DIRICHLET PROCESS MIXTURE MODELS WITH MULTIPLE MODALITIES

John Paisley and Lawrence Carin

Duke University
Department of Electrical & Computer Engineering
Durham, NC 27708

ABSTRACT

The Dirichlet process can be used as a nonparametric prior for an infinite-dimensional probability mass function on the parameter space of a mixture model. The set of parameters over which it is defined is generally used for a single, parametric distribution. We extend this idea to parameter spaces that characterize multiple distributions, or modalities. In this framework, observations containing multiple, incompatible pieces of information can be mixed upon, allowing for all information to inform the final clustering result. We provide a general MCMC sampling scheme and demonstrate this framework on a Gaussian-HMM mixture model applied to synthetic and Major League Baseball data.

Index Terms— Dirichlet process, Bayesian hierarchical models, hidden Markov model, Gaussian mixture model

1. INTRODUCTION

The Dirichlet process [1] has proven useful in the machine learning and signal processing communities [2][3][4] as a nonparametric Bayesian prior for mixture models [5]. The infinite extent of the Dirichlet process allows for a robust prior definition on a parameter space that can accommodate an unlimited number of components. The Dirichlet process takes two input parameters: a positive scalar, α, and a base probability measure, G_0. Using Sethuraman's constructive definition [6] and truncating to a level, K, which can be set to produce an arbitrarily small error [7], the full generative process can be written as follows:

$$\begin{aligned} X_i &\sim F(\theta_{c_i}) && (1)\\ c_i &\overset{iid}{\sim} Mult(\boldsymbol{\pi}) && (2)\\ \pi_j &= V_j \prod_{l<j}(1-V_l) && (3)\\ V_j &\overset{iid}{\sim} Beta(1,\alpha) && (4)\\ \theta_j &\overset{iid}{\sim} G_0 && (5) \end{aligned}$$

where $j = 1, \ldots, K$ and π_K is replaced by $1 - \sum_{j=1}^{K-1} \pi_j$. We will use the notation $\boldsymbol{\pi} = \phi_K(\boldsymbol{V})$ to represent the truncated function in (3). The hidden data, $c_i \in \{1, \ldots, K\}$, acts as an indicator of which set of parameters, θ_{c_i}, are used to parameterize the distribution $F(\theta_{c_i})$ from which observation X_i is drawn. Because draws from a Dirichlet process, being the process in (3), (4) and (5), are discrete when $\alpha < \infty$, these parameters will repeat, meaning there will be multiple observations drawn from the same distribution function. This leads to a *clustering* of the data, as two observations that share the same parameters will have similar statistical properties as defined by $F(\theta)$. Therefore, the Dirichlet process has found use in the partitioning of data sets into groups where data within a group is considered similar and data across groups dissimilar. As K is a variable, it can be set to an arbitrarily large number, providing a potentially unlimited number of clusters, or ways in which data can be manifested according to $F(\theta)$. Furthermore, when G_0 is conjugate to $F(\theta)$, inference is fully analytical and straightforward.

Many developments of this framework have been proposed in the literature, e.g. [8][4][2], that vary or add to elements of the generative process of (1)-(5). Each addresses a potential aspect of mixture modeling not accounted for in (1)-(5), but easily handled via slight modifications. We present here our own modification that accounts for the desire to model data sets where each observation is *itself* a data set of multiple modalities, i.e., multiple statistically irreducible distribution functions, $F_m(\theta)$. In this case, each X_i is a set of observations and each contributes to characterizing the object of interest. In such cases where multiple pieces of information are available with which objects can be clustered, it is useful to modify the Dirichlet process to account for *all* information when partitioning data into groups. We call this general framework a *Dirichlet process with product base measure* (DP-PBM) as it requires multiple base measures combined in product form to parameterize the Dirichlet process.

This paper is organized as follows: In Section 2 we present the DP-PBM framework and discuss some of its theoretical properties. In Section 3, we outline a general MCMC inference algorithm for DP-PBM mixture models. Experimental results are given in Section 4, where we focus on a Gaussian-HMM mixture model – one instantiation of the DP-PBM framework. Results are shown for both synthesized and Major League Baseball data for the 2007 season.

2. THE DIRICHLET PROCESS WITH PRODUCT BASE MEASURE

In this section, we discuss a variant of the Dirichlet process that incorporates a product base measure, called a DP-PBM, where rather than drawing parameters for one parametric distribution, $\theta \sim G_0$, parameters are drawn for *multiple* distributions, $\theta_m \sim G_{0,m}$ for $m = 1, \ldots, M$. In other words, rather than having a connection between data, $\{X_i\}_{i=1}^N$, and their respective parameters, $\{\theta_{c_i}\}_{i=1}^N$, through a parametric distribution, $\{F(\theta_{c_i})\}_{i=1}^N$, *sets* of data, $\{X_{1,i}, \ldots, X_{M,i}\}_{i=1}^N$ have respective *sets* of parameters, $\{\theta_{1,c_i}, \ldots, \theta_{M,c_i}\}_{i=1}^N$, used in inherently different and generally incompatible distribution functions, $\{F_1(\theta_{1,c_i}), \ldots, F_M(\theta_{M,c_i})\}_{i=1}^N$.

The DP-PBM is so called because it utilizes a product base measure to achieve this end, $G_0 = G_{0,1} \times G_{0,2} \times \cdots \times G_{0,M}$, where in this case, M modalities are considered. The space over which this process is defined is now $\left(\prod_{m=1}^M \Theta_m, \bigotimes_{m=1}^M \mathcal{B}_m, \prod_{m=1}^M G_{0,m}\right)$. Though this construction implicitly takes place in all mixture models that attempt to estimate multiple parameters, for example the multivariate Gaussian mixture model, we believe our use of these parameters in multiple, incompatible distributions (or modalities) is novel. The full generative process can be written as follows:

$$
\begin{aligned}
X_{m,i} &\sim F_m(\theta_{m,c_i}) && (6)\\
c_i &\overset{iid}{\sim} Mult(\boldsymbol{\pi}) && (7)\\
\pi_j &= V_j \prod_{l<j}(1 - V_l) && (8)\\
V_j &\overset{iid}{\sim} Beta(1, \alpha) && (9)\\
\theta_{m,j} &\sim G_{0,m} && (10)
\end{aligned}
$$

for $m = 1, \ldots, M$, where $\theta_{m,j}$ are drawn iid from $G_{0,m}$ for a fixed m and independently under varying m. Again, this process requires truncation to K,with $\pi_K = 1 - \sum_{j=1}^{K-1} \pi_j$. To make a clarifying observation, we note that if each $G_{0,m}$ is a univariate normal-gamma prior, this model reduces to a multivariate GMM with a forced diagonal covariance matrix. As previously stated, we are more interested in cases where each X_m is inherently incompatible, but is still linked by the structure of the data set.

For example, consider a set of observations, $\{O_i\}_{i=1}^N$, where each $O_i = \{X_{1,i}, X_{2,i}\}$ with $X_1 \in \mathbb{R}^d$ and X_2 a sequence of time-series data. In this case, a single density function, $f(X_1, X_2|\theta_1, \theta_2)$ cannot analytically accommodate O_i, making inference difficult. However, if these densities can be considered as *independent*, that is $f(X_1, X_2|\theta_1, \theta_2) = f(X_1|\theta_1) \cdot f(X_2|\theta_2)$, then this problem becomes analytically tractable and, furthermore, no more difficult to solve than for the standard Dirichlet process. One might choose to model X_1 with a Gaussian distribution, with $G_{0,1}$ the appropriate prior and X_2 by an HMM [9], with $G_{0,2}$ its respective prior. In this case, this model becomes a hybrid Gaussian-HMM mixture, where each component is *both* a Gaussian *and* a hidden Markov model.

2.1. Capturing Correlations Across Modalities

As alluded to in the previous section, the analytical nature of the DP-PBM framework depends upon a factorization of the likelihood function. That is, for the likelihood function of our M-modality data, we assume that we can write $f(X_1, \ldots, X_M|\theta_1, \ldots, \theta_M) = \prod_{m=1}^M f_m(X_m|\theta_m)$, where $f_m(X_m|\theta_m)$ is the likelihood function and θ_m the parameter (or set of parameters) for the m^{th} modality. As will be seen in the next section, inference then becomes analytical, provided the appropriate priors, $p(\theta_m)$, are selected, as the different modalities are all drawn independently conditioned upon the latent indicator, c, which selects the set of parameters for all M distribution functions.

Because of this independence assumption, it might seem that the model will not capture any correlations within the data across modalities. While it is true that this ability is not given to the prior, the *posterior* will capture correlations. For example, given the posterior for N observations, consider an $N + 1^{st}$ observation where the first $M - 1$ modalities are present, but the M^{th} is missing. If we wish to infer its associated latent indicator, c_{N+1} (or which component it came from), we can simply calculate for the first $M - 1$ modalities

$$
P(c_{N+1} = j|\boldsymbol{X}, \boldsymbol{\theta}) \propto \pi_j \prod_{m=1}^{M-1} f_m(x_{m,N+1}|\theta_{m,j}) \qquad (11)
$$

thereby effectively integrating out the M^{th} modality. We see that, given the distribution on c_{N+1}, we can then interpolate or make predictions as to the missing modality, $x_{M,N+1}$. An MCMC inference algorithm in the next section will allow for a closer look at the functioning of the model.

3. MCMC INFERENCE FOR DP-PBM MIXTURE MODELS

In this section, we outline a general method for performing Markov chain Monte Carlo (MCMC) [10] inference for DP-PBM models. We let $f_m(x_m|\theta_m)$ be the likelihood function for the m^{th} modality of an observation given the parameters, θ_m, and $p(\theta_m)$ the prior density of θ_m. For compactness, we refer to the DP-PBM as G (as is typical in the literature), where $G = \sum_{j=1}^{K+1} \pi_j \prod_{m=1}^M \delta_{\theta_{m,j}}$. We also observe that this sampling method is unbounded in the potential number of components, but only requires the K occupied components plus a $K + 1^{st}$ proposal component for any given iteration.

Initialization: Select a truncation level, $K + 1$, and initialize the model, G, by sampling $\theta_{m,k} \sim G_{m,0}$ for $k = 1, \ldots, K + 1$, $m = 1, \ldots, M$ and $V_k \sim Beta(1, \alpha)$ for $k = 1, \ldots, K$ and construct $\boldsymbol{\pi} = \phi_K(\boldsymbol{V})$.

Step 1: Sample the indicators, $c_1, \ldots, c_N$, independently from their respective conditional posteriors, $p(c_j|\{x_{m,1}\}_{m=1}^M) \propto \prod_{m=1}^M f(x_{m,j}|\theta_{m,c_j})p(\theta_{m,c_j}|G)$,

$$c_j \sim \sum_{k=1}^{K+1} \frac{\pi_k \prod_{m=1}^M f_m(x_{m,j}|\theta_{m,k})}{\sum_l \pi_l \prod_{m=1}^M f_m(x_{m,j}|\theta_{m,l})}\delta_k \qquad (12)$$

Set K to be the number of unique values among $c_1, \ldots, c_N$ and relabel from 1 to K.

Step 2: Sample $\{\theta_{m,1}\}_{m=1}^M, \ldots, \{\theta_{m,K}\}_{m=1}^M$ from their respective posteriors conditioned on $c_1, \ldots, c_N$ and $x_1, \ldots, x_N$,

$$\theta_{m,k} \sim p\left(\theta_{m,k}|\{c_j\}_{j=1}^N, x_{m,1}, \ldots, x_{m,N}\right) \qquad (13)$$

$$p\left(\theta_{m,k}|\{c_j\}_{j=1}^N, x_{m,1}, \ldots, x_{m,N}\right) \propto \prod_{j=1}^N \left(f(x_{m,j}|\theta_m)^{\delta_{c_j}(k)}\right) p(\theta_m) \qquad (14)$$

where $\delta_{c_j}(k)$ is a delta function equal to one if $c_j = k$ and zero otherwise, simply picking out which $\{x_{m,j}\}_{m=1}^M$ belong to component k. Sample $\theta_{m,K+1} \sim G_{0,m}$ for $m = 1, \ldots M$. These M posteriors are calculated independently of one another given the relevant data for that modality extracted from the observations assigned to that component. We stress that when an "observation" is assigned to a component (via the indicator, c) it is actually *all* of the data that comprise that observation that is being assigned to the component.

Step 3: Construct the $(K+1)$-dimensional weight vector, $\boldsymbol{\pi} = \phi_K(\boldsymbol{V})$, using $V_1, \ldots, V_K$ sampled from their Beta-distributed posteriors conditioned on $c_1, \ldots, c_N$,

$$V_k \sim Beta\left(1 + \sum_{j=1}^N \delta_{c_j}(k), \alpha + \sum_{l=k+1}^K \sum_{j=1}^N \delta_{c_j}(l)\right) \qquad (15)$$

Set $\pi_{K+1} = \prod_{k=1}^K (1 - V_k)$.

Repeat Steps 1 – 3 for a desired number of iterations. The convergence of this Markov chain can be assessed [10], after which point uncorrelated samples (properly spaced out in the chain) of the values in Steps 1 – 3 are considered iid samples from the posterior. As can be seen, inference for DP-PBM models is fairly straightforward and, when each $p(\theta_m)$ is conjugate to $f(x_m|\theta_m)$, fully analytical.

4. APPLICATIONS: THE GAUSSIAN-HMM MIXTURE MODEL

We look at a concrete example of a DP-PBM model, a Gaussian-HMM mixture model, where modality one is data $X_1 \in \mathbb{R}^d$ and modality two is a sequence drawn from a hidden Markov model [9], $X_2 \sim HMM(A, B, \pi')$. Our experiments are performed on synthesized and Major League Baseball (MLB) data sets.

4.1. Experiment with Synthesized Data

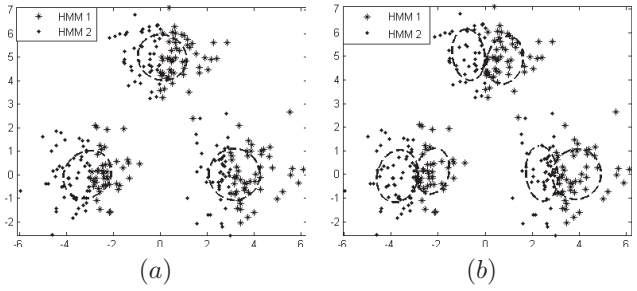
We define three, two-dimensional Gaussian distributions with respective means $\mu_1 = (-3, 0)$, $\mu_2 = (3, 0)$ and $\mu_3 = (0, 5)$ and each having the identity as the covariance matrix. Two hidden Markov models are defined as below,

$$\mathbf{A_1} = \begin{bmatrix} 0.05 & 0.9 & 0.05 \\ 0.05 & 0.05 & 0.9 \\ 0.9 & 0.05 & 0.05 \end{bmatrix} \quad \mathbf{A_2} = \begin{bmatrix} 0.9 & 0.05 & 0.05 \\ 0.05 & 0.9 & 0.05 \\ 0.05 & 0.05 & 0.9 \end{bmatrix}$$

$$\mathbf{B_1}, \mathbf{B_2} = \begin{bmatrix} 0.9 & 0.05 & 0.05 \\ 0.05 & 0.9 & 0.05 \\ 0.05 & 0.05 & 0.9 \end{bmatrix}$$

with the initial state vector $\boldsymbol{\pi'}_1, \boldsymbol{\pi'}_2 = [1/3, 1/3, 1/3]$. Data was generated as follows: We sampled 300 observations, 100 from each Gaussian, constituting $X_{1,i}$ for $i = 1, \ldots, 300$. For each sample, if the observation was on the right half of its respective Gaussian, a sequence of length 50 was drawn from HMM 1, if on the left, from HMM 2. For display purposes, we select a typical sample from MCMC inference.

This precisely defined data set allows the model to clearly display the benefits of its design. If one were to build a Gaussian mixture model on the X_1 data alone, three components would be uncovered, as shown in Figure 1(a). If an HMM mixture were built alone on the X_2 data, only two components would be uncovered. Using *all* of the data, that is, mixing on $\{O_i\}_{i=1}^{300}$ rather than just $\{X_{1,i}\}_{i=1}^{300}$ or $\{X_{2,i}\}_{i=1}^{300}$ alone, the correct number of six components was uncovered, as shown in Figure 1(b).



(a) (b)

Fig. 1. An example of a mixed Gaussian-HMM data set. (a) Gaussian mixture model results. (b) Gaussian-HMM mixture results. Each ellipse corresponds to a cluster.

The results show that, as was required by the data, the DP-PBM prior uncovered six distinct clusters of data. The DP-PBM framework allowed for the incorporation of *all* information of the data set to be included, thus providing more precise clustering results.

4.2. Major League Baseball Data Set

Using the complete bat-by-bat statistics for the 2007 season[1], we processed our data set as follows. We created a 3-dimensional vector, X_1, of the batting average, on-base percentage and slugging percentage. We then quantized the plate appearances for a given player into the following codes: 1. Strikeout, 2. Fielded out, 3. Hit, where walks and other results were ignored. We limited our set to the 252 players with a sequence length greater than 300. For MCMC, we used 1000 burn-in and 3000 burn-out iterations and selected an iteration of median likelihood for presentation below. We also show results for an HMM mixture model [11] without using the spatial data. The component membership results, or the number of observations that were assigned to a given indexed component, are shown in Figure 2 for both models. We see that using additional information produces a more refined clustering result, as was the case in the synthetic result.

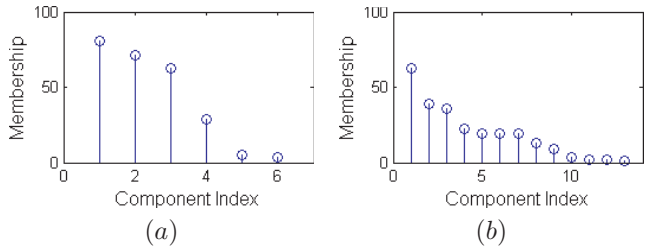


Fig. 2. Component membership results for MLB data when (a) X_1 data is ignored – the HMM mixture model. (b) both X_1 and X_2 data is used – the Gaussian-HMM mixture model.

We next ask whether the increase in the number of clusters results in a more precise and informative clustering result. To do this we consider two measures, first the average differential entropy of the Gaussian component, where we empirically calculated the covariance from the HMM mixture results.

$$h_{avg}(X_1) = \sum_{i=1}^{K} \pi_i \frac{1}{2} \ln \left((2\pi e)^3 |\Sigma_i| \right) \tag{16}$$

We recall that differential entropy can be negative and that $h_{avg}(X_1) \to -\infty$ as the uncertainty tends to zero. Using this measure for the HMM mixture, $h_{avg}(X_1) = -6.31$, while for the Gaussian-HMM mixture, $h_{avg}(X_1) = -7.11$, indicating that the Gaussian-HMM more precisely represented the spatial information, thus improving clustering.

As a second measure, we consider the average entropy of each HMM, which is estimated using the original data

$$H_{avg}(X_2) = -\sum_{i=1}^{K} \pi_i \sum_{n=1}^{N_i} \frac{1}{N_i} \ln P(X_{2,\rho_i(n)} | A_i, B_i, \pi_i') \tag{17}$$

[1]Data was obtained from www.retrosheet.org

where N_i is the number of data in component i, with $\rho_i(n)$ selecting the appropriate X_2. Using this measure, for the HMM mixture we found that, $H_{avg}(X_2) = 477.4$, and for the Gaussian-HMM mixture, $H_{avg}(X_2) = 476.7$. Therefore, performance for the HMM is comparable. This is reasonable when viewed in light of the synthetic example. We've therefore seen that clustering with all data tends to improve the overall result as it refines the clustering in a meaningful way.

5. CONCLUSIONS

We have derived an extension of the Dirichlet process that mixes on all data in an observation by using a product base distribution, which can accommodates multiple modalities. As an example, we developed the Gaussian-HMM mixture model, where each component generated data from both a multivariate Gaussian distribution and a hidden Markov model, which comprised the complete observation. Experimental results showed the functioning of this model on both synthesized and MLB data for clustering.

6. REFERENCES

[1] T. Ferguson, "A bayesian analysis of some nonparametric problems," *Annals of Statistics*, vol. 1, pp. 209–230, 1973.

[2] Y. Xue, X. Liao, L. Carin, and B. Krishnapuram, "Multi-task learning for classification with dirichlet process priors," *The Journal of Machine Learning Research*, vol. 8, pp. 35–63, 2007.

[3] L. Ren, D. Dunson, S. Lindroth, and L. Carin, "Dynamic nonparametric bayesian models for analysis of music," *Journal of the American Statistical Association*, submitted.

[4] Y.H. Teh, M.I. Jordan, M.J. Beal, and D.M. Blei, "Hierarchical dirichlet processes," *Journal of the American Statistical Association*, vol. 101, pp. 1566–1581, 2006.

[5] C.E. Antoniak, "Mixtures of dirichlet processes with applications to bayesian nonparametric problems," *Annals of Statistics*, vol. 2, pp. 1152–1174, 1974.

[6] J. Sethuraman, "A constructive definition of dirichlet priors," *Statistica Sinica*, vol. 4, pp. 639–650, 1994.

[7] H. Ishwaran and L.F. James, "Gibbs sampling methods for stick-breaking priors," *Journal of the American Statistical Association*, vol. 96, pp. 161–173, 2001.

[8] A. Rodriquez, D.B. Dunson, and A.E. Gelfand, "The nested dirichlet process," *Journal of the American Statistical Association*, vol. to appear, 2008.

[9] L.R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77 no. 2, pp. 257–286, 1989.

[10] D. Gamerman and H.F. Lopes, *Markov Chain Monte Carlo: Stochastic Simulation for Bayesian Inference*, Chapman & Hall, 2nd edition, 2006.

[11] Y. Qi, J. Paisley, and L. Carin, "Music analysis using hidden markov mixture models," *IEEE Trans. on Signal Processing*, vol. 55, pp. 5209–5224, 2007.