Gatsby Computational Neuroscience Unit
University College London
http://www.gatsby.ucl.ac.uk

17 Queen Square, London
WC1N 3AR, United Kingdom
+44 20 7679 1176

Funded in part by the Gatsby Charitable Foundation.

May 5, 2005

GCNU TR 2005–001

Infinite Latent Feature Models and the Indian Buffet Process

Thomas L. Griffiths
Cognitive and Linguistic Sciences
Brown University

Zoubin Ghahramani
Gatsby Unit

Abstract

We define a probability distribution over equivalence classes of binary matrices with a finite number of rows and an unbounded number of columns. This distribution is suitable for use as a prior in probabilistic models that represent objects using a potentially infinite array of features. We derive the distribution by taking the limit of a distribution over $N \times K$ binary matrices as $K \rightarrow \infty$, a strategy inspired by the derivation of the Chinese restaurant process (Aldous, 1985; Pitman, 2002) as the limit of a Dirichlet-multinomial model. This strategy preserves the exchangeability of the rows of matrices. We define several simple generative processes that result in the same distribution over equivalence classes of binary matrices, one of which we call the Indian buffet process. We illustrate the use of this distribution as a prior in an infinite latent feature model, deriving a Markov chain Monte Carlo algorithm for inference in this model and applying this algorithm to an artificial dataset.

Infinite Latent Feature Models and the Indian Buffet Process

Thomas L. Griffiths
Cognitive and Linguistic Sciences
Brown University

Zoubin Ghahramani
Gatsby Unit

1 Introduction

Unsupervised learning aims to recover the latent structure responsible for generating the observed properties of a set of objects. The statistical models typically used in unsupervised learning draw upon a relatively small repertoire of representations for this latent structure. The simplest representation, used in mixture models, associates each object with a single latent class. This approach is appropriate when objects can be partitioned into relatively homogeneous subsets. However, the properties of many objects are better captured by representing each object as possessing multiple latent features. For example, when describing a friend, we might characterize him as married, a Democrat, and a Red Sox fan. Each of these features may be useful in explaining aspects of his behavior, and is not necessarily directly observable.

Several methods exist for representing objects in terms of latent features. One approach is to associate each object with a probability distribution over features. This approach has proven successful in modeling the content of documents, where each feature indicates one of the topics that appears in the document (e.g., Blei, Ng, & Jordan, 2003). However, using a probability distribution over features introduces a conservation constraint: the more an object expresses one feature, the less it can express others. This constraint is inappropriate in many settings – in the example above, it would imply that the more our friend appreciates the Red Sox, the less he would be married – and is not imposed by other feature-based representation schemes. For instance, we could choose to represent each object as a binary vector, with entries indicating the presence or absence of each feature (e.g., Ueda & Saito, 2003), allow each feature to take on a continuous value, representing objects with points in a latent space (e.g., Jolliffe, 1986), or define a factorial model, in which each feature takes on one of a discrete set of values (e.g., Zemel & Hinton, 1994; Ghahramani, 1995).

Regardless of the form the representation takes, a critical question in all of these approaches is the dimensionality of that representation: how many classes or features are needed to express the latent structure responsible for the observed data. Often, this is treated as a model selection problem, choosing the model with the dimensionality that results in the best performance. This treatment of the problem assumes that there is a single, finite-dimensional representation that correctly characterizes the properties of the observed objects. An alternative is to assume that the number of classes or features is actually potentially unbounded, and that the observed objects only manifest a sparse subset of those classes or features (Rasmussen & Ghahramani, 2001). This assumption seems appropriate when describing our friend the Red Sox fan: it is possible to imagine an arbitrarily large set of features that could be used to describe people, and which subset of features we actually use will depend upon the properties we want to explain.

The assumption that the observed objects manifest a sparse subset of an unbounded number of latent classes is often used in nonparametric Bayesian statistics. In particular, this assumption is made in Dirichlet process mixture models, which are used for nonparametric density estimation (Antoniak, 1974; Escobar & West, 1995; Ferguson, 1983; Neal, 2000). Under one interpretation of a Dirichlet process mixture model, each datapoint is assigned to a latent class, and each class is associated with a distribution over observable properties. The prior distribution over assignments of datapoints to classes is specified in such a way that the number of classes used by the model is bounded only by the number of objects, making Dirichlet process mixture models "infinite" mixture models (Rasmussen, 2000). Recent work has extended these methods to models in which each object is represented by a distribution over features (Blei, Griffiths, Jordan, & Tenenbaum, 2004; Teh, Jordan, Beal, & Blei, 2004). However, there are no equivalent methods for dealing with other feature-based representations, be they binary vectors, factorial structures, or vectors of continuous feature values.

In this paper, we take the idea of defining priors over infinite combinatorial structures from nonparametric Bayesian statistics, and use it to develop methods for unsupervised learning in which each object is represented by a sparse subset of an unbounded number of features. These features can be binary, take on multiple discrete values, or have continuous weights. In all of these representations, the difficult problem is deciding which features an object should possess. The set of features possessed by a set of objects can be expressed in the form of a binary matrix, where each row is an object, each column is a feature, and an entry of 1 indicates that a particular objects possesses a particular feature. We thus focus on the problem of defining a distribution on infinite sparse binary matrices. This distribution can be used to define probabilistic models that represent objects with infinitely many binary features, and can be combined with priors on feature values to produce factorial and continuous representations.

The plan of the paper is as follows. Section 2 reviews the principles behind infinite mixture models, focusing on the prior on class assignments assumed in these models, which can be defined in terms of a simple stochastic process – the *Chinese restaurant process*. Section 3 discusses the role of a prior on infinite binary matrices in defining infinite latent feature models. Section 4 describes such a prior, corresponding to a stochastic process we call the *Indian buffet process*. Section 5 illustrates how this prior can be used, defining an infinite-dimensional linear-Gaussian model, deriving a sampling algorithm for inference in this model, and applying it to a simple dataset. Section 6 discusses conclusions and future work.

2 Latent class models

Assume we have N objects, with the ith object having D observable properties represented by a row vector $\mathbf{x}_i$. In a latent class model, such as a mixture model, each object is assumed to belong to a single class, c_i, and the properties $\mathbf{x}_i$ are generated from a distribution determined by that class. Using the matrix $\mathbf{X} = \left[\mathbf{x}_1^T \, \mathbf{x}_2^T \, \cdots \, \mathbf{x}_N^T\right]^T$ to indicate the properties of all N objects, and the vector $\mathbf{c} = [c_1 \; c_2 \; \cdots \; c_N]^T$ to indicate their class assignments, the model is specified by a prior over assignment vectors $P(\mathbf{c})$, and a distribution over property matrices conditioned on those assignments, $p(\mathbf{X}|\mathbf{c})$.[1] These two distributions can be dealt with separately: $P(\mathbf{c})$ specifies the number of classes and their relative probability, while $p(\mathbf{X}|\mathbf{c})$ determines how these classes relate to the properties of objects. In this section, we will focus on the prior over assignment vectors, $P(\mathbf{c})$, showing how such a prior can be defined without placing an upper bound on the number of classes.

[1] We will use $P(\cdot)$ to indicate probability mass functions, and $p(\cdot)$ to indicate probability density functions. We will assume that $\mathbf{x}_i \in \mathbb{R}^D$, and $p(\mathbf{X}|\mathbf{c})$ is thus a density.

2.1 Finite mixture models

Mixture models assume that the assignment of an object to a class is independent of the assignments of all other objects. If there are K classes, we have

$$P(\mathbf{c}|\theta) = \prod_{i=1}^{N} P(c_i|\theta) = \prod_{i=1}^{N} \theta_{c_i}, \tag{1}$$

where θ is a multinomial distribution over those classes, and θ_k is the probability of class k under that distribution. Under this assumption, the probability of the properties of all N objects $\mathbf{X}$ can be written as

$$p(\mathbf{X}|\theta) = \prod_{i=1}^{N} \sum_{k=1}^{K} p(\mathbf{x}_i|c_i = k)\, \theta_k. \tag{2}$$

The distribution from which each $\mathbf{x}_i$ is generated is thus a *mixture* of the K class distributions $p(\mathbf{x}_i|c_i = k)$, with θ_k determining the weight of class k.

The mixture weights θ can either be treated as a parameter to be estimated, or a variable with prior distribution $p(\theta)$. In Bayesian approaches to mixture modeling, a standard choice for $p(\theta)$ is a symmetric Dirichlet distribution. The Dirichlet distribution on multinomials over K classes has parameters $\alpha_1, \alpha_2, \ldots, \alpha_K$, and is conjugate to the multinomial (e.g., Bernardo & Smith, 1994). The probability of any multinomial distribution θ is given by

$$p(\theta) = \frac{\prod_{k=1}^{K} \theta_k^{\alpha_k - 1}}{D(\alpha_1, \alpha_2, \ldots, \alpha_K)}, \tag{3}$$

in which $D(\alpha_1, \alpha_2, \ldots, \alpha_K)$ is the Dirichlet normalizing constant

$$D(\alpha_1, \alpha_2, \ldots, \alpha_K) = \int_{\Delta_K} \prod_{k=1}^{K} \theta_k^{\alpha_k - 1}\, d\theta \tag{4}$$

$$= \frac{\prod_{k=1}^{K} \Gamma(\alpha_k)}{\Gamma(\sum_{k=1}^{K} \alpha_k)}, \tag{5}$$

where Δ_K is the simplex of multinomials over K classes, and $\Gamma(\cdot)$ is the generalized factorial function, with $\Gamma(m) = (m-1)!$ for any non-negative integer m. In a *symmetric* Dirichlet distribution, all α_k are equal. For example, we could take $\alpha_k = \frac{\alpha}{K}$ for all k. In this case, Equation 5 becomes

$$D(\tfrac{\alpha}{K}, \tfrac{\alpha}{K}, \ldots, \tfrac{\alpha}{K}) = \frac{\Gamma(\frac{\alpha}{K})^K}{\Gamma(\alpha)}, \tag{6}$$

and the mean of θ is the multinomial that is uniform over all classes.

The probability model that we have defined is

$$\begin{aligned} \theta \,|\, \alpha &\sim \text{Dirichlet}(\tfrac{\alpha}{K}, \tfrac{\alpha}{K}, \ldots, \tfrac{\alpha}{K}) \\ c_i \,|\, \theta &\sim \text{Discrete}(\theta) \end{aligned}$$

where Discrete(θ) is the multiple-outcome analogue of a Bernoulli event, where the probabilities of the outcomes are specified by θ (i.e. $c_i \,|\, \theta \sim \text{Multinomial}(\theta, 1)$). The dependencies among variables in this model are shown in Figure 1 (a). Having defined a prior on θ, we can simplify this model by integrating over all values of θ rather than representing them explicitly.

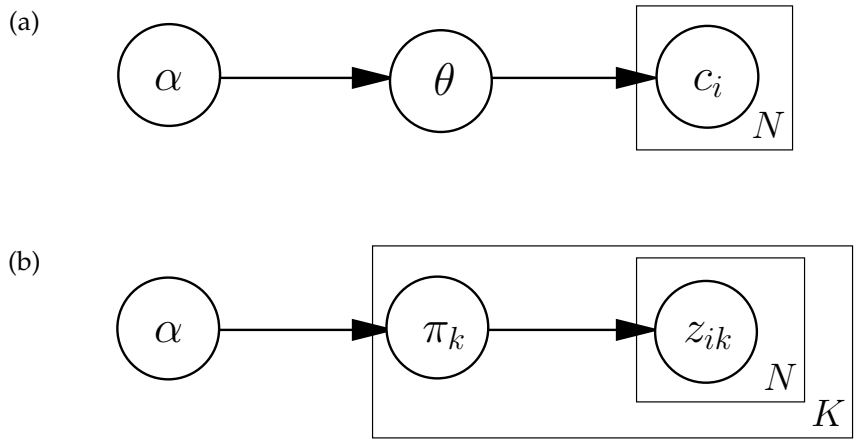

Figure 1: Graphical models for different priors. Nodes are variables, arrows indicate dependencies, and plates (Buntine, 1994) indicate replicated structures. (a) The Dirichlet-multinomial model used in defining the Chinese restaurant process. (b) The beta-binomial model used in defining the Indian buffet process.

The marginal probability of an assignment vector $\mathbf{c}$, integrating over all values of θ, is

$$
\begin{aligned}
P(\mathbf{c}) &= \int_{\Delta_K} \prod_{i=1}^{n} P(c_i|\theta) p(\theta)\, d\theta && (7)\\
&= \int_{\Delta_K} \frac{\prod_{k=1}^{K} \theta_k^{m_k+\alpha_k-1}}{D(\alpha_1, \alpha_2, \ldots, \alpha_K)}\, d\theta && (8)\\
&= \frac{D(m_1 + \frac{\alpha}{K}, m_2 + \frac{\alpha}{K}, \ldots, m_k + \frac{\alpha}{K})}{D(\frac{\alpha}{K}, \frac{\alpha}{K}, \ldots, \frac{\alpha}{K})} && (9)\\
&= \frac{\prod_{k=1}^{K} \Gamma(m_k + \frac{\alpha}{K})}{\Gamma(\frac{\alpha}{K})^K} \frac{\Gamma(\alpha)}{\Gamma(N+\alpha)}, && (10)
\end{aligned}
$$

where $m_k = \sum_{i=1}^{N} \delta(c_i = k)$ is the number of objects assigned to class k. The tractability of this integral is a result of the fact that the Dirichlet is conjugate to the multinomial.

Equation 10 defines a probability distribution over the class assignments $\mathbf{c}$ as an ensemble. Individual class assignments are no longer independent. Rather, they are *exchangeable* (Bernardo & Smith, 1994), with the probability of an assignment vector remaining the same when the indices of the objects are permuted. Exchangeability is a desirable property in a distribution over class assignments, because the indices labelling objects are typically arbitrary. However, the distribution on assignment vectors defined by Equation 10 assumes an upper bound on the number of classes of objects, since it only allows assignments of objects to up to K classes.

2.2 Infinite mixture models

Intuitively, defining an infinite mixture model means that we want to specify the probability of $\mathbf{X}$ in terms of infinitely many classes, modifying Equation 2 to become

$$
p(\mathbf{X}|\theta) = \prod_{i=1}^{N} \sum_{k=1}^{\infty} p(\mathbf{x}_i|c_i = k)\, \theta_k, \qquad (11)
$$

where θ is an infinite-dimensional multinomial distribution. In order to repeat the argument above, we would need to define a prior, $p(\theta)$, on infinite-dimensional multinomials, and com-

pute the probability of $\mathbf{c}$ by integrating over θ. This is essentially the strategy that is taken in deriving infinite mixture models from the Dirichlet process (Antoniak, 1974; Ferguson, 1983; Ishwaran & James, 2001; Sethuraman, 1994). Instead, we will work directly with the distribution over assignment vectors given in Equation 10, considering its limit as the number of classes approaches infinity (c.f. Green & Richardson, 2001; Neal, 1992, 2000).

Expanding the gamma functions in Equation 10 using the recursion $\Gamma(x) = (x-1)\Gamma(x-1)$ and cancelling terms produces the following expression for the probability of an assignment vector $\mathbf{c}$:

$$P(\mathbf{c}) = \left(\frac{\alpha}{K}\right)^{K_+} \left(\prod_{k=1}^{K^+} \prod_{j=1}^{m_k-1} (j + \tfrac{\alpha}{K})\right) \frac{\Gamma(\alpha)}{\Gamma(N+\alpha)}, \tag{12}$$

where K_+ is the number of classes for which $m_k > 0$, and we have re-ordered the indices such that $m_k > 0$ for all $k \leq K_+$. There are K^N possible values for $\mathbf{c}$, which diverges as $K \to \infty$. As this happens, the probability of any single set of class assignments goes to 0. Since $K_+ \leq N$ and N is finite, it is clear that $P(\mathbf{c}) \to 0$ as $K \to \infty$, since $\frac{1}{K} \to 0$. Consequently, we will define a distribution over equivalence classes of assignment vectors, rather than the vectors themselves.

Specifically, we will define a distribution on *partitions* of objects. In our setting, a partition is a division of the set of N objects into subsets, where each object belongs to a single subset and the ordering of the subsets does not matter. Two assignment vectors that result in the same division of objects correspond to the same partition. For example, if we had three objects, the class assignments $\{c_1, c_2, c_3\} = \{1, 1, 2\}$ would correspond to the same partition as $\{2, 2, 1\}$, since all that differs between these two cases is the labels of the classes. A partition thus defines an equivalence class of assignment vectors, which we denote $[\mathbf{c}]$, with two assignment vectors belonging to the same equivalence class if they correspond to the same partition. A distribution over partitions is sufficient to allow us to define an infinite mixture model, since these equivalence classes of class assignments are the same as those induced by identifiability: $p(\mathbf{X}|\mathbf{c})$ is the same for all assignment vectors $\mathbf{c}$ that correspond to the same partition, so we can apply statistical inference at the level of partitions rather than the level of assignment vectors.

Assume we have a partition of N objects into K_+ subsets, and we have $K = K_0 + K_+$ class labels that can be applied to those subsets. Then there are $\frac{K!}{K_0!}$ assignment vectors $\mathbf{c}$ that belong to the equivalence class defined by that partition, $[\mathbf{c}]$. We can define a probability distribution over partitions by summing over all class assignments that belong to the equivalence class defined by each partition. The probability of each of those class assignments is equal under the distribution specified by Equation 12, so we obtain

$$\begin{aligned} P([\mathbf{c}]) &= \sum_{\mathbf{c} \in [\mathbf{c}]} P(\mathbf{c}) && (13) \\ &= \frac{K!}{K_0!} \left(\frac{\alpha}{K}\right)^{K_+} \left(\prod_{k=1}^{K^+} \prod_{j=1}^{m_k-1} (j + \tfrac{\alpha}{K})\right) \frac{\Gamma(\alpha)}{\Gamma(N+\alpha)}. && (14) \end{aligned}$$

Rearranging the first two terms, we can compute the limit of the probability of a partition as $K \to \infty$, which is

$$\begin{aligned} &\lim_{K\to\infty} \alpha^{K_+} \cdot \frac{K!}{K_0!\, K^{K_+}} \cdot \left(\prod_{k=1}^{K_+} \prod_{j=1}^{m_k-1} (j + \tfrac{\alpha}{K})\right) \cdot \frac{\Gamma(\alpha)}{\Gamma(N+\alpha)} \\ &= \alpha^{K_+} \cdot \quad 1 \quad \cdot \left(\prod_{k=1}^{K_+} (m_k - 1)!\right) \cdot \frac{\Gamma(\alpha)}{\Gamma(N+\alpha)}. \end{aligned} \tag{15}$$

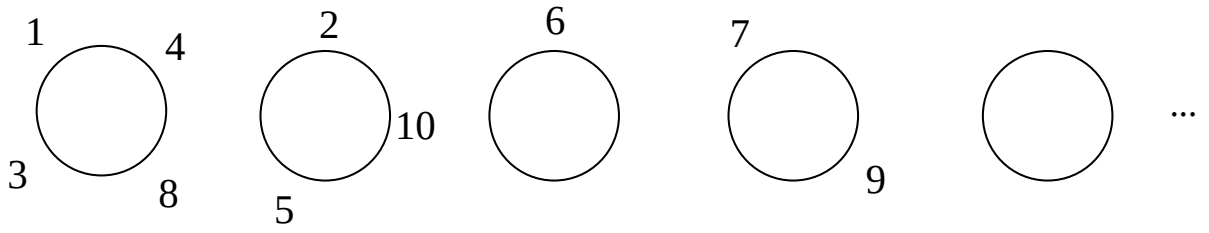

Figure 2: A partition induced by the Chinese restaurant process. Numbers indicate customers (objects), circles indicate tables (classes).

The details of the steps taken in computing this limit are given in the Appendix. These limiting probabilities define a valid distribution over partitions, and thus over equivalence classes of class assignments, providing a prior over class assignments for an infinite mixture model. Objects are exchangeable under this distribution, just as in the finite case: the probability of a partition is not affected by the ordering of the objects, since it depends only on the counts m_k.

As noted above, the distribution over partitions specified by Equation 15 can be derived in a variety of ways – by taking limits (Green & Richardson, 2001; Neal, 1992; 2000), from the Dirichlet process (Blackwell & McQueen, 1973), or from other equivalent stochastic processes (Ishwaran & James, 2001; Sethuraman, 1994). We will briefly discuss a simple process that produces the same distribution over partitions: the Chinese restaurant process.

2.3 The Chinese restaurant process

The Chinese restaurant process (CRP) was named by Jim Pitman and Lester Dubins, based upon a metaphor in which the objects are customers in a restaurant, and the classes are the tables at which they sit (the process first appears in Aldous, 1985, where it is attributed to Pitman). Imagine a restaurant with an infinite number of tables, each with an infinite number of seats.[2] The customers enter the restaurant one after another, and each choose a table at random. In the CRP with parameter α, each customer chooses an occupied table with probability proportional to the number of occupants, and chooses the next vacant table with probability proportional to α. For example, Figure 2 shows the state of a restaurant after 10 customers have chosen tables using this procedure. The first customer chooses the first table with probability $\frac{\alpha}{\alpha} = 1$. The second customer chooses the first table with probability $\frac{1}{1+\alpha}$, and the second table with probability $\frac{\alpha}{1+\alpha}$. After the second customer chooses the second table, the third customer chooses the first table with probability $\frac{1}{2+\alpha}$, the second table with probability $\frac{1}{2+\alpha}$, and the third table with probabililty $\frac{\alpha}{2+\alpha}$. This process continues until all customers have seats, defining a distribution over allocations of people to tables, and, more generally, objects to classes. Extensions of the CRP and connections to other stochastic processes are pursued in depth by Pitman (2002).

The distribution over partitions induced by the CRP is the same as that given in Equation 15. If we assume an ordering on our N objects, then we can assign them to classes sequentially using the method specified by the CRP, letting objects play the role of customers and classes play the role of tables. The ith object would be assigned to the kth class with probability

$$P(c_i = k | c_1, c_2, \ldots, c_{i-1}) = \begin{cases} \frac{m_k}{i-1+\alpha} & k \leq K_+ \\ \frac{\alpha}{i-1+\alpha} & \text{otherwise} \end{cases} \tag{16}$$

where m_k is the number of objects currently assigned to class k, and K_+ is the number of classes for which $m_k > 0$. If all N objects are assigned to classes via this process, the probability

[2] Pitman and Dubins, both statisticians at UC Berkeley, were inspired by the apparently infinite capacity of Chinese restaurants in San Francisco when they named the process.

of a partition of objects $\mathbf{c}$ is that given in Equation 15. The CRP thus provides an intuitive means of specifying a prior for infinite mixture models, as well as revealing that there is a simple sequential process by which exchangeable class assignments can be generated.

2.4 Inference by Gibbs sampling

Inference in an infinite mixture model is only slightly more complicated than inference in a mixture model with a finite, fixed number of classes. The standard algorithm used for inference in infinite mixture models is Gibbs sampling (Escobar & West, 1995; Neal, 2000). Gibbs sampling is a Markov chain Monte Carlo (MCMC) method, in which variables are successively sampled from their distributions when conditioned on the current values of all other variables (Geman & Geman, 1984). This process defines a Markov chain, which ultimately converges to the distribution of interest (see Gilks, Richardson, & Spiegelhalter, 1996).

Implementing a Gibbs sampler requires deriving the full conditional distribution for all variables to be sampled. In a mixture model, these variables are the class assignments $\mathbf{c}$. The relevant full conditional distribution is $P(c_i|\mathbf{c}_{-i}, \mathbf{X})$, the probability distribution over c_i conditioned on the class assignments of all other objects, $\mathbf{c}_{-i}$, and the data, $\mathbf{X}$. By applying Bayes' rule, this distribution can be expressed as

$$P(c_i = k|\mathbf{c}_{-i}, \mathbf{X}) \propto p(\mathbf{X}|\mathbf{c})P(c_i = k|\mathbf{c}_{-i}), \tag{17}$$

where only the second term on the right hand side depends upon the distribution over class assignments, $P(\mathbf{c})$.

In a finite mixture model with $P(\mathbf{c})$ defined as in Equation 10, we can compute $P(c_i = k|\mathbf{c}_{-i})$ by integrating over θ, obtaining

$$\begin{aligned} P(c_i = k|\mathbf{c}_{-i}) &= \int P(c_i = k|\theta)p(\theta|\mathbf{c}_{-i})\, d\theta \\ &= \frac{m_{-i,k} + \frac{\alpha}{K}}{N - 1 + \alpha}, \end{aligned} \tag{18}$$

where $m_{-i,k}$ is the number of objects assigned to class k, not including object i. This is the posterior predictive distribution for a multinomial distribution with a Dirichlet prior.

In an infinite mixture model with a distribution over class assignments defined as in Equation 15, we can use exchangeability to find the full conditional distribution. Since it is exchangeable, $P([\mathbf{c}])$ is unaffected by the ordering of objects. Thus, we can choose an ordering in which the ith object is the last to be assigned to a class. It follows directly from the definition of the Chinese restaurant process that

$$P(c_i = k|\mathbf{c}_{-i}) = \begin{cases} \frac{m_{-i,k}}{N-1+\alpha} & m_{-i,k} > 0 \\ \frac{\alpha}{N-1+\alpha} & k = K_{-i,+} + 1 \\ 0 & \text{otherwise} \end{cases} \tag{19}$$

where $K_{-i,+}$ is the number of classes for which $m_{-i,k} > 0$. The same result can be found by taking the limit of the full conditional distribution in the finite model, given by Equation 18 (Neal, 2000).

When combined with some choice of $p(\mathbf{X}|\mathbf{c})$, Equations 18 and 19 are sufficient to define Gibbs samplers for finite and infinite mixture models respectively. Demonstrations of Gibbs sampling in infinite mixture models are provided by Neal (2000) and Rasmussen (2000). Similar MCMC algorithms are presented in Bush and MacEachern (1996), West, Muller, and Escobar (1994), Escobar and West (1995) and Ishwaran and James (2001). Algorithms that go beyond the local changes in class assignments allowed by a Gibbs sampler are given by Jain and Neal (2004) and Dahl (2003).

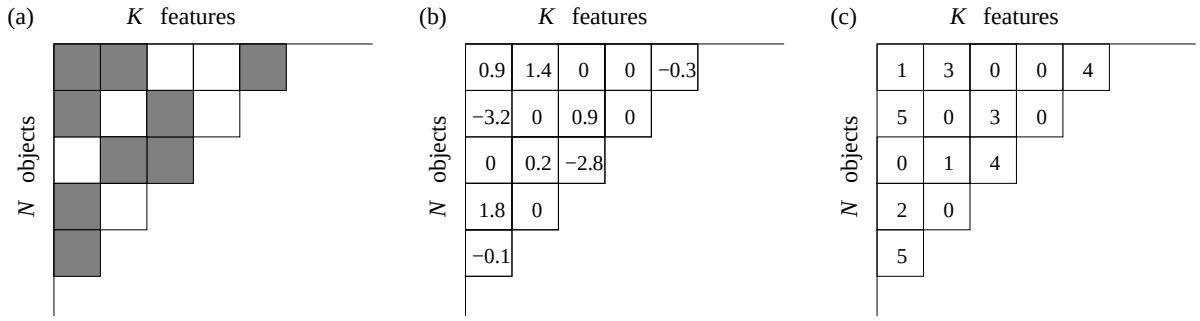

Figure 3: Feature matrices. A binary matrix $\mathbf{Z}$, as shown in (a), can be used as the basis for sparse infinite latent feature models, indicating which features take non-zero values. Elementwise multiplication of $\mathbf{Z}$ by a matrix $\mathbf{V}$ of continuous values gives a representation like that shown in (b). If $\mathbf{V}$ contains discrete values, we obtain a representation like that shown in (c).

2.5 Summary

Our review of infinite mixture models serves three purposes: it shows that infinite statistical models can be defined by specifying priors over infinite combinatorial objects; it illustrates how these priors can be derived by taking the limit of priors for finite models; and it demonstrates that inference in these models can remain possible, despite the large hypothesis spaces they imply. However, infinite mixture models are still fundamentally limited in their representation of objects, assuming that each object can only belong to a single class. In the remainder of the paper, we use the insights underlying infinite mixture models to derive methods for representing objects in terms of infinitely many latent features.

3 Latent feature models

In a latent feature model, each object is represented by a vector of latent feature values $\mathbf{f}_i$, and the properties $\mathbf{x}_i$ are generated from a distribution determined by those latent feature values. Latent feature values can be continuous, as in principal component analysis (PCA; Jolliffe, 1986), or discrete, as in cooperative vector quantization (CVQ; Zemel & Hinton, 1994; Ghahramani, 1995). In the remainder of this section, we will assume that feature values are continuous. Using the matrix $\mathbf{F} = \left[\mathbf{f}_1^T \, \mathbf{f}_2^T \, \cdots \, \mathbf{f}_N^T\right]^T$ to indicate the latent feature values for all N objects, the model is specified by a prior over features, $p(\mathbf{F})$, and a distribution over observed property matrices conditioned on those features, $p(\mathbf{X}|\mathbf{F})$. As with latent class models, these distributions can be dealt with separately: $p(\mathbf{F})$ specifies the number of features, their probability, and the distribution over values associated with each feature, while $p(\mathbf{X}|\mathbf{F})$ determines how these features relate to the properties of objects. Our focus will be on $p(\mathbf{F})$, showing how such a prior can be defined without placing an upper bound on the number of features.

We can break the matrix $\mathbf{F}$ into two components: a binary matrix $\mathbf{Z}$ indicating which features are possessed by each object, with $z_{ik} = 1$ if object i has feature k and 0 otherwise, and a second matrix $\mathbf{V}$ indicating the value of each feature for each object. $\mathbf{F}$ can be expressed as the elementwise (Hadamard) product of $\mathbf{Z}$ and $\mathbf{V}$, $\mathbf{F} = \mathbf{Z} \otimes \mathbf{V}$, as illustrated in Figure 3. In many latent feature models, such as PCA and CVQ, objects have non-zero values on every feature, and every entry of $\mathbf{Z}$ is 1. In *sparse* latent feature models (e.g., sparse PCA; d'Aspremont, Ghaoui, Jordan, & Lanckriet, 2004; Jolliffe & Uddin, 2003; Zou, Hastie, & Tibshirani, in press) only a subset of features take on non-zero values for each object, and $\mathbf{Z}$ picks out these subsets.

A prior on $\mathbf{F}$ can be defined by specifying priors for $\mathbf{Z}$ and $\mathbf{V}$ separately, with $p(\mathbf{F}) = P(\mathbf{Z})p(\mathbf{V})$. We will focus on defining a prior on $\mathbf{Z}$, since the effective dimensionality of a latent feature model is determined by $\mathbf{Z}$. Assuming that $\mathbf{Z}$ is sparse, we can define a prior

for infinite latent feature models by defining a distribution over infinite binary matrices. Our analysis of latent class models provides two desiderata for such a distribution: objects should be exchangeable, and inference should be tractable. It also suggests a method by which these desiderata can be satisfied: start with a model that assumes a finite number of features, and consider the limit as the number of features approaches infinity.

4 A distribution on infinite binary matrices

In this section, we derive a distribution on infinite binary matrices by starting with a simple model that assumes K features, and then taking the limit as $K \to \infty$. The resulting distribution corresponds to a simple generative process, which we term the Indian buffet process.

4.1 A finite feature model

We have N objects and K features, and the possession of feature k by object i is indicated by a binary variable z_{ik}. Each object can possess multiple features. The z_{ik} thus form a binary $N \times K$ feature matrix, $\mathbf{Z}$. We will assume that each object possesses feature k with probability π_k, and that the features are generated independently. In contrast to the class models discussed above, for which $\sum_k \theta_k = 1$, the probabilities π_k can each take on any value in $[0, 1]$. Under this model, the probability of a matrix $\mathbf{Z}$ given $\pi = \{\pi_1, \pi_2, \ldots, \pi_K\}$, is

$$P(\mathbf{Z}|\pi) = \prod_{k=1}^{K} \prod_{i=1}^{N} P(z_{ik}|\pi_k) = \prod_{k=1}^{K} \pi_k^{m_k} (1 - \pi_k)^{N - m_k}, \tag{20}$$

where $m_k = \sum_{i=1}^{N} z_{ik}$ is the number of objects possessing feature k.

We can define a prior on π by assuming that each π_k follows a beta distribution. The beta distribution has parameters r and s, and is conjugate to the binomial. The probability of any π_k under the Beta(r, s) distribution is given by

$$p(\pi_k) = \frac{\pi_k^{r-1}(1 - \pi_k)^{s-1}}{B(r, s)}, \tag{21}$$

where $B(r, s)$ is the beta function,

$$B(r, s) = \int_0^1 \pi_k^{r-1}(1 - \pi_k)^{s-1} \, d\pi_k \tag{22}$$

$$= \frac{\Gamma(r)\Gamma(s)}{\Gamma(r + s)}. \tag{23}$$

We will take $r = \frac{\alpha}{K}$ and $s = 1$, so Equation 23 becomes

$$B(\tfrac{\alpha}{K}, 1) = \frac{\Gamma(\frac{\alpha}{K})}{\Gamma(1 + \frac{\alpha}{K})} = \frac{K}{\alpha}, \tag{24}$$

exploiting the recursive definition of the gamma function.

The probability model we have defined is

$$\begin{aligned} \pi_k \,|\, \alpha &\sim \text{Beta}(\tfrac{\alpha}{K}, 1) \\ z_{ik} \,|\, \pi_k &\sim \text{Bernoulli}(\pi_k) \end{aligned}$$

Each z_{ik} is independent of all other assignments, conditioned on π_k, and the π_k are generated independently. A graphical model illustrating the dependencies among these variables is

shown in Figure 1 (b). Having defined a prior on π, we can simplify this model by integrating over all values for π rather than representing them explicitly. The marginal probability of a binary matrix $\mathbf{Z}$ is

$$P(\mathbf{Z}) = \prod_{k=1}^{K} \int \left(\prod_{i=1}^{N} P(z_{ik}|\pi_k) \right) p(\pi_k)\, d\pi_k \tag{25}$$

$$= \prod_{k=1}^{K} \frac{B(m_k + \frac{\alpha}{K}, N - m_k + 1)}{B(\frac{\alpha}{K}, 1)} \tag{26}$$

$$= \prod_{k=1}^{K} \frac{\frac{\alpha}{K}\Gamma(m_k + \frac{\alpha}{K})\Gamma(N - m_k + 1)}{\Gamma(N + 1 + \frac{\alpha}{K})}. \tag{27}$$

Again, the result follows from conjugacy, this time between the binomial and beta distributions. This distribution is exchangeable, depending only on the counts m_k.

This model has the important property that the expectation of the number of non-zero entries in the matrix $\mathbf{Z}$, $E\left[\mathbf{1}^T\mathbf{Z}\mathbf{1}\right] = E\left[\sum_{ik} z_{ik}\right]$, has an upper bound for any K. Since each column of $\mathbf{Z}$ is independent, the expectation is K times the expectation of the sum of a single column, $E\left[\mathbf{1}^T\mathbf{z}_k\right]$. This expectation is easily computed,

$$E\left[\mathbf{1}^T\mathbf{z}_k\right] = \sum_{i=1}^{N} E(z_{ik}) = \sum_{i=1}^{N} \int_0^1 \pi_k p(\pi_k)\, d\pi_k = N\frac{\frac{\alpha}{K}}{1 + \frac{\alpha}{K}}, \tag{28}$$

where the result follows from the fact that the expectation of a Beta(r, s) random variable is $\frac{r}{r+s}$. Consequently, $E\left[\mathbf{1}^T\mathbf{Z}\mathbf{1}\right] = KE\left[\mathbf{1}^T\mathbf{z}_k\right] = \frac{N\alpha}{1+\frac{\alpha}{K}}$. For finite K, the expectation of the number of entries in $\mathbf{Z}$ is bounded above by $N\alpha$.

4.2 Equivalence classes

In order to find the limit of the distribution specified by Equation 27 as $K \to \infty$, we need to define equivalence classes of binary matrices – the analogue of partitions for assignment vectors. Our equivalence classes will be defined with respect to a function on binary matrices, $lof(\cdot)$. This function maps binary matrices to *left-ordered* binary matrices. $lof(\mathbf{Z})$ is obtained by ordering the columns of the binary matrix $\mathbf{Z}$ from left to right by the magnitude of the binary number expressed by that column, taking the first row as the most significant bit. The left-ordering of a binary matrix is shown in Figure 4. In the first row of the left-ordered matrix, the columns for which $z_{1k} = 1$ are grouped at the left. In the second row, the columns for which $z_{2k} = 1$ are grouped at the left of the sets for which $z_{1k} = 1$. This grouping structure persists throughout the matrix.

The *history* of feature k at object i is defined to be $(z_{1k}, \ldots, z_{(i-1)k})$. Where no object is specified, we will use *history* to refer to the full history of feature k, $(z_{1k}, \ldots, z_{Nk})$. We will individuate the histories of features using the decimal equivalent of the binary numbers corresponding to the column entries. For example, at object 3, features can have one of four histories: 0, corresponding to a feature with no previous assignments, 1, being a feature for which $z_{2k} = 1$ but $z_{1k} = 0$, 2, being a feature for which $z_{1k} = 1$ but $z_{2k} = 0$, and 3, being a feature possessed by both previous objects were assigned. K_h will denote the number of features possessing the history h, with K_0 being the number of features for which $m_k = 0$ and $K_+ = \sum_{h=1}^{2^N-1} K_h$ being the number of features for which $m_k > 0$, so $K = K_0 + K_+$. This method of denoting histories also facilitates the process of placing a binary matrix in left-ordered form, as it is used in the definition of $lof(\cdot)$.

$lof(\cdot)$ is a many-to-one function: many binary matrices reduce to the same left-ordered form, and there is a unique left-ordered form for every binary matrix. We can thus use $lof(\cdot)$

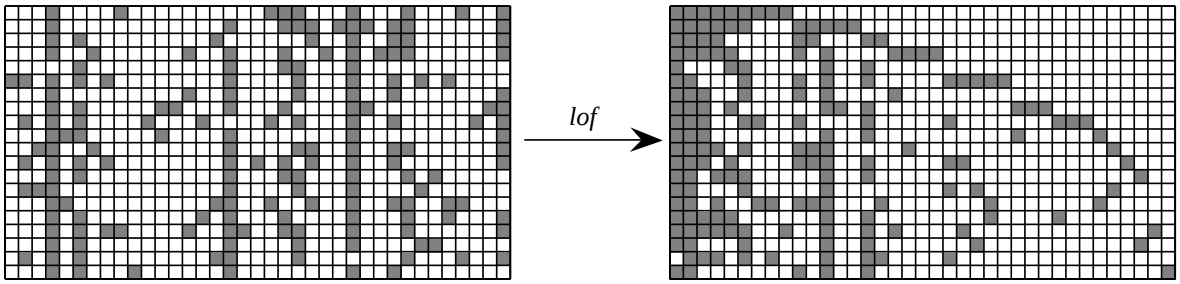

Figure 4: Binary matrices and the left-ordered form. The binary matrix on the left is transformed into the left-ordered binary matrix on the right by the function $lof(\cdot)$. This left-ordered matrix was generated from the exchangeable Indian buffet process with $\alpha = 10$. Empty columns are omitted from both matrices.

to define a set of equivalence classes. Any two binary matrices $\mathbf{Y}$ and $\mathbf{Z}$ are lof-equivalent if $lof(\mathbf{Y}) = lof(\mathbf{Z})$, that is, if $\mathbf{Y}$ and $\mathbf{Z}$ map to the same left-ordered form. The lof-equivalence class of a binary matrix $\mathbf{Z}$, denoted $[\mathbf{Z}]$, is the set of binary matrices that are lof-equivalent to $\mathbf{Z}$. lof-equivalence classes are preserved through permutation of either the rows or the columns of a matrix, provided the same permutations are applied to the other members of the equivalence class. Performing inference at the level of lof-equivalence classes is appropriate in models where feature order is not identifiable, with $p(\mathbf{X}|\mathbf{F})$ being unaffected by the order of the columns of $\mathbf{F}$. Any model in which the probability of $\mathbf{X}$ is specified in terms of a linear function of $\mathbf{F}$, such as PCA or CVQ, has this property.

We need to evaluate the cardinality of $[\mathbf{Z}]$, being the number of matrices that map to the same left-ordered form. The columns of a binary matrix are not guaranteed to be unique: since an object can possess multiple features, it is possible for two features to be possessed by exactly the same set of objects. The number of matrices in $[\mathbf{Z}]$ is reduced if $\mathbf{Z}$ contains identical columns, since some re-orderings of the columns of $\mathbf{Z}$ result in exactly the same matrix. Taking this into account, the cardinality of $[\mathbf{Z}]$ is $\binom{K}{K_0 \ldots K_{2^N-1}} = \frac{K!}{\prod_{h=0}^{2^N-1} K_h!}$, where K_h is the count of the number of columns with full history h.

lof-equivalence classes play the same role for binary matrices as partitions do for assignment vectors: they collapse together all binary matrices (assignment vectors) that differ only in column ordering (class labels). This relationship can be made precise by examining the lof-equivalence classes of binary matrices constructed from assignment vectors. Define the *class matrix* generated by an assignment vector $\mathbf{c}$ to be a binary matrix $\mathbf{Z}$ where $z_{ik} = 1$ if and only if $c_i = k$. It is straightforward to show that the class matrices generated by two assignment vectors that correspond to the same partition belong to the same lof-equivalence class, and vice versa.

4.3 Taking the infinite limit

Under the distribution defined by Equation 27, the probability of a particular lof-equivalence class of binary matrices, $[\mathbf{Z}]$, is

$$P([\mathbf{Z}]) = \sum_{\mathbf{Z} \in [\mathbf{Z}]} P(\mathbf{Z}) \tag{29}$$

$$= \frac{K!}{\prod_{h=0}^{2^N-1} K_h!} \prod_{k=1}^{K} \frac{\frac{\alpha}{K}\Gamma(m_k + \frac{\alpha}{K})\Gamma(N - m_k + 1)}{\Gamma(N + 1 + \frac{\alpha}{K})}. \tag{30}$$

In order to take the limit of this expression as $K \to \infty$, we will divide the columns of $\mathbf{Z}$ into two subsets, corresponding to the features for which $m_k = 0$ and the features for which $m_k > 0$.

Re-ordering the columns such that $m_k > 0$ if $k \leq K_+$, and $m_k = 0$ otherwise, we can break the product in Equation 30 into two parts, corresponding to these two subsets. The product thus becomes

$$\prod_{k=1}^{K} \frac{\frac{\alpha}{K}\Gamma(m_k + \frac{\alpha}{K})\Gamma(N - m_k + 1)}{\Gamma(N + 1 + \frac{\alpha}{K})}$$

$$= \left(\frac{\frac{\alpha}{K}\Gamma(\frac{\alpha}{K})\Gamma(N+1)}{\Gamma(N+1+\frac{\alpha}{K})}\right)^{K-K_+} \prod_{k=1}^{K_+} \frac{\frac{\alpha}{K}\Gamma(m_k + \frac{\alpha}{K})\Gamma(N - m_k + 1)}{\Gamma(N + 1 + \frac{\alpha}{K})} \tag{31}$$

$$= \left(\frac{\frac{\alpha}{K}\Gamma(\frac{\alpha}{K})\Gamma(N+1)}{\Gamma(N+1+\frac{\alpha}{K})}\right)^{K} \prod_{k=1}^{K_+} \frac{\Gamma(m_k + \frac{\alpha}{K})\Gamma(N - m_k + 1)}{\Gamma(\frac{\alpha}{K})\Gamma(N+1)} \tag{32}$$

$$= \left(\frac{N!}{\prod_{j=1}^{N} j + \frac{\alpha}{K}}\right)^{K} \left(\frac{\alpha}{K}\right)^{K_+} \prod_{k=1}^{K_+} \frac{(N - m_k)! \prod_{j=1}^{m_k - 1}(j + \frac{\alpha}{K})}{N!}. \tag{33}$$

Substituting Equation 33 into Equation 30 and rearranging terms, we can compute our limit

$$\lim_{K\to\infty} \frac{\alpha^{K_+}}{\prod_{h=1}^{2^N-1} K_h!} \cdot \frac{K!}{K_0!\, K^{K_+}} \cdot \left(\frac{N!}{\prod_{j=1}^{N}(j + \frac{\alpha}{K})}\right)^{K} \cdot \prod_{k=1}^{K_+} \frac{(N - m_k)! \prod_{j=1}^{m_k-1}(j + \frac{\alpha}{K})}{N!}$$

$$= \frac{\alpha^{K_+}}{\prod_{h=1}^{2^N-1} K_h!} \cdot \quad 1 \quad \cdot \quad \exp\{-\alpha H_N\} \quad \cdot \prod_{k=1}^{K_+} \frac{(N - m_k)!(m_k - 1)!}{N!}, \tag{34}$$

where H_N is the Nth harmonic number, $H_N = \sum_{j=1}^{N} \frac{1}{j}$. The details of the steps taken in computing this limit are given in the Appendix. Again, this distribution is exchangeable: neither the number of identical columns nor the column sums are affected by the ordering on objects.

4.4 The Indian buffet process

The probability distribution defined in Equation 34 can be derived from a simple stochastic process. As with the CRP, this process assumes an ordering on the objects, generating the matrix sequentially using this ordering. We will also use a culinary metaphor in defining our stochastic process, appropriately adjusted for geography. Many Indian restaurants in London offer lunchtime buffets with an apparently infinite number of dishes. We can define a distribution over infinite binary matrices by specifying a procedure by which customers (objects) choose dishes (features).

In our Indian buffet process (IBP), N customers enter a restaurant one after another. Each customer encounters a buffet consisting of infinitely many dishes arranged in a line. The first customer starts at the left of the buffet and takes a serving from each dish, stopping after a Poisson(α) number of dishes as his plate becomes overburdened. The ith customer moves along the buffet, sampling dishes in proportion to their popularity, serving himself with probability $\frac{m_k}{i}$, where m_k is the number of previous customers who have sampled a dish. Having reached the end of all previous sampled dishes, the ith customer then tries a Poisson($\frac{\alpha}{i}$) number of new dishes.

We can indicate which customers chose which dishes using a binary matrix $\mathbf{Z}$ with N rows and infinitely many columns, where $z_{ik} = 1$ if the ith customer sampled the kth dish. Figure 5 shows a matrix generated using the IBP with $\alpha = 10$. The first customer tried 17 dishes. The second customer tried 7 of those dishes, and then tried 3 new dishes. The third customer tried 3 dishes tried by both previous customers, 5 dishes tried by only the first customer, and 2

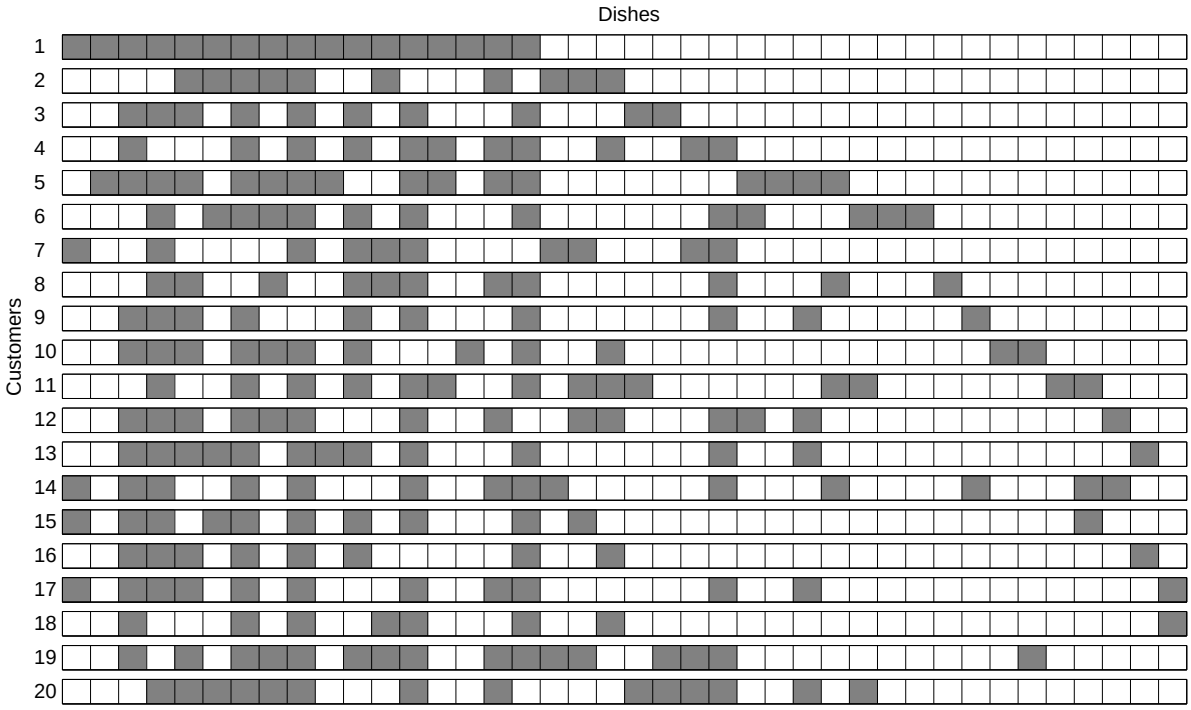

Figure 5: A binary matrix generated by the Indian buffet process with $\alpha = 10$.

new dishes. Vertically concatenating the choices of the customers produces the binary matrix shown in the figure.

Using $K_1^{(i)}$ to indicate the number of new dishes sampled by the ith customer, the probability of any particular matrix being produced by this process is

$$P(\mathbf{Z}) = \frac{\alpha^{K_+}}{\prod_{i=1}^{N} K_1^{(i)}!} \exp\{-\alpha H_N\} \prod_{k=1}^{K_+} \frac{(N - m_k)!(m_k - 1)!}{N!}. \tag{35}$$

As can be seen from Figure 5, the matrices produced by this process are generally not in left-ordered form. However, these matrices are also not ordered arbitrarily because the Poisson draws always result in choices of new dishes that are to the right of the previously sampled dishes. Customers are not exchangeable under this distribution, as the number of dishes counted as $K_1^{(i)}$ depends upon the order in which the customers make their choices. However, if we only pay attention to the lof-equivalence classes of the matrices generated by this process, we obtain the exchangeable distribution $P([\mathbf{Z}])$ given by Equation 34: $\frac{\prod_{i=1}^{N} K_1^{(i)}!}{\prod_{h=1}^{2^N-1} K_h!}$ matrices generated via this process map to the same left-ordered form, and $P([\mathbf{Z}])$ is obtained by multiplying $P(\mathbf{Z})$ from Equation 35 by this quantity.

It is possible to define a similar sequential process that directly produces a distribution on left-ordered binary matrices in which customers are exchangeable, but this requires more effort on the part of the customers. In the *exchangeable* Indian buffet process, the first customer samples a Poisson(α) number of dishes, moving from left to right. The ith customer moves along the buffet, and makes a single decision for each set of dishes with the same history. If there are K_h dishes with history h, under which m_h previous customers have sampled each of those dishes, then the customer samples a Binomial($\frac{m_h}{i}, K_h$) number of those dishes, starting at the left. Having reached the end of all previous sampled dishes, the ith customer then tries a Poisson($\frac{\alpha}{i}$) number of new dishes. Attending to the history of the dishes and always sampling from the left guarantees that the resulting matrix is in left-ordered form, and it is easy to show

that the matrices produced by this process have the same probability as the corresponding lof-equivalence classes under Equation 34.

4.5 A distribution over collections of histories

In Section 4.2, we noted that lof-equivalence classes of binary matrices generated from assignment vectors correspond to partitions. Likewise, lof-equivalence classes of general binary matrices correspond to simple combinatorial structures: vectors of non-negative integers. Fixing some ordering of N objects, a collection of feature histories on those objects can be represented by a frequency vector $\mathbf{K} = (K_1, \ldots, K_{2^N-1})$, indicating the number of times each history appears in the collection.[3] A collection of feature histories can be translated into a left-ordered binary matrix by horizontally concatenating an appropriate number of copies of the binary vector representing each history into a matrix. A left-ordered binary matrix can be translated into a collection of feature histories by counting the number of times each history appears in that matrix. Since partitions are a subset of all collections of histories – namely those collections in which each object appears in only one history – this process is strictly more general than the CRP.

This connection between lof-equivalence classes of feature matrices and collections of feature histories suggests another means of deriving the distribution specified by Equation 34, operating directly on the frequencies of these histories. We can define a distribution on vectors of non-negative integers $\mathbf{K}$ by assuming that each K_h is generated independently from a Poisson distribution with parameter $\alpha B(m_h, N - m_h + 1) = \alpha\frac{(m_h-1)!(N-m_h)!}{N!}$ where m_h is the number of non-zero elements in the history h. This gives

$$P(\mathbf{K}) = \prod_{h=1}^{2^N-1} \frac{\left(\alpha\frac{(m_h-1)!(N-m_h)!}{N!}\right)^{K_h}}{K_h!} \exp\left\{-\alpha\frac{(m_h-1)!(N-m_h)!}{N!}\right\} \quad (36)$$

$$= \frac{\alpha^{\sum_{h=1}^{2^N-1} K_h}}{\prod_{h=1}^{2^N-1} K_h!} \exp\{-\alpha H_N\} \prod_{h=1}^{2^N-1} \left(\frac{(m_h-1)!(N-m_h)!}{N!}\right)^{K_h}, \quad (37)$$

which is easily seen to be the same as $P([\mathbf{Z}])$ in Equation 34. The harmonic number in the exponential term is obtained by summing $\frac{(m_h-1)!(N-m)!}{N!}$ over all histories h. There are $\binom{N}{j}$ histories for which $m_h = j$, so we have

$$\sum_{h=1}^{2^N-1} \frac{(m_h-1)!(N-m_h)!}{N!} = \sum_{j=1}^{N} \binom{N}{j} \frac{(j-1)!(N-j)!}{N!} = \sum_{j=1}^{N} \frac{1}{j} = H_N. \quad (38)$$

4.6 Some properties of this distribution

These different views of the distribution specified by Equation 34 make it straightforward to derive some of its properties. First, the effective dimension of the model, K_+, follows a Poisson(αH_N) distribution. This is most easily shown using the generative process described in Section 4.5: $K_+ = \sum_{h=1}^{2^N-1} K_h$, and under this process is thus the sum of a set of Poisson distributions. The sum of a set of Poisson distributions is a Poisson distribution with parameter equal to the sum of the parameters of its components. Using Equation 38, this is αH_N.

A second property of this distribution is that the number of features possessed by each object follows a Poisson(α) distribution. This follows from the definition of the exchangeable IBP. The first customer chooses a Poisson(α) number of dishes. By exchangeability, all other

[3] While $\mathbf{K}$ is technically a vector of non-negative integers, it is not a particularly generic example of such a vector as most of its entries will be 0 or 1.

customers must also choose a Poisson(α) number of dishes, since we can always specify an ordering on customers which begins with a particular customer.

Finally, it is possible to show that $\mathbf{Z}$ remains sparse as $K \to \infty$. The simplest way to do this is to exploit the previous result: if the number of features possessed by each object follows a Poisson(α) distribution, then the expected number of entries in $\mathbf{Z}$ is $N\alpha$. This is consistent with the quantity obtained by taking the limit of this expectation in the finite model, which is given in Equation 28: $\lim_{K\to\infty} E\left[\mathbf{1}^T\mathbf{Z}\mathbf{1}\right] = \lim_{K\to\infty} \frac{N\alpha}{1+\frac{\alpha}{K}} = N\alpha$. More generally, we can use the property of sums of Poisson random variables described above to show that $\mathbf{1}^T\mathbf{Z}\mathbf{1}$ will follow a Poisson($N\alpha$) distribution. Consequently, the probability of values higher than the mean decreases exponentially.

4.7 Inference by Gibbs sampling

We have defined a distribution over infinite binary matrices that satisfies one of our desiderata – objects (the rows of the matrix) are exchangeable under this distribution. It remains to be shown that inference in infinite latent feature models is tractable, as was the case for infinite mixture models. We will derive a Gibbs sampler for latent feature models in which the exchangeable IBP is used as a prior. The critical quantity needed to define the sampling algorithm is the full conditional distribution

$$P(z_{ik} = 1|\mathbf{Z}_{-(ik)}, \mathbf{X}) \propto p(\mathbf{X}|\mathbf{Z})P(z_{ik} = 1|\mathbf{Z}_{-(ik)}), \tag{39}$$

where $\mathbf{Z}_{-(ik)}$ denotes the entries of $\mathbf{Z}$ other than z_{ik}, and we are leaving aside the issue of the feature values $\mathbf{V}$ for the moment. The prior on $\mathbf{Z}$ contributes to this probability by specifying $P(z_{ik} = 1|\mathbf{Z}_{-(ik)})$.

In the finite model, where $P(\mathbf{Z})$ is given by Equation 27, it is straightforward to compute the full conditional distribution for any z_{ik}. Integrating over π_k gives

$$\begin{aligned} P(z_{ik} = 1|\mathbf{z}_{-i,k}) &= \int_0^1 P(z_{ik}|\pi_k)p(\pi_k|\mathbf{z}_{-i,k})\, d\pi_k \\ &= \frac{m_{-i,k} + \frac{\alpha}{K}}{N + \frac{\alpha}{K}}, \end{aligned} \tag{40}$$

where $\mathbf{z}_{-i,k}$ is the set of assignments of other objects, not including i, for feature k, and $m_{-i,k}$ is the number of objects possessing feature k, not including i. We need only condition on $\mathbf{z}_{-i,k}$ rather than $\mathbf{Z}_{-(ik)}$ because the columns of the matrix are generated independently under this prior.

In the infinite case, we can derive the conditional distribution from the exchangeable IBP. Choosing an ordering on objects such that the ith object corresponds to the last customer to visit the buffet, we obtain

$$P(z_{ik} = 1|\mathbf{z}_{-i,k}) = \frac{m_{-i,k}}{N}, \tag{41}$$

for any k such that $m_{-i,k} > 0$. The same result can be obtained by taking the limit of Equation 40 as $K \to \infty$. Similarly the number of new features associated with object i should be drawn from a Poisson($\frac{\alpha}{N}$) distribution. This can also be derived from Equation 40, using the same kind of limiting argument as that presented above to obtain the terms of the Poisson.

5 A latent feature model with binary features

We have derived a prior for infinite sparse binary matrices, and indicated how statistical inference can be done in models defined using this prior. In this section, we will show how this prior can be put to use in models for unsupervised learning, illustrating some of the issues

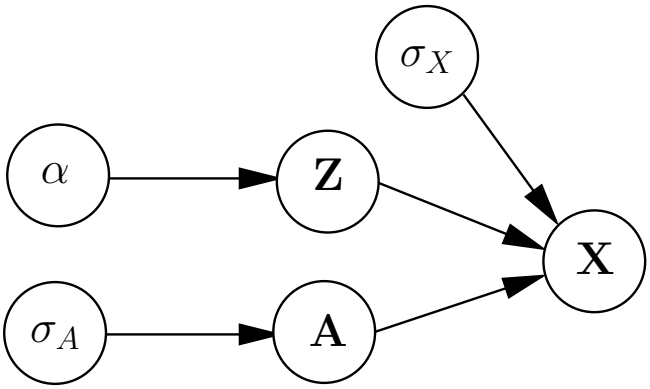

Figure 6: Graphical model for the linear-Gaussian model with binary features.

that can arise in this process. We will describe a simple linear-Gaussian latent feature model, in which the features are binary. As above, we will start with a finite model and then consider the infinite limit.

5.1 A finite linear-Gaussian model

In our finite model, the D-dimensional vector of properties of an object i, $\mathbf{x}_i$ is generated from a Gaussian distribution with mean $\mathbf{z}_i\mathbf{A}$ and covariance matrix $\mathbf{\Sigma}_X = \sigma_X^2\mathbf{I}$, where $\mathbf{z}_i$ is a K-dimensional binary vector, and $\mathbf{A}$ is a $K \times D$ matrix of weights. In matrix notation, $E[\mathbf{X}] = \mathbf{ZA}$. If $\mathbf{Z}$ is a feature matrix, this is a form of binary factor analysis. The distribution of $\mathbf{X}$ given $\mathbf{Z}$, $\mathbf{A}$, and σ_X is matrix Gaussian:

$$p(\mathbf{X}|\mathbf{Z}, \mathbf{A}, \sigma_X) = \frac{1}{(2\pi\sigma_X^2)^{ND/2}} \exp\{-\frac{1}{2\sigma_X^2}\text{tr}((\mathbf{X} - \mathbf{ZA})^T(\mathbf{X} - \mathbf{ZA}))\} \tag{42}$$

where $\text{tr}(\cdot)$ is the trace of a matrix. This makes it easy to integrate out the model parameters $\mathbf{A}$. To do so, we need to define a prior on $\mathbf{A}$, which we also take to be matrix Gaussian:

$$p(\mathbf{A}|\sigma_A) = \frac{1}{(2\pi\sigma_A^2)^{KD/2}} \exp\{-\frac{1}{2\sigma_A^2}\text{tr}(\mathbf{A}^T\mathbf{A})\}, \tag{43}$$

where σ_A is a parameter setting the diffuseness of the prior. The dependencies among the variables in this model are shown in Figure 6.

Combining Equations 42 and 43 results in an exponentiated expression involving the trace of

$$\begin{aligned}
&\frac{1}{\sigma_X^2}(\mathbf{X} - \mathbf{ZA})^T(\mathbf{X} - \mathbf{ZA}) + \frac{1}{\sigma_A^2}\mathbf{A}^T\mathbf{A} \\
&\qquad = \frac{1}{\sigma_X^2}\mathbf{X}^T\mathbf{X} - \frac{1}{\sigma_X^2}\mathbf{X}^T\mathbf{ZA} - \frac{1}{\sigma_X^2}\mathbf{A}^T\mathbf{Z}^T\mathbf{X} + \mathbf{A}^T(\frac{1}{\sigma_X^2}\mathbf{Z}^T\mathbf{Z} + \frac{1}{\sigma_A^2}\mathbf{I})\mathbf{A} &(44)\\
&\qquad = \frac{1}{\sigma_X^2}(\mathbf{X}^T(\mathbf{I} - \mathbf{ZMZ}^T)\mathbf{X}) + (\mathbf{MZ}^T\mathbf{X} - \mathbf{A})^T(\sigma_X^2\mathbf{M})^{-1}(\mathbf{MZ}^T\mathbf{X} - \mathbf{A}), &(45)
\end{aligned}$$

where $\mathbf{I}$ is the identity matrix, $\mathbf{M} = (\mathbf{Z}^T\mathbf{Z} + \frac{\sigma_X^2}{\sigma_A^2}\mathbf{I})^{-1}$, and the last line is obtained by completing

the square for the quadratic term in $\mathbf{A}$ in the second line. We can then integrate out $\mathbf{A}$ to obtain

$$
\begin{aligned}
& p(\mathbf{X}|\mathbf{Z}, \sigma_X, \sigma_A) \\
&= \int p(\mathbf{X}|\mathbf{Z}, \mathbf{A}, \sigma_X) p(\mathbf{A}|\sigma_A)\, d\mathbf{A} && (46) \\
&= \frac{1}{(2\pi)^{(N+K)D/2} \sigma_X^{ND} \sigma_A^{KD}} \exp\{-\frac{1}{2\sigma_X^2} \mathrm{tr}(\mathbf{X}^T(\mathbf{I} - \mathbf{Z}\mathbf{M}\mathbf{Z}^T)\mathbf{X})\} \\
&\qquad \int \exp\{-\frac{1}{2}\mathrm{tr}((\mathbf{M}\mathbf{Z}^T\mathbf{X} - \mathbf{A})^T(\sigma_X^2\mathbf{M})^{-1}(\mathbf{M}\mathbf{Z}^T\mathbf{X} - \mathbf{A}))\}\, d\mathbf{A} && (47) \\
&= \frac{|\sigma_X^2\mathbf{M}|^{D/2}}{(2\pi)^{ND/2} \sigma_X^{ND} \sigma_A^{KD}} \exp\{-\frac{1}{2\sigma_X^2} \mathrm{tr}(\mathbf{X}^T(\mathbf{I} - \mathbf{Z}\mathbf{M}\mathbf{Z}^T)\mathbf{X})\} && (48) \\
&= \frac{1}{(2\pi)^{ND/2} \sigma_X^{(N-K)D} \sigma_A^{KD} |\mathbf{Z}^T\mathbf{Z} + \frac{\sigma_X^2}{\sigma_A^2}\mathbf{I}|^{D/2}} \\
&\qquad \exp\{-\frac{1}{2\sigma_X^2} \mathrm{tr}(\mathbf{X}^T(\mathbf{I} - \mathbf{Z}(\mathbf{Z}^T\mathbf{Z} + \frac{\sigma_X^2}{\sigma_A^2}\mathbf{I})^{-1}\mathbf{Z}^T)\mathbf{X})\}. && (49)
\end{aligned}
$$

This result is intuitive: the exponentiated term is the difference between the inner product matrix of the raw values of $\mathbf{X}$ and their projections onto the space spanned by $\mathbf{Z}$, regularized to an extent determined by the ratio of the variance of the noise in $\mathbf{X}$ to the variance of the prior on $\mathbf{A}$.

We can use this derivation of $p(\mathbf{X}|\mathbf{Z}, \sigma_X, \sigma_A)$ to infer $\mathbf{Z}$ from a set of observations $\mathbf{X}$, provided we have a prior on $\mathbf{Z}$. The finite feature model discussed as a prelude to the IBP is such a prior. The full conditional distribution for z_{ik} is given by:

$$P(z_{ik}|\mathbf{X}, \mathbf{Z}_{-(i,k)}, \sigma_X, \sigma_A) \propto p(\mathbf{X}|\mathbf{Z}, \sigma_X, \sigma_A) P(z_{ik}|\mathbf{z}_{-i,k}). \tag{50}$$

While evaluating $p(\mathbf{X}|\mathbf{Z}, \sigma_X, \sigma_A)$ always involves matrix multiplication, it need not always involve a matrix inverse. $\mathbf{Z}^T\mathbf{Z}$ can be rewritten as $\sum_i \mathbf{z}_i^T\mathbf{z}_i$, allowing us to use rank one updates to efficiently compute the inverse when only one $\mathbf{z}_i$ is modified. Defining $\mathbf{M}_{-i} = (\sum_{j \neq i} \mathbf{z}_j^T\mathbf{z}_j + \frac{\sigma_X^2}{\sigma_A^2}\mathbf{I})^{-1}$, we have

$$
\begin{aligned}
\mathbf{M}_{-i} &= (\mathbf{M}^{-1} - \mathbf{z}_i^T\mathbf{z}_i)^{-1} && (51) \\
&= \mathbf{M} - \frac{\mathbf{M}\mathbf{z}_i^T\mathbf{z}_i\mathbf{M}}{\mathbf{z}_i\mathbf{M}\mathbf{z}_i^T - 1} && (52) \\
\mathbf{M} &= (\mathbf{M}_{-i}^{-1} + \mathbf{z}_i^T\mathbf{z}_i)^{-1} && (53) \\
&= \mathbf{M}_{-i} - \frac{\mathbf{M}_{-i}\mathbf{z}_i^T\mathbf{z}_i\mathbf{M}_{-i}}{\mathbf{z}_i\mathbf{M}_{-i}\mathbf{z}_i^T + 1}. && (54)
\end{aligned}
$$

Iteratively applying these updates allows $p(\mathbf{X}|\mathbf{Z}, \sigma_X, \sigma_A)$, to be computed via Equation 49 for different values of z_{ik} without requiring an excessive number of inverses, although a full rank update should be made occasionally to avoid accumulating numerical errors. The second part of Equation 50, $P(z_{ik}|\mathbf{z}_{-i,k})$, can be evaluated using Equation 40.

5.2 Taking the infinite limit

To make sure that we can define an infinite version of this model, we need to check that $p(\mathbf{X}|\mathbf{Z}, \sigma_X, \sigma_A)$ remains well-defined if $\mathbf{Z}$ has an unbounded number of columns. $\mathbf{Z}$ appears in two places in Equation 49: in $|\mathbf{Z}^T\mathbf{Z} + \frac{\sigma_X^2}{\sigma_A^2}\mathbf{I}|$ and in $\mathbf{Z}(\mathbf{Z}^T\mathbf{Z} + \frac{\sigma_X^2}{\sigma_A^2}\mathbf{I})^{-1}\mathbf{Z}^T$. We will examine how these behave as $K \to \infty$.

If $\mathbf{Z}$ is in left-ordered form, we can write it as $[\mathbf{Z}_+ \ \mathbf{Z}_0]$, where $\mathbf{Z}_+$ consists of K_+ columns with sums $m_k > 0$, and $\mathbf{Z}_0$ consists of K_0 columns with sums $m_k = 0$. It follows that the first of the two expressions we are concerned with reduces to

$$\left|\mathbf{Z}^T\mathbf{Z} + \frac{\sigma_X^2}{\sigma_A^2}\mathbf{I}\right| = \left|\begin{bmatrix} \mathbf{Z}_+^T\mathbf{Z}_+ & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} + \frac{\sigma_X^2}{\sigma_A^2}\mathbf{I}_K\right| \tag{55}$$

$$= \left(\frac{\sigma_X^2}{\sigma_A^2}\right)^{K_0} \left|\mathbf{Z}_+^T\mathbf{Z}_+ + \frac{\sigma_X^2}{\sigma_A^2}\mathbf{I}_{K_+}\right|. \tag{56}$$

The appearance of K_0 in this expression is not a problem, as we will see shortly. The abundance of zeros in $\mathbf{Z}$ leads to a direct reduction of the second expression to

$$\mathbf{Z}(\mathbf{Z}^T\mathbf{Z} + \frac{\sigma_X^2}{\sigma_A^2}\mathbf{I})^{-1}\mathbf{Z}^T = \mathbf{Z}_+(\mathbf{Z}_+^T\mathbf{Z}_+ + \frac{\sigma_X^2}{\sigma_A^2}\mathbf{I}_{K_+})^{-1}\mathbf{Z}_+^T, \tag{57}$$

which only uses the finite portion of $\mathbf{Z}$. Combining these results yields the likelihood for the infinite model

$$p(\mathbf{X}|\mathbf{Z}, \sigma_X, \sigma_A) = \frac{1}{(2\pi)^{ND/2}\sigma_X^{(N-K_+)D}\sigma_A^{K_+D}|\mathbf{Z}_+^T\mathbf{Z}_+ + \frac{\sigma_X^2}{\sigma_A^2}\mathbf{I}_{K_+}|^{D/2}}$$
$$\exp\{-\frac{1}{2\sigma_X^2}\mathrm{tr}(\mathbf{X}^T(\mathbf{I} - \mathbf{Z}_+(\mathbf{Z}_+^T\mathbf{Z}_+ + \frac{\sigma_X^2}{\sigma_A^2}\mathbf{I}_{K_+})^{-1}\mathbf{Z}_+^T)\mathbf{X})\}. \tag{58}$$

The K_+ in the exponents of σ_A and σ_X appears as a result of introducing $D/2$ multiples of the factor of $\left(\frac{\sigma_X^2}{\sigma_A^2}\right)^{K_0}$ from Equation 56. The likelihood for the infinite model is thus just the likelihood for the finite model defined on the first K_+ columns of $\mathbf{Z}$.

The Gibbs sampler for this model is now straightforward. Assignments to classes for which $m_{-i,k} > 0$ are drawn in the same way as for the finite model, via Equation 50, using Equation 58 to obtain $p(\mathbf{X}|\mathbf{Z}, \sigma_X, \sigma_A)$ and Equation 41 for $P(z_{ik}|\mathbf{z}_{-i,k})$. As in the finite case, Equations 52 and 54 can be used to compute inverses efficiently. The distribution over the number of new features can be approximated by truncation, computing probabilities for a range of values of $K_1^{(i)}$ up to some reasonable upper bound. For each value, $p(\mathbf{X}|\mathbf{Z}, \sigma_X, \sigma_A)$ can be computed from Equation 58, and the prior on the number of new classes is Poisson($\frac{\alpha}{N}$).

5.3 Demonstration

We applied the Gibbs sampler for the infinite binary linear-Gaussian model to a simulated dataset, $\mathbf{X}$, consisting of 100 6×6 images. Each image, $\mathbf{x}_i$, was represented as a 36-dimensional vector of pixel intensity values. The images were generated from a representation with four latent features, corresponding to the image elements shown in Figure 7 (a). These image elements correspond to the rows of the matrix $\mathbf{A}$ in the model introduced in Section 5.1, specifying the pixel intensity values associated with each binary feature. The non-zero elements of $\mathbf{A}$ were set to 1.0, and are indicated with white pixels in the figure. A feature vector, $\mathbf{z}_i$, for each image was sampled from a distribution under which each feature was present with probability 0.5. Each image was then generated from a Gaussian distribution with mean $\mathbf{z}_i\mathbf{A}$ and covariance $\sigma_X\mathbf{I}$, where $\sigma_X = 0.5$. Some of these images are shown in Figure 7 (b), together with the feature vectors, $\mathbf{z}_i$, that were used to generate them.

The Gibbs sampler was initialized with $K_+ = 1$, choosing the feature assignments for the first column by setting $z_{i1} = 1$ with probability 0.5. σ_A, σ_X, and α were initially set to 1.0, and then sampled by adding Metropolis steps to the MCMC algorithm (see Gilks et al., 1996). Figure 7 shows trace plots for the first 1000 iterations of MCMC for the log joint probability of

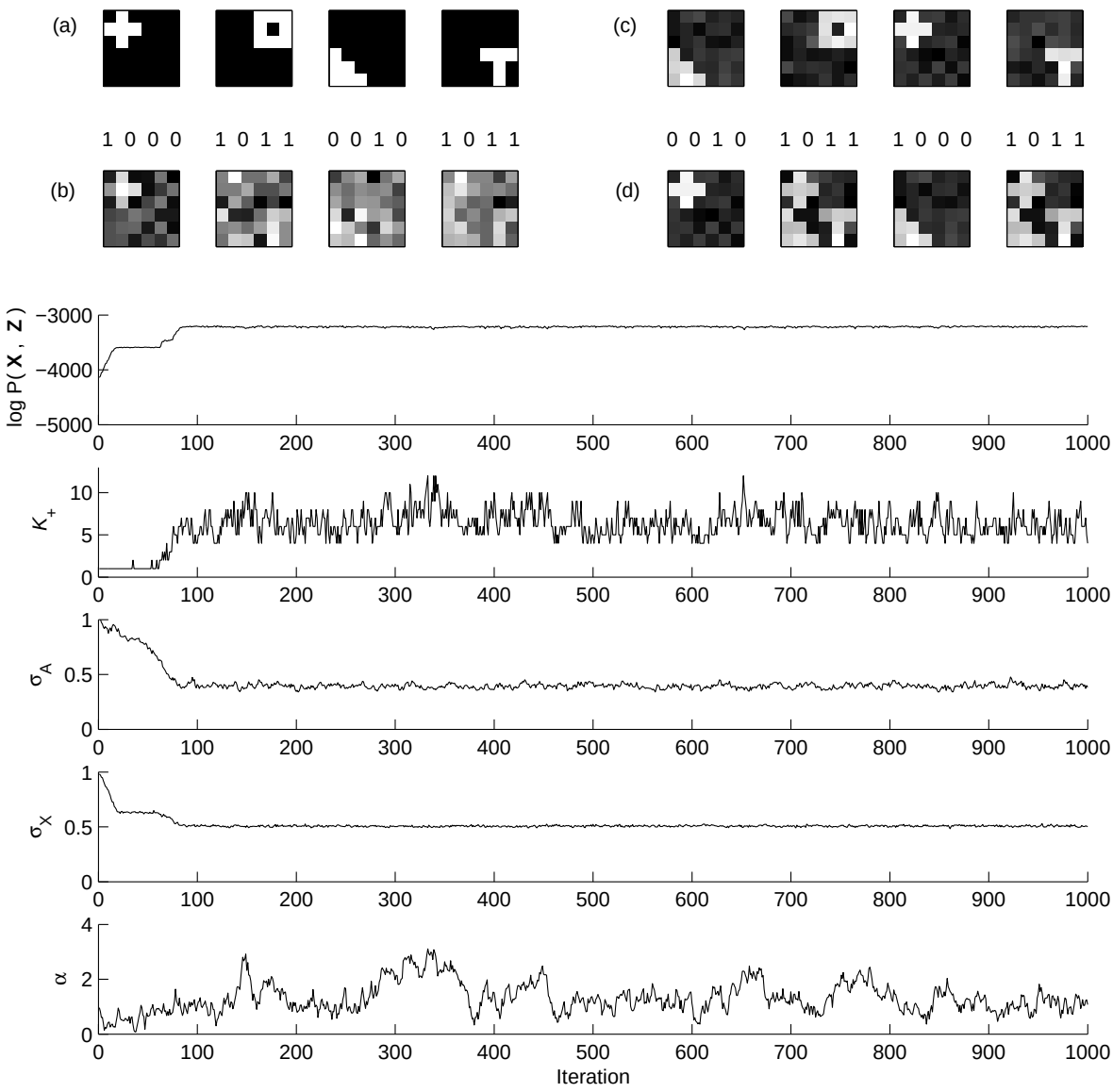

Figure 7: Stimuli and results for the demonstration of the infinite binary linear-Gaussian model. (a) Image elements corresponding to the four latent features used to generate the data. (b) Sample images from the dataset. (c) Image elements corresponding to the four features possessed by the most objects in the 1000th iteration of MCMC. (d) Reconstructions of the images in (b) using the output of the algorithm. The lower portion of the figure shows trace plots for the MCMC simulation, which are described in more detail in the text.

the data and the latent features, $\log p(\mathbf{X}, \mathbf{Z})$, the number of features used by at least one object, K_+, and the model parameters σ_A, σ_X, and α. The algorithm reached relatively stable values for all of these quantities after approximately 100 iterations, and our remaining analyses will use only samples taken from that point forward.

The latent feature representation discovered by the model was extremely consistent with that used to generate the data. Figure 8 (a) shows the distribution over K_+ computed from the samples. While the mode of the distribution is around six, samples tended to include four features that were used by a large number of objects, and then a few features used by only one or two objects. Figure 8 (b) shows the mean frequency with which objects tended to possess the different features, ordering features by these frequencies in each sample. The first four features averaged around 40 objects, while the remainder averaged less than five. Figure 8 (c) shows the distribution of the number of features possessed by each object. Most objects had one or two features, but no objects had more than six. The model thus tended to use a latent feature representation dominated by four features, consistent with the representation used to generate the data. Figure 8 (d) and (e) show the same quantities for the feature matrix that was actually used to generate the data, illustrating the close correspondence between the posterior distribution and the true representation.

The posterior mean of the feature weights, $\mathbf{A}$, given $\mathbf{X}$ and $\mathbf{Z}$ is

$$E[\mathbf{A}|\mathbf{X}, \mathbf{Z}] = (\mathbf{Z}^T\mathbf{Z} + \frac{\sigma_X^2}{\sigma_A^2}\mathbf{I})^{-1}\mathbf{Z}^T\mathbf{X}. \tag{59}$$

Figure 7 (c) shows the posterior mean of $\mathbf{a}_k$ for the four most frequent features in the 1000th sample produced by the algorithm, ordered to match the features shown in Figure 7 (a). These features pick out the image elements used in generating the data. Figure 7 (d) shows the feature vectors $\mathbf{z}_i$ from this sample for the four images in Figure 7(b), together with the posterior means of the reconstructions of these images for this sample, $E[\mathbf{z}_i\mathbf{A}|\mathbf{X}, \mathbf{Z}]$. Similar reconstructions are obtained by averaging over all values of $\mathbf{Z}$ produced by the Markov chain. The reconstructions provided by the model clearly pick out the relevant features, despite the high level of noise in the original images.

6 Conclusions and future work

We have shown that the methods that have been used to define infinite latent class models can be extended to models in which objects are represented in terms of a set of latent features, deriving a distribution on infinite binary matrices that can be used as a prior for such models. While we derived this prior as the infinite limit of a simple distribution on finite binary matrices, we have shown that the same distribution can be specified in terms of a simple stochastic process – the Indian buffet process. This distribution satisfies our two desiderata for a prior for infinite latent feature models: objects are exchangeable, and inference remains tractable.

There are a number of directions in which this work can be extended. First, while we have focussed on the distribution over the binary matrix $\mathbf{Z}$ indicating the features possessed by different objects, our intent is that this be combined with a prior over feature values $\mathbf{V}$ to define richer infinite latent feature models, as discussed in Section 3. We anticipate that MCMC algorithms similar to that described above in Section 4.7 can be applied in such models, and have developed such an algorithm for a simple model using discrete feature values – an infinite version of cooperative vector quantization (Zemel & Hinton, 1994). However, introducing feature values into the model raises some significant technical issues: in models where feature values have to be represented explicitly, and the structure of the model does not permit the use of conjugate priors, care has to be taken to ensure that posterior distributions remain proper and inference algorithms are well-defined. Similar issues arise in infinite mixture models, and are discussed by Neal (2000).

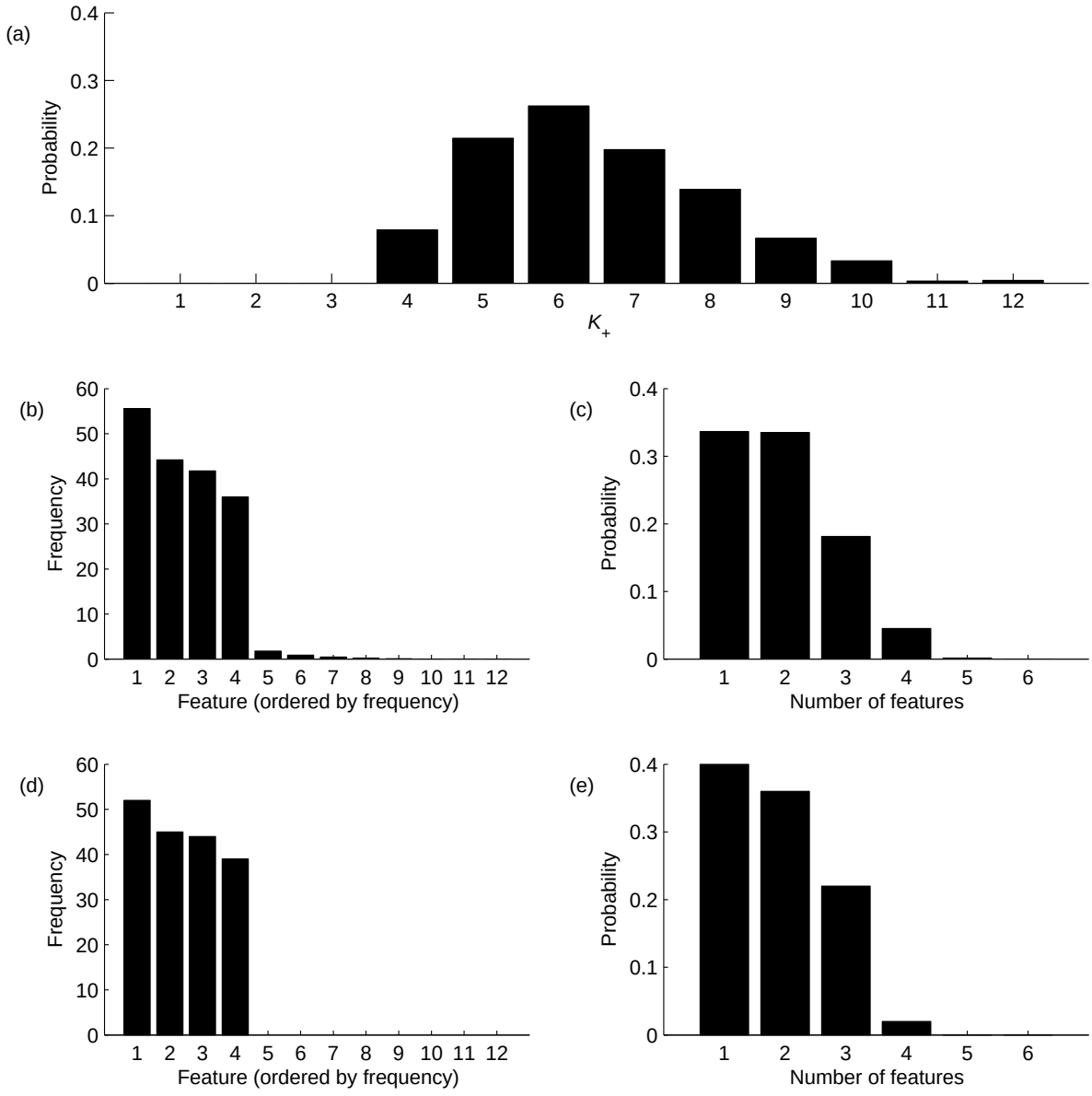

Figure 8: Statistics derived from the MCMC simulation, compared with the representation used to generate the data. (a) Posterior distribution over K_+, the number of features possessed by at least one object. (b) Mean frequencies with which objects were assigned to features, ordered from highest to lowest in each sample. (c) Distribution over number of features possessed by each object. (d)-(e) show the same statistics as (b)-(c), but computed from the representation that was actually used to generate the data.

A second direction in which this work can be extended is in considering other models in which such priors can be used. In particular, infinite latent feature models in which the relationship between data and features are non-linear may be useful in manifold-learning problems. There are also a number of applications beyond infinite latent feature models in which distributions on binary matrices with N rows and infinitely many columns are useful. For example, such matrices can be used to represent the relations that hold between two classes of entities, where one class contains a known number of entities, and the other class contains an unknown number. Cases like this arise in causal learning, where the dependencies among a fixed set of observable variables might be explained by the relationships between those variables and an unknown number of hidden causes. The distribution defined in this paper can be used as a prior over graph structures for causal learning problems of this kind.

Large-scale applications of these models will require developing more sophisticated inference algorithms. The Gibbs sampling algorithms discussed in this paper rely on local changes to class or feature assignments to move through the space of representations. These methods are slow to converge on large problems, and tend to get stuck in local maxima of the posterior distribution – while the sampler used in our demonstration in Section 5.3 stabilized rapidly, it only explored one of the modes of the posterior, never switching the order of the features in $\mathbf{Z}$. Inference in infinite mixture models can be improved by supplementing the local changes produced by the Gibbs sampler with a Metropolis-Hastings step that occasionally produces global changes (Dahl, 2003; Jain & Neal, 2004). Similar algorithms may be beneficial for inference in infinite latent feature models.

Finally, there is the question of whether the methods used in this paper can lead to other priors on infinite combinatorial structures. One obvious extension of the current work is to explore distributions on infinite binary matrices produced by making different assumptions about the generation of π_k, such as a two-parameter model in which π_k is generated from a Beta$(\frac{\alpha}{K}, \beta)$ distribution. However, there are a range of other possibilities. Our success in transferring the strategy of taking the limit of a finite model from latent classes to latent features suggests that the same strategy might be fruitfully applied with other representations, broadening the kinds of latent structure that can be recovered through unsupervised learning.

References

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

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

Bernardo, J. M., & Smith, A. F. M. (1994). *Bayesian theory.* New York: Wiley.

Blackwell, D., & MacQueen, J. (1973). Ferguson distributions via Polya urn schemes. *The Annals of Statistics*, *1*, 353-355.

Blei, D., Griffiths, T., Jordan, M., & Tenenbaum, J. (2004). Hierarchical topic models and the nested Chinese restaurant process. In *Advances in Neural Information Processing Systems 16.* Cambridge, MA: MIT Press.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet Allocation. *Journal of Machine Learning Research*, *3*, 993-1022.

Bush, C. A., & MacEachern, S. N. (1996). A semi-parametric Bayesian model for randomized block designs. *Biometrika*, *83*, 275-286.

Dahl, D. B. (2003). *An improved merge-split sampler for conjugate Dirichlet process mixture models* (Tech. Rep. No. 1086). Department of Statistics, University of Wisconsin.

d'Aspremont, A., Ghaoui, L. E., Jordan, I., & Lanckriet, G. R. G. (2004). *A direct formulation for sparse PCA using semidefinite programming* (Tech. Rep. No. UCB/CSD-04-1330). Computer Science Division, University of California, Berkeley.

Escobar, M. D., & West, M. (1995). Bayesian density estimation and inference using mixtures. *Journal of the American Statistical Association*, *90*, 577-588.

Ferguson, T. S. (1983). Bayesian density estimation by mixtures of normal distributions. In M. Rizvi, J. Rustagi, & D. Siegmund (Eds.), *Recent advances in statistics* (p. 287-302). New York: Academic Press.

Geman, S., & Geman, D. (1984). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *6*, 721-741.

Ghahramani, Z. (1995). Factorial learning and the EM algorithm. In *Advances in Neural Information Processing Systems 7.* San Francisco, CA: Morgan Kaufmann.

Gilks, W., Richardson, S., & Spiegelhalter, D. J. (Eds.). (1996). *Markov chain Monte Carlo in practice.* Suffolk: Chapman and Hall.

Green, P., & Richardson, S. (2001). Modelling heterogeneity with and without the Dirichlet process. *Scandinavian Journal of Statistics*, *28*, 355-377.

Ishwaran, H., & James, L. F. (2001). Gibbs sampling methods for stick-breaking priors. *Journal of the American Statistical Association*, *96*, 1316-1332.

Jain, S., & Neal, R. M. (2004). A split-merge Markov chain Monte Carlo procedure for the Dirichlet Process mixture model. *Journal of Computational and Graphical Statistics*, *13*, 158-182.

Jolliffe, I. T. (1986). *Principal component analysis.* New York: Springer.

Jolliffe, I. T., & Uddin, M. (2003). A modified principal component technique based on the lasso. *Journal of Computational and Graphical Statistics*, *12*, 531-547.

Neal, R. M. (1992). Bayesian mixture modeling. In *Maximum Entropy and Bayesian methods: Proceedings of the 11th International Workshop on Maximum Entropy and Bayesian Methods of Statistical Analysis* (p. 197-211). Dordrecht: Kluwer.

Neal, R. M. (2000). Markov chain sampling methods for Dirichlet process mixture models. *Journal of Computational and Graphical Statistics*, *9*, 249-265.

Pitman, J. (2002). *Combinatorial stochastic processes.* (Notes for Saint Flour Summer School)

Rasmussen, C. (2000). The infinite gaussian mixture model. In *Advances in Neural Information Processing Systems 12.* Cambridge, MA: MIT Press.

Rasmussen, C. E., & Ghahramani, Z. (2001). Occam's razor. In *Advances in Neural Information Processing Systems 13.* Cambridge, MA: MIT Press.

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

Teh, Y., Jordan, M., Beal, M., & Blei, D. (2004). Hierarchical Dirichlet processes. In *Advances in Neural Information Processing Systems 17.* Cambridge, MA: MIT Press.

Ueda, N., & Saito, K. (2003). Parametric mixture models for multi-labeled text. In *Advances in Neural Information Processing Systems 15.* Cambridge: MIT Press.

West, M., Muller, P., & Escobar, M. (1994). Hierarchical priors and mixture models, with application in regression and density estimation. In P. Freeman & A. Smith (Eds.), *Aspects of uncertainty* (p. 363-386). New York: Wiley.

Zemel, R. S., & Hinton, G. E. (1994). Developing population codes by minimizing description length. In *Advances in Neural Information Processing Systems 6.* San Francisco, CA: Morgan Kaufmann.

Zou, H., Hastie, T., & Tibshirani, R. (in press). Sparse principal component analysis. *Journal of Computational and Graphical Statistics*.

Appendix: Details of limits

This Appendix contains the details of the limits of three expressions that appear in Equations 15 and 34.

The first expression is

$$\frac{K!}{K_0!\,K^{K_+}} = \frac{\prod_{k=1}^{K_+}(K-k+1)}{K^{K_+}} \tag{60}$$

$$= \frac{K^{K_+} - \frac{(K_+-1)K_+}{2}K^{K_+-1} + \cdots + (-1)^{K_+-1}(K_+-1)!K}{K^{K_+}} \tag{61}$$

$$= 1 - \frac{(K_+-1)K_+}{2K} + \cdots + \frac{(-1)^{K_+-1}(K_+-1)!}{K^{K_+-1}}. \tag{62}$$

For finite K_+, all terms except the first go to zero as $K \to \infty$.

The second expression is

$$\prod_{j=1}^{m_k-1}(j+\tfrac{\alpha}{K}) = (m_k-1)! + \tfrac{\alpha}{K}\sum_{j=1}^{m_k-1}\frac{(m_k-1)!}{j} + \cdots + \left(\tfrac{\alpha}{K}\right)^{m_k-1}. \tag{63}$$

For finite m_k and α, all terms except the first go to zero as $K \to \infty$.

The third expression is

$$\left(\frac{N!}{\prod_{j=1}^{N}(j+\frac{\alpha}{K})}\right)^K = \left(\frac{\prod_{j=1}^{N} j}{\prod_{j=1}^{N}(j+\frac{\alpha}{K})}\right)^K \tag{64}$$

$$= \left(\prod_{j=1}^{N}\frac{j}{(j+\frac{\alpha}{K})}\right)^K \tag{65}$$

$$= \prod_{j=1}^{N}\left(\frac{1}{1+\frac{\alpha\frac{1}{j}}{K}}\right)^K. \tag{66}$$

We can now use the fact that

$$\lim_{K\to\infty}\left(\frac{1}{1+\frac{x}{K}}\right)^K = \exp\{-x\} \tag{67}$$

to compute the limit of Equation 66 as $K \to \infty$, obtaining

$$\lim_{K\to\infty}\prod_{j=1}^{N}\left(\frac{1}{1+\frac{\alpha\frac{1}{j}}{K}}\right)^K = \prod_{j=1}^{N}\exp\{-\alpha\tfrac{1}{j}\} \tag{68}$$

$$= \exp\{-\alpha\sum_{j=1}^{N}\tfrac{1}{j}\} \tag{69}$$

$$= \exp\{-\alpha H_N\}, \tag{70}$$

as desired.