Beta Diffusion Trees

Creighton Heaukulani[†] CKH28@CAM.AC.UK
David A. Knowles[⋆] DAVIDKNOWLES@CS.STANFORD.EDU
Zoubin Ghahramani[†] ZOUBIN@ENG.CAM.AC.UK
[†]University of Cambridge, Department of Engineering, Cambridge, UK
[⋆]Stanford University, Department of Computer Science, Stanford, CA, USA

Abstract

We define the *beta diffusion tree*, a random tree structure with a set of leaves that defines a collection of overlapping subsets of objects, known as a *feature allocation*. The generative process for the tree is defined in terms of particles (representing the objects) diffusing in some continuous space, analogously to the *Dirichlet* and *Pitman–Yor diffusion trees* (Neal, 2003b; Knowles & Ghahramani, 2011), both of which define tree structures over clusters of the particles. With the beta diffusion tree, however, multiple copies of a particle may exist and diffuse to multiple locations in the continuous space, resulting in (a random number of) possibly overlapping clusters of the objects. We demonstrate how to build a hierarchically-clustered factor analysis model with the beta diffusion tree and how to perform inference over the random tree structures with a Markov chain Monte Carlo algorithm. We conclude with several numerical experiments on missing data problems with data sets of gene expression arrays, international development statistics, and intranational socioeconomic measurements.

1. Introduction

Latent feature models assume that there are a set of non-overlapping subsets (called *features*) of a collection of objects underlying a data set. This is an appropriate assumption for a variety of statistical tasks, for example, in visual scene analyses, images could be assigned to the following features: “image contains a chair”, “image contains a table”, “image is of a kitchen”, etc. The *Indian buffet process* (IBP; Griffiths & Ghahramani (2011)) defines a prior on such clusterings, called *feature allocations*. With the IBP, objects are assigned or not assigned to a feature with a feature-specific probability that is independent of the other features. In the scene example, however, the features are structured into a hierarchy: tables and chairs are likely to appear in scenes together, and if the scene is in a kitchen, then possessing both tables and chairs are highly probable. In order to model hierarchically related feature allocations, we define the *beta diffusion tree* a random tree structure whose set of leaves define a feature allocation for a collection of objects. As with the IBP, the number of leaves (features) is random and unbounded, but will be almost surely finite for a finite set of objects.

Models for hierarchically structured partitions (non-overlapping subsets) of a collection of objects can be constructed by the *Dirichlet and Pitman–Yor diffusion trees* (Neal, 2003b; Knowles, 2012; Knowles & Ghahramani, 2014), in which a collection of particles (representing the objects) diffuse in some continuous space $\mathcal{X}$ (for example, as Brownian motion in Euclidean space) over some interval of time. Particles start at a fixed point and sequentially follow the paths of previous particles, potentially diverging from a path at random times. At the end of the time interval, the clusters of particles define a partition of the objects, and the paths taken by the particles define a tree structure over the partitions. The beta diffusion tree proceeds analogously to the Dirichlet diffusion tree, except that multiple copies of a particle (corresponding to multiple copies of an object) may be created (or removed) at random times. Therefore, at the end of the time interval, objects may correspond to particles in multiple clusters, and each cluster is interpreted as a feature.

The article is organized as follows: In Section 2, we describe a generative process for the beta diffusion tree and investigate its properties. In Section 3, we construct a hierarchically-clustered factor analysis model with the beta diffusion tree and review related work. In Section 4, we describe a Markov chain Monte Carlo procedure to integrate over the tree structures, which we apply in Section 5 to experiments on missing data problems.

Proceedings of the 31^{st} International Conference on Machine Learning, Beijing, China, 2014. JMLR: W&CP volume 32. Copyright 2014 by the author(s).

2. A generative process

We describe a collection of particles, each labelled with one of N objects, diffusing in a continuous space $\mathcal{X}$ over a hypothetical time interval $t \in [0, 1]$. If a particle is labeled with object n, then we call it an *n-particle*, multiple of which may exist at time $t > 0$. In this work, we take $\mathcal{X} = \mathbb{R}^D$ for some dimension D, and let the (random) diffusion paths be distributed as Brownian motion with variance $\sigma_X^2 \boldsymbol{I}_D$. In particular, if an n-particle is at position $\boldsymbol{x}(t)$ in $\mathbb{R}^D$ at time t, then it will reach position

$$\boldsymbol{x}(t + \mathrm{d}t) \mid \boldsymbol{x}(t) \sim \mathcal{N}(\boldsymbol{x}(t), \sigma_X^2 \boldsymbol{I}_D \mathrm{d}t), \qquad (1)$$

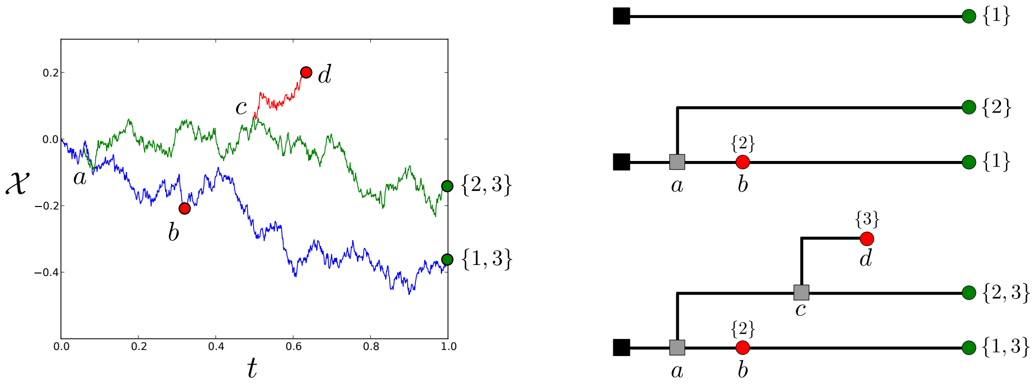
at time $t + \mathrm{d}t$. Sequentially for every data point $n = 1, \ldots, N$, we begin with one n-particle at the origin $\mathbf{0}$, which follows the paths of previous particles. At random times $t^\star$ throughout the process, an n-particle travelling a path may perform one of two actions:

1. **stop:** The particle stops diffusing at time $t^\star$.
2. **replicate:** A copy of the n-particle is created at time $t^\star$. One copy continues along the original path and the other diverges from the path and diffuses independently of all other particles.

More precisely, let λ_s, λ_r, θ_s, and θ_r be positive, finite constants that parameterize the generative process, which proceeds as follows:

- $\boldsymbol{n = 1}$: A 1-particle starts at the origin and diffuses as Brownian motion for $t > 0$.
  - The particle *stops* in the next infinitesimal time interval $[t, \mathrm{d}t]$ with probability
  $$\lambda_s \, \mathrm{d}t. \qquad (2)$$
  - The particle *replicates* in $[t, \mathrm{d}t]$ with probability
  $$\lambda_r \, \mathrm{d}t, \qquad (3)$$
  creating a copy of the 1-particle. Both particles diffuse (independently of each other) for $t > 1$, each stopping or replicating with the probabilities given by Eq. (2) and Eq. (3), respectively. Arbitrarily label one of the paths as the "original path" and the other as the "divergent path".
- $\boldsymbol{n \geq 2}$: For every $n \geq 2$, a single n-particle starts at the origin and follows the path initially taken by the previous particles. For a particle travelling on a path along which m particles have previously travelled:
  - The particle stops in $[t, \mathrm{d}t]$ with probability
  $$\frac{\theta_s}{\theta_s + m} \lambda_s \, \mathrm{d}t. \qquad (4)$$
  - The particle replicates in $[t, \mathrm{d}t]$ with probability
  $$\frac{\theta_r}{\theta_r + m} \lambda_r \, \mathrm{d}t, \qquad (5)$$
  creating a copy of the n-particle. One copy follows the original path, and the other copy diverges from the path and diffuses independently of all previous particles, stopping or replicating with the probabilities in Eq. (2) and Eq. (3), respectively. The newly created path is labeled as the "divergent path".
  - If the particle reaches an existing stop point (a point on the path where at least one previous particle has stopped), it also stops at this point with probability
  $$\frac{n_s}{\theta_s + m}, \qquad (6)$$
  where n_s is the number of particles that have previously stopped at this location.
  - If the particle reaches an existing replicate point (a point on the path where a particle has previously replicated), the particle also replicates at this point with probability
  $$\frac{n_r}{\theta_r + m}, \qquad (7)$$
  where n_r is the number of particles that have previously replicated at this point (and taken the divergent path). In this case, one copy of the particle follows the original path and the other follows the divergent path. If the particle does not replicate, then it continues along the original path.

The process terminates at $t = 1$, at which point all particles stop diffusing. The times until stopping or replicating on a path along which m particles have previously travelled are exponentially distributed with rates $\lambda_s \theta_s / (\theta_s + m)$ and $\lambda_r \theta_r / (\theta_r + m)$, respectively, and it is therefore straightforward to simulate a beta diffusion tree in practice. In Fig. 1, we show a beta diffusion tree with $N = 3$ objects in $D = 1$ dimension, along with its corresponding tree structure, in which the origin is the *root node*, stop points are *stop nodes*, replicate points are *replicate nodes*, and the points at $t = 1$ are *leaf nodes*. We call segments between nodes *branches*. Because multiple copies of a particle (all corresponding to the same object) can follow multiple branches to multiple leaves in the tree, the leaves define a feature allocation of the N objects. For example, adopting the notation of Broderick et al. (2013), the beta diffusion tree in Fig. 1 determines a feature allocation with two features $\{1, 3\}$ and $\{2, 3\}$. The number of (non-empty) features is therefore the number of leaves in the tree structure, which is unbounded, however, in Section 2.2 we will see that this number is (almost surely) finite for any finite number of objects.



(a) A beta diffusion tree with $N = 3$ objects

(b) The corresponding tree structure

Figure 1. (a) A beta diffusion tree with $N = 3$ objects, where b and d are stop points and a and c are replicate points, resulting in the feature allocation $\{\{1,3\},\{2,3\}\}$. (b) The corresponding tree structures as each object is included in the generative process, where the origin is the root node, the replicate and stop points are internal nodes, the features are leaf nodes, and segments between nodes are branches. Here, the stop nodes have been additionally annotated with the (labels of) the particles that have stopped at the node.

2.1. The generative process is exchangeable

Let $\mathcal{T}_{[N]}$ denote the *tree structure* (i.e., the collection of nodes, associated node times, and branches) of the beta diffusion tree with ordered set of objects $[N] := (1,\dots,N)$, and let $\boldsymbol{x}_{\mathcal{T}_{[N]}}$ denote the set of node locations in $\mathcal{X}$. While the generative process depends on the ordering of $[N]$, we will now show that the density $p(\mathcal{T}_{[N]}, \boldsymbol{x}_{\mathcal{T}_{[N]}})$ does not depend on this ordering. Because the times until stopping or replicating on a branch are exponentially distributed, it follows that the probability of neither replicating nor stopping between times t and t' (with $t < t'$) on a branch along which m previous particles have traversed is given by

$$\Psi_m(t,t') := \mathbb{P}\{\text{ not replicating and not stopping in } [t,t']\,\}$$
$$= \exp\Big\{-\frac{\theta_r}{\theta_r+m}\lambda_r\,(t'-t) - \frac{\theta_s}{\theta_s+m}\lambda_s\,(t'-t)\Big\}.$$

For example, consider the tree in Fig. 1, consisting of nodes a, b, c, d, $\{1,3\}$, and $\{2,3\}$, with corresponding node times t_a, t_b, etc. From the tree structure, we can determine that there is one 1-particle, two 2-particles, and two 3-particles. The 1-particle contributes a factor of $\Psi_0(0,1)$ to the density $p(\mathcal{T}_{[N]}, \boldsymbol{x}_{\mathcal{T}_{[N]}})$ for no "event" (i.e., for not stopping or replicating) in $t \in (0,1)$. The 2-particles contribute:

1. $\Psi_1(0,t_a)$ for no event in $t \in (0,t_a)$,
2. $\frac{\theta_r}{\theta_r+1}\lambda_r$ for replicating at $t = t_a$,
3. $\Psi_1(t_a,t_b)$ for no event in $t \in (t_a,t_b)$,
4. $\frac{\theta_s}{\theta_s+1}\lambda_s$ for stopping at $t = t_b$, and
5. $\Psi_0(t_a,1)$ for no event in $t \in (t_a,1)$.

The 3-particles contribute:

1. $\Psi_2(0,t_a)$ for no event in $t \in (0,t_a)$,
2. $\frac{1}{\theta_r+2}$ for taking the divergent path at $t = t_a$,
3. $\Psi_2(t_a,t_b)$ for no event in $t \in (t_a,t_b)$,
4. $1 - \frac{1}{\theta_s+2}$ for not stopping at $t = t_b$,
5. $\Psi_1(t_b,1)$ for no event in $t \in (t_b,1)$,
6. $\Psi_1(t_a,t_c)$ for no event in $t \in (t_a,t_c)$,
7. $\frac{\theta_r}{\theta_r+1}\lambda_r$ for replicating at $t = t_c$,
8. $\Psi_1(t_c,1)$ for no event in $t \in (t_c,1)$,
9. $\Psi_0(t_c,t_d)$ for no event in $t \in (t_c,t_d)$, and
10. λ_s for stopping at $t = t_d$.

Finally, the components of the density resulting from the node locations x_a, x_b, etc., are

$$\mathcal{N}(x_a;0,\sigma_X^2 t_a)\,\mathcal{N}(x_b;x_a,\sigma_X^2(t_b-t_a)) \qquad (8)$$
$$\times\,\mathcal{N}(x_c;x_a,\sigma_X^2(t_c-t_a))\,\mathcal{N}(x_d;x_c,\sigma_X^2(t_d-t_c))$$
$$\times\,\mathcal{N}(x_{\{1,3\}};x_b,\sigma_X^2(1-t_b))\,\mathcal{N}(x_{\{2,3\}};x_c,\sigma_X^2(1-t_c)).$$

There is one Gaussian term for each branch. Because the behavior of objects in the generative process depends on previous objects, the terms above depend on their ordering. However, this dependence is superficial; if we were to multiply these terms together, we would find that the resulting expression for $p(\mathcal{T}_{[3]}, \boldsymbol{x}_{\mathcal{T}_{[3]}})$, does not depend on the ordering of the objects. In Heaukulani et al. (2014), we generalize this expression to an arbitrary number of objects N and find that the density function is given by

$$p(\mathcal{T}_{[N]}, \boldsymbol{x}_{\mathcal{T}_{[N]}}) \qquad (9)$$
$$= \prod_{u\in\mathcal{R}(\mathcal{T}_{[N]})}\Big[\theta_r\lambda_r\,\frac{\Gamma(n_r(u))\,\Gamma(\theta_r+m(u)-n_r(u))}{\Gamma(\theta_r+m(u))}\Big]$$
$$\times \prod_{v\in\mathcal{S}(\mathcal{T}_{[N]})}\Big[\theta_s\lambda_s\,\frac{\Gamma(n_r(v))\,\Gamma(\theta_s+m(v)-n_r(v))}{\Gamma(\theta_s+m(v))}\Big]$$
$$\times \prod_{[uv]\in\mathcal{B}(\mathcal{T}_N)}\Big[\exp\Big\{(\lambda_r H^{\theta_r}_{m(v)-1}+\lambda_s H^{\theta_s}_{m(v)-1})(t_u-t_v)\Big\}$$
$$\times\,\mathcal{N}(\boldsymbol{x}_v;\boldsymbol{x}_u,\sigma_X^2(t_v-t_u)\boldsymbol{I}_D)\Big],$$

where $\mathcal{R}(\mathcal{T}_{[N]}), \mathcal{S}(\mathcal{T}_{[N]})$, and $\mathcal{B}(\mathcal{T}_{[N]})$ denote the sets of replicate nodes, stop nodes, and branches in $\mathcal{T}_{[N]}$, respectively; for every non-root node u in the tree structure, $m(u)$ denotes the number of particles that have traversed the branch ending at u; for every $u \in \mathcal{R}(\mathcal{T}_{[N]})$, $n_r(u)$ denotes the number of particles that have replicated at u; for every $v \in \mathcal{S}(\mathcal{T}_{[N]})$, $n_s(v)$ denotes the number of particles that have stopped at v; and finally $H_n^\alpha := \sum_{i=1}^n \alpha/(\alpha + i)$. Because this expression does not depend on the ordering of the objects, the stochastic process $(\mathcal{T}_{[N]}, \boldsymbol{x}_{\mathcal{T}_{[N]}})$ is exchangeable with respect to the ordering of the objects. Stated more formally:

Theorem 1 (Exchangeability). *Let $\sigma([N])$ be any permutation of $[N]$. Then*

$$(\mathcal{T}_{[N]}, \boldsymbol{x}_{\mathcal{T}_{[N]}}) \overset{d}{=} (\mathcal{T}_{\sigma([N])}, \boldsymbol{x}_{\mathcal{T}_{\sigma([N])}}). \tag{10}$$

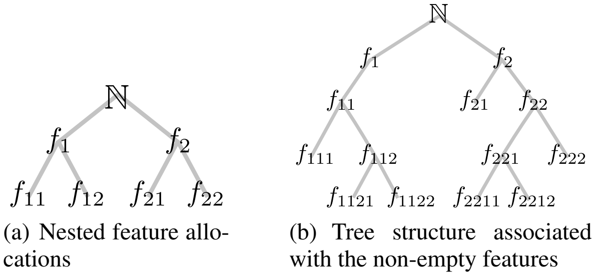
Because the ordering of the sequence $[N]$ is irrelevant, we will henceforth simply write $\mathcal{T}_N$. By its sequential construction, the generative process is projective, and we may therefore define a stochastic process by a beta diffusion tree with set of objects $\mathbb{N}$, the associated tree structure $\mathcal{T}_{\mathbb{N}}$ of which is a tree structure over feature allocations of $\mathbb{N}$.

2.2. A nested feature allocation scheme

Let there be L levels in a nested feature allocation scheme of N objects. Associate each level $\ell \leq L$ of the scheme with a discrete time $t_\ell = (\ell - 1)/L \in [0, 1]$, and let $p_1^{(\ell)}$ and $p_2^{(\ell)}$ be independent random variables with

$$\begin{aligned} p_1^{(\ell)} &\sim \text{beta}(\theta_s(1 - \lambda_s/L), \theta_s\lambda_s/L), \\ p_2^{(\ell)} &\sim \text{beta}(\theta_r\lambda_r/L, \theta_r(1 - \lambda_r/L)). \end{aligned} \tag{11}$$

At the first level, we allocate the N objects to two different features $f_1^{(1)}$ and $f_2^{(1)}$ independently with the level one-specific probabilities $p_1^{(1)}$ and $p_2^{(1)}$, respectively. At the next level, we allocate the objects in $f_1^{(1)}$ to two different features, $f_{11}^{(2)}$ and $f_{12}^{(2)}$ at level two, independently with probabilities $p_1^{(2)}$ and $p_2^{(2)}$, respectively. The objects in $f_2^{(1)}$ are likewise allocated to two features $f_{21}^{(2)}$ and $f_{22}^{(2)}$ at level two. A figure depicting this scheme for $L = 2$ levels is shown in Fig. 2(a). Continue this scheme recursively for L levels, where we allocate the objects in every (non-empty) feature at level $\ell - 1$ to two features in level ℓ, independently with the level ℓ-specific probabilities given by Eq. (11). Define a binary branching, discrete tree structure where every non-empty feature represents a node, as depicted in Fig. 2(b). Let segments between nodes be branches and let $\mathcal{T}_{\mathbb{N},L}$ denote the collection of nodes and branches. In Heaukulani et al. (2014), we show that in the continuum limit $L \to \infty$, we obtain the tree structure of the beta diffusion tree:



(a) Nested feature allocations

(b) Tree structure associated with the non-empty features

Figure 2. Depictions of a nested feature allocation scheme. In (a) we show the nested scheme for $L = 2$ levels, where the objects in each feature at level one are allocated to two features at level two. In (b) we show the tree structure corresponding to an $L = 4$ level deep scheme, where the nodes in the tree are non-empty features.

Theorem 2 (continuum limit). *Let $\mathcal{T}_{\mathbb{N}}$ be the tree structure of a beta diffusion tree with set of objects $\mathbb{N}$. Then*

$$\lim_{L\to\infty} \mathcal{T}_{\mathbb{N},L} \overset{d}{=} \mathcal{T}_{\mathbb{N}}. \tag{12}$$

From the perspective of the nested feature allocation scheme and Theorem 2, it is clear that the de Finetti measure for the beta diffusion tree with index set $\mathbb{N}$ is characterized by the countable collection

$$\mathcal{F} := \{(p_1^{(1)}, p_2^{(2)}), (p_1^{(1)}, p_2^{(2)}), \dots\}, \tag{13}$$

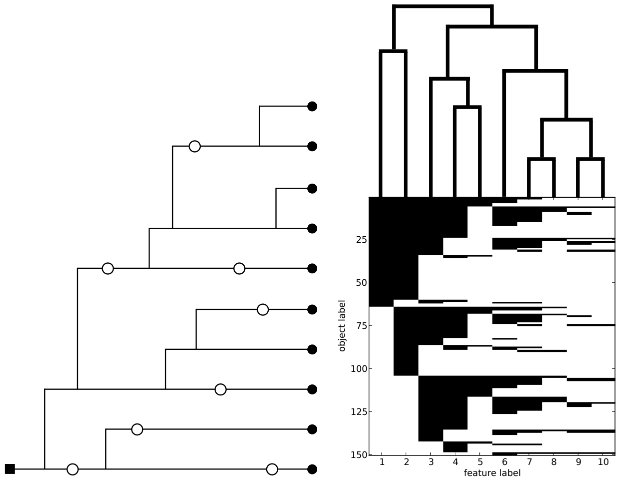
of (tuples of) beta random variables, motivating our name for the stochastic process. In Heaukulani et al. (2014), we use this identification of $\mathcal{F}$ to provide yet another characterization of the beta diffusion tree as a *multitype continuous-time Markov branching process* (Mode, 1971; Athreya & Vidyashankar, 2001; Harris, 2002). Taking advantage of these well-studied stochastic processes, we show:

Theorem 3. *Let $\mathcal{T}_N$ be the tree structure of a beta diffusion tree with a finite set of N objects. If $\lambda_s, \lambda_r, \theta_s, \theta_r < \infty$, then the number of leaves in $\mathcal{T}_N$ is almost surely finite.*

This is a reassuring property for any stochastic process employed as a non-parametric latent variable model, the translation in this case being that the number of latent features will be (almost surely) finite for any finite data set. Furthermore, we also characterize the expected number of leaves in a beta diffusion tree.

2.3. Correlating features and related work

We show another beta diffusion tree with $N = 150$ objects in Fig. 3(a). Let K denote the number of leaf nodes (features), which we have seen is unbounded yet almost surely finite. In this larger example, it is convenient to represent the feature allocation as a binary matrix, which we will denote as $\boldsymbol{Z}$, where the n-th row $\boldsymbol{z}_n \in \{0, 1\}^K$ indicates the



(a) Tree structure (b) Latent feature matrix

Figure 3. A simulated beta diffusion tree with $N = 150$ objects and its corresponding representation as a binary feature matrix, whose columns are interpreted as the features. The hierarchical clustering defined by the tree structure is shown over the columns. For visualization, the rows have been sorted according to their binary representations.

features to which object n is allocated, i.e., $z_{nk} = 1$ indicates object n is allocated to feature k. Then each column of $\boldsymbol{Z}$ represents a feature, and the tree structure defines a hierarchical clustering of the columns, depicted in Fig. 3(b).

The *Indian buffet process* (IBP; (Griffiths & Ghahramani, 2006; Ghahramani et al., 2007)) was originally described in terms of such binary matrices with an unbounded number of *independent* columns. A class of *correlated IBP* models appeared in (Doshi-Velez & Ghahramani, 2009), which cluster the columns of an IBP-distributed matrix in order to induce (sparse) dependencies between the features. For example, let $\boldsymbol{Z}^{(1)}$ be an IBP-distributed matrix, and conditioned on $\boldsymbol{Z}^{(1)}$, let $\boldsymbol{Z}^{(2)}$ be another IBP-distributed matrix whose rows corresponds to the columns of $\boldsymbol{Z}^{(1)}$. This scheme is extended to an arbitrary number of iterations by the *cascading Indian buffet process* (Adams et al., 2010b), in which the rows in an IBP-distributed matrix $\boldsymbol{Z}^{(m)}$ at iteration m correspond to the columns in the IBP-distributed matrix $\boldsymbol{Z}^{(m-1)}$ at iteration $m - 1$. While the beta diffusion tree generalizes the "flat clustering" of the correlated IBP to a hierarchical clustering, it does not obtain the general network structure obtained with the cascading IBP.

These stochastic processes all model continuous tree structures, which are most useful when modeling continuous variables associated with the hierarchy. We will see examples using the beta diffusion tree in Section 3. Probabilistic models for non-parametric, *discrete* tree structures are also widespread (Blei et al., 2010; Rodriguez et al., 2008; Paisley et al., 2012; Adams et al., 2010a; Steinhardt & Ghahramani, 2012), which would be appropriate for modeling only discrete variables associated with the tree structure. Relevant examples of non-probabilistic models for non-parametric tree structures include (Heller & Ghahramani, 2005; Blundell et al., 2010).

Models based on the beta diffusion tree are not to be confused with the *phylogenetic Indian buffet process* (Miller et al., 2008), which hierarchically clusters the *rows* (objects) in an IBP-distributed matrix. Alternatively, the *distance-dependent IBP* (Gershman et al., 2011) assumes that there is an observed distance metric between objects. If two objects are close, they tend to share the same features. Both of these models, unlike models based on the beta diffusion tree and the correlated IBP, assume that the features themselves are *a priori* independent.

3. Application: Linear Gaussian models with hierarchical factor loadings

In applications, we typically associate the objects allocated to a feature with a set of feature-specific latent parameters. The objects can be observed data that depend on the latent parameters, or the objects can themselves be unobserved variables in the model. A convenient choice for a set of continuous-valued latent parameters associated with each feature (leaf node in the beta diffusion tree) are the locations of the leaf nodes in $\mathcal{X}$. Consider the following example: Let $\boldsymbol{Z}$ be the binary matrix representation of the feature allocation corresponding to a beta diffusion tree with K leaf nodes. Recall that the k-th column of $\boldsymbol{Z}$ corresponds to a leaf node in the tree with diffusion location $\boldsymbol{x}_k$ in $\mathcal{X} = \mathbb{R}^D$ at time $t = 1$ (c.f. Fig. 1). We model a collection of N data points $\boldsymbol{y}_1, \dots, \boldsymbol{y}_N$ in $\mathbb{R}^D$ by

$$\boldsymbol{y}_n = \boldsymbol{z}_n^T \boldsymbol{X} + \varepsilon_n, \qquad n \leq N, \tag{14}$$

where $\boldsymbol{X}$ is a $K \times D$ factor loading matrix whose k-th row is given by $\boldsymbol{x}_k$, and $\boldsymbol{\epsilon}_1, \dots, \boldsymbol{\epsilon}_n$ are i.i.d. Gaussian noise vectors with zero mean and covariance $\sigma_Y^2 \boldsymbol{I}_D$. Here σ_Y^2 is the noise variance and $\boldsymbol{I}_D$ denotes the $D \times D$ identity matrix. Let $\boldsymbol{Y}$ be the $N \times D$ matrix with its n-th row given by $\boldsymbol{y}_n$. Then $\boldsymbol{Y}$ is matrix Gaussian and we may write $\mathbb{E}[\boldsymbol{Y} \mid \mathcal{T}_N, \boldsymbol{X}] = \boldsymbol{Z}\boldsymbol{X}$. This is a type of factor analysis model that generalizes the linear Gaussian models utilized by Griffiths & Ghahramani (2011) and Doshi-Velez & Ghahramani (2009). In the former, the latent features (columns of $\boldsymbol{Z}$) are independent and in the latter, the features are correlated via a flat clustering. In both models, the factor loadings $\boldsymbol{x}_1, \dots, \boldsymbol{x}_K$ are mutually independent. With the beta diffusion tree, on the other hand, *both* the latent features and factor loadings are hierarchically related through the tree structure.

Because the particles in the beta diffusion tree diffuse as Brownian motion, we may analytically integrate out the specific paths that were taken, along with the locations of the internal (non-leaf) nodes in the tree structure. Furthermore, because $\boldsymbol{Y}$ and $\boldsymbol{X}$ are both Gaussian, we may follow the derivations by Griffiths & Ghahramani (2011) to analytically integrate out the factor loadings $\boldsymbol{X}$ from the model, giving the resulting likelihood function

$$p(\boldsymbol{Y} \mid \mathcal{T}_N) = \frac{1}{(2\pi)^{ND/2}\sigma_Y^{(N-K)D}\sigma_X^{DK}\big|\boldsymbol{Q}\big|^{D/2}} \exp\Big\{-\frac{1}{2\sigma_Y^2}\mathrm{tr}\Big[\boldsymbol{Y}^T\Big(\boldsymbol{I}_N - \boldsymbol{Z}\boldsymbol{Q}^{-1}\boldsymbol{V}\boldsymbol{Z}^T\Big)\boldsymbol{Y}\Big]\Big\}, \tag{15}$$

where $\boldsymbol{Q} := \boldsymbol{V}\boldsymbol{Z}^T\boldsymbol{Z} + \frac{\sigma_X^2}{\sigma_Y^2}\boldsymbol{I}_K$, and $\boldsymbol{V}$ is a $K \times K$ matrix with entries given by

$$\nu_{\ell,k} = \begin{cases} t_{a(\ell,k)}, & \text{if } \ell \neq k, \\ 1, & \text{if } \ell = k, \end{cases} \tag{16}$$

where $t_{a(\ell,k)}$ is the time of the most recent common ancestor node to leaf nodes ℓ and k in the tree, and we recall that σ_X^2 is the variance of the Brownian motion (c.f. Eq. (1)).

4. Inference

In Heaukulani et al. (2014), we describe a series of Markov Chain Monte Carlo steps to integrate over the random tree structures of the beta diffusion tree. These moves are summarized as the following proposals:

Resample subtrees: Randomly select a subtree rooted at a non-leaf node in the tree, and resample the paths of one or more particles down the subtree according to the prior.

Add and remove replicate and stop nodes: Randomly propose an internal (either replicate or stop) node in the tree structure to remove. If the removed node is a replicate node, then the entire subtree emerging from the divergent branch is removed. If the node is a stop node, then the particles that stopped at the node need to be resample down the remaining tree according to the prior. Conversely, propose adding replicate and stop nodes to branches in the tree.

Resample configurations at internal nodes: Randomly select an internal node in the tree and propose changing the decisions that particles take at the node (i.e., the decisions to either replicate at replicate nodes or stop at stop nodes).

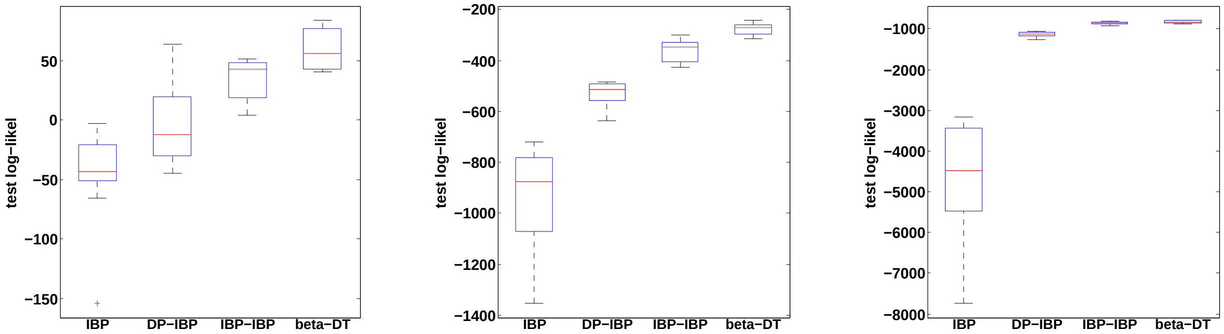
Heuristics to prune or thicken branches: Propose removing replicate (or stop) nodes at which a small proportion of the particles through the node have decided to replicate (or stop).

Each proposal is accepted or rejected with a Metropolis–Hastings step. In Heaukulani et al. (2014), we provide the results on joint distribution tests (Geweke, 2004) ensuring that the first three moves sample from the correct posterior distributions. The fourth move is a heuristic that does not leave the steady state distribution of the Markov chain invariant, though we found it critical for efficient mixing and good performance of the procedure. All hyperparameters were given broad prior distributions and integrated out with slice sampling (Neal, 2003a).

5. Numerical comparisons on test data

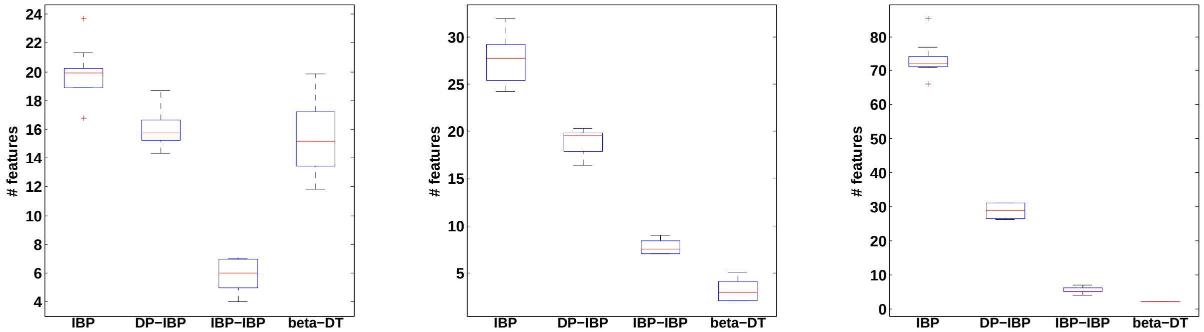
We implement our MCMC procedure on the linear Gaussian model and evaluate the log-likelihood of the inferred model given test sets of held-out data on an ***E. Coli*** dataset of the expression levels of $N = 100$ genes measured at $D = 24$ time points (Kao et al., 2004), a **UN** dataset of human development statistics for $N = 161$ countries on $D = 15$ variables (UN Development Programme, 2013), and an **India** dataset of socioeconomic measurements for $N = 400$ Indian households on $D = 15$ variables (Desai & Vanneman, 2013). We compare this performance against baselines modeling $\boldsymbol{Z}$ with the two parameter Indian buffet process (IBP; Ghahramani et al. (2007)) and two correlated latent feature models introduced by Doshi-Velez & Ghahramani (2009). All three baselines model the factor loadings (independently from the factors) as mutually independent Gaussian vectors $\boldsymbol{x}_k \sim \mathcal{N}(\boldsymbol{0}, \sigma_X^2\boldsymbol{I}_D)$, $k = 1, \ldots, K$, where K is the number of non-empty features. For each data set, we created 10 different test sets, each one holding out a different 10% of the data. In Fig. 4, we display the box-plots of the test log-likelihood scores over the 10 test sets, where the score for a single set is averaged over 3,000 samples (of the latent variables and parameters of the model) collected following the burn-in period of each method. The beta diffusion tree achieved the highest median score in every experiment, with the IBP-IBP achieving the second best performance in each instance. The difference between these two sets of scores is statistically significant in each case, based on a t-test at a 0.05 significance level. The p-values for the null hypothesis that the means of the two sets are the same were 5.3×10^{-3}, 1.5×10^{-4}, and 7.9×10^{-3} for the *E. Coli*, UN, and India data sets, respectively. In Fig. 5, we display box plots of the number of features inferred for each test set (averaged over the 3,000 samples following the burn-in). The superior performance on the test log-likelihood metric therefore suggests that a hierarchical feature allocation is an appropriate model for these data sets.

We can extend the qualitative analysis by Doshi-Velez & Ghahramani (2009) on the UN development statistics. Here we display the maximum *a posteriori* probability sample (among 2,000 samples collected after a burn-in period on the data set with no missing entries) of the feature matrix



(a) *E. Coli* dataset (b) UN dataset (c) India dataset

Figure 4. Box plots of the test log-likelihoods for the four different models on three different data sets. See the text for descriptions of the data sets and methods. The beta diffusion tree achieves the best performance in each case.



(a) *E. Coli* dataset (b) UN dataset (c) India dataset

Figure 5. Box plots of the average number of inferred features in the numerical experiments.

and tree structure over the features. For visualization, the rows (corresponding to different countries) are sorted from highest human development index (HDI – a score computed by the UN) to lowest. We also display the HDI scores for five ranges of equal sizes, along with the names of the top and bottom 10 countries in each range. We can see that a hierarchical structure is present; many highly developed countries are assigned to the third feature, with a more refined set belonging to the fourth feature. An even finer subset belongs to the fifth feature. On the other hand, the less developed countries have high prevalence in the second feature, with a broader set belonging to the first. This subset is not strict; many countries belonging to the second feature do not belong to the first. We have also displayed the posterior mean of the factor loading matrix. The third feature places higher weight on the variables we expect to be positively correlated with the highly developed countries, for example, GDP per capita, the number of broadband subscribers, and life expectancy. On the other hand, these features place lower weight on the variables we expect to be negatively correlated with the highly developed countries, notably, the rates for homicide and infant mortality. The first and second features are the reverse.

Similarly, in Fig. 7 we display the maximum *a posteriori* probability feature matrix and corresponding hierarchy over the features for the *E. Coli* data set when no data is held out. We note that, in this figure, the features are not necessarily ordered with the divergent branches to the right like in the previous figures in the document. In this case, the individual genes are not as interpretable as the countries in the UN data set, however, the hierarchical structure is reflected in the feature allocation matrix.

6. Conclusion

The beta diffusion tree is an expressive new model class of tree-structured feature allocations of $\mathbb{N}$, where the number of features are unbounded. The superior performance of this model class in our experiments, compared to independent or flatly-clustered features, provides evidence that hierarchically-structured feature allocations are appropriate for a wide range of statistical applications, and that the beta diffusion tree can successfully capture this structure.

There are many future directions to be explored. The features in the beta diffusion tree are not exchangeable (at a

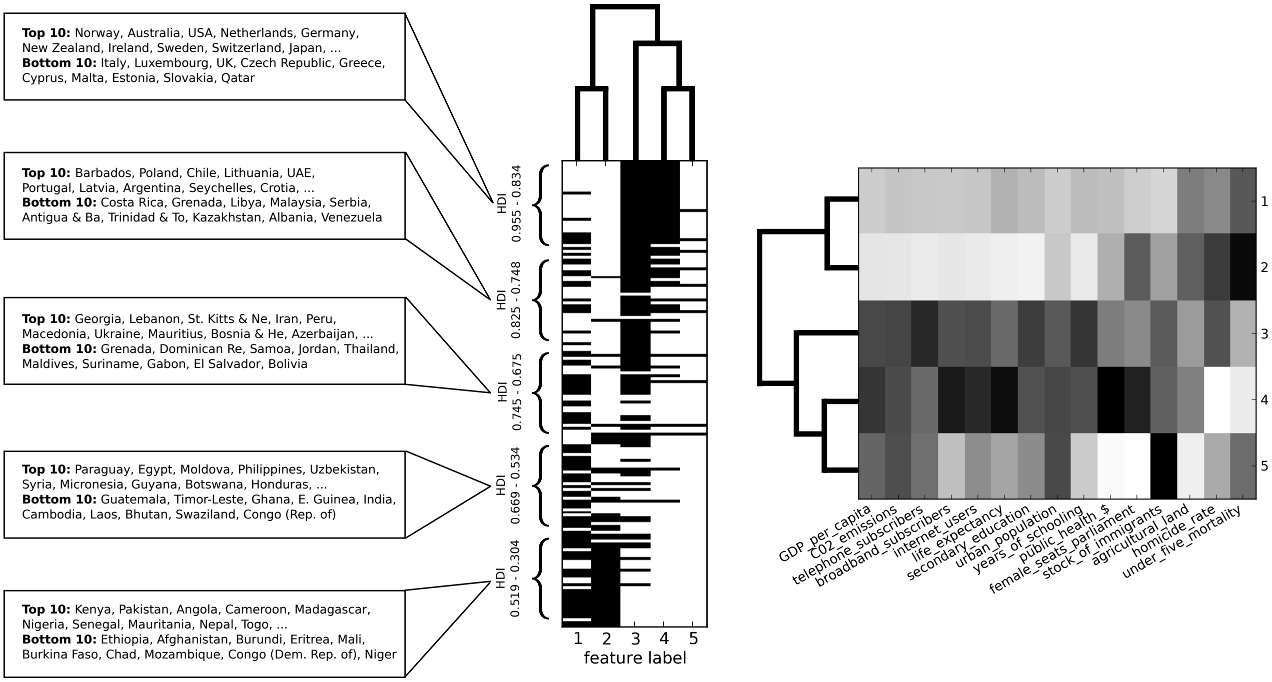


Figure 6. Left: Inferred factor matrix and corresponding hierarchy over the features for the UN data set. Black entries correspond to a one, and white entries correspond to a zero. The rows correspond to countries, which are ranked by their Human Development Indices (HDI). The names of the top and bottom 10 countries in five different ranges of the HDIs are displayed. Right: The posterior mean of the factor loading matrix, along with the hierarchy over the features. Darker values correspond to larger values.

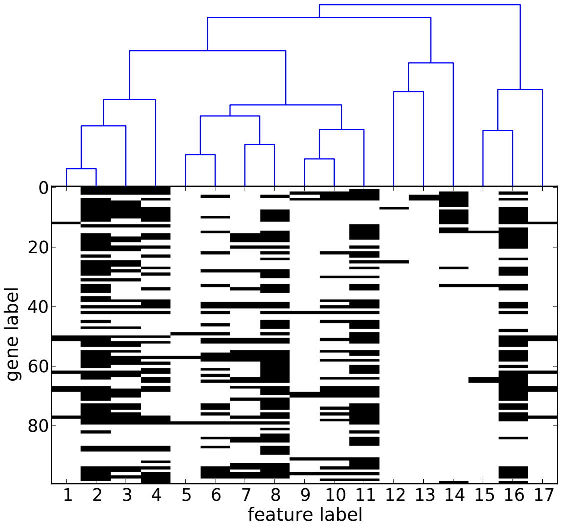


Figure 7. Inferred factor matrix and feature hierarchy for the *E. Coli* data set. Note that, unlike previously in the document, the features here are not necessarily ordered with the divergent branches to the right.

replicate point, particles are guaranteed to follow an original branch but not necessarily a divergent branch). This is reflected by the fact that the feature allocation probabilities $p_1^{(\ell)}$ and $p_2^{(\ell)}$ in the nested feature allocation scheme are not exchangeable. In contrast, the exchangeability of the feature allocation probabilities obtained from the beta process (Hjort, 1990) implies the exchangeability of the features in the IBP (Thibaux & Jordan, 2007). One could investigate if there is a variant or generalization of the beta diffusion tree in which the features are exchangeable. This could be a desirable modeling assumption in some applications and may enable the development of new inference procedures.

Teh et al. (2011); Elliott & Teh (2012) showed that the Dirichlet diffusion tree (Neal, 2003b) is the fragementation-coagulation dual (Pitman, 2006; Bertoin, 2006) of the Kingman coalescent (Kingman, 1982; Teh et al., 2007). The stochastic process introduced therein can be viewed as combining the dual processes in order to model a time-evolving partition of objects. One could investigate if such a dual process exists for the beta diffusion tree or some variant thereof, from which a model for time evolving feature allocations could be obtained.

Acknowledgments

We thank Daniel M. Roy, Christian Steinrueken, and anonymous reviewers for feedback on drafts. This work was funded in part by EPSRC grant EP/I036575/1. CH is funded by the Stephen Thomas studentship at Queens' College, Cambridge.

References

Adams, R. P., Ghahramani, Z., and Jordan, M. I. Tree-structured stick breaking for hierarchical data. In *Proc. NIPS*, 2010a.

Adams, R. P., Wallach, H. M., and Ghahramani, Z. Learning the structure of deep sparse graphical models. In *Proc. AISTATS*, 2010b.

Athreya, K. B. and Vidyashankar, A. N. Branching processes. *Stochastic Processes: Theory and Methods*, 19:35–53, 2001.

Bertoin, J. *Random fragmentation and coagulation processes*. Cambridge University Press, 2006.

Blei, D. M., Griffiths, T. L., and Jordan, M. I. The nested Chinese restaurant process and Bayesian nonparametric inference of topic hierarchies. *Journal of the ACM*, 57(2):7, 2010.

Blundell, C., Heller, K. A., and Teh, Y. W. Bayesian rose trees. In *Proc. UAI*, 2010.

Broderick, T., Pitman, J., and Jordan, M. I. Feature allocations, probability functions, and paintboxes. *Bayesian Analysis*, 8(4): 801–836, 2013.

Desai, S. and Vanneman, R. India human development survey (ihds), 2005. Technical Report ICPSR22626-v8, National Council of Applied Economic Research, New Delhi, 2013. Report available at: http://ihds.umd.edu/index.html; data available at: https://www.icpsr.umich.edu/icpsrweb/DSDR/studies/22626.

Doshi-Velez, F. and Ghahramani, Z. Correlated non-parametric latent feature models. In *Proc. UAI*, 2009.

Elliott, L. T. and Teh, Y. W. Scalable imputation of genetic data with a discrete fragmentation-coagulation process. In *Proc. NIPS*, 2012.

Gershman, S. J., Frazier, P. I., and Blei, D. M. Distance dependent infinite latent feature models. *arXiv preprint arXiv:1110.5454*, 2011.

Geweke, J. Getting it right: Joint distribution tests of posterior simulators. *Journal of the American Statistical Association*, 99 (467):799–804, 2004.

Ghahramani, Z., Griffiths, T. L., and Sollich, P. Bayesian nonparametric latent feature models. *Bayesian Statistics*, 8:201–226, 2007. See also the discussion and rejoinder.

Griffiths, T. and Ghahramani, Z. The Indian buffet process: An introduction and review. *Journal of Machine Learning Research*, 12:1185–1224, 2011.

Griffiths, T. L. and Ghahramani, Z. Infinite latent feature models and the Indian buffet process. In *Proc. NIPS*, 2006.

Harris, T. E. *The theory of branching processes*. Courier Dover Publications, 2002.

Heaukulani, C., Knowles, D. A., and Ghahramani, Z. Beta diffusion trees. *Preprint*, 2014. Extended version of this paper. Contact the author(s).

Heller, K. A. and Ghahramani, Z. Bayesian hierarchical clustering. In *Proc. ICML*, 2005.

Hjort, N. L. Nonparametric Bayes estimators based on beta processes in models for life history data. *The Annals of Statistics*, 18(3):985–1500, 1990.

Kao, K. C., Yang, Y., Boscolo, R., Sabatti, C., Roychowdhury, V., and Liao, J. C. Transcriptome-based determination of multiple transcription regulator activities in escherichia coli by using network component analysis. *Proceedings of the National Academy of Sciences of the United States of America*, 101(2): 641–646, 2004.

Kingman, J. F. C. The coalescent. *Stochastic Processes and Their Applications*, 13(3):235–248, 1982.

Knowles, D. A. *Bayesian non-parametric models and inference for sparse and hierarchical latent structure*. PhD thesis, University of Cambridge, 2012.

Knowles, D. A. and Ghahramani, Z. Pitman-Yor diffusion trees. In *Proc. UAI*, 2011.

Knowles, D. A. and Ghahramani, Z. Pitman–Yor diffusion trees for Bayesian hierarchical clustering. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, XX(XX):To appear, 2014.

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

Mode, C. J. *Multitype branching processes: theory and applications*. Elsevier, 1971.

Neal, R. M. Slice sampling. *Annals of Statistics*, 31(3):705–741, 2003a.

Neal, R. M. Density modeling and clustering using Dirichlet diffusion trees. *Bayesian Statistics*, 7, 2003b.

Paisley, J., Wang, C., Blei, D. M., and Jordan, M. I. Nested hierarchical dirichlet processes. *arXiv preprint arXiv:1210.6738*, 2012.

Pitman, J. *Combinatorial stochastic processes*. Springer–Verlag, 2006.

Rodriguez, A., Dunson, D. B., and Gelfand, A. E. The nested Dirichlet process. *Journal of the American Statistical Association*, 103(483), 2008.

Steinhardt, J. and Ghahramani, Z. Flexible martingale priors for deep hierarchies. In *Proc. AISTATS*, 2012.

Teh, Y. W., III, H. Daumé, and Roy, D. M. Bayesian agglomerative clustering with coalescents. In *Proc. NIPS*, 2007.

Teh, Y. W., Blundell, C., and Elliott, L. T. Modelling genetic variations with fragmentation-coagulation processes. In *Proc. NIPS*, 2011.

Thibaux, R. and Jordan, M. I. Hierarchical beta processes and the Indian buffet process. In *Proc. AISTATS*, 2007.

UN Development Programme. Human development report 2013: The rise of the south: Human progress in a diverse world. Technical report, United Nations, 2013. Report available at: http://hdr.undp.org/en/2013-report; data available at: http://hdr.undp.org/en/data.