*BAYESIAN STATISTICS 9,*
*J. M. Bernardo, M. J. Bayarri, J. O. Berger, A. P. Dawid,*
*D. Heckerman, A. F. M. Smith and M. West (Eds.)*


# Nonparametrics and Graphical Models: Discussion of Ickstadt et al.

Michael I. Jordan
*University of California, Berkeley, USA*
jordan@stat.berkeley.edu

Summary

Bayesian nonparametrics often has a strong combinatorial flavor, where inference is made over explicit "allocation" variables that associate data points to parameters. Taking this point of view I consider graphical models that are nonparametric in the sense of containing infinite numbers of nodes, and describe some specific allocation schemes for associating data points to nodes in such models.

*Keywords and Phrases:* Graphical Models; Bayesian Nonparametrics

## 1. INTRODUCTION

Many interesting issues arise when one considers Bayesian nonparametric variations on the graphical models theme, and I'm pleased that Ickstadt et al. have given us the opportunity to think through some of these issues.

While the great majority of graphical models deployed in practice are parametric, it is important to keep in mind that the general definition of graphical models has a strongly nonparametric flavor. In defining a graphical model on a collection of random variables $X = (X_1, X_2, \ldots, X_N)$ one begins with a graph, $G = (V, E)$, where the vertices $V = (V_1, V_2, \ldots, V_N)$ are in one-to-one correspondence with the random variables $X$, and where $E$ are the edges in the graph. A given graph expresses a set of conditional independence statements, via the pattern of missing edges in the graph. In particular, in the undirected graphical model formalism, a missing edge between nodes $i$ and $j$ asserts that $X_i$ and $X_j$ are independent conditional on the remaining random variables. This is a nonparametric modeling statement—it refers to *all* probability distributions that respect that conditional independence statement. In the general definition, to each graph $G$ we associate a *family* of probability distributions that respect all of the conditional independence assertions encoded by the edges $E$ in the graph. Such families are not generally captured via a finite set of parameters. For a thorough discussion of conditional independence and graphical models, see Lauritzen (1996).

Despite the nonparametric freedom inherent in this definition, practical applications of graphical models tend to collapse these nonparametric families to parametric sub-families. These sub-families are generally obtained by associating potential functions with the cliques of the graph, where each potential function is often of exponential family form. Indeed, most graphical models used in practice are either multivariate Gaussian or multinomial families.

While exponential graphical families provide a rich class of probabilistic models, given the large number of patterns of missing edges, inevitably the restriction to parametric models will seem limiting, and it is valuable to consider some of the ways to bring graphical models into the domain of Bayesian nonparametrics.

One way to do this is to consider countably infinite mixtures of graphical models. This is the approach considered by Ickstadt et al. Each individual mixture component (i.e., graph) can either have its own structure (i.e., pattern of missing edges) or the structure can be shared among the mixture components. The driving motivation for this approach is a familiar one in the mixture modeling community—for reasons of simplicity (interpretational or computational) it is desired to use simple distributions such as multivariate Gaussians, but the data appear to be multi-modal. This suggests mixtures, and in situations where there is significant uncertainty about the number of modes, it is natural to turn to nonparametric mixtures.

While I feel that this is a valuable contribution, I do have a concern about the use of graphical models as mixture components. In particular, the motivation for using graphical models is that their structure (i.e., the set of conditional independence relations) has a clean probabilistic interpretation. But this interpretation is lost when one takes mixtures. Indeed, even if all of component graphical models have the same structure (i.e., the same pattern of missing edges), it is not generally the case that the overall mixture model has *any* of the conditional independence relations expressed by the single underlying graph.

One can take two attitudes towards this fact. The first is that one may believe that a mixture of graphical models is a faithful expression of the generative process behind the data. In particular, in modeling interacting collections of proteins, one might imagine that there are a number of latent "states" that the biological system can be in, and given the state, the proteins have a particular pattern of interaction. In taking this point of view, one must presumably be prepared to do a significant amount of biological validation; in particular, one would like to give biological meaning to the underlying states. But in this context, it is not clear why one would want to consider an *infinite* number of underlying states.

The other possible attitude is to simply treat the mixture as a flexible formalism for fitting densities to data. From this point of view, the value of graphical models as components is not necessarily due to their clean conditional independence semantics, but rather because they provide a way to encode a sparse set of interactions among variables. In particular, in the directed Gaussian context that is the focus of Ickstadt et al., a graphical model is a set of sparse linear regressions. While this is a reasonable point of view, it is not clear why one should prefer graphical model inference procedures over other options for obtaining sparse linear regressions.

## 2. NONPARAMETRICS AND GRAPHICAL MODELS

In the remainder of this article, I wish to consider a broader perspective on the merger of nonparametrics and graphical models.

In doing so, it is useful to consider what one means by "nonparametrics." In the Bayesian setting, a rough definition is that one simply replaces the prior distribution

by a prior stochastic process. This definition is of limited value, however; indeed, a classical parametric graphical model can be viewed as a stochastic process where the nodes of the graph are the index set. More useful definitions can be based on notions such as "support" and "locality." In frequentist nonparametrics, the historical origins of the field was the notion that various statements should hold in a "distribution-free" sense; this is a statement about support. Further progress was made as researchers aimed to characterize rates of convergence; here constraints were imposed on the distributions, but the constraints were weak enough such that the resulting families were still "large." Additional perspective was obtained by defining notions of "locality," where a nonparametric estimator was defined in terms of a growing number of degrees of freedom (i.e., parameters), where each parameter has an influence on a shrinking fraction of the data points.

In Bayesian nonparametrics the notion of "locality" has become quite important, even if it not always explicitly acknowledged. Indeed, the workhorse of Bayesian nonparametrics is the countably infinite mixture model, which can be viewed in terms of a collection of "allocation" variables that explicitly associate each data point to a specific parameter in the model. Thus locality becomes an explicit object of inference in Bayesian nonparametrics. What is appealing about this approach is that despite the large collection of allocation variables there are combinatorial properties of the underlying stochastic processes that make it possible to derive efficient inference algorithms. Indeed, the field of Bayesian nonparametric has tended to focus on *combinatorial stochastic processes* where these properties are exploited systematically.

Thus, in considering nonparametric variations on the graphical model theme it is useful to consider the role of allocation variables. In the simplest case, there are no allocation variables, and data points are associated a priori with particular nodes in the graph. We wish to consider more flexible ways of mapping data to the nodes in a graph.

## 3. MODELS FOR PARTIALLY EXCHANGEABLE DATA

The allocation problem is brought into relief if we make an exchangeability assumption; indeed, in the exchangeable case there is nothing that allows us to wire in the association of a given data point to a particular parameter. Thus in this section we discuss models based on exchangeability, focusing for concreteness on models for document collections, where we assume that the words in a document are exchangeable (the "bag-of-words assumption"). Words are not exchangeable between documents, and thus we have an assumption of partial exchangeability.

The background for our discussion is a sequence of three historical steps in the modeling of document collections: (1) finite mixture models, (2) latent Dirichlet allocation (Blei, Ng, and Jordan, 2003), and (3) the hierarchical Dirichlet process (Teh, et al., 2006). All of these models are instances of discrete mixture models. Each of them associate each word in each document to one of a number of mixture components. In the document modeling literature, these mixture components are known as "topics," and they are defined by a vector parameter that lies in the simplex of discrete probability distributions on words. Thus these mixture models associate data points (words) to parameters (topics).

The finite mixture model makes such an association once per document; all words in a given document are assigned to the same topic. Latent Dirichlet allocation (LDA) allows the words in a given document to be assigned to different topics. In particular, LDA involves selecting a probability distribution $\theta$ across the topics on

a per-document basis. Each word in the document is generating by first selecting a topic based on $\theta$ and then generating a word from the selected topic. Finally, the hierarchical Dirichlet process (HDP) can be viewed as the nonparametric version of LDA which allows a countably infinite number of topics instead of the finite number of topics assumed by LDA. All of these models can be viewed as graphical models that include nodes that explicitly represent the allocation of words to topics.

There are two problems with these models. First, the topics estimated by these models tend to be heterogeneous and redundant. In particular, function words ("and," "the," "of," etc) appear with high probability across many topics; this is also true for other kinds of abstract words. Second, although these models essentially provide a clustering at the level of words by assigning words to topics, they do not provide a clustering at the level of documents, and the latter clustering is often desired in practice. One can use the per-document posterior distribution on topics obtained from LDA or the HDP as a "signature" for a document, and cluster the signatures via an adhoc clustering algorithm, but this algorithm has no interpretation in terms of the original model.

As we now show, both of these problems can be fixed by considering more complex (Bayesian nonparametric) graphical models. For simplicity, we will consider tree-structured graphical models. These models will be nonparametric in the sense that the underlying graphs will be infinite. In particular, we will use the directed graphical model formalism and the trees we will consider are rooted trees that have infinite depth and infinite branching factor. The question will be how to associate a finite data set to this infinite object.

### 3.1. *Hierarchical Latent Dirichlet Allocation*

*Hierarchical latent Dirichlet allocation* (hLDA) is a model for exchangeable data that aims to produce topics that are organized according to a notion of level of abstraction (Blei, Griffiths and Jordan, 2010). That is, the goal of hLDA to obtain some topics that are abstract and others that are concrete. The basic structure of the model is an infinite tree in which there is a topic associated with each node. (Recall that a topic is a parameter; it is a probability distribution across words.) The issue is how to map the words in a document to these topics.

The hLDA model assumes that each document is associated with a path down the infinite tree. The association of documents to paths is made via a probability model referred to as the *nested Chinese restaurant process* (nCRP). In the nCRP, there is a Chinese restaurant at each node of the infinite tree. A customer enters the restaurant at the root and sits at a table according to the classical Chinese restaurant process (i.e., selecting a table with probability proportional to the number of customers who have previously selected that table). The choice of table indicates which of the (infinite) collection of outgoing branches the customer follows. Following that branch, the customer arrives at a restaurant at the next level of the tree and the process recurses. The result is that a customer (which represents a document) is associated with a path down the tree. A collection of documents picks out a collection of paths down the tree.

Having associated a document with a path down the tree, the remaining problem is to associate the words in the document with nodes along the path. This is done using a stick-breaking process (cf. Sethuraman, 1994). Specifically, hLDA is based on the classical GEM distribution that underlies the Dirichlet process (but it is also possible to consider more general stick-breaking processes such as the Ongaro-Cattaneo distributions discussed by Ickstadt, et al.). A draw from the GEM

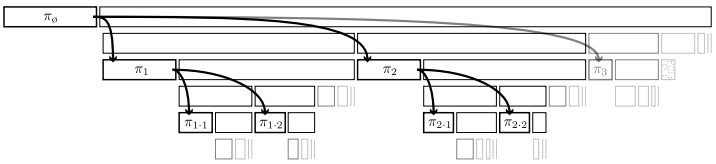


**Figure** 1: The TSSB is based on an interleaved pair of stick-breaking processes. In rows 1, 3 and 5, the first stick-breaking process assigns a fraction of the mass arriving at a node to the node itself (the boxes labeled with the $\pi$ values) and the remaining mass to the children. In rows 2, 4 and 6, the second stick-breaking process subdivides the remaining mass among the children.

distribution yields an infinite probability vector $\rho$. For each word in the document, we select a node along the designated path by treating $\rho$ as a distribution on levels in the tree. The vector $\rho$ is selected once per document and words are allocated to nodes in the tree by repeatedly drawing from $\rho$.

Experimental results show that hLDA model produces trees in which nodes near the root encode abstract topics and nodes far from the root encode concrete topics. The reason for this is that nodes nearer the root are used by larger subsets of documents (e.g., the root is used by all documents). Thus there is statistical pressure to force the topics at nodes near the root to concentrate on words that are useful across larger collections of documents; these tend to be abstract words.

### 3.2. *Tree-Structured Stick Breaking*

While in the hLDA model the words in a document are allocated along a path in a tree, the *tree-structured stick breaking* (TSSB) model (Adams, Ghahramani and Jordan, 2010) represents the opposite extreme in which all of the words in a given document are generated from a single node in the tree; moreover, multiple documents can reside at a single node. This model aims to solve the second of the problems discussed above—it provides a model for hierarchical clustering of collections of exchangeable data.

The TSSB model is based on a random process that assigns probability mass to each node in an infinitely-deep, infinitely-branching tree. Formally, if we let $\boldsymbol{\epsilon}$ index the nodes in the tree, and let $\pi_{\boldsymbol{\epsilon}}$ denote the random mass assigned to node $\boldsymbol{\epsilon}$, then the TSSB model defines a joint distribution on collections $\{\pi_{\boldsymbol{\epsilon}}\}$ that sum to one over the tree. As depicted in Fig. 3.2, this is achieved via an interleaved pair of stick-breaking processes that recursively allocate probability mass down the tree, beginning with the root. The first stick-breaking process uses beta random variables to decide how much of the mass arriving at a node should remain at the node and how much should be allocated to the children of the node. The second stick-breaking process subdivides the latter mass among the children.

Conditioning on $\pi$, the allocation procedure that assigns documents to nodes in the tree is straightforward; we simply place a document at node $\boldsymbol{\epsilon}$ with probability $\pi_{\boldsymbol{\epsilon}}$. As in the case of hLDA, this procedure has an interpretation as an urn model. In particular, as documents pass down the tree they either stop at a given node or continue descending; the decision to stop is made with probability proportional to one plus the number of previous documents that have previously arrived at the node and stopped there. If a document continues descending it chooses an outgoing

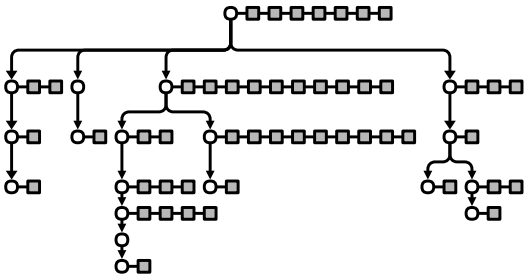

**Figure** 2: A random allocation of documents to nodes in a tree according to the TSSB model. The circles are represented nodes, and the squares are the documents.

branch according to the Chinese restaurant process.

We complete the TSSB model by placing a "topic" at each node in the tree. This can be done in a variety of ways; one natural choice is a "Dirichlet diffusion" in which we let $\theta_{\boldsymbol{\epsilon}} \sim \mathrm{Dir}(\kappa\theta_{\rho(\boldsymbol{\epsilon})})$, where $\rho(\boldsymbol{\epsilon})$ is the parent of node $\boldsymbol{\epsilon}$ and where $\kappa$ is a parameter. We now generate the words in a given document by choosing a node according to the urn model and then generating all words in the document from the topic found at that node.

## 4. DISCUSSION

We have focused on the role played by allocation variables in Bayesian nonparametrics, and we have exhibited ways in which these allocation variables can be used to define nonparametric graphical models containing infinite numbers of nodes.

The process of allocation can be viewed yet more broadly. In particular, the problem of *parsing* in natural language processing can be viewed as the problem of associating data points (words) to nodes in a tree. Conditional on the parse we obtain a graphical model, but obtaining the parse is itself a non-trivial inference problem (generally solved by dynamic programming). For further discussion of Bayesian nonparametrics and grammars, see Liang et al (2010).

## REFERENCES

Adams, R., Ghahramani, Z., and Jordan, M. I. (2010). Tree-structured stick breaking for hierarchical data. Technical Report arXiv:1006.1062.

Blei, D. M., Griffiths, T., and Jordan, M. I. (2010). The nested Chinese restaurant process and Bayesian inference of topic hierarchies. *Journal of the ACM*, *57*, 1–30.

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

Lauritzen, S. L. (1996). *Graphical Models*. Oxford: University Press.

Liang, P., Jordan, M. I., and Klein, D. (2010). Probabilistic grammars and hierarchical Dirichlet processes. In T. O'Hagan and M. West (Eds.), *The Handbook of Applied Bayesian Analysis*. Oxford: University Press.

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

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