dirichlet process mixing. *Journal of the American Statistical Association*, 100(471):1021–1035, 2005.

Matthew D Hoffman, David M Blei, Chong Wang, and John Paisley. Stochastic variational inference. *The Journal of Machine Learning Research*, 14(1):1303–1347, 2013.

Michael C Hughes and Erik Sudderth. Memoized online variational inference for dirichlet process mixture models. In *Advances in Neural Information Processing Systems*, pages 1133–1141, 2013.

Kenichi Kurihara, Max Welling, and Nikos A Vlassis. Accelerated variational dirichlet process mixtures. In *Advances in Neural Information Processing Systems*, pages 761–768, 2006.

Kenichi Kurihara, Max Welling, and Yee Whye Teh. Collapsed variational dirichlet process mixture models. In *IJCAI*, volume 7, pages 2796–2801, 2007.

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

R.M. Neal. Markov chain sampling methods for Dirichlet process mixture models. *Journal of computational and graphical statistics*, 9(2):249–265, 2000.

J. Pitman. Poisson–Dirichlet and GEM invariant distributions for split-and-merge transformations of an interval partition. *Combinatorics, Probability and Computing*, 11(05):501–514, 2002. ISSN 1469-2163.

Jim Pitman and Marc Yor. The two-parameter poisson-dirichlet distribution derived from a stable subordinator. *Annals of Probability*, 25:855–900, 1997.

J. Sethuraman. A constructive definition of Dirichlet priors. *Statistica Sinica*, 4(2):639–650, 1994.

Rick Smolan and Jennifer Erwitt. *The human face of big data*. Against All Odds Productions, 2013.

Alex Tank, Nicholas Foti, and Emily Fox. Streaming variational inference for bayesian nonparametric mixture models. In *Proc. of Int. Conf. on Artificial Intelligence and Statistics (AISTAT)*, pages 977–985, 2015.

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

Chong Wang and David M Blei. Truncation-free online variational inference for bayesian nonparametric models. In *Advances in neural information processing systems*, pages 413–421, 2012.

Max Welling and Kenichi Kurihara. Bayesian k-means as a "maximization-expectation" algorithm. In *SDM*, pages 474–478. SIAM, 2006.