Bayesian Statistics: Indian Buffet Process

Ilker Yildirim
Department of Brain and Cognitive Sciences
University of Rochester
Rochester, NY 14627

August 2012

Reference: Most of the material in this note is taken from: Griffiths, T. L. & Ghahramani, Z. (2005). Infinite latent feature models and the Indian buffet process. Gatsby Unit Technical Report GCNU-TR-2005-001.

1. Introduction

A common goal of unsupervised learning is to discover the latent variables responsible for generating the observed properties of a set of objects. For example, factor analysis attempts to find a set of latent variables (or factors) that explain the correlations among the observed variables. A problem with factor analysis, however, is that the user has to specify the number of latent variables when using this technique. Should one use 5, 10, or 15 latent variables (or some other number)? There are at least two ways one can address this question. One way is by performing model selection. Another way is to use a Bayesian nonparametric method. In general, Bayesian nonparametric methods grow the number of parameters as the size and complexity of the data set grow. An important example of a Bayesian nonparametric method is the Indian Buffet Process. (Another important example is the Dirichlet Process Mixture Model; see the Computational Cognition Cheat Sheet on Dirichlet Processes.)

An Indian buffet process (IBP) is a stochastic process that provides a probability distribution over equivalence classes of binary matrices of bounded rows and potentially infinite columns. Adoption of the IBP as a prior distribution in Bayesian modeling has led to successful Bayesian nonparametric models of human cognition including models of latent feature learning (Austerweil & Griffiths, 2009), causal induction (Wood, Griffiths, & Ghahramani, 2006), similarity judgements (Navarro & Griffiths, 2008), and the acquisition of multisensory representations (Yildirim & Jacobs, 2012).

Two aspects of the Indian buffet process — that it shares with other Bayesian nonparametric methods — underlie its successful application in modeling human cognition. First, it defines a probability distribution over very rich, combinatoric structures. Indeed, depending on the goal of the modeler, one can interpret binary matrices as feature ownership tables, adjacency matrices representing graphs, or other complex, structured representations.

Second, under IBP, binary matrices can grow or shrink with more data, effectively letting models adapt to the complexity of the observations. For example, a binary matrix is often thought of as a feature ownership matrix in which there is a row for each object, and a column for each possible feature. An entry of 1 at element (i, k) of the matrix means that object i posseses feature k. Otherwise this entry is 0. Importantly, the user does not need to specify the number of columns (that is, the number of features). Instead, the number of

columns can grow with the size and complexity of the data set. That is, *a priori*, IBP — due to its stochastic process foundations — has support over feature ownership matrices with any number of columns. Upon observing data, *a posteriori*, IBP concentrates its mass over a subset of binary matrices with finitely many columns via probabilistic inference (assuming the use of a finite data set). Because IBP lends itself to Bayesian inference, the posterior IBP will maintain the variability regarding the exact number of features arising from the observations, thereby solving (or at least alleviating) the difficult problems of model selection and model comparison.

As stated above, IBP does not make assumptions about the exact number of latent features underlying a finite set of observed objects, although it does make other assumptions about these units (Griffiths & Ghahramani, 2005, 2006). It assumes that the latent features are binary. Thus, an object either does or does not possess a feature. It also assumes that latent features are statistically independent, meaning that knowledge that an object possesses one feature does not provide information about whether it possesses other features. Lastly, it assumes that the latent features are a finite subset of an unbounded or infinite set of features.

More formally, the probability of a binary matrix Z under IBP can be written as the following:

$$p(Z) = \frac{\alpha^K}{\prod_{h=1}^{2^N-1} K_h!} \exp\{-\alpha H_N\} \prod_{k=1}^{K} \frac{(N-m_k)!(m_k-1)!}{N!} \tag{1}$$

where N is the number of objects, K is the number of multisensory features, K_h is the number of features with history h (the history of a feature is the matrix column for that feature interpreted as a binary number), H_N is the N^{th} harmonic number, m_k is the number of objects with feature k, and α is a variable influencing the number of features (a derivation of this equation can be found in Griffiths & Ghahramani, 2005). We denote $Z \sim \text{IBP}(\alpha)$, meaning that Z is distributed according to the IBP with parameter α. Below, we provide intuition for α, the only parameter of the process.

2. Constructing the Indian Buffet Process

There are three well studied constructions for the Indian buffet process – constructions based on an implicit representation, a finite representation, and an explicit "stick-breaking" representation. Here we mainly study the implicit representation before we briefly introduce the other two constructions.

An implicit representation: In the field of Bayesian nonparametric models (and the study of stochastic processes, in general), implicit representations are a way of constructing models which exploit *de Finetti's Theorem*. If a sequence of observations, $X_1, \ldots, X_n$, are exchangeable (i.e., its probability distribution is invariant to permutations of the variables in the sequence), then – by de Finetti's theorem – there is a stochastic element (also known as the mixing distribution) that underlies these observations. For such cases, one can choose to work only with the observations, X_i's, (e.g., in defining conditional distributions) and thereby address the underlying stochastic process implicitly. A well know example of such a representation is the Chinese Restaurant Process which implies the Dirichlet Process as

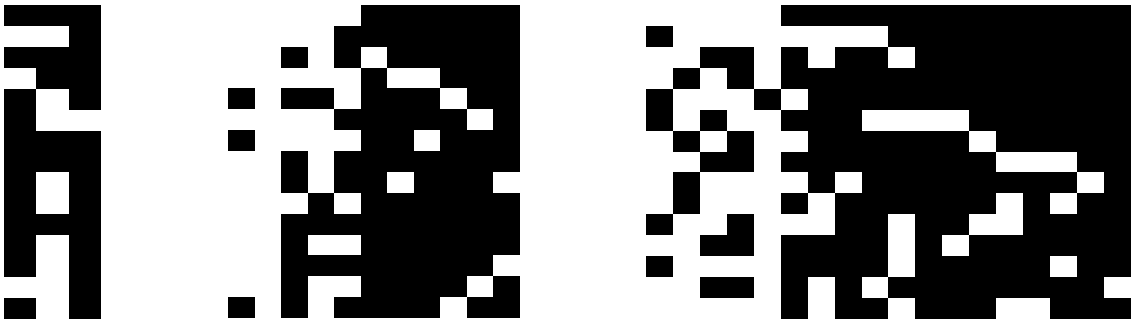

Figure 1: Simulations of the Indian buffet process with different α values. The binary matrices depict samples from the IBP with $\alpha = 1$ resulting in a total of 3 dishes sampled (Left), with $\alpha = 3$ resulting in 12 dishes sampled (Middle), and with $\alpha = 5$ resulting in 18 dishes sampled (Right). In all simulations, the number of customers (i.e., the number of rows) was fixed to 15. White pixels are 1 and black pixels are 0.

its mixing distribution (Aldous, 1985; also see the Computational Cognition Cheat Sheet on Dirichlet Processes.).

Griffiths and Ghahramani (2005, 2006) introduced the following culinary metaphor that results in an IBP while keeping the mixing distribution implicit. In this metaphor, customers (observations) sequentially arrive at an Indian buffet (hence the name) with an apparently infinite number of dishes (features). The first customer arrives and samples the first Poisson(α) dishes, with $\alpha > 0$ being the only parameter of the process. The i^{th} customer samples one of the already sampled dished with probability proportional to the popularity of that dish prior to her arrival (that is, proportional to $\frac{n_{-i,k}}{i}$ where $n_{-i,k}$ is the number of previous customers who sampled dish k). When she is done sampling dishes previously sampled, customer i further samples Poisson($\frac{\alpha}{i}$) number of new dishes. This process continues until all N customers visit the buffet.

We can represent the outcome of this process in a binary matrix Z where the rows of the matrix are customers (observations) and the columns are dishes (features) ($z_{i,k}$ is 1 if observation i possesses feature k). The probability distribution over binary matrices induced by this process is indeed as expressed in Equation 1. To get a better intuition for the parameter α, we simulated samples from IBP with three different α values. As illustrated in Figure 1, the smaller the α, the smaller the number of features with $\sum_i z_{i,k} > 0$. In other words, for equal number of observations, the parameter α influences how likely it is that multiple observations will share the same features. Based on this property of the process, α is called the concentration parameter (similar to the α parameter in Dirichlet Processes).

Thibaux and Jordan (2007) showed that Beta processes are the underlying mixing distribution for IBP. (Similar to the relationship between the Chinese Restaurant Process and the Dirichlet Processes.) We do not provide details of this derivation here. However, we note that the availability of the underlying mixing distribution is advantageous in at least two ways: (1) one can devise new constructions (e.g., stick-breaking constructions) of the

same stochastic process which can lead to new and more efficient inference techniques; and (2) it can lead to derivations of more sophisticated models from the same family, such as hierarchical extensions of IBP.

Finally, another way of constructing Bayesian nonparametric methods is by first defining a finite latent feature model, and then take the limit of this model as the number of latent features goes to infinity (see Griffiths & Ghahramani, 2005, 2006).

Conditional distributions for Inference: We need to derive the prior conditional distributions to perform Gibbs sampling for the IBP. In doing so, we exploit the exchangeability of IBP. For non-singleton features (i.e., features possessed by at least one other observation), we imagine that the i^{th} observation is the last observation, allowing us to write the conditional distribution in the following way:

$$P(z_{ik} = 1|\mathbf{z}_{-i,k}) = \frac{n_{-i,k}}{N} \tag{2}$$

where $\mathbf{z}_{-i,k}$ is the feature assignments for column k except for observation i, and $n_{-i,k}$ is the number of observations that possesses feature k excluding observation i.

3. Infinite Linear-Gaussian Model

Now that we have a prior over binary matrices, we put it in action by outlining a model of unsupervised latent feature learning. Here we consider a simple linear-Gaussian model (Griffiths & Ghahramani, 2005, 2006).

Model: An important goal of the model is to find a set of latent features, denoted Z, "explaining" a set of observations, denoted X. Let Z be a binary feature ownership matrix, where $z_{i,k} = 1$ indicates that object i possesses feature k. Let X be real-valued observations (e.g., $X_{i,j}$ is the value of feature j for object i). The problem of inferring Z from X can be solved via Bayes' rule:

$$p(Z|X) = \frac{p(X|Z)\ p(Z)}{\sum_{Z'} p(X|Z')\ p(Z')} \tag{3}$$

where $p(Z)$ is the prior probability of the feature ownership matrix, and $p(X|Z)$ is the likelihood of the observed objects, given the features. We assign IBP as a prior over binary feature ownership matrices, $P(Z)$, which is described above. We now describe the likelihood function.

The likelihood function is based on a linear-Gaussian model. Let z_i be the feature values for object i, and let x_i be the observed values for object i. Then x_i is drawn from a Gaussian distribution whose mean is a linear function of the features, $z_i W$, and whose covariance matrix equals $\sigma_X^2 I$, where W is a weight matrix (the weight matrix itself is drawn from zero-mean Gaussian distribution with covariance $\sigma_W^2 I$). Given these assumptions, the likelihood for a feature matrix is:

$$p(X|Z, W, \sigma_X^2) = \frac{1}{(2\pi\sigma_X^2)^{ND/2}}\ \exp\{-\frac{1}{2\sigma_X^2}\mathrm{tr}((X - ZW)^T(X - ZW))\} \tag{4}$$

where D is the dimensionality of X, and $\mathrm{tr}(\cdot)$ denotes the trace operator. A graphical model representation of the model can be found in Figure 2.

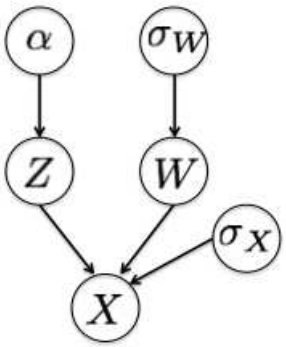

Figure 2: A graphical model for the infinite linear Gaussian model.

Inference: Exact inference in the model is computationally intractable and, thus, approximate inference must be performed using Markov chain Monte Carlo (MCMC) sampling methods (Gelman et al., 1995; Gilks, Richardson, & Spiegelhalter, 1996). We describe the MCMC sampling algorithm of Griffiths and Ghahramani (2005). A Matlab implementation of this sampling algorithm can be found on the Computational Cognition Cheat Sheet website.

This inference algorithm is a combination of Gibbs and Metropolis-Hastings (MH) steps. We take Gibbs steps when sampling the cells in the feature ownership matrix. For all i with $n_{-i,k} > 0$ (non-singleton features), the conditional distribution can be computed using Equation 4 (the likelihood) and Equation 2 (the prior). To sample the number of new features for observation i, we compute a truncated distribution for a limited number of proposals (e.g., letting the number of new features range from 0 up to 6) using Equation 4 (the likelihood) and the prior over new features, Poisson($\frac{\alpha}{N}$). Then we sample the number of new features for observation i from this truncated distribution (after normalizing it).

We take Gibbs steps when sampling α. The hyper-prior over the α parameter is a vague Gamma distribution, α Gamma$(1, 1)$. The resulting posterior conditional for α is still a Gamma distribution, $\alpha|Z \sim G(1 + K_+, 1 + \sum_{i=1}^{N} H_i)$ (Görur, Jaekel, & Rasmussen, 2006).

We take MH steps when sampling σ_X. Proposals are generated by perturbing the current value by $\epsilon \sim$Uniform$(-0.05, 0.05)$. We compute the acceptance function by computing the likelihood (Equation 4) twice, once with the current value (the denominator of the acceptance function) and once with the proposed value (the nominator of the acceptance function).

Simulations: We applied the model to a synthetic data set of 100 images, denoted X. Each image x_i was a real-valued vector of length $6 \times 6 = 36$. Each image was a superposition of a subset of four base images. These base images corresponded to the rows of the weight matrix W. For each row of the binary feature matrix, z_i, each entry was set to 1 with probability 0.5 and set to 0 otherwise. Then we generated each x_i by adding white noise with variance $\sigma_x^2 = 0.5$ to the linear combination of the weight matrix based on the binary feature matrix, $z_i W$. Figure 3 illustrates this process: the four base images (top row), binary feature vectors for a subsample, and the resulting images from the data set (bottom row).

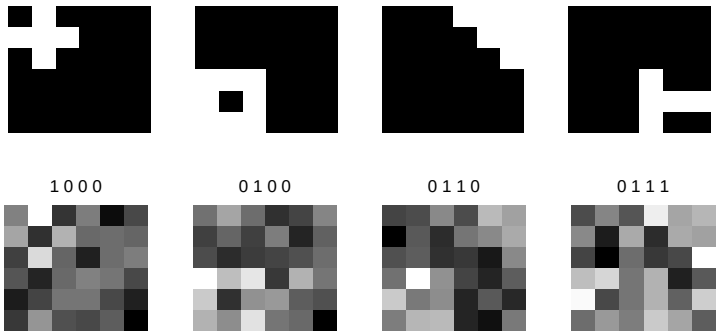

Figure 3: (Top row) Four basis images underlying the data set. These binary images of length $6 \times 6 = 36$ constitute each of the four rows of the weight matrix. (Bottom row) Binary vectors of length 4 at the titles are four rows from the feature matrix, Z. Present features are denoted 1. Corresponding images from the dataset are depicted. Each image is a linear combination of the present features and white noise.

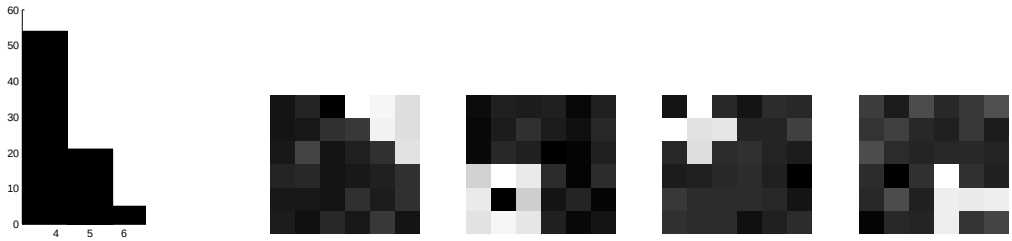

Figure 4: (Left) The posterior distribution of the number of features based on the samples of the MCMC chain. (Right) The expected posterior weight matrix computed in the following way: $E[W|Z,X] = (Z^TZ + \frac{\sigma_X^2}{\sigma_A^2}I)^{-1}Z^TX$.

In our Gibbs sampler, we initialized $\alpha = 1$, and initialized Z to a random draw from IBP(α). We ran our simulation for 1000 steps. We discarded the first 200 samples as burn-in. We thinned the remaining 800 samples by retaining only every 10^{th} sample. Our results are based on this remaining 80 samples. (We ran many other chains initialized differently. Results reported were typical across all simulations.)

Figure 4 illustrates the results. The left panel in Figure 4 illustrates the distribution of the number of latent features discovered by the model. Clearly, the model captured that there were 4 latent features underlying the observations. The four images on the right panel in Figure 4 illustrate the weights for the four most frequent features across the chain. Notice how remarkably close the latent features discovered by the model are to the weights used to generate the data in the first place (top row of Figure 3).

Aldous, D. (1985). Exchangeability and related topics. In *École d'été de probabilités de Saint-Flour, XIII—1983* (pp. 1-198). Berlin: Springer.

Austerweil, J. L. & Griffiths, T. L. (2009). Analyzing human feature learning as nonparametric Bayesian inference. In D. Koller, Y. Bengio, D. Schuurmans, & L. Bottou (Eds.), *Advances in Neural Information Processing Systems 21*. Cambridge, MA: MIT Press.

Gelman, A., Carlin, J. B., Stern. H. S., & Rubin, D. B. (1995). *Bayesian Data Analysis*. London: Chapman and Hall.

Gilks, W. R., Richardson, S., & Spiegelhalter, D. J. (1996). *Markov Chain Monte Carlo in Practice*. London: Chapman and Hall.

Görur, D., Jaekel, F., & Rasmussen, C. E. (2006). A choice model with infinitely many features. *Proceedings of the International Conference on Machine Learning*.

Griffiths, T. L. & Ghahramani, Z. (2005). Infinite latent feature models and the Indian buffet process. Gatsby Unit Technical Report GCNU-TR-2005-001.

Griffiths, T. L. & Ghahramani, Z. (2006). Infinite latent feature models and the Indian buffet process. In B. Schölkopf, J. Platt, & T. Hofmann (Eds.), *Advances in Neural Information Processing Systems 18*. Cambridge, MA: MIT Press.

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

Thibaux, R. & Jordan, M. I. (2007). Hierarchical beta processes and the Indian buffet process. *Proceedings of the Tenth Conference on Artificial Intelligence and Statistics (AISTATS)*.

Wood, F., Griffiths, T. L., & Ghahramani, Z. (2006). A non-parametric Bayesian method for inferring hidden causes. Proceedings of the 2006 conference on *Uncertainty in Artificial Intelligence*.

Yildirim, I. & Jacobs, R. A. (2012). A rational analysis of the acquisition of multisensory representations. *Cognitive Science*, **36**, 305-332.