Bayesian Agglomerative Clustering with Coalescents

Yee Whye Teh
Gatsby Unit
University College London
ywteh@gatsby.ucl.ac.uk

Hal Daumé III
School of Computing
University of Utah
me@hal3.name

Daniel Roy
CSAIL
MIT
droy@mit.edu

Abstract

We introduce a new Bayesian model for hierarchical clustering based on a prior over trees called Kingman's coalescent. We develop novel greedy and sequential Monte Carlo inferences which operate in a bottom-up agglomerative fashion. We show experimentally the superiority of our algorithms over the state-of-the-art, and demonstrate our approach in document clustering and phylolinguistics.

1 Introduction

Hierarchically structured data abound across a wide variety of domains. It is thus not surprising that hierarchical clustering is a traditional mainstay of machine learning [1]. The dominant approach to hierarchical clustering is agglomerative: start with one cluster per datum, and greedily merge pairs until a single cluster remains. Such algorithms are efficient and easy to implement. Their primary limitations—a lack of predictive semantics and a coherent mechanism to deal with missing data— can be addressed by probabilistic models that handle partially observed data, quantify goodness-of-fit, predict on new data, and integrate within more complex models, all in a principled fashion.

Currently there are two main approaches to probabilistic models for hierarchical clustering. The first takes a direct Bayesian approach by defining a prior over trees followed by a distribution over data points conditioned on a tree [2, 3, 4, 5]. MCMC sampling is then used to obtain trees from their posterior distribution given observations. This approach has the advantages and disadvantages of most Bayesian models: averaging over sampled trees can improve predictive capabilities, give confidence estimates for conclusions drawn from the hierarchy, and share statistical strength across the model; but it is also computationally demanding and complex to implement. As a result such models have not found widespread use. [2] has the additional advantage that the distribution induced on the data points is exchangeable, so the model can be coherently extended to new data. The second approach uses a flat mixture model as the underlying probabilistic model and structures the posterior hierarchically [6, 7]. This approach uses an agglomerative procedure to find the tree giving the best posterior approximation, mirroring traditional agglomerative clustering techniques closely and giving efficient and easy to implement algorithms. However because the underlying model has no hierarchical structure, there is no sharing of information across the tree.

We propose a novel class of Bayesian hierarchical clustering models and associated inference algorithms combining the advantages of both probabilistic approaches above. 1) We define a prior and compute the posterior over trees, thus reaping the benefits of a fully Bayesian approach; 2) the distribution over data is hierarchically structured allowing for sharing of statistical strength; 3) we have efficient and easy to implement inference algorithms that construct trees agglomeratively; and 4) the induced distribution over data points is exchangeable. Our model is based on an exchangeable distribution over trees called Kingman's coalescent [8, 9]. Kingman's coalescent is a standard model from population genetics for the genealogy of a set of individuals. It is obtained by tracing the genealogy backwards in time, noting when lineages *coalesce* together. We review Kingman's coalescent in Section 2. Our own contribution is in using it as a prior over trees in a hierarchical clustering model (Section 3) and in developing novel inference procedures for this model (Section 4).
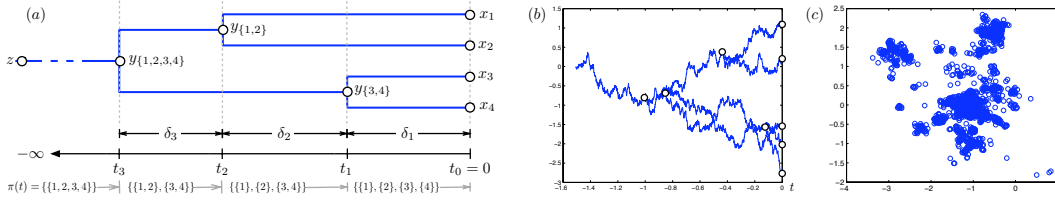
Figure 1: (a) Variables describing the n-coalescent. (b) Sample path from a Brownian diffusion coalescent process in 1D, circles are coalescent points. (c) Sample observed points from same in 2D, notice the hierarchically clustered nature of the points.

2 Kingman's coalescent

Kingman's coalescent is a standard model in population genetics describing the common genealogy (ancestral tree) of a set of individuals [8, 9]. In its full form it is a distribution over the genealogy of a countably infinite set of individuals. Like other nonparametric models (e.g. Gaussian and Dirichlet processes), Kingman's coalescent is most easily described and understood in terms of its finite dimensional marginal distributions over the genealogies of n individuals, called n-coalescents. We obtain Kingman's coalescent as $n \to \infty$.

Consider the genealogy of n individuals alive at the present time $t = 0$. We can trace their ancestry backwards in time to the distant past $t = -\infty$. Assume each individual has one parent (in genetics, *haploid* organisms), and therefore genealogies of $[n] = \{1, \dots, n\}$ form a *directed forest*. In general, time $t < 0$, there are m ($1 < m < n$) ancestors alive. Identify these ancestors with their corresponding sets $\rho_1, \dots, \rho_m$ of descendants (we will make this identification throughout the paper). Note that $\pi(t) = \{\rho_1, \dots, \rho_m\}$ form a *partition* of $[n]$, and interpret $t \mapsto \pi(t)$ as a function from $(-\infty, 0]$ to the set of partitions of $[n]$. This function is piecewise constant, left-continuous, monotonic ($s \leq t$ implies that $\pi(t)$ is a refinement of $\pi(s)$), and $\pi(0) = \{\{1\}, \dots, \{n\}\}$ (see Figure 1a). Further, π *completely and succinctly* characterizes the genealogy; we shall henceforth refer to π as *the genealogy* of $[n]$.

Kingman's n-coalescent is simply a distribution over genealogies of $[n]$, or equivalently, over the space of partition-valued functions like π. More specifically, the n-coalescent is a continuous-time, partition-valued, Markov process, which starts at $\{\{1\}, \dots, \{n\}\}$ at present time $t = 0$, and evolves *backwards in time*, merging (coalescing) lineages until only one is left. To describe the Markov process in its entirety, it is sufficient to describe the jump process (i.e. the embedded, discrete-time, Markov chain over partitions) and the distribution over coalescent times. Both are straightforward and their simplicity is part of the appeal of Kingman's coalescent. Let ρ_{li}, ρ_{ri} be the ith pair of lineages to coalesce, $t_{n-1} < \cdots < t_1 < t_0 = 0$ be the coalescent times and $\delta_i = t_{i-1} - t_i > 0$ be the duration between adjacent events (see Figure 1a). Under the n-coalescent, every pair of lineages merges independently with exponential rate 1. Thus the first pair amongst m lineages merge with rate $\binom{m}{2} = \frac{m(m-1)}{2}$. Therefore $\delta_i \sim \text{Exp}\left(\binom{n-i+1}{2}\right)$ independently, the pair ρ_{li}, ρ_{ri} is chosen from among those right after time t_i, and with probability one a random draw from the n-coalescent is a binary tree with a single root at $t = -\infty$ and the n individuals at time $t = 0$. The genealogy is:

$$\pi(t) = \begin{cases} \{\{1\}, \dots, \{n\}\} & \text{if } t = 0; \\ \pi_{t_{i-1}} - \rho_{li} - \rho_{ri} + (\rho_{li} \cup \rho_{ri}) & \text{if } t = t_i; \\ \pi_{t_i} & \text{if } t_{i+1} < t < t_i. \end{cases} \quad (1)$$

Combining the properties of the durations and choices of lineages, the probability of π is simply:

$$p(\pi) = \prod_{i=1}^{n-1} \binom{n-i+1}{2} \exp\left(-\binom{n-i+1}{2}\delta_i\right) / \binom{n-i+1}{2} = \prod_{i=1}^{n-1} \exp\left(-\binom{n-i+1}{2}\delta_i\right) \quad (2)$$

The n-coalescent has some interesting statistical properties [8, 9]. The marginal distribution over tree topologies is uniform and independent of the coalescent times. Secondly, it is infinitely exchangeable: given a genealogy drawn from an n-coalescent, the genealogy of any m contemporary individuals alive at time $t \leq 0$ embedded within the genealogy is a draw from the m-coalescent. Thus, taking $n \to \infty$, there is a distribution over genealogies of a countably infinite population for which the marginal distribution of the genealogy of any n individuals gives the n-coalescent. Kingman called this *the coalescent*.

3 Hierarchical clustering with coalescents

We take a Bayesian approach to hierarchical clustering, placing a coalescent prior on the latent tree and modeling the observed data with a tree structured Markov process evolving *forward in time*. We will alter our terminology from genealogy to tree, from n individuals at present time to n observed data points, and from individuals on the genealogy to latent variables on the tree-structured distribution. Let $\mathbf{x} = \{x_1, \ldots, x_n\}$ be n observed data points at the leaves of a tree π drawn from the n-coalescent. π has $n-1$ coalescent points, the ith occuring when ρ_{li} and ρ_{ri} merge at time t_i to form $\rho_i = \rho_{li} \cup \rho_{ri}$. Let t_{li} and t_{ri} be the times at which ρ_{li} and ρ_{ri} are themselves formed.

We use a continuous-time Markov process to define the distribution over the n data points $\mathbf{x}$ given the tree π. The Markov process starts in the distant past, evolves forward in time, splits at each coalescent point, and evolves independently down both branches until we reach time 0, when n data points are observations of the process at the n leaves of the tree. The joint distribution described by this process respects the conditional independences implied by the structure of the directed tree π. Let y_{ρ_i} be a latent variable that takes on the value of the Markov process at ρ_i just before it splits Let $y_{\{i\}} = x_i$ at leaf i. See Figure 1a.

To complete the description of the likelihood model, let $q(z)$ be the initial distribution of the Markov process at time $t = -\infty$, and $k_{st}(x, y)$ be the transition probability from state x at time s to state y at time t. This Markov process need be neither stationary nor ergodic. Marginalizing over paths of the Markov process, the joint probability over the latent variables and the observations is:

$$p(\mathbf{x}, \mathbf{y}, z | \pi) = q(z) k_{-\infty\, t_{n-1}}(z, y_{\rho_{n-1}}) \prod_{i=1}^{n-1} k_{t_i t_{li}}(y_{\rho_i}, y_{\rho_{li}}) k_{t_i t_{ri}}(y_{\rho_i}, y_{\rho_{ri}}) \tag{3}$$

Notice that the marginal distributions for each observation $p(x_i | \pi)$ are identical and given by the Markov process at time 0. However the observations are not independent as they share the same sample path down the Markov process until it splits. In fact the amount of dependence between two observations is a function of the time at which the observations coalesce. A more recent coalescent time implies larger dependence. The overall distribution induced on the observations $p(\mathbf{x})$ inherits the infinite exchangeability of the n-coalescent. We consider in Section 4.3 a brownian diffusion (Figures 1(b,c)) and a simple independent sites mutation process on multinomial vectors.

4 Agglomerative sequential Monte Carlo and greedy inference

We develop two classes of efficient and easily implementable inference algorithms for our hierarchical clustering model based on sequential Monte Carlo (SMC) and greedy schemes respectively. In both classes, the latent variables are integrated out, and the trees are constructed in a bottom-up fashion. The full tree π can be expressed as a series of $n-1$ coalescent events, ordered backwards in time. The ith coalescent event involves the merging of the two subtrees with leaves ρ_{li} and ρ_{ri} and occurs at a time δ_i before the previous coalescent event. Let $\theta_i = \{\delta_j, \rho_{lj}, \rho_{rj}$ for $j \leq i\}$ denote the first i coalescent events. θ_{n-1} is equivalent to π and we shall use them interchangeably.

We assume that the form of the Markov process is such that the latent variables $\{y_{\rho_i}\}_{i=1}^{n-1}$ and z can be efficiently integrated out using an upward pass of belief propagation on the tree. Let $M_{\rho_i}(y)$ be the message passed from y_{ρ_i} to its parent; $M_{\{i\}}(y) = \delta_{x_i}(y)$ is point mass at x_i for leaf i. $M_{\rho_i}(y)$ is proportional to the likelihood of the observations at the leaves below coalescent event i, given that $y_{\rho_i} = y$. Belief propagation computes the messages recursively up the tree; for $i = 1, \ldots, n-1$:

$$M_{\rho_i}(y) = Z_{\rho_i}^{-1}(\mathbf{x}, \theta_i) \prod_{b=l,r} \int k_{t_i t_{bi}}(y, y_b) M_{\rho_{bi}}(y_b)\, dy_b \tag{4}$$

where $Z_{\rho_i}(\mathbf{x}, \theta_i)$ is a normalization constant. The choice of Z does not affect the computed probability of $\mathbf{x}$, but does impact the accuracy and efficiency of our inference algorithms. We found that $Z_{\rho_i}(\mathbf{x}, \theta_i) = \iint q(z) k_{-\infty t_i}(z, y) M_{\rho_i}(y)\, dy\, dz$ worked well. At the root, we have:

$$Z_{-\infty}(\mathbf{x}, \theta_{n-1}) = \iint q(z) k_{-\infty\, t_{n-1}}(z, y) M_{\rho_{n-1}}(y)\, dy\, dz \tag{5}$$

The marginal probability $p(\mathbf{x} | \pi)$ is now given by the product of normalization constants:

$$p(\mathbf{x} | \pi) = Z_{-\infty}(\mathbf{x}, \theta_{n-1}) \prod_{i=1}^{n-1} Z_{\rho_i}(\mathbf{x}, \theta_i) \tag{6}$$

Multiplying in the prior (2) over π, we get the joint probability for the tree π and observations $\mathbf{x}$:

$$p(\mathbf{x}, \pi) = Z_{-\infty}(\mathbf{x}, \theta_{n-1}) \prod_{i=1}^{n-1} \exp\left(-\binom{n-i+1}{2}\delta_i\right) Z_{\rho_i}(\mathbf{x}, \theta_i) \tag{7}$$

Our inference algorithms are based upon (7). The sequential Monte Carlo (SMC) algorithms approximate the posterior over the tree θ_{n-1} using a weighted sum of samples, while the greedy algorithms construct θ_{n-1} by maximizing local terms in (7). Both proceeds by iterating over $i = 1, ..., n-1$, choosing a duration δ_i and a pair of subtrees ρ_{li}, ρ_{ri} to coalesce at each iteration. This choice is based upon the ith term in (7), interpreted as the product of a local prior $\exp\left(-\binom{n-i+1}{2}\delta_i\right)$ and a local likelihood $Z_{\rho_i}(\mathbf{x}, \theta_i)$ for choosing δ_i, ρ_{li} and ρ_{ri} given θ_{i-1}.

4.1 Sequential Monte Carlo algorithms

SMC algorithms approximate the posterior by iteratively constructing a weighted sum of point masses. At iteration $i - 1$, particle s consists of $\theta_{i-1}^s = \{\delta_j^s, \rho_{lj}^s, \rho_{rj}^s \text{ for } j < i\}$, and has weight w_{i-1}^s. At iteration i, s is extended by sampling δ_i^s, ρ_{li}^s and ρ_{ri}^s from a proposal distribution $f_i(\delta_i^s, \rho_{li}^s, \rho_{ri}^s|\theta_{i-1}^s)$, and the weight is updated by:

$$w_i^s = w_{i-1}^s \exp\left(-\binom{n-i+1}{2}\delta_i^s\right) Z_{\rho_i}(\mathbf{x}, \theta_i^s)/f_i(\delta_i^s, \rho_{li}^s, \rho_{ri}^s|\theta_{i-1}^s) \tag{8}$$

After $n - 1$ iterations, we obtain a set of trees θ_{n-1}^s and weights w_{n-1}^s. The joint distribution is approximated by: $p(\pi, \mathbf{x}) \approx \sum_s w_{n-1}^s \delta_{\theta_{n-1}^s}(\pi)$, while the posterior is approximated with the weights normalized. An important aspect of SMC is resampling, which places more particles in high probability regions and prunes particles stuck in low probability regions. We resample as in Algorithm 5.1 of [10] when the effective sample size ratio as estimated in [11] falls below one half.

SMC-PriorPrior. The simplest proposal distribution is to sample δ_i^s, ρ_{li}^s and ρ_{ri}^s from the local prior. δ_i^s is drawn from an exponential with rate $\binom{n-i+1}{2}$ and ρ_{li}^s, ρ_{ri}^s are drawn uniformly from all available pairs. The weight updates (8) reduce to multiplying by $Z_{\rho_i}(\mathbf{x}, \theta_i^s)$. This approach is computationally very efficient, but performs badly with many objects due to the uniform draws over pairs. **SMC-PriorPost**. The second approach addresses the suboptimal choice of pairs to coalesce. We first draw δ_i^s from its local prior, then draw ρ_{li}^s, ρ_{ri}^s from the local posterior:

$$f_i(\rho_{li}^s, \rho_{ri}^s|\delta_i^s, \theta_{i-1}^s) \propto Z_{\rho_i}(\mathbf{x}, \theta_{i-1}^s, \delta_i^s, \rho_{li}^s, \rho_{ri}^s); \quad w_i^s = w_{i-1}^s \sum_{\rho_l', \rho_r'} Z_{\rho_i}(\mathbf{x}, \theta_{i-1}^s, \delta_i^s, \rho_l', \rho_r') \tag{9}$$

This approach is more computationally demanding since we need to evaluate the local likelihood of every pair. It also performs significantly better than SMC-PriorPrior. We have found that it works reasonably well for small data sets but fails in larger ones for which the local posterior for δ_i is highly peaked. **SMC-PostPost**. The third approach is to draw all of δ_i^s, ρ_{li}^s and ρ_{ri}^s from their posterior:

$$f_i(\delta_i^s, \rho_{li}^s, \rho_{ri}^s|\theta_{i-1}^s) \propto \exp\left(-\binom{n-i+1}{2}\delta_i^s\right) Z_{\rho_i}(\mathbf{x}, \theta_{i-1}^s, \delta_i^s, \rho_{li}^s, \rho_{ri}^s)$$
$$w_i^s = w_{i-1}^s \sum_{\rho_l', \rho_r'} \int \exp\left(-\binom{n-i+1}{2}\delta'\right) Z_{\rho_i}(\mathbf{x}, \theta_{i-1}^s, \delta', \rho_l', \rho_r') \, d\delta' \tag{10}$$

This approach requires the fewest particles, but is the most computationally expensive due to the integral for each pair. Fortunately, for the case of Brownian diffusion process described below, these integrals are tractable and related to generalized inverse Gaussian distributions.

4.2 Greedy algorithms

SMC algorithms are attractive because they can produce an arbitrarily accurate approximation to the full posterior as the number of samples grow. However in many applications a single good tree is often sufficient. We describe a few greedy algorithms to construct a good tree.

Greedy-MaxProb: the obvious greedy algorithm is to pick δ_i, ρ_{li} and ρ_{ri} maximizing the ith term in (7). We do so by computing the optimal δ_i for each pair of ρ_{li}, ρ_{ri}, and then picking the pair maximizing the ith term at its optimal δ_i. **Greedy-MinDuration**: pick the pair to coalesce whose optimal duration is minimum. Both algorithms require recomputing the optimal duration for each pair at each iteration, since the prior rate $\binom{n-i+1}{2}$ on the duration varies with the iteration i. The total computational cost is thus $O(n^3)$. We can avoid this by using the alternative view of the n-coalescent as a Markov process where each pair of lineages coalesces at rate 1. **Greedy-Rate1**: for each pair ρ_{li} and ρ_{ri} we determine the optimal δ_i, replacing the $\binom{n-i+1}{2}$ prior rate with 1. We coalesce the pair with most recent time (as in Greedy-MinDuration). This reduces the complexity to $O(n^2)$. We found that all three performed similarly, and use Greedy-Rate1 in our experiments as it is faster.

4.3 Examples

Brownian diffusion. Consider the case of continuous data evolving via Brownian diffusion. The transition kernel $k_{st}(y, \cdot)$ is a Gaussian centred at y with variance $(t - s)\Lambda$, where Λ is a symmetric positive definite covariance matrix. Because the joint distribution (3) over $\mathbf{x}$, $\mathbf{y}$ and z is Gaussian, we can express each message $M_{\rho_i}(y)$ as a Gaussian with mean $\widehat{y}_{\rho_i}$ and covariance Λv_{ρ_i}. The local likelihood is:

$$Z_{\rho_i}(\mathbf{x}, \theta_i) = |2\pi\widehat{\Lambda}_i|^{-\frac{1}{2}} \exp\left(-\tfrac{1}{2}\|\widehat{y}_{\rho li} - \widehat{y}_{\rho ri}\|^2_{\widehat{\Lambda}_i}\right); \qquad \widehat{\Lambda}_i = \Lambda(v_{\rho li} + v_{\rho ri} + t_{li} + t_{ri} - 2t_i) \quad (11)$$

where $\|x\|_\Psi = x^\top \Psi^{-1} x$ is the Mahanalobis norm. The optimal duration δ_i can also be solved for,

$$\delta_i = \tfrac{1}{4}\binom{n-i+1}{2}^{-1}\left(\sqrt{4\binom{n-i+1}{2}\|\widehat{y}_{\rho li} - \widehat{y}_{\rho ri}\|^2_\Lambda + D^2} - D\right) - \tfrac{1}{2}(v_{\rho li} + v_{\rho ri} + t_{li} + t_{ri} - 2t_{i-1}) \quad (12)$$

where D is the dimensionality. The message at the newly coalesced point has parameters:

$$v_{\rho_i} = \left((v_{\rho li} + t_{li} - t_i)^{-1} + (v_{\rho ri} + t_{ri} - t_i)^{-1}\right)^{-1}; \widehat{y}_{\rho_i} = \left(\frac{\widehat{y}_{\rho li}}{v_{\rho li} + t_{li} - t_i} + \frac{\widehat{y}_{\rho ri}}{v_{\rho ri} + t_{ri} - t_i}\right)v_{\rho_i} \quad (13)$$

Multinomial vectors. Consider a Markov process acting on multinomial vectors with each entry taking one of K values and evolving independently. Entry d evolves at rate λ_d and has equilibrium distribution vector q_d. The transition rate matrix is $Q_d = \lambda_d(q_h^\top \mathbf{1}_K - I_k)$ where $\mathbf{1}_K$ is a vector of K ones and I_K is identity matrix of size K, while the transition probability matrix for entry d in a time interval of length t is $e^{Q_d t} = e^{-\lambda_d t} I_K + (1 - e^{-\lambda_d t})q_d^\top \mathbf{1}_K$. Representing the message for entry d from ρ_i to its parent as a vector $M_{\rho_i}^d = [M_{\rho_i}^{d1}, \ldots, M_{\rho_i}^{dK}]^\top$, normalized so that $q_d \cdot M_{\rho_i}^d = 1$, the local likelihood terms and messages are computed as,

$$Z_{\rho_i}^d(\mathbf{x}, \theta_i) = 1 - e^{\lambda_h(2t_i - t_{li} - t_{ri})}\left(1 - \sum_{k=1}^K q_{dk}M_{\rho li}^{dk}M_{\rho ri}^{dk}\right) \quad (14)$$

$$M_{\rho_i}^d = (1 - e^{\lambda_d(t_i - t_{li})}(1 - M_{\rho li}^d))(1 - e^{\lambda_d(t_i - t_{ri})}(1 - M_{\rho ri}^d))/Z_{\rho_i}^d(\mathbf{x}, \theta_i) \quad (15)$$

Unfortunately the optimal δ_i cannot be solved analytically and we use Newton steps to compute it.

4.4 Hyperparameter estimation

We perform hyperparameter estimation by iterating between estimating a tree, and estimating the hyperparameters. In the Brownian case, we place an inverse Wishart prior on Λ and the MAP posterior $\hat{\Lambda}$ is available in a standard closed form. In the multinomial case, the updates are not available analytically and are solved iteratively. Further information on hyperparameter estimation, as well predictive densities and more experiments are available in a longer technical report.

5 Experiments

Synthetic Data Sets. In Figure 2 we compare the various SMC algorithms and Greedy-Rate1 on a range of synthetic data sets drawn from the Brownian diffusion coalescent process itself ($\Lambda = I_D$) to investigate the effects of various parameters on the efficacy of the algorithms[1]. Generally SMC-PostPost performed best, followed by SMC-PriorPost, SMC-PriorPrior and Greedy-Rate1. With increasing D the amount of data given to the algorithms increases and all algorithms do better, especially Greedy-Rate1. This is because the posterior becomes concentrated and the Greedy-Rate1 approximation corresponds well with the posterior. As n increases, the amount of data increases as well and all algorithms perform better. However, the posterior space also increases and SMC-PriorPrior which simply samples from the prior over genealogies does not improve as much. We see this effect as well when S is small. As S increases all SMC algorithms improve. Finally, the algorithms were surprisingly robust when there is mismatch between the generated data sets' λ and the λ used by the model. We expected all models to perform worse with SMC-PostPost best able to maintain its performance (though this is possibly due to our experimental setup).

MNIST and SPAMBASE. We compare the performance of Greedy-Rate1 to two other hierarchical clustering algorithms: average-linkage and Bayesian hierarchical clustering (BHC) [6]. In MNIST,

---

[1]Each panel was generated from independent runs. Data set variance affected all algorithms, varying overall performance across panels. However, trends in each panel are still valid, as they are based on the same data.
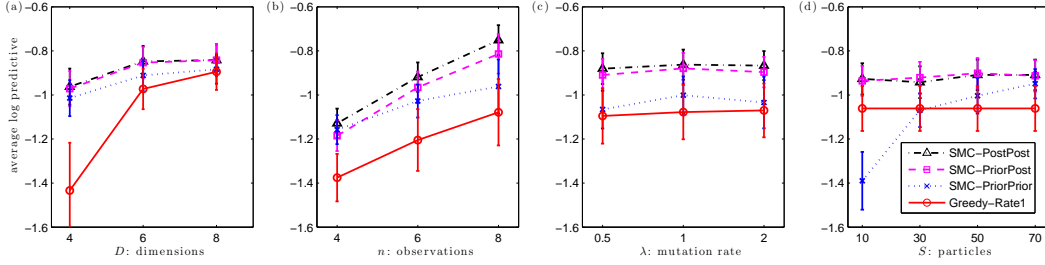
Figure 2: Predictive performance of algorithms as we vary (a) the numbers of dimensions D, (b) observations n, (c) the mutation rate λ ($\Lambda = \lambda I_D$), and (d) number of samples S. In each panel other parameters are fixed to their middle values (we used $S = 50$) in other panels, and we report log predictive probabilities on one unobserved entry, averaged over 100 runs.

	MNIST			SPAMBASE		
	Avg-link	**BHC**	**Coalescent**	**Avg-link**	**BHC**	**Coalescent**
Purity	$.363\pm.004$	$.392\pm.006$	$\mathbf{.412\pm.006}$	$.616\pm.007$	$\mathbf{.711\pm.010}$	$.689\pm.008$
Subtree	$.581\pm.005$	$.579\pm.005$	$\mathbf{.610\pm.005}$	$.607\pm.011$	$.549\pm.015$	$\mathbf{.661\pm.012}$
LOO-acc	$.755\pm.005$	$.763\pm.005$	$\mathbf{.773\pm.005}$	$.846\pm.010$	$.832\pm.010$	$\mathbf{.861\pm.008}$

Table 1: Comparative results. Numbers are averages and standard errors over 50 and 20 repeats.

we use 20 exemplars from each of 10 digits from the MNIST data set, reduced via PCA to 20 dimensions, repeating the experiment 50 times. In SPAMBASE, we use 100 examples of 57 binary attributes from each of 2 classes, repeating 20 times. We present purity scores [6], subtree scores ($\#\{$interior nodes with all leaves of same class$\}/(n - \#\text{classes})$) and leave-one-out accuracies (all scores between 0 and 1, higher better). The results are in Table 1; except for purity on SPAMBASE, ours gives the best performance. Experiments not presented here show that all greedy algorithms perform about the same and that performance improves with hyperparameter updates.

Phylolinguistics. We apply Greedy-Rate1 to a phylolinguistic problem: language evolution. Unlike previous research [12] which studies only phonological data, we use a full typological database of 139 binary features over 2150 languages: the *World Atlas of Language Structures* (WALS) [13]. The data is *sparse*: about 84% of the entries are unknown. We use the same version of the database as extracted by [14]. Based on the Indo-European subset of this data for which at most 30 features are unknown (48 languages total), we recover the coalescent tree shown in Figure 3(a). Each language is shown with its genus, allowing us to observe that it teases apart Germanic and Romance languages, but makes a few errors with respect to Iranian and Greek.

Next we compare predictive abilities to other algorithms. We take a subset of WALS and tested on 5% of withheld entries, restoring these with various techniques: Greedy-Rate1; nearest neighbors (use value from nearest observed neighbor); average-linkage (nearest neighbor in the tree); and probabilistic PCA (latent dimensions in $5, 10, 20, 40$, chosen optimistically). We use five subsets of the WALS database, obtained by sorting both the languages and features of the database according to sparsity and using a varying percentage ($10\% - 50\%$) of the densest portion. The results are in Figure 3(b). Our approach performed reasonably well.

Indo-European Data			
	Avg-link	**BHC**	**Coalescent**
Purity	0.510	0.491	**0.813**
Subtree	0.414	0.414	**0.690**
LOO-acc	0.538	0.590	**0.769**
Whole World Data			
	Avg-link	**BHC**	**Coalescent**
Purity	0.162	0.160	**0.269**
Subtree	0.227	0.099	**0.177**
LOO-acc	0.080	0.248	**0.369**

Table 2: Comparative performance of various algorithms on phylolinguistics data.

Finally, we compare the trees generated by Greedy-Rate1 with trees generated by either average-linkage or BHC, using the same evaluation criteria as for MNIST and SPAMBASE, using language genus as classes. The results are in Table 5, where we can see that the coalescent significantly outperforms the other methods.
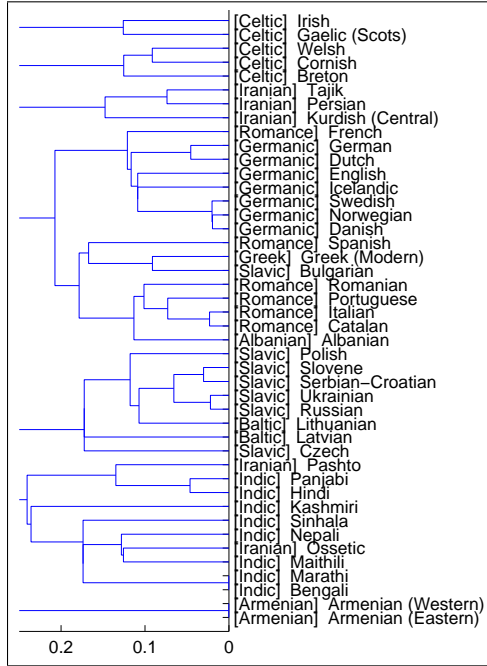
(a) Coalescent for a subset of Indo-European languages from WALS.
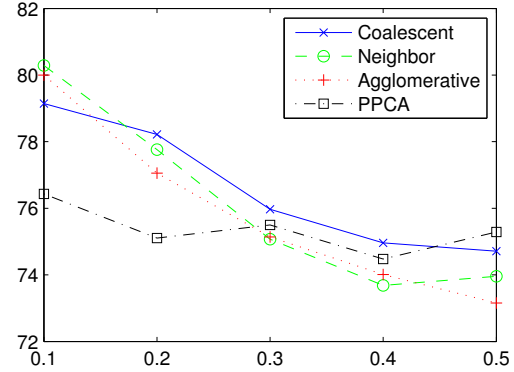
(b) Data restoration on WALS. Y-axis is accuracy; X-axis is percentage of data set used in experiments. At 10%, there are $N = 215$ languages, $H = 14$ features and $p = 94\%$ observed data; at 20%, $N = 430$, $H = 28$ and $p = 80\%$; at 30%: $N = 645$, $H = 42$ and $p = 66\%$; at 40%: $N = 860$, $H = 56$ and $p = 53\%$; at 50%: $N = 1075$, $H = 70$ and $p = 43\%$. Results are averaged over five folds with a different 5% hidden each time. (We also tried a "mode" prediction, but its performance is in the 60% range in all cases, and is not depicted.)

Figure 3: Results of the phylolinguistics experiments.

LLR	(t)	Top Words	Top Authors (# papers)
32.7	(-2.71)	bifurcation attractors hopfield network saddle	Mjolsness (9) Saad (9) Ruppin (8) Coolen (7)
0.106	(-3.77)	voltage model cells neurons neuron	Koch (30) Sejnowski (22) Bower (11) Dayan (10)
83.8	(-2.02)	chip circuit voltage vlsi transistor	Koch (12) Alspector (6) Lazzaro (6) Murray (6)
140.0	(-2.43)	spike ocular cells firing stimulus	Sejnowski (22) Koch (18) Bower (11) Dayan (10)
2.48	(-3.66)	data model learning algorithm training	Jordan (17) Hinton (16) Williams (14) Tresp (13)
31.3	(-2.76)	infomax image ica images kurtosis	Hinton (12) Sejnowski (10) Amari (7) Zemel (7)
31.6	(-2.83)	data training regression learning model	Jordan (16) Tresp (13) Smola (11) Moody (10)
39.5	(-2.46)	critic policy reinforcement agent controller	Singh (15) Barto (10) Sutton (8) Sanger (7)
23.0	(-3.03)	network training units hidden input	Mozer (14) Lippmann (11) Giles (10) Bengio (9)

Table 3: Nine clusters discovered in NIPS abstracts data.

NIPS. We applied Greedy-Rate1 to all NIPS abstracts through NIPS12 (1740, total). The data was preprocessed so that only words occuring in at least 100 abstracts were retained. The word counts were then converted to binary. We performed one iteration of hyperparameter re-estimation. In the supplemental material, we depict the top levels of the coalescent tree. Here, we use the tree to generate a flat clustering. To do so, we use the log likelihood ratio at each branch in the coalescent to determine if a split should occur. If the log likelihood ratio is greater than zero, we break the branch; otherwise, we recurse down. On the NIPS abstracts, this leads to nine clusters, depicted in Table 3. Note that clusters two and three are quite similar—had we used a slightly higher log likelihood ratio, they would have been merged (the LLR for cluster 2 was only 0.105). Note that the clustering is able to tease apart Bayesian learning (cluster 5) and non-bayesian learning (cluster 7)—both of which have Mike Jordan as their top author!

6 Discussion

We described a new model for Bayesian agglomerative clustering. We used Kingman's coalescent as our prior over trees, and derived efficient and easily implementable greedy and SMC inference algorithms for the model. We showed empirically that our model gives better performance than other

agglomerative clustering algorithms, and gives good results on applications to document modeling and phylolinguistics.

Our model is most similar in spirit to the Dirichlet diffusion tree of [2]. Both use infinitely exchangeable priors over trees. While [2] uses a fragmentation process for trees, our prior uses the reverse—a coalescent process instead. This allows us to develop simpler inference algorithms than those in [2] (we have not compared our model against the Dirichlet diffusion tree due to the complexity of implementing it). It will be interesting to consider the possibility of developing similar agglomerative style algorithms for [2]. [3] also describes a hierarchical clustering model involving a prior over trees, but his prior is not infinitely exchangeable. [5] uses tree-consistent partitions to model relational data; it would be interesting to apply our approach to their setting. Another related work is the Bayesian hierarchical clustering of [6], which uses an agglomerative procedure returning a tree structured approximate posterior for a Dirichlet process mixture model. As opposed to our work [6] uses a flat mixture model and does not have a notion of distributions over trees.

There are a number of unresolved issues with our work. Firstly, our algorithms take $O(n^3)$ computation time, except for Greedy-Rate1 which takes $O(n^2)$ time. Among the greedy algorithms we see that there are no discernible differences in quality of approximation thus we recommend Greedy-Rate1. It would be interesting to develop SMC algorithms with $O(n^2)$ runtime, and compare these against Greedy-Rate1 on real world problems. Secondly, there are unanswered statistical questions. For example, since our prior is infinitely exchangeable, by de Finetti's theorem there is an underlying random distribution for which our observations are i.i.d. draws. What is this underlying random distribution, and how do samples from this distribution look like? We know the answer for at least a simple case: if the Markov process is a mutation process with mutation rate $\alpha/2$ and new states are drawn i.i.d. from a base distribution H, then the induced distribution is a Dirichlet process $\mathrm{DP}(\alpha, H)$ [8]. Another issue is that of consistency—does the posterior over random distributions converge to the true distribution as the number of observations grows? Finally, it would be interesting to generalize our approach to varying mutation rates, and to non-binary trees by using generalizations to Kingman's coalescent called Λ-coalescents [15].

References

[1] R. O. Duda and P. E. Hart. *Pattern Classification And Scene Analysis*. Wiley and Sons, New York, 1973.

[2] R. M. Neal. Defining priors for distributions using Dirichlet diffusion trees. Technical Report 0104, Department of Statistics, University of Toronto, 2001.

[3] C. K. I. Williams. A MCMC approach to hierarchical mixture modelling. In *Advances in Neural Information Processing Systems*, volume 12, 2000.

[4] C. Kemp, T. L. Griffiths, S. Stromsten, and J. B. Tenenbaum. Semi-supervised learning with trees. In *Advances in Neural Information Processing Systems*, volume 16, 2004.

[5] D. M. Roy, C. Kemp, V. Mansinghka, and J. B. Tenenbaum. Learning annotated hierarchies from relational data. In *Advances in Neural Information Processing Systems*, volume 19, 2007.

[6] K. A. Heller and Z. Ghahramani. Bayesian hierarchical clustering. In *Proceedings of the International Conference on Machine Learning*, volume 22, 2005.

[7] N. Friedman. Pcluster: Probabilistic agglomerative clustering of gene expression profiles. Technical Report Technical Report 2003-80, Hebrew University, 2003.

[8] J. F. C. Kingman. On the genealogy of large populations. *Journal of Applied Probability*, 19:27–43, 1982. Essays in Statistical Science.

[9] J. F. C. Kingman. The coalescent. *Stochastic Processes and their Applications*, 13:235–248, 1982.

[10] P. Fearnhead. *Sequential Monte Carlo Method in Filter Theory*. PhD thesis, Merton College, University of Oxford, 1998.

[11] R. M. Neal. Annealed importance sampling. Technical Report 9805, Department of Statistics, University of Toronto, 1998.

[12] A. McMahon and R. McMahon. *Language Classification by Numbers*. Oxford University Press, 2005.

[13] M. Haspelmath, M. Dryer, D. Gil, and B. Comrie, editors. *The World Atlas of Language Structures*. Oxford University Press, 2005.

[14] H. Daumé III and L. Campbell. A Bayesian model for discovering typological implications. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2007.

[15] J. Pitman. Coalescents with multiple collisions. *Annals of Probability*, 27:1870–1902, 1999.