Dependent Indian Buffet Processes

Sinead Williamson **Peter Orbanz** **Zoubin Ghahramani**
Computational and Biological Learning Laboratory
Department of Engineering
University of Cambridge

Abstract

Latent variable models represent hidden structure in observational data. To account for the distribution of the observational data changing over time, space or some other covariate, we need generalizations of latent variable models that explicitly capture this dependency on the covariate. A variety of such generalizations has been proposed for latent variable models based on the Dirichlet process. We address dependency on covariates in binary latent feature models, by introducing a dependent Indian buffet process. The model generates, for each value of the covariate, a binary random matrix with an unbounded number of columns. Evolution of the binary matrices over the covariate set is controlled by a hierarchical Gaussian process model. The choice of covariance functions controls the dependence structure and exchangeability properties of the model. We derive a Markov Chain Monte Carlo sampling algorithm for Bayesian inference, and provide experiments on both synthetic and real-world data. The experimental results show that explicit modeling of dependencies significantly improves accuracy of predictions.

1 Introduction

Latent variables models are widely used to identify hidden structure in data. Classic examples of such models include finite mixture models (McLachlan and Peel, 2000) and factor analyzers (Bartholomew, 1987). These parametric models suffer from the restriction that the number of features has to be specified a priori. For a model that attempts to identify unknown features, the assumption that the number of features is known beforehand is hard to justify. Bayesian nonparametric models have emerged as a powerful approach for addressing this problem. Prominent examples include the Dirichlet process (DP; Ferguson, 1973) for mixture models, and the Indian buffet process (IBP; Griffiths and Ghahramani, 2006) for binary latent feature models.

In this paper, we build upon the IBP model, which is a random distribution on binary matrices with a variable number of columns. When combined with a suitable likelihood, the matrix entries act as switching variables which activate or deactivate a feature's influence for each observation. Application examples include nonparametric Bayesian representations of binary factor analysis (Griffiths and Ghahramani, 2006), tree-structured models, (Miller *et al.*, 2008), and sparse nonparametric factor analysis (Knowles and Ghahramani, 2007).

A problem that has recently received much attention in the context of DP models is dependence of data on a covariate, such as latent feature models that evolve over time, or exhibit varying behavior according to their position in space. MacEachern (1999) proposed a model, called the Dependent Dirichlet Process (DDP), which combines Dirichlet and Gaussian process models to achieve dependence on a covariate: If the cluster parameters of the components in a Dirichlet process mixture model are of dimension d, each such parameter is substituted by a d-dimensional function drawn from a Gaussian process, with the covariate as its argument. If the covariate is time, for example, this can be regarded as a nonparametric mixture model in which the characteristics of each component evolve over time. The DDP idea has triggered a flurry of publications in nonparametric Bayesian and spatial statistics, see e.g. Duan *et al.* (2007) and Griffin and Steel (2006). Dunson and Park (2008) model dependence by introducing kernel-defined weights in the "stick-breaking construc-

Appearing in Proceedings of the 13^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2010, Chia Laguna Resort, Sardinia, Italy. Volume 6 of JMLR: W&CP 6.

tion" of the DP. Sudderth and Jordan (2009) have extended the DDP idea to hierarchical models.

In this paper, we address the problem of dependence for binary latent feature models, and derive a dependent form of the IBP. The DDP idea does not immediately carry over to the IBP, as the model involves only binary variables. Roughly speaking, we couple a set of IBPs by simulating the binary variables as transformed Gaussian variables. The Gaussian variables can be represented as marginals of a GP, and the binary variables couple through the GP's mean and covariance function. We show that Bayesian inference for the dependent IBP model can be conducted by Markov Chain Monte Carlo (MCMC) sampling. The modular character of the model permits the combination of different MCMC techniques to improve efficiency, such as truncated Gibbs sampling for the model's IBP components, and Hybrid Monte Carlo for the GP hyperparameters. The advantage of explicitly modeling dependence on a covariate is demonstrated by experiments on both synthetic data and two real-world data sets.

2 Background: Indian Buffet Process

The Indian buffet process (IBP; Griffiths and Ghahramani, 2006) is a distribution on binary matrices with N rows and an infinite number of columns. To generate such a random matrix $\mathbf{Z} = (z_{nk})$, we choose a scalar parameter $\alpha \in \mathbb{R}_+$. The distribution IBP(α) can be sampled as follows:

$$\begin{aligned} v_j &\sim \text{Beta}(\alpha, 1) \\ b_k &:= \prod_{j=1}^{k} v_j \\ z_{nk} &\sim \text{Bernoulli}(b_k) \ . \end{aligned} \tag{1}$$

This is the "stick-breaking construction" of the IBP (Teh *et al.*, 2007), where the variables b_k represent a set of "sticks" of decaying length. As the construction shows, a given matrix $\mathbf{Z}$ will have non-zero probability under the IBP if and only if the number of columns containing non-zero entries is finite. By removing all columns containing only zeros, $\mathbf{Z}$ can therefore be regarded as a random binary $N \times K$ matrix, for which the number of non-empty columns K is itself a random variable that varies from draw to draw.

The IBP can be used as a Bayesian prior in latent feature models, where the binary entries z_{nk} of an IBP random matrix encode whether feature k is used to explain observation n. The IBP is combined with a parametric likelihood $p(\mathbf{X}^{(t)}|\mathbf{Z}^{(t)}, \Theta)$. A common example of such a likelihood is a linear Gaussian model (see e.g. Griffiths and Ghahramani, 2006). Given a binary $N \times K$ matrix $\mathbf{Z}^{(t)}$, a data $\mathbf{X}^{(t)}$ is assumed to be generated drawing each row $\mathbf{X}_n^{(t)}$ as

$$\mathbf{X}_n^{(t)} \sim \mathcal{N}(\mathbf{Z}_n^{(t)}\mathbf{A}, \epsilon_X^2\mathbf{I}) \ , \tag{2}$$

where the $K \times D$ feature matrix $\mathbf{A}$ is generated column-wise as $\mathbf{A}_d \sim \mathcal{N}(\mathbf{0}, \epsilon_A^2\mathbf{I})$. Such a likelihood represents a binary factor analysis model. It is parametrized by $\Theta := (\epsilon_X, \epsilon_A)$. A number of other likelihood models have been used with the IBP, see for example Navarro and Griffiths (2008) and Miller *et al.* (2009).

Inference in IBP models can be conducted by Gibbs sampling. The MCMC algorithm samples the latent variables z_{nk} and b_k, the parameters Θ of the likelihood, and the IBP hyperparameter α. To cope with the variable number of active features, and hence the variable dimension of the sampling space, samplers may truncate the number of features at some reasonably large value, in a manner similar to truncated sampling methods for the Dirichlet process. As an alternative to Markov chain sampling for the IBP, Doshi-Velez *et al.* (2009) propose a variational algorithm for approximate inference, which permits application of IBP methods to large data sets.

3 Dependent Indian Buffet Processes

The dependent Indian buffet process (dIBP) derived in this section substitutes the individual binary matrix $\mathbf{Z}$ generated by an IBP with a set of matrices $\mathbf{Z}(t)$ for $t \in T$. Each matrix $\mathbf{Z}(t)$ contains a column $\mathbf{Z}_k(t)$ corresponding to feature k at covariate t. For each feature, these columns for different values of t are correlated. The covariate t may be time, space, or another suitable index set.[1]For example, if t represents a time index, $\mathbf{Z}(t)$ is a sequence of matrices, and each feature k evolves over time. The number $N(t)$ of items (rows) in the matrix $\mathbf{Z}(t)$ may vary for different values of t.

Coupling over T is modeled by representing the Bernoulli variables $z_{nk}(t)$ as transformed Gaussian variables, and assembling these into a Gaussian process indexed by t. More precisely, an arbitrary Bernoulli variable $z \sim \text{Bernoulli}(\beta)$ can be represented as

$$\begin{aligned} u &\sim \mathcal{N}(\mu, \sigma^2) \\ z &:= \mathbb{I}\{u < F^{-1}(\beta \,|\, \mu, \sigma^2)\} \ , \end{aligned} \tag{3}$$

[1]The special case of dependence of an IBP on a discrete time variable has been recently considered by Van Gael *et al.* (2009) and used to define a generalization of hidden Markov models. In contrast to their model, the dIBP introduced in this section allows dependence on arbitrary, possibly vector-valued, covariates, does not have Markov and time-invariance restrictions, and has IBP marginals at any covariate value.

with normal cumulative distribution function $F(\,.\,|\,\mu,\sigma)$ and indicator function $\mathbb{I}$. (This representation follows Sudderth and Jordan (2009), and is due to Duan *et al.* (2007) and, more generally, MacEachern (2000).)

To generate a coupled set of variables $z(t)$ for $t \in T$, the individual Gaussian distributions can be regarded as the t-marginals of a Gaussian process $\mathrm{GP}(m, \Sigma)$ indexed by T. Draws from the GP are functions $g : T \to \mathbb{R}$. For fixed t, the Bernoulli variable $z(t)$ is generated according to (3) using the marginal of the process at t, which is just $\mathcal{N}(m(t), \Sigma(t,t))$. To generate a set of dependent IBPs, a collection $\{h_{nk}\}$ of random functions is drawn from Gaussian processes, and is then used to generate the random variables $z_{nk}(t)$.

We define the dependent IBP model as follows: First generate stick length parameters b_k, which are not dependent on t, as $v_j \sim \mathrm{Beta}(\alpha, 1)$ and $b_k := \prod_{j=1}^{k} v_j$ according to (1). Then generate the variables $z_{nk}(t)$ for all t as

$$\begin{aligned} g_k &\sim \mathrm{GP}(0, \Sigma_k) \\ h_{nk} &\sim \mathrm{GP}(g_k, \Gamma_{nk}) \\ z_{nk}(t) &:= \mathbb{I}\{h_{nk}(t) < F^{-1}(b_k|0, \mathbf{\Sigma}_k^{(t,t)} + \mathbf{\Gamma}_{nk}^{(t,t)})\}\ . \end{aligned} \tag{4}$$

The model (4) defines a *hierarchical Gaussian process*, similar to the hierarchical Dirichlet process (Teh *et al.*, 2006). By means of the hierarchy, information can be shared within columns (features). Different choices of the GP covariance structure define different model properties:

General model: Hierarchical sharing. The hierarchy in (4) shares information within features. The covariances Σ_k define a general profile for each feature k, with the second layer modeling individual variations per item. If, for example, Σ_k is chosen as a large-scale covariance function, and Γ_{nk} to model small-scale fluctuations, h_{nk} will vary significantly for different values of k, but on a smaller scale between different items n within a fixed feature k. The GP draws coupled features over T. Consequently, for two index values t_1 and t_2, item n at t_1 has to correspond to item n at t_2.

Exchangeable model: Bag of items. If Γ_{nk} is chosen independently of n, e.g. $\Gamma_{nk} = \rho^2 I$, items become exchangeable, i.e. the item index set $\{1, \ldots, N\}$ may be permuted by any fixed permutation applied over all $t \in T$, without changing the overall probability of the observed process. We obtain a model with a "bag of items" property at each value of t. For each t, the marginal distribution of $\mathbf{Z}(t)$ is an IBP, and IBP matrices have exchangeable rows. For covariance $\rho^2 I$, the second layer GP can be interpreted as adding noise variables, which do not couple over T. The items (rows) within a matrix $\mathbf{Z}(t_1)$ can therefore be permuted *independently* of $\mathbf{Z}(t_2)$, without changing the probability of the overall sample. In other words, item n at t_1 does not generally correspond to item n at t_2 for this choice of Γ_{nk}. Such a parametrization would be suitable, for example, to model data sets representing the development of scientific journal articles over a given period of time, where a journal issue is a bag of articles, and there is no obvious correspondence between articles at time t_1 and articles at time t_2. At different times t_1 and t_2, the respective numbers $N(t_1)$ and $N(t_2)$ need not be identical.

Collapsed model: Tracking individual items. If the objective is to model the evolution of features over T specific to each item, the covariance Σ_k in the first layer (and hence the coupling structure shared over items) can be set to zero. In the zero covariance limit, the hierarchy collapses into

$$\begin{aligned} h_{nk} &\sim \mathrm{GP}(0, \Gamma_{nk}) \\ z_{nk}(t) &:= \mathbb{I}\{h_{nk}(t) < F^{-1}(b_k|0, \mathbf{\Gamma}_{nk}^{(t,t)})\}\ . \end{aligned} \tag{5}$$

Without the hierarchy, the model does not impose a shared covariance structure of the latent functions h_{nk}. For a fixed feature k, the variables $z_{nk}(t_1)$ and $z_{n'k}(t_2)$ are coupled by their shared stick length b_k, which controls the probability of feature k being selected, and does not depend on n or t. Additionally, for a fixed item n and feature k, $z_{nk}(t_1)$ and $z_{nk}(t_2)$ are correlated through the latent function h_{nk}. This version of the model would, for example, be applicable to the tracking of patients over time in a longitudinal study.

4 MCMC Inference

The dIBP can be used as a prior in conjunction with a parametric data likelihood $p(\mathbf{X}^{(t)}|\mathbf{Z}^{(t)}, \Theta)$. The objective of an inference algorithm is to estimate the latent variables b_k, g_k and h_{nk} (which define the matrices $\mathbf{Z}(t)$), as well as the model hyperparameters, from the data $\mathbf{X}$. For the sake of simplicity, we derive the MCMC steps for the case $\Gamma_{nk} = \rho^2 I$.

The random draws generated in the dIBP model (4) are the stick lengths b_k and the GP draws g_k and h_{nk}. Evaluation of the functions on a finite set of data yields a finite set of real-valued variables $g_k(t)$ and $h_{nk}(t)$. If the hyperparameters s_k of the GP models are to be estimated from data, these have to be sampled as well. An MCMC sampler then performs a random walk in the space spanned by the variables b_k, s_k, $g_k(t)$ and $h_{nk}(t)$. The number of active features is a random variable of unbounded value. The unbounded number of features is accomodated by an infinite-dimensional space. As for many nonparametric Bayesian models, infinite dimensionality can be addressed either by approximate methods, e.g. by truncating the number of

represented features at a suitably large value K, or by asymptotically exact inference algorithms, such as slice sampling (Neal, 2003).

To achieve computational efficiency and make the dIBP model applicable to real-world data, we use an approximate inference strategy based on Gibbs sampling. Derivation of a Gibbs sampler means derivation of the full conditionals of the target variables, as given below. The Gibbs sampler on a truncated set of variables is combined with a hybrid Monte Carlo approach (Duane *et al.*, 1987) for updates of the GP hyperparameters.

We assume that the number of features is truncated at K. For a finite data set consisting of measurements at index values $\{t_1, \ldots, t_M\} =: \mathbf{T}$, all continuous objects on T have to be evaluated on $\mathbf{T}$. We will generally denote the resulting vectors by bold letters, such that g_k is represented as the vector $\mathbf{g}_k = (g_k(t_1), \ldots, g_k(t_n))$, Σ_k as the matrix $\mathbf{\Sigma}_k = \big(\Sigma_k(t,t')\big)_{t,t'\in\mathbf{T}}$ etc. For removal of an individual element from a vector $\mathbf{x}$, we use the notation $\mathbf{x}_{-i} := (x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_I)$.

Stick lengths. The full conditional of a stick length b_k is the distribution $p(b_k|\mathbf{b}_{-k}, \mathbf{Z}(t), \mathbf{g}_k)$, and can be obtained as the posterior of the likelihood $p(z_{nk}^{(t)}|b_k, \mathbf{g}_k)$ for all n, and prior $p(b_k|\alpha)$. As shown in Appendix A, the full conditional is

$$p(b_k|\mathbf{b}_{-k}, \mathbf{h}_{nk}, \mathbf{g}_k) \propto \frac{b_K^\alpha}{b_k} \prod_{t\in\mathbf{T}} \prod_{n=1}^{N(t)} (\gamma_k^{(t)})^{z_{nk}(t)} (1-\gamma_k^{(t)})^{1-z_{nk}(t)} \ , \quad (6)$$

where $\gamma_k^{(t)} := F\big(F^{-1}(b_k|0, \mathbf{\Sigma}_k^{(t,t)} + \rho^2) - \mathbf{g}_k^{(t)}|0, \rho^2\big)$. Dependence on $\mathbf{h}_{nk}$ is encoded by the binary variables $z_{nk}(t)$, which are deterministic functions of $\mathbf{h}_{nk}$. We use a Metropolis-Hastings algorithm in the space of $F^{-1}(b_k)$ to sample from (6).

Gaussian process draws. The Gaussian process random functions g_k and h_{nk} are evaluated on $\mathbf{T}$ as the random vectors $\mathbf{g}_k \sim \mathcal{N}(0, \mathbf{\Sigma}_k)$ and $\mathbf{h}_{nk} \sim \mathcal{N}(\mathbf{g}_k, \rho^2\mathbf{I})$. Given all $\mathbf{h}_{nk}$, the full conditional of $\mathbf{g}_k$ is

$$p(\mathbf{g}_{nk}|\{\mathbf{h}_{nk}\}, \mathbf{b}) \propto p_{\text{Normal}}(\mathbf{g}_k|0, \mathbf{\Sigma}_k) \cdot \prod_{t\in\mathbf{T}} \prod_{n=1}^{N(t)} p_{\text{Normal}}(\mathbf{h}_{nk}^{(t)}|\mathbf{g}_k^{(t)}, \rho^2) \ . \quad (7)$$

Sampling $\mathbf{h}_{nk}$ is slightly more complicated: These latent variables provide the actual link to the data, as the data likelihood $p(\mathbf{X}^{(t)}|\mathbf{Z}^{(t)}, \Theta)$ is parametrized by the values $z_{nk}^{(t)}$, which are in turn functions of $\mathbf{h}_{nk}$. However, since the Gaussian values enter in the likelihood only indirectly, through the binary values $z_{nk}^{(t)}$, sampling has to proceed in two steps, by first obtaining estimates of $z_{nk}^{(t)}$, and then sampling $\mathbf{h}_{nk}$ conditional on those.

For the stick lengths estimates, we already derived the distributions $p(z_{nk}^{(t)}|b_k, \mathbf{g}_k)$ as Bernoulli$(\gamma_k^{(t)})$. If these are used as priors for $z_{nk}^{(t)}$, the full conditionals $p(z_{nk}^{(t)}|b_k, \mathbf{g}_k^{(t)}, \mathbf{X}^{(t)}, \Theta)$ are given by the corresponding posteriors under the data likelihood. Given samples of $z_{nk}^{(t)}$, we now obtain samples of $\mathbf{h}_{nk}$. In (4), $z_{nk}^{(t)}$ is defined by thresholding $\mathbf{h}_{nk}^{(t)}$ on the threshold value $\tilde{b}_k^{(t)} := F^{-1}(b_k|0, \mathbf{\Sigma}_k^{(t,t)} + \rho^2)$. Conditioning on $z_{nk}^{(t)}$ restricts the possible values of $\mathbf{h}_{nk}^{(t)}$ to either above or below the threshold, which turns the Gaussian distribution into a truncated Gaussian on the corresponding half-line. Hence, the full conditional $p(\mathbf{h}_{nk}^{(t)}|\mathbf{g}_k^{(t)}, b_k, z_{nk}^{(t)})$ is obtained by restricting the Gaussian density $p_{\text{Normal}}(\mathbf{h}_{nk}^{(t)}|\mathbf{g}_k^{(t)}, \rho^2)$, either to $(-\infty, \tilde{b}_k^{(t)}]$ for $z_{nk}^{(t)} = 1$, or to $(\tilde{b}_k^{(t)}, +\infty)$ for $z_{nk}^{(t)} = 0$. An efficient approach for sampling truncated Gaussians is described by Damien and Walker (2001).

Hyperparameters. The model hyperparameters consist of the beta distribution parameter α, and the parameters of the GP covariance functions. The beta parameter α is sampled using Metropolis-Hastings steps. All experiments reported in Sec. 5 use exponential kernels,

$$\Sigma_k(t,t') := \sigma^2 \exp\Big(-\frac{(t-t')^2}{s_k^2}\Big) \ . \quad (8)$$

Due to the smoothness and special geometry of the problem, GP hyperparameters can be sampled much more efficiently than with general Metropolis-Hastings algorithms, and we sample s_k^2 by means of a Hybrid Monte Carlo method (Duane *et al.*, 1987; Neal, 1997).

5 Results

We experimentally evaluate the dIBP model on both synthetic and real-world data. The likelihood $p(\mathbf{X}^{(t)}|\mathbf{Z}^{(t)}, \Theta)$ in all experiments reported here is the linear Gaussian model described in Sec. 2. Both the likelihood and the MCMC inference method are chosen to make experiments comparable to other methods, and to focus on the properties of the dIBP model. Other likelihoods developed for the IBP, such as the ICA model of Knowles and Ghahramani (2007) and the relational model of Miller *et al.* (2009), could be used with the dIBP as well. Similarly, efficient approximate inference methods developed for the GP could be adapted to the dIBP.

Since the likelihood parameters $\Theta := (\epsilon_X, \epsilon_A)$ are scatter parameters, vague gamma priors are used in the sampler. As covariance functions Σ_k and Γ_{nk} in model

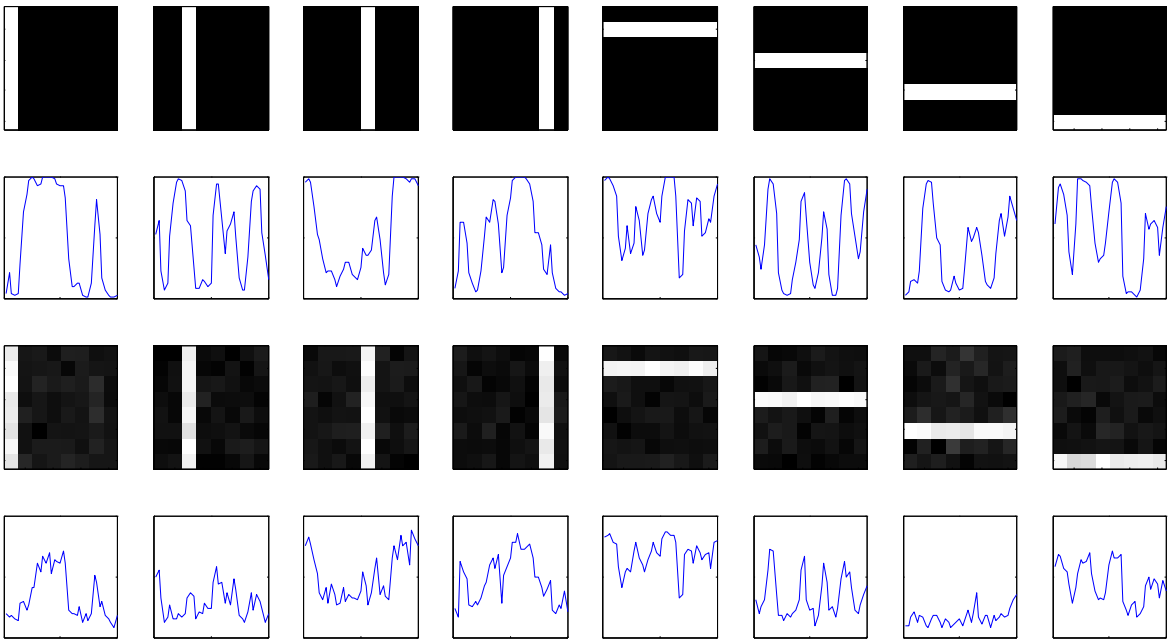

Figure 1: Hierarchical model, synthetic data: The true generating features (top) and their time-varying probabilities (second row), compared to dIBP estimates of the features (third row) and posterior sample of the probabilities (fourth row).

(4), we use squared exponential covariance kernels of the form (8) for Σ_k, with individual length scale parameters for each feature, and $\Gamma_{nk} := \rho^2\mathbf{I}$ independently of n and k.

5.1 Synthetic Data

Hierarchical model: Bag of items. To illustrate the behavior of the hierarchical model (4), we use the eight 8×8 images in Fig. 1 (top row) as features. Data is generated by collecting the eight images in a $K \times D = 8 \times 64$ feature matrix $\mathbf{A}$. Synthetic binary matrices $\mathbf{Z}^{(t)}$ are obtained by first drawing K samples with $T = 40$ time steps from a GP with Matern kernel ($\nu = \frac{5}{2}$) and passing them through a sigmoid function. At each time value t, the resulting functions represent the probability for $z_{nk}^{(t)} = 1$, and are shown for each feature over the 40 time steps in the second row in Fig. 1. From these functions, the matrices $\mathbf{Z}^{(t)}$ for $t = 1, \ldots, 40$ are generated as in (4), and the synthetic data is generated by superimposing the images as $\mathbf{X}^{(t)} = \mathbf{Z}^{(t)}\mathbf{A} + \text{noise}$, where the noise term is Gaussian with variance 0.25.

For inference, we run the sampler derived in Sec. 4 with a truncation level of 20. The third row in Fig. 1 shows the 8 most commonly observed features learned by our model, arranged to correspond to the true features in the first row. The fourth row shows a sample from the posterior of the time-varying probability associated with each of these features. We note that the posterior samples capture the shape, but not the scale of the true probabilities shown above. The different scale of the estimates represents the remaining uncertainty of the model after observing only a limited amount of data. Values close to zero or one in the true probabilities (in the second row) express certainty. The posterior under limited sample size concentrates its mass on estimate probabilities that avoid certainty.

Collapsed model: Evolving item. The collapsed version (5) of the model is used to track the evolution of a single item (N=1) over 100 time steps. We generate a Markov chain of binary matrices $\mathbf{Z}^{(t)}$, by drawing transition probabilities q_k for each feature at random as $q_k \sim \text{Beta}(1, 10)$, and generating the initial matrix $\mathbf{Z}^{(t_1)}$ according to a standard IBP. From these and the features used in the previous experiment, data $\mathbf{X}^{(t)}$ is generated by the same steps as above, resulting in 100 time steps of 64-dimensional observations. Fig. 2 shows the recovered features, that is, the posterior mean (over all time steps) of $\mathbf{A}$, given the data $\mathbf{X}$ and conditional on the model's estimate of the $\mathbf{Z}^{(t)}$. As shown in Fig. 3, the model's recovery of the matrices $\mathbf{Z}^{(t)}$ is almost perfect.

5.2 Real-world data

To evaluate whether the incorporation of covariate data leads to an improvement in model quality, we

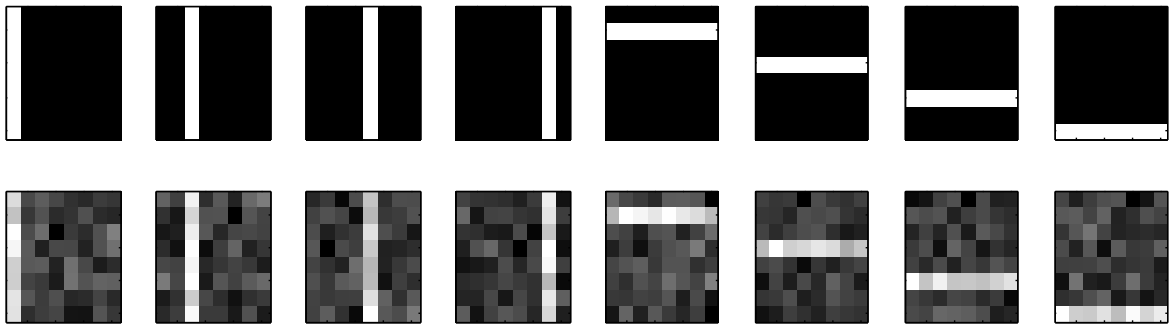

Figure 2: Collapsed model, synthetic data: True features (top row) and recovered features (bottom row).

compared the dIBP model to the basic IBP model on two real-world datasets where each data-point is associated with a position in some covariate space.

Piano music data. This data set, which is due to Poliner and Ellis (2007), consists of features extracted from audio recordings of piano music. It consists of 161 continuous-valued spectral features sampled at 10ms intervals. We scaled and centered a subset of the dataset consisting of 120 consecutive time-points. The data was randomly divided into ten folds, each consisting of twelve time-points, selecting nine folds as the training set and one as the test set. For each datapoint in the test set, ten features (selected at random) were retained, and the prediction task was to predict the values of the remaining features.

We modeled the data using both an IBP and a dIBP model with linear Gaussian likelihood. The covariate is time. Inference in the IBP model follows the truncated inference scheme of Teh *et al.* (2007), and both the IBP and the dIBP sampler use a truncation level of 50. We used 100 samples from the resulting posterior distribution to predict the held-out features of the test dataset, and calculated the RMSE of these predictions against the true values. Table 1 shows that, by incorporating temporal information, the dIBP achieves significantly lower root mean-squared error (RMSE) on the held-out test data.

	IBP	dIBP
Music data	1.08 ± 0.06	0.69 ± 0.04
UN data	0.93 ± 0.06	0.79 ± 0.07

Table 1: Root mean-squared error (RMSE) obtained with the dIBP and IBP on real-world data. The covariates are time (music data) and gross domestic product (UN data).

UN development data. The second data set consists of 13 development indicators, such as public health spending and illiteracy rates, recorded for 144 countries. This dataset was obtained from the UN Human Development Statistics database. The covariate in this case is each country's GDP, and the linear Gaussian likelihood is applied to the logarithms of the indicator variables. The data was randomly split into a training set of 130 countries, and a test set of 14 countries. For each test country, one randomly selected indicator was observed, and the remainder held out for prediction. Inference was conducted in the same manner as for the piano music data described above. The dIBP achieves lower RMSE on each of the ten folds. The average RMSE with standard deviation, obtained by 10-fold cross validation, is compared for both models in Table 1.

6 Conclusion

We have presented a framework for dependent modeling in Indian buffet processes, drawing a parallel with the considerable body of work on dependent modeling in Dirichlet processes (MacEachern, 1999; Griffin and Steel, 2006; Duan *et al.*, 2007; Sudderth and Jordan, 2009). By using Gaussian processes to model the dependency, we draw on the flexibility of GPs, and more generally kernel methods, in creating the dependence structure. Moreover, we can leverage the well-developed toolbox for learning the parameters of GP covariance functions from data.

Our model uses GPs with the stick-breaking construction of the IBP (Teh *et al.*, 2007) to create a countably infinite collection of dependent sticks, each of which is marginally drawn from a beta distribution. When coupled with a base measure B_0, this infinite collection of beta random variables $\{b_k\}$ can be used to define a beta process (Hjort, 1990), which is a stationary independent increment (i.e. Lévy) process with beta distributed increments. The base measure defines the location of independently drawn points with masses b_k. An alternative method for generating dependent IBPs would be by means of *dependent beta processes*. We leave this for future work.

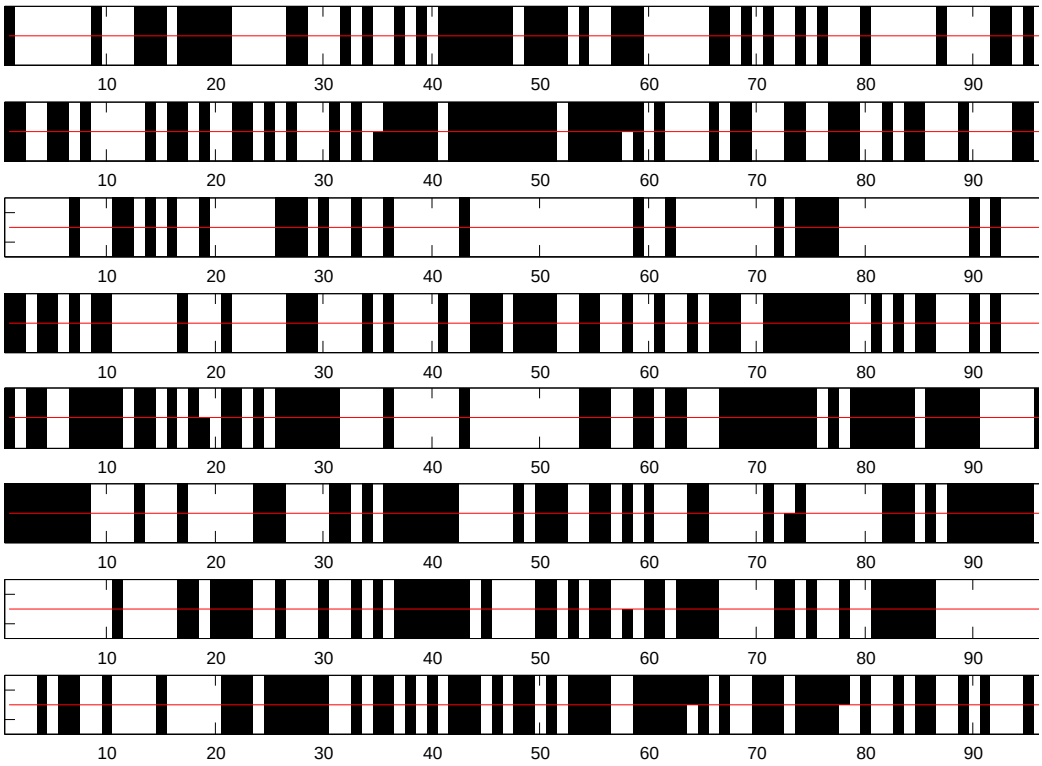

Figure 3: Collapsed model, synthetic data: Recovery of the matrices $\mathbf{Z}^{(t)}$ over time. Rows correspond to features, the horizontal axis is time. Each row is split in a top half (column k of true $\mathbf{Z}^{(t)}$) and a bottom half (column k of estimated $\mathbf{Z}^{(t)}$). Each estimate is a single sample of $\mathbf{Z}^{(t)}$ from the MCMC run.

A Stick Lengths Conditionals

To derive the probability $p(z_{nk}^{(t)}|b_k, \mathbf{g}_k)$, we note that for an arbitrary Gaussian variable y with CDF F_y, the variable $z := \mathbb{I}\{y < F_y^{-1}(b)\}$ is Bernoulli(b)-distributed. According to the hierarchical generation of the functions h_{nk} in (4), the variable $\mathbf{h}_{nk}^{(t)}$ can be represented as a sum $\mathbf{h}_{nk}^{(t)} = \mathbf{g}_k^{(t)} + y$ with an auxiliary Gaussian variable $y \sim \mathcal{N}(0, \rho^2)$. Then

$$\begin{aligned} z_{nk}^{(t)} &= \mathbb{I}\{\mathbf{h}_{nk}^{(t)} < F^{-1}(b_k|0, \mathbf{\Sigma}_k^{(t,t)} + \mathbf{\Gamma}_{nk}^{(t,t)})\} \\ &= \mathbb{I}\{\mathbf{g}_k^{(t)} + y < F^{-1}(b_k|0, \mathbf{\Sigma}_k^{(t,t)} + \mathbf{\Gamma}_{nk}^{(t,t)})\} \\ &= \mathbb{I}\{y < F^{-1}(b_k|0, \mathbf{\Sigma}_k^{(t,t)} + \mathbf{\Gamma}_{nk}^{(t,t)}) - \mathbf{g}_k^{(t)}\} \, . \end{aligned}$$

Since y is again Gaussian with CDF $F(y|0, \rho^2)$, and since $\Gamma_{nk}^{(t,t)} = \rho^2$, this means that z_{nk} is Bernoulli with parameter

$$\gamma_k^{(t)} := F\big(F^{-1}(b_k|0, \mathbf{\Sigma}_k^{(t,t)} + \rho^2) - \mathbf{g}_k^{(t)}\big|0, \rho^2\big) \, . \quad (9)$$

Since the stick lengths b_k are generated sequentially according to (1), the probability of the complete set $\mathbf{b} = (b_1, \ldots, b_K)$ is

$$p(\mathbf{b}|\alpha) = p(b_1|\alpha) \prod_{k=2}^{K} p(b_k|b_{k-1}, \alpha) = \alpha^K b_K^{\alpha} \prod_{k=1}^{K} b_k^{-1} \, . \quad (10)$$

Combined with the likelihood $p(z_{nk}^{(t)}|b_k, \mathbf{g}_k)$, the corresponding posterior, and hence the full conditional of b_k, is

$$\begin{aligned} &p(b_k|\mathbf{b}_{-k}, \mathbf{h}_{nk}, \mathbf{g}_k) \\ &\qquad \propto \frac{b_K^{\alpha}}{b_k} \prod_{t \in \mathbf{T}} \prod_{n=1}^{N(t)} (\gamma_k^{(t)})^{z_{nk}^{(t)}} (1 - \gamma_k^{(t)})^{1 - z_{nk}^{(t)}} \, . \end{aligned} \quad (11)$$

References

Bartholomew, D. J. (1987). The foundations of factor analysis. *Biometrika*, **71**(2), 221–232.

Damien, P. and Walker, S. G. (2001). Sampling truncated normal, beta, and gamma densities. *Journal of Computational and Graphical Statistics*, **10**(2), 206–215.

Doshi-Velez, F., Miller, K. T., van Gael, J., and Teh, Y. W. (2009). Variational inference for the Indian buffet process. In *Proceedings of the 12th Inter-*

national Conference on Artificial Intelligence and Statistics (AISTATS).

Duan, J. A., Guindani, M., and Gelfand, A. E. (2007). Generalized spatial Dirichlet process models. *Biometrika*, **94**(4), 809–825.

Duane, S., Kennedy, A. D., Pendleton, B. J., and Roweth, D. (1987). Hybrid Monte Carlo. *Physical Review Letters B*, **195**(2), 216–222.

Dunson, D. B. and Park, J.-H. (2008). Kernel stick-breaking processes. *Biometrika*, **95**(2), 307–323.

Ferguson, T. S. (1973). A Bayesian analysis of some nonparametric problems. *Annals of Statistics*, **1**(2).

Griffin, J. E. and Steel, M. F. J. (2006). Order-based dependent Dirichlet processes. *Journal of the American Statistical Association*, **101**, 179–194.

Griffiths, T. L. and Ghahramani, Z. (2006). Infinite latent features models and the Indian buffet process. In *Advances in Neural Information Processing Systems (NIPS) 2005*.

Hjort, N. L. (1990). Nonparametric Bayes estimators based on beta processes in models for life history data. *Annals of Statistics*, **18**(3), 1259–1294.

Knowles, D. and Ghahramani, Z. (2007). Infinite sparse factor analysis and infinite independent component analysis. In *7th International Conference on Independent Component Analysis and Signal Separation (ICA)*.

MacEachern, S. N. (1999). Dependent nonparametric processes. In *Proc. Bayesian Statist. Sci. Sect.*

MacEachern, S. N. (2000). Dependent Dirichlet processes. Technical report, Ohio State University.

McLachlan, G. and Peel, D. (2000). *Finite Mixture Models*. John Wiley & Sons.

Miller, K. T., Griffiths, T. L., and Jordan, M. I. (2008). The phylogenetic Indian buffet process: A non-exchangeable nonparametric prior for latent features. In *Uncertainty in Artificial Intelligence (UAI)*.

Miller, K. T., Griffiths, T. L., and Jordan, M. I. (2009). Nonparametric latent feature models for link prediction. In *Advances in Neural Information Processing Systems (NIPS) 2009*.

Navarro, D. J. and Griffiths, T. L. (2008). Latent features in similarity judgment: A nonparametric Bayesian approach. *Neural Computation*, **20**, 2597–2628.

Neal, R. M. (1997). Monte Carlo implementation of Gaussian process models for Bayesian regression and classification. Technical report, University of Toronto.

Neal, R. M. (2003). Slice sampling. *Annals of Statistics*, **31**(3), 705–767.

Poliner, G. E. and Ellis, D. P. W. (2007). A discriminative model for polyphonic piano transcription. *EURASIP J. Appl. Signal Process.*, **2007**(1), 154–162.

Sudderth, E. B. and Jordan, M. I. (2009). Shared segmentation of natural scenes using dependent Pitman-Yor processes. In *Advances in Neural Information Processing Systems (NIPS) 2008*.

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

Teh, Y. W., Görür, D., and Ghahramani, Z. (2007). Stick-breaking construction for the Indian buffet process. In *Proceedings of the 11th Conference on Artificial Intelligence and Statistics*.

Van Gael, J., Teh, Y. W., and Ghahramani, Z. (2009). The infinite factorial hidden Markov model. In *Advances in Neural Information Processing Systems (NIPS) 2008*.