Computational Statistics and Data Analysis 176 (2022) 107566

Contents lists available at ScienceDirect

Computational Statistics and Data Analysis

www.elsevier.com/locate/csda

Agglomerative and divisive hierarchical Bayesian clustering

Elliot Burghardt [a,b,*], Daniel Sewell [a], Joseph Cavanaugh [a]

[a] Department of Biostatistics, 145 N. Riverside Dr., Iowa City, 52242, IA, United States
[b] Carver College of Medicine, 375 Newton Rd., Iowa City, 52242, IA, United States

ARTICLE INFO

Article history:
Received 22 September 2021
Received in revised form 29 June 2022
Accepted 7 July 2022
Available online 15 July 2022

Keywords:
Hierarchical clustering algorithms
Finite mixture model
Dirichlet concentration parameter
Dirichlet distribution

ABSTRACT

Cluster analysis methods are designed to discover groups of subjects or objects in datasets by uncovering latent patterns in data. Two model-based Bayesian hierarchical clustering algorithms are presented—divisive and agglomerative—that return nested clustering configurations and provide guidance on the plausible number of clusters in a principled way. These algorithms outperform many existing clustering methods on benchmark data. The methods are applied to identify subpopulations among Parkinson's disease subjects using only baseline data, and differing patterns of progression in the few years following diagnosis are demonstrated in the identified clusters.

1. Introduction

Unsupervised learning methods are used to expose hidden structure in datasets. Among these methods, cluster analysis methods are used to reveal subgroups of subjects or objects by their similarities. By recognizing and identifying subgroups, data can be organized for better understanding, increased efficiency, or both.

Because classification is the basis of understanding, there are many applications of cluster analysis across scientific disciplines. Examples in medicine include: differentiating between tissues in medical imaging (Selvan et al., 2017; Vianney Kinani et al., 2017); identifying cell types by their markers, size, and/or morphology in flow cytometry (Lo et al., 2009; Qian et al., 2010); explaining the heterogeneity within a disease population by identifying subpopulations—a step toward better treatments and personalized medicine (Sweeney et al., 2018; Knox et al., 2015; Pikoula et al., 2019; Wang et al., 2013). Chen et al. (2018) harness large amounts of cancer-omics data to identify cancer subtypes with consensus clusters. In sociology, a religiosity scale was built using hierarchical cluster analysis (Filsinger et al., 1979). In psychology, clustering was used to develop classes of alcoholism and to better direct treatment (DiMartini et al., 2008). Clustering has been using to find structural similarity in chemical compounds (Gallet and Pietrucci, 2013) and to identify patterns in climate modeling (Huth et al., 2008).

A broad class of clustering methods is hierarchical clustering, which aims to generate reasonable nested cluster configurations for each number of clusters, ranging from one to the number of observations (N). This hierarchy of clusters can be visualized via a dendrogram. Hierarchical clustering can either be: agglomerative, starting with N clusters and merging clusters at each step, or divisive, starting with one cluster and splitting clusters at each step. Agglomerative methods are far more prevalent than divisive (Kaufman and Rousseeuw, 2009). Traditionally, hierarchical clustering algorithms determine merges or splits based on proximity measures. To use these methods, users must select from a large collection of distance or dissimilarity measures and linkage methods, as none have been shown to perform uniformly better than the others (Everitt

* Corresponding author at: Department of Biostatistics, 145 N. Riverside Dr., Iowa City, 52242, IA, United States.
E-mail address: elliot-burghardt@uiowa.edu (E. Burghardt).

https://doi.org/10.1016/j.csda.2022.107566
0167-9473/

et al., 2011). These methods each come with implicit assumptions about cluster size and shape. The number of clusters must be selected in order to cut the dendrogram and uncover the final clustering configuration, but the hierarchical nature allows nested configurations from multiple numbers of clusters to be considered, often in the context of scientific understanding. In other words, hierarchical clustering returns nested clusters, which improves an investigator's ability to choose the number of clusters by the nested structure, ensuring results are sensible. For example, if observations are in the same cluster when there are four clusters, they will certainly be clustered when there are three or two clusters. In methods that are not hierarchical, one could obtain observations that are clustered together when there are two and four clusters but not when there are three clusters. These difficult to justify results will be avoided with hierarchical methods.

Another common clustering method is model-based clustering. Unlike the traditional hierarchical clustering methods, model-based clustering places a formal statistical model on the population, making the assumptions about the structure of the data explicit (Bouveyron et al., 2019). The population is assumed to arise from a mixture density, where each cluster has its own probability density function or probability mass function (Banfield and Raftery, 1993; Celeux and Govaert, 1993). This framework makes assumptions about cluster structure explicit. Model-based clustering approaches can provide interpretability and measures of uncertainty; however, they do not eliminate the need to specify the correct number of clusters (Bouveyron et al., 2019).

Model-based hierarchical clustering approaches have been established, combining the strengths of each. Agglomerative model-based clustering methods, which can be found in Heard et al. (2006), Heller and Ghahramani (2005), Iwayama and Tokunaga (1995), Vaithyanathan and Dom (2013), are more common than divisive model-based clustering methods (Sharma et al., 2017). Fraley and Raftery (2002) and Banfield and Raftery (1993) describe a classification likelihood approach as an agglomerative method called MBHAC (model based hierarchical agglomerative clustering), merging the two clusters that lead to the greatest increase in classification likelihood at each step and incorporating an expectation-maximization algorithm to refine the preliminary hierarchical results. These methods feature a reparameterization of covariance matrices in order to select features (i.e. shape, size, orientation) to be the same between clusters and they are implemented for multivariate normal data in the mclust package. Iwayama and Tokunaga (1995) propose an agglomerative algorithm for text classification that merges clusters in order to maximize the probability of the cluster configuration, given the data. Heller and Ghahramani (2005) present an agglomerative method based on a Dirichlet process model (DPM) that proceeds by merging partial trees. At each step, the algorithm selects the best partial trees to merge based on the probability of the data of interest having been generated from one tree (vs. more than one tree). Heard et al. (2006) perform agglomerative clustering based on a finite mixture model with decreasing number of components per merge. Their method includes a times series extension with regression splines. At each step, the algorithm selects the merge that maximizes the change in the log marginal posterior of the number of components/clusters (C) and clustering configuration. The optimal number of components is based on the maximum marginal posterior on the number of components. Sharma et al. (2017) implement a divisive method based on the maximum likelihood of Gaussian mixtures. This method forms clusters by removing one observation at a time by maximum distance until removal of additional observations decreases the likelihood (rough cluster), tuning the rough cluster by performing 2-means on the distances between observations and the mean of the rough cluster, and removing tuned clusters from the sample after each split.

Regardless of the clustering method used, cluster identification depends on the number of partitions. The process of selecting the appropriate number of clusters can be both challenging and subjective. Much work has been done to address this challenge, two of the most well-known techniques being average silhouette width and gap statistic. These methods are global methods, aiming to optimize the number of clusters over the entire dataset. With the exception of the gap statistic, most global methods are undefined at a single cluster ($k = 1$) and cannot provide information about whether or not data should be clustered (Tibshirani et al., 2001). The average sihoutte methods and the gap statistic both rely on within cluster distance, and the selection of distance measure (e.g. Euclidean) contains implicit assumptions about data distributions. Further, these methods do not take advantage of explicit data distributional assumptions. Many other methods exist, but they tend to be data-type dependent (e.g. only for continuous data) and have restrictive underlying assumptions (e.g. well separated and round), and little is known about their statistical properties (Everitt et al., 2011; Milligan and Cooper, 1985).

For model-based hierarchical clustering, the process of determining a reasonable number of clusters (and associated configurations) becomes a model selection problem. According to Steele and Raftery (2010), the Bayesian Information Criterion (BIC) can be used for this purpose, outperforming posterior probabilities for a well-known proper prior, as well as the Akaike Information Criterion (AIC), Deviance Information Criterion (DIC), and Integrated Complete Likelihood (ICL). While these criteria are general and can be informative for determining the number of clusters, there are concerns about their performance and suitability for choosing the number of clusters from a mixture distribution. For AIC and BIC, regularity conditions are often not met for mixture models, and both criteria tend to overestimate the number of clusters (Fraley and Raftery, 1998). ICL underestimates the number of clusters when clusters are poorly separated (Celeux et al., 2019). DIC is often unstable with mixture models (Celeux et al., 2019).

We propose agglomerative and divisive Bayesian hierarchical model-based clustering methods that provide nested configurations of clusters. Like many model-based clustering approaches, we apply the Dirichlet distribution as a prior over the cluster weights, and we utilize the Dirichlet shape hyperparameter in two ways: 1) to guide the number of clusters along the hierarchy and 2) to provide an informative metric on reasonable numbers of clusters via the favorability of each merge or split. Van Havre et al. (2015) developed a model-based clustering approach that also leveraged this Dirichlet shape hyperparameter, but our method contrasts with theirs in that our methods are hierarchical, do not require MCMC, and solve

for optimal Dirichlet parameters a posteriori rather than pre-specifying a fixed sequence for which there is not principled guidance. Other Bayesian methods for clustering with finite and infinite mixture models that harness on the impact of the Dirichlet concentration parameter on number of clusters have been proposed, but these methods are not hierarchical, tend to model each cluster through normal or mixtures of normal distributions, and depend on MCMC (Frühwirth-Schnatter et al., 2020; Malsiner-Walli et al., 2017; Miller and Harrison, 2018; Medvedovic et al., 2004). Fuentes-García et al. (2019) describe methods to identify a Bayesian point estimate for latent classifiers by maximizing of full conditional distributions. Our approach similarly seeks to identify maximum a posteriori (MAP) estimators for classifiers and provides flexibility in structure of the likelihood, but our framework provides and relies on nested clusters, avoiding complexities like label switching and the need for multiple starting points for classifiers and simplifying the optimization process. While the model-based hierarchical clustering framework in the form of mixture models is amenable to mixtures of a variety of distributions, most current methods and their implementations are limited to multivariate normal data. The methods of Heller and Ghahramani (2005) and Heard et al. (2006) depend on conjugacy, whereas our methods can be applied with or without conjugate priors on component parameters. Not only does this allow for more choices of priors, but it also allows for flexibility in additional and indeed multiple data types. Unlike Iwayama and Tokunaga (1995), Heard et al. (2006), and Banfield and Raftery (1993), our methods allow for empty components and update the Dirichlet concentration parameter in order to harness the utility of the Dirichlet prior on the finite mixture model. This parameter update guides the number of clusters, defined as non-empty components of the mixture. The methods of Sharma et al. (2017) are restricted to Gaussian mixtures, and their method is incompatible with a complete dendrogram (1 to N hierarchy). Unlike other hierarchical methods, each partition within the nested hierarchy of our algorithms is a meaningful clustering result, even without constraining the data type. With the exception of the divisive method of Sharma et al. (2017), all hierarchical model-based methods described are exclusively agglomerative. In summary, we propose both agglomerative and divisive hierarchical clustering methods which are applicable to any data type, are compatible with non-conjugate priors, provide meaningful partitions at each step of the hierarchy, and use the Dirichlet prior structure as a principled way to guide researchers to plausible numbers of clusters.

In Section 2, we describe the modeling framework used, the role of the Dirichlet concentration parameter, and the steps of the agglomerative and divisive algorithms. Section 3 compares our method to existing approaches on common clustering benchmark datasets, and Section 4 provides a simulation study for additional evaluation. Section 5 provides a detailed analysis using the Parkinson's Progression Markers Initiative database to identify Parkinson's disease subtypes. Finally, Section 6 gives a brief conclusion.

2. Methods

2.1. Finite mixture model framework

In our hierarchical clustering algorithms, we determine hard cluster assignments and cluster parameters from overfitted finite mixture models. Due to the model specifications, we use the term "cluster" to represent non-empty mixture components. We start from the classification likelihood, where the data consisting of N observations are denoted as $y := (y_1, y_2, \ldots, y_N)$. We define $Z = [Z_{ik}]$, $i = 1, \ldots, N$, $k = 1, \ldots, K$, as classifiers such that Z_{ik} equals one if individual i belongs to cluster k, $k = 1, 2, \ldots, K$, and zero otherwise. Data from the k^{th} cluster are modeled through the likelihood function, f, parameterized by $\theta_k \in \Theta$. The classification likelihood can be expressed as

$$\pi(y|Z, \theta) = \prod_{i=1}^{N} \prod_{k=1}^{K} [f(y_i|\theta_k)]^{Z_{ik}}, \tag{1}$$

where $\theta = (\theta_1, \ldots, \theta_K)$. While not technically necessary, in practice for multivariate y_i's we set f such that variables of differing distributions are conditionally independent given cluster assignments, with the exception of multivariate normal features within y_i.

We set a Dirichlet-multinomial prior on the component weights, $\xi := (\xi_1, \ldots, \xi_K)$, and the classifiers, $\{Z_i\}$, and a uniform prior on the Dirichlet shape parameter, α. Although our methods are not dependent on this specific hyperprior on α, we encourage its use. A uniform prior on α aligns with our expectations that α will range from very small to very large along the hierarchy. That is,

$$\begin{aligned} Z_i = (Z_{i1}, \ldots, Z_{iK}) &\overset{iid}{\sim} Mult(1, \xi) \\ \xi|\alpha &\sim Dir(\alpha) \\ \alpha &\sim Unif(0, \alpha_{max}), \quad 0 < \alpha_{max} < \infty. \end{aligned} \tag{2}$$

However, in what follows we integrate out ξ so that

$$\pi(Z|\alpha) = \frac{\Gamma(K\alpha)}{[\Gamma(\alpha)]^K} \frac{\prod_{k=1}^{K} \Gamma(n_k + \alpha)}{\Gamma(N + K\alpha)},$$

where n_k is the number of individuals belonging to cluster k, i.e., $n_k = \sum_i Z_{ik}$.

Our framework assumes a fixed, large number K of components (usually $K = N$), with the intention that many or most will be empty. Because of this, much care is needed in the prior structure of the component-specific parameters θ_k, else the prior $\pi(\theta_k)$ can potentially dominate the determination of how many clusters- defined to be non-empty components- are plausible. To avoid any such deleterious effects, we set the prior on θ_k to be dependent on whether or not a component is empty, i.e., our joint prior is structured as

$$\begin{aligned}\pi(\theta, Z, \alpha) &= \pi(\theta|Z, \alpha)\pi(Z|\alpha)\pi(\alpha) \\ &= \left[\prod_{k=1}^{K} \pi\left(\theta_k|1_{[n_k>0]}\right)\right]\pi(Z|\alpha)\pi(\alpha). \end{aligned} \tag{3}$$

Let $\pi_1(\theta_k)$ denote the prior density function given that there is at least one individual in cluster k, and $\pi_0(\theta_k)$ given that the cluster is empty. Our methods are not dependent on a specific form or the conjugacy of $\pi_1(\theta_k)$, although vague priors have been selected for the implementations that follow. Specifically, for gamma shape and rate parameters, we used gamma priors with shape = 1.01 and rate = 0.01; for the Bernoulli probability parameter we used a beta(1.01, 1.01) prior; for multivariate normal parameters we used a normal-Wishart($\vec{0}_p$, .001, $10\mathcal{I}_p$, $p+3$) prior, where p is the number of variables and $\mathcal{I}_p$ is the identity matrix of dimension p. We set π_0 to be a uniform prior over $\mathcal{B} := \widetilde{\mathcal{B}} \cap \Theta$, where $\widetilde{\mathcal{B}}$ is some ball around an interior point of the parameter space, Θ. Then the conditional prior over the component parameters is given by

$$\pi(\theta_k|Z) = \begin{cases} \pi_1(\theta) & n_k > 0 \\ \pi_0(\theta) = \frac{1_{[\theta\in\mathcal{B}]}}{\int_{\mathcal{B}} d\theta} & n_k = 0. \end{cases} \tag{4}$$

The prior π_0 is selected to be uniform, since a shaped distribution makes little sense once it is known that no observations will come from this mixture component. However, as we will see later, the volume of this ball has the potential to play an oversized role in determining the number of clusters. However, if Θ is unbounded (e.g., $\Theta = [0,1] \times \Re^+$), we can set $\pi_0(\theta_k) = c$ for any $c > 0$. (See the supplemental materials A for the proof.) In the analyses we present throughout this paper we set $\pi_0(\theta_k) = 1$. It is important to note, however, that this constant can be changed to penalize or support empty components so that the posterior may respectively support splitting or merging clusters. As $|\mathcal{B}| \to \infty \Leftrightarrow c \to 0$, the prior tends to disfavor merges (favors splits) as $c < \pi_1(\hat{\theta}_k)$ for any reasonable estimate $\hat{\theta}_k$ of θ_k, whereas as $|\mathcal{B}| \to 0 \Leftrightarrow c \to \infty$, the prior tends to favor merges (disfavors splits) as $c > \pi_1(\hat{\theta}_k)$.

At a high level, our proposed agglomerative (divisive) hierarchical clustering algorithm iterates between the following steps: (1) for fixed values of θ and Z, fix α at the maximum of the conditional posterior $\pi(\alpha|\theta, Z, y)$; and (2) for the newly fixed α, update Z through merges (splits) and θ to maximize the conditional posterior $\pi(\theta, Z|\alpha, y)$. This is nearly identical to a coordinate ascent approach with the difference being the nested clustering structure imposed throughout the algorithm. This nested structure can be formalized via the following definition.

Definition. For two $N \times K$ clustering assignment matrices $Z^{(1)}$ and $Z^{(2)}$, let $\mathcal{I}_k^{(m)} := \left\{i : Z_{ik}^{(m)} = 1\right\}$ for $k = 1, \ldots, K$, and $m = 1, 2$. We say $Z^{(1)}$ is <u>nested</u> in $Z^{(2)}$ if for $j, k \in \{1, 2, \ldots, K\}$, $\mathcal{I}_j^{(1)} \cap \mathcal{I}_k^{(2)}$ equals either $\mathcal{I}_j^{(1)}$ or $\emptyset$, and we denote this by $Z^{(1)} \subset Z^{(2)}$.

Importantly, after each iteration we have a (local) posterior mode of the clustering assignments and component parameters given a fixed hyperparameter value of α, thereby imbuing meaning into the clustering assignations at each step of the hierarchy. To assist in determining the number of clusters, after completion of the two-step iterative process we run through the hierarchical clustering structure to examine the plausibility of each merge (split) using an approach based on the Dirichlet concentration parameter which we will describe in the next section.

2.2. Dirichlet concentration parameter

Our methods highlight the role of the Dirichlet concentration parameter, α, as it relates to emptying out extra components of overfitted mixture models. As described in detail by Rousseau and Mengersen (2011), this α provides a sliding scale along the hierarchy of cluster configurations. For larger α, the component weights will be more evenly distributed, leading to fewer empty components. Smaller α leads to the component weights falling into a corner of the K-simplex, leading to more empty components.

The Dirichlet shape parameter, then, plays a critical role in our proposed hierarchical clustering algorithms. We utilize this concentration parameter in two distinct ways: (1) to guide the merging or splitting of clusters at each step of the algorithm; and (2) to guide the selection of reasonable cluster numbers and configurations along the hierarchy.

As mentioned in the brief synopsis at the end of Section 2.1, at each step of our algorithm we find the $\hat{\alpha}$ which maximizes the conditional posterior mode given the cluster assignments and component parameters. That is, suppose at the start of the ℓ^{th} iteration we have an estimate of Z, $Z^{(\ell-1)}$, and of each θ_k, $\theta_k^{(\ell-1)}$; we then find

$$\hat{\alpha}^{(\ell)} = \text{argmax}_{\alpha}\, \pi(\alpha|\theta^{(\ell-1)}, Z^{(\ell-1)}, y) = \text{argmax}_{\alpha}\, \frac{\Gamma(K\alpha)}{[\Gamma(\alpha)]^K} \frac{\prod_{k=1}^{K} \Gamma(n_k^{(\ell-1)} + \alpha)}{\Gamma(N + K\alpha)} \pi(\alpha), \tag{5}$$

where $n_k^{(\ell)} = \sum_i Z_{ik}^{(\ell)}$. Note that if $\pi(\alpha)$ has non-zero probability mass over an unbounded space, this equation may not have a valid maximum. For example, there is no valid maximum when there are no empty components. (See the supplemental materials B for the proof—which shows that this function has a positive slope everywhere—and additional guidance on the prior on α based on results in Rossi (2014).) This new value $\hat{\alpha}^{(\ell)}$ is then used to update

$$(\theta^{(\ell)}, Z^{(\ell)}) = \underset{\theta \in \Theta^K, Z \supset Z^{(\ell-1)}}{\text{argmax}}\ \pi(\theta, Z|\hat{\alpha}^{(\ell)}, y), \tag{6}$$

for the agglomerative method and

$$(\theta^{(\ell)}, Z^{(\ell)}) = \underset{\theta \in \Theta^K, Z \subset Z^{(\ell-1)}}{\text{argmax}}\ \pi(\theta, Z|\hat{\alpha}^{(\ell)}, y), \tag{7}$$

for the divisive method. The values $(\theta^{(\ell)}, Z^{(\ell)})$ maximize the posterior conditioned on $\alpha = \hat{\alpha}^{(\ell)}$, and hence for each hyperparameter value $\alpha \in \{\hat{\alpha}^{(\ell)}\}$ our algorithm produces the MAP estimators of (Z, θ), although this estimator is very likely only a local mode due to the algorithmic nesting constraints on $\{Z^{(\ell)}\}$.

We now describe how we may use α to help determine the plausibility of a given number of clusters/non-empty components. When evaluating the potential merging of two clusters, we wish to find the concentration parameter, α_{root}, at which the current merge is neither favored nor disfavored a posteriori. At step ℓ, we find α_{root} which satisfies

$$\log[\pi(\theta^{(\ell)}, Z^{(\ell)}|\alpha, y)] - \log[\pi(\theta^{(\ell-1)}, Z^{(\ell-1)}|\alpha, y)] = 0. \tag{8}$$

The value of α_{root} is informative regarding which numbers of clusters, k, and thus which cluster configurations, are favorable. This is seen from the theorem below.

Theorem. *Suppose we have two nested cluster assignment matrices $Z^{(\ell-1)} \subset Z^{(\ell)}$ and the associated parameter estimates $\theta^{(\ell-1)}$ and $\theta^{(\ell)}$ as described in the main text. Then if α_{root} from (8) exists, for $\alpha < \alpha_{root}$,*

$$\log[\pi(\theta^{(\ell)}, Z^{(\ell)}|\alpha, y)] > \log[\pi(\theta^{(\ell-1)}, Z^{(\ell-1)}|\alpha, y)],$$

and for $\alpha > \alpha_{root}$

$$\log[\pi(\theta^{(\ell)}, Z^{(\ell)}|\alpha, y)] < \log[\pi(\theta^{(\ell-1)}, Z^{(\ell-1)}|\alpha, y)].$$

This theorem, which is given for the agglomerative method yet can be trivially adapted for the divisive, shows that any concentration hyperparameter $\alpha < \alpha_{root}$ favors a posteriori a merge whereas $\alpha > \alpha_{root}$ favors a split. In other words, when α_{root} is small, in order to favor this merge, the concentration hyperparameter must shrink further still, thus placing high prior probability on few clusters. When α_{root} is large, this merge is already favorable without shrinking the concentration hyperparameter much, so that priors with large probability on a larger number of clusters may still favor the merge. Further, we have the following corollary.

Corollary. *Suppose we have two nested cluster assignment matrices $Z^{(\ell-1)} \subset Z^{(\ell)}$ and the associated parameter estimates $\theta^{(\ell-1)}$ and $\theta^{(\ell)}$ as described in the main text. Then if α_{root} from (8) does not exist, there does not exist a Dirichlet prior on the component weights ξ such that*

$$\log[\pi(\theta^{(\ell)}, Z^{(\ell)}|\alpha, y)] > \log[\pi(\theta^{(\ell-1)}, Z^{(\ell-1)}|\alpha, y)],$$

i.e., the posterior favors the merged clustering assignments.

(See proof in the supplemental materials C; the idea is that if the conditional posterior of θ given the merge is larger than that given the split, then no Dirichlet prior can overcome this discrepancy to a posteriori favor the split.) Based on this, we may use the existence of α_{root} as a rule of thumb to guide the number of clusters. The smallest k such that α_{root} does not exist (i.e. before the first disfavored split in divisive method or after the last favored merge in the agglomerative method) is a reasonable location to cut the dendrogram, as no prior beliefs (described by the Dirichlet family of distributions) on the component weights would lead us to prefer the split. Yet we strongly emphasize that because our methods are hierarchical, this is simply a guide and we encourage the consideration of other reasonable numbers of clusters and configurations based on scientific insight.

Unlike $\hat{\alpha}$, α_{root} will not necessarily decrease monotonically with the number of empty components, because the simplified root equation does not even include the number of empty components not involved in the merge/split. This means α_{root} is highly local, and thus will not be monotonic throughout the hierarchical clustering algorithm. As a local method for

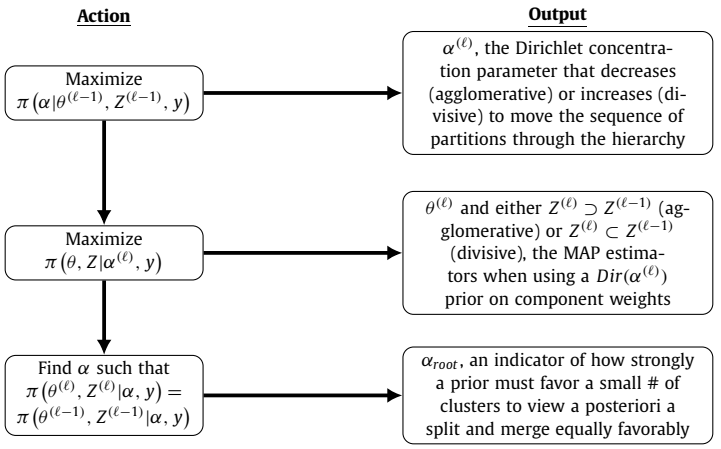

Fig. 1. Conceptual schematic for a single stage of the A-HBC or D-HBC algorithms. The input is assumed to be the current parameter and cluster estimates, $\theta^{(\ell-1)}$ and $Z^{(\ell-1)}$.

determining the number of clusters, α_{root} only provides information about each configuration, specifically by providing a measure of the favorability of each merge/split by comparing the previous to the current configuration. The notion of α_{root} is similar conceptually to that of a Bayes factor as described in supplemental materials D. While the values of α_{root} are not perfectly comparable, looking at the existence of α_{root} does provide a method to compare any number of clusters from 1 to N.

2.3. Overview of clustering algorithms

To provide context to the descriptions of the agglomerative and divisive hierarchical clustering algorithms that follow, we would like to first present an overview of the methods. In Fig. 1, we provide a schematic of our algorithms. At each number of clusters starting with 1 (divisive) or N (agglomerative), we solve for $\hat{\alpha}$, then identify MAP estimators for nested cluster assignments and cluster parameters given $\hat{\alpha}$, and finally solve for α_{root} to determine whether the most recent merge or split was favorable. We repeat this process until we have N clusters (divisive) or 1 cluster (agglomerative) or reach some other stopping criteria.

2.4. Agglomerative hierarchical Bayesian clustering algorithm

The goal of our agglomerative hierarchical Bayesian clustering (A-HBC) is to find the "best" cluster configuration for each number of clusters by merging clusters, one by one, under the constraint that the partitions corresponding to different numbers of clusters are nested, i.e., $Z^{(\ell-1)} \subset Z^{(\ell)}$. This approach begins with the number of clusters equal to the number of observations, N, and selects the merge that leads to the greatest increase (or smallest decrease) in the posterior. After determining cluster assignments and parameters, we find both α_{root} and $\hat{\alpha}$, at each step. These steps iterate until all components are merged.

We initialize $Z^{(1)} = I_N$ directly, and $\theta_k^{(1)}$ is maximized as specified by the posterior distribution for a single observation per cluster. In considering merges, we calculate the change in posterior for each potential merge. For the initial merges that involve two singletons, this change will need to be fully calculated for each potential merge. For all other merges, evaluating this posterior change involves updating posteriors for only those merges involving newly merged and newly emptied components, as well as adjusting for changes in $\hat{\alpha}$.

At any merge step, the candidate configurations Z^* are updated by moving all observations in some column Z_j into another column Z_k (i.e. $Z_k^* = Z_j^{(\ell-1)} + Z_k^{(\ell-1)}$ and $Z_j^* = 0$), and all other columns of Z^* remaining as they were prior to the merge. Then component parameters MAPs are determined given the candidate configuration:

$$\theta_k^* = \arg\max_{\theta_k} \pi(y_{\mathcal{I}_k^{(\ell-1)} \cup \mathcal{I}_j^{(\ell-1)}}|\theta_k, Z^*)\pi(\theta_k|Z^*),$$

where for a set $\mathcal{I} \subseteq \{1, \ldots, N\}$, $y_{\mathcal{I}} := (y_{\mathcal{I}_1}, \ldots, y_{\mathcal{I}_{|\mathcal{I}|}})$. Then we evaluate the change in the log posterior due to the candidate merge:

$$\begin{aligned} M_{kj} &= \log\left(\pi(\theta^*, Z^*|\hat{\alpha}^{(\ell-1)}, y)\right) - \log\left(\pi(\theta^{(\ell-1)}, Z^{(\ell-1)}|\hat{\alpha}^{(\ell-1)}, y)\right) \\ &= \log\left(\pi(\theta_k^*, Z_k^*|\hat{\alpha}^{(\ell-1)}, y)\right) + \log\left(\pi(\theta_j^*, Z_j^*|\hat{\alpha}^{(\ell-1)}, y)\right) \\ &\quad - \log\left(\pi(\theta_k^{(\ell-1)}, Z_k^{(\ell-1)}|\hat{\alpha}^{(\ell-1)}, y)\right) - \log\left(\pi(\theta_j^{(\ell-1)}, Z_j^{(\ell-1)}|\hat{\alpha}^{(\ell-1)}, y)\right), \end{aligned}$$

where $\pi(\theta_j^*, Z_j^* | \hat{\alpha}^{(\ell-1)}, y) = \pi_0(\theta)\pi(Z_j^* = (0 \dots 0)^T | \hat{\alpha}^{(\ell-1)})$ is the posterior of an empty component.

The merge (k, j) that leads to the greatest increase in posterior, M_{kj}, is selected as best. We solve (5) to get $\hat{\alpha}^{(\ell)}$ and set $\alpha_{root}^{(\ell)}$ as the root of (8). Merging is repeated until all observations are in a single cluster. In practice, $\hat{\alpha}$ is constrained to a large α_{max} upper bound, and we selected $\alpha_{max} = 10^7$, noting that any $\alpha > \frac{|\Theta|}{2}$ tends to lead to larger numbers of non-empty components than true clusters (Rousseau and Mengersen, 2011). The general agglomerative algorithm is outlined below.

initialize: number of clusters, $nclust = N$
  $\hat{\alpha} = \alpha_{max}$
while $nclust > 1$ **do**
  for $j \neq k$: $n_j \neq 0, n_k \neq 0$ **do**
    Find optimal θ_{kj}^* by merging j and k.
    Solve for change in posterior by merging clusters j and k, M_{kj} for $k, j \in \{1, 2, ..., K | j > k\}$.
  end
  Merge $\{\tilde{k}, \tilde{j}\} = \text{argmax}_{\{k,j\}} M_{kj}$.
  Update cluster assignments and parameters, Z and θ.
  Update $nclust = \sum_k 1_{[n_k > 0]}$.
  Compute optimizing $\hat{\alpha}$.
  Solve for α_{root}.
end

Algorithm 1: Agglomerative Hierarchical Bayesian Clustering (A-HBC)

In the algorithm, most cluster assignments and parameters will remain unchanged between iterations, so candidate component parameters and assignments, θ^* and Z^*, will only need to be optimized for merges involving the most recently merged clusters, $\tilde{j}$ and $\tilde{k}$. That is, only for the first merge will these values need to be calculated for all $\binom{nclust}{2}$ potential merges; the remaining $N - 2$ iterations require $nclust - 1$ comparisons. Thus the overall number of parameter updates is $\mathcal{O}(N^2)$. A more detailed version of the A-HBC algorithm is provided in the supplemental materials E.

2.5. Divisive hierarchical Bayesian clustering algorithm

While agglomerative algorithms can consider all possible merges, divisive methods cannot reasonably consider all possible splits. There are only $\binom{nclust}{2}$ possible merges at each merging step, which is quadratic in $nclust$, the number of clusters (at maximum $nclust$ is the number of subjects). All possible splits would be $2^{n_k - 1} - 1$ combinations at each splitting step, which grows exponentially with n_k, the number of subjects per cluster (at maximum n_k is the overall number of subjects). For example, with 10 subjects, there are 45 merges and 511 splits possible for agglomerative and divisive methods at the step with the largest number of possibilities, respectively. For 100 subjects, this is 4950 merges and roughly 6.34×10^{29} splits. However, we can still find a posterior mode without conducting an exhaustive search through all possible splits.

The goal of the divisive hierarchical Bayesian clustering (D-HBC) algorithm is to find the "best" cluster configuration for each number of clusters by bisecting clusters (so $Z^{(\ell)} \supset Z^{(\ell-1)}$). This approach begins with one cluster and selects the split that leads to the greatest increase in posterior. As with A-HBC, we find both α_{root} and $\hat{\alpha}$ at each step. Splitting and α determination steps can repeat until each observation is in its own cluster. However, stopping criteria relating to the maximum number of clusters or α_{root} can be applied to shorten the runtime while providing all reasonable cluster configurations within the hierarchy.

We initialize Z as a column vector of 1's (all observations in one cluster) and 0's elsewhere, and θ_k is maximized as specified by the posterior distribution for all observations in a single cluster. In considering splitting, we calculate the change in posterior for each candidate split.

At any split step, the MAP candidate cluster assignments and parameters are determined by coordinate ascent. Briefly, for each k such that $n_k^{(\ell)} > 1$, $\mathcal{I}_k$ is initialized into two preliminary clustering assignments by a rapid heuristic clustering approach (e.g. 2-medoids) into $\mathcal{I}_{k1}^*$ and $\mathcal{I}_{k2}^*$. We then iterate until convergence between updating the two newly formed cluster components' parameters and subsequently updating the cluster assignments. To stabilize this iterative process, each observation's assignment is not permitted to change back and forth in successive iterations. Rather, upon changing clusters an observation will remain in the newly assigned cluster for the next iteration, regaining its ability to defect to the previous assignment in the iteration that follows. If all observations are updated to the same cluster, it is strongly recommended to discontinue splitting this cluster. However, the algorithm can continue by forcing a split. This is done by splitting off the observation for which the difference between the value of the posterior in the two possible cluster assignments is smallest (i.e. the observation with the weakest preference for the assignment of all of the other observations).

Next we evaluate the change in the log posterior due to the candidate split, M_k, which follows the same formula as that of A-HBC. The split of cluster k that leads to the greatest increase in the posterior, M_k is selected as best, and we then solve for $\hat{\alpha}^{(\ell)}$ and $\alpha_{root}^{(\ell)}$. These steps are iterated until splitting is no longer recommended (as described in the previous paragraph) or all observations are in their own cluster. In practice, $\hat{\alpha}$ is constrained to a small α_{min} lower bound, and we

selected α_{min} as the smallest positive double-precision number of the machine in use, noting that any $\alpha < |\Theta|/2$ tends to lead to fewer numbers of non-empty components than true clusters (Rousseau and Mengersen, 2011). The general divisive algorithm is outlined below.

initialize: number of clusters, $nclust = 1$
$\hat{\alpha} = \alpha_{min}$
while $nclust < N$ **do**
  for *each candidate split* $k : |\mathcal{I}_k| > 1$ **do**
    Use 2-medoids to initialize candidate split assignments into $\mathcal{I}^*_{k1}$ and $\mathcal{I}^*_{k2}$.
    Update θ^*_{k1}, θ^*_{k2} and $\mathcal{I}^*_{k1}$ and $\mathcal{I}^*_{k2}$ via coordinate ascent.
    Solve for change in posterior by splitting k into $k1$ and $k2$, M_k.
  end
  Split cluster $\{\tilde{k}\} = \text{argmax}_{\{k\}} M_k$.
  Update cluster assignments and parameters, Z and θ.
  Update $nclust = \sum_k 1_{[\sum_i z_{ik} > 0]}$.
  Compute optimizing $\hat{\alpha}$.
  Solve for α_{root}.
end

Algorithm 2: Divisive Hierarchical Bayesian Clustering (D-HBC)

In the algorithm, most cluster assignments and parameters will remain unchanged between iterations, so candidate parameters and assignments, θ^* and Z^*, will only need to be optimized for merges involving the two clusters resulting from the most recent split, $\tilde{k}$ and $nclust$. Additionally, the algorithm can be stopped at a reasonable maximum number of clusters or criteria involving α_{root}. Even if the algorithm runs until all observations are in their own cluster, the number of splits is $\mathcal{O}(N)$. A more detailed version of the D-HBC algorithm is provided in the supplemental materials F. An example to illustrate how the cluster assignments proceed is included the supplemental materials G.

3. Benchmark data

We compared the performance of our algorithms to that of common hierarchical clustering methods; we also compared our approach to k-means, as this is arguably the most common clustering algorithm in use and provides a natural reference. All analyses were completed in R R Core Team (2020). For hierarchical clustering methods, we selected hclust (traditional agglomerative methods by average linkage using Euclidean distance for all continuous variables or Gower distance for a combination of categorical/dichotomous and continuous variables) (Müllner, 2013), DIANA (DIvisive ANAlysis Clustering) (Kaufman and Rousseeuw, 2009), and MBHAC (model-based hierarchical agglomerative clustering based on maximum likelihood criteria for Gaussian mixture models parameterized by eigenvalue decomposition with BIC to select shape/size/orientation) (Fraley and Raftery, 2002). To select the number of clusters, we used the gap statistic (first within 1 SE of local maximum) for hclust, DIANA, and k-means, BIC for MBHAC, and the existence of α_{root} for our methods. Because multiple methods of determining the number of clusters may be used, we also included performance metrics for each method at the true number of clusters (i.e. cutting the tree at the true k or providing the true k to k-means).

We considered dataset Hepta ($N = 212$, $k = 7$, $p = 3$) from the Fundamental Clustering Problem Suite (FCPS), which includes seven clearly defined clusters of simulated multivariate normal data with different variances (Ultsch, 2005). For A-HBC and D-HBC, the form of the likelihood was specified as multivariate normal for the 3 variables. We also considered three real datasets. The Iris flower dataset ($N = 150$, $k = 3$, $p = 4$) includes four continuous attributes from 50 samples for each of three species of irises (Fisher, 1936; Anderson, 1936). For A-HBC and D-HBC, the form of the likelihood was specified as multivariate normal for the 4 variables. The Reaven diabetes dataset ($N = 145$, $k = 3$, $p = 5$) includes five continuous attributes for subjects with chemical and overt diabetes and healthy controls (Reaven and Miller, 1979). For A-HBC and D-HBC, the form of the likelihood was specified as multivariate normal for the "relwt", "glufast", "gluest", and "sspg" variables and gamma for the skewed "instest" variable. The Dermatology dataset includes clinical and histological features of six erythemato-squamous diseases and expert diagnosis (Dua and Graff, 2017). For the Dermatology dataset ($N = 358$, $k = 6$, $p = 12$), the 12 clinical attributes were dichotomized as present vs. absent, family history (dichotomous), and age (continuous). The form of the likelihood was specified as Bernoulli for dichotomous variables and normal for the "age" variable for A-HBC and D-HBC. In addition to these datasets described above, 6 other datasets were analyzed with similar results which are presented in the Supplementary Material.

For all data sets and clustering methods, only complete data were used, data were centered (subtracted off mean) and scaled (divided by standard deviation) for all multivariate normal covariates and data were scaled (divided by root mean square) for skewed continuous covariates. The form of the likelihood was visually assessed by viewing the marginal densities by cluster for continuous variables. For example, very skewed densities per cluster of a variable where the likelihood had been specified as normal may indicate that the likelihood has been misspecified. Clustering accuracy may be improved by more appropriately specifying the likelihood.

To evaluate clustering performance, we used several external criteria of clusters vs. gold standard classes. These external criteria metrics are the estimated number of clusters ($\hat{k}$), Adjusted Rand Index (ARI), Normalized Mutual Information (NMI), and Normalized Variation of Information (NVI). ARI is the accuracy of clustering algorithm adjusted to account for chance

Table 1
External validation of clustering on benchmark data.

Unknown k							
Dataset	Metric	A-HBC	D-HBC	HClust	DIANA	MBHAC	k-means
Hepta	$\hat{k}$	**7**	**7**	**7**	1	**7**	1
$k=7$	ARI	**1**	**1**	**1**	0	**1**	0
$n=212$	NMI	**1**	**1**	**1**	0	**1**	0
$p=3$	VI	**0**	**0**	**0**	1	**0**	1
Iris	$\hat{k}$	**3**	**3**	2	**3**	2	2
$k=3$	ARI	0.8180	**0.9410**	0.5681	0.5923	0.5438	0.5681
$n=150$	NMI	0.8130	**0.9192**	0.7337	0.6427	0.6925	0.7337
$p=4$	NVI	0.3150	**0.1495**	0.4206	0.5265	0.4704	0.4206
Diabetes	$\hat{k}$	**3**	**3**	1	2	**3**	**3**
$k=3$	ARI	0.5192	**0.5446**	0	0.3122	0.4824	0.3392
$n=145$	NMI	0.5217	**0.5442**	0	0.429	0.4577	0.3571
$p=5$	NVI	0.6471	**0.6262**	1	0.7269	0.7032	0.7826
Derm	$\hat{k}$	33	**6**	1	4	NA	24
$k=6$	ARI	0.2584	**0.6066**	0	0.1931	NA	0.1899
$n=358$	NMI	0.5499	**0.6831**	0	0.3523	NA	0.4812
$p=12$	NVI	0.6208	**0.4813**	1	0.7862	NA	0.6832
Known k							
Dataset	Metric	A-HBC	D-HBC	HClust	DIANA	MBHAC	k-means
Hepta	$\hat{k}$	✓	✓	✓	-	✓	-
$k=7$	ARI	**1**	**1**	**1**	0.8306	**1**	0.6422
$n=212$	NMI	**1**	**1**	**1**	0.9296	**1**	0.8405
$p=3$	NVI	**0**	**0**	**0**	0.377	**0**	0.2751
Iris	$\hat{k}$	✓	✓	-	✓	-	-
$k=3$	ARI	0.8180	**0.9410**	0.5621	0.5923	0.6304	0.6201
$n=150$	NMI	0.8130	**0.9192**	0.7131	0.6427	0.7166	0.6595
$p=4$	NVI	0.3150	**0.1495**	0.4459	0.5265	0.4417	0.508
Diabetes	$\hat{k}$	✓	✓	-	-	✓	✓
$k=3$	ARI	0.5192	**0.5446**	0.3947	0.2831	0.4824	0.3392
$n=145$	NMI	0.5217	**0.5442**	0.5274	0.398	0.4577	0.3571
$p=5$	NVI	0.6471	**0.6262**	0.6419	0.7516	0.7032	0.7826
Derm	$\hat{k}$	-	✓	-	-	NA	-
$k=6$	ARI	0.5251	**0.6066**	0.5253	0.2165	NA	0.3573
$n=358$	NMI	0.5367	**0.6831**	0.6660	0.4442	NA	0.4935
$p=12$	NVI	0.6333	**0.4813**	0.5007	0.7145	NA	0.6724

correctness (1 is best, 0 is equal to expected correctness by chance, and < 0 is worse than random guessing); NMI is the reduction in entropy of classifications given the cluster labels, normalized for sum of class and clusters entropy to allow for comparing cluster configurations with different numbers of clusters (1 is best and 0 is worst); and NVI is the information change (loss and gain) between classes and cluster assignments, which is also normalized for joint entropy of classes and clusters to allow for comparing cluster configurations with different numbers of clusters (0 is best and 1 is worst) (Meilă, 2007).

Table 1 provides the results from the benchmark analyses. For all of the benchmark datasets, D-HBC outperforms the competing methods or else clusters perfectly according to the gold standard. In all four cases D-HBC correctly selects the number of clusters using the existence rule of α_{root} for each dataset, and still performs better than all other tested methods even when the true k is provided.

With respect to ARI, NMI, and NVI, A-HBC was the next best performer with unknown k, and is only narrowly outperformed by HClust when the true k is provided for one dataset (Dermatology). However, it is clear that A-HBC, using the existence of α_{root} criterion, dramatically overestimates the number of clusters for that dataset. In such cases where a very large $\hat{k}$ is selected by the α_{root} existence rule of thumb, the values of all α_{root}'s and the hierarchical results from A-HBC, as well as scientific background, could be used to select a more reasonable number of clusters.

In fact, the Dermatology dataset provides a good case study on this issue. Fig. 2 represents the dendrogram of the A-HBC results for the Dermatology dataset. For 358 observations, 33 clusters would warrant further investigation. Closer evaluation of the values of α_{root} reveals a drastic change between the last three merges ($\alpha_{root} = 3.055 \times 10^{-7}$, 8.017×10^{-7}, and 9.658×10^{-7}) and fourth-to-last ($\alpha_{root} = 2.931 \times 10^{-4}$) and fifth-to-last ($\alpha_{root} = 1.982$) merges. This means that at each of the final three merges, keeping the clusters apart (not merging) is very strongly supported by the data, whereas at the fourth-to-last merge the evidence against merging clusters is less strong and at the fifth-to-last merge and beyond the

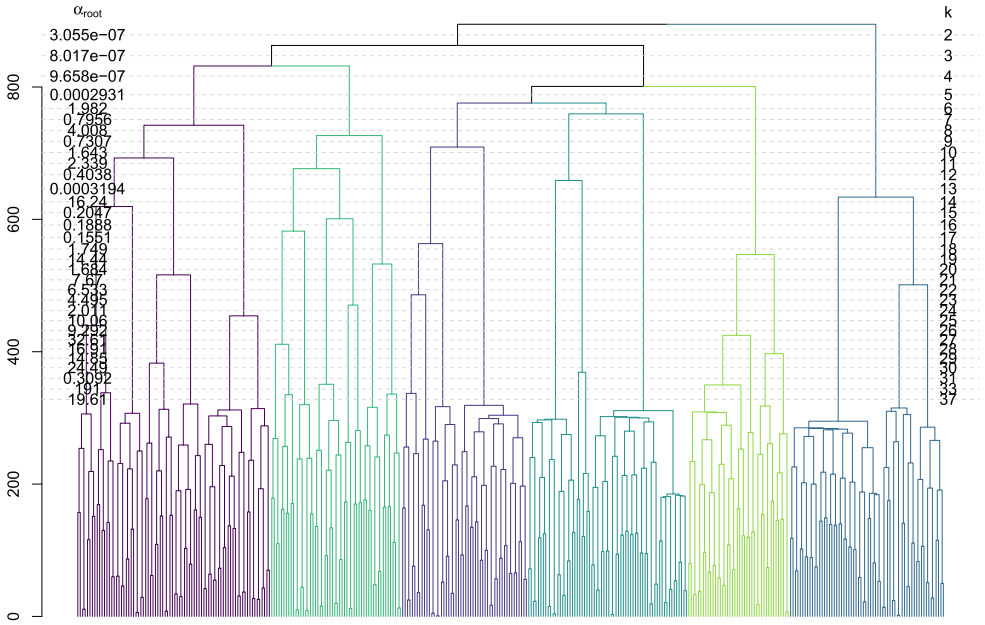

Fig. 2. Dendrogram of A-HBC from Dermatology dataset. The colorings correspond to the 6 true clusters. (Color online.)

evidence is even weaker. Considering this context and how unlikely it would be to find 33 meaningful clusters from 358 observations, more reasonable cut points would be $\hat{k} = 4$ or $\hat{k} = 5$. While the performance of Agglomerative HBC is already second only to Divisive HBC in the Dermatology dataset with $\hat{k} = 33$, its performance is improved at the revised cutpoints. For $\hat{k} = 4$, ARI = 0.4665, NMI = 0.5167, NVI = 0.6517 and for $\hat{k} = 5$, ARI = 0.4858, NMI = 0.5403, NVI = 0.6298.

4. Simulation study

A simulation study based on a factorial design was performed to further characterize the performance of A-HBC and D-HBC in comparison to other clustering methods. Fifty datasets were generated for each of the simulation scenarios resulting from all 12 possible combinations of the following attributes: total dataset size $(N) = 200, 500$; number of clusters $(k) = 2, 5, 10$; likelihood $= 5$ multivariate normal and 5 Bernoulli, 10 multivariate normal. Given N and k, the number of subjects in each cluster were simulated from a multinomial distribution with k categories each having a (1/k) assignment probability. Data points were generated according to the specified distribution, and multivariate normal data were centered and scaled as described in Section 3 prior to clustering. With the exception of spectral clustering, the methods for the comparisons of cluster analyses and the methods for the selection of the number of clusters are as described in Section 3. Spectral clustering approaches cluster analysis as a graph partition problem, focusing on connectivity over distance. While it is neither hierarchical nor model-based, it is another popular clustering method with generally good performance (Shi and Malik, 2000; Ng et al., 2001).

Table 2 and Table 3 present the results of the simulations in the form of the medians and interquartile ranges (IQR) for number of clusters selected ($\hat{k}$), ARI for the selected number of clusters ($ARI_{\hat{k}}$), and ARI for the true number of clusters (ARI_k). For each metric, the median(s) representing the best performing method(s) are listed in bold, rounded to the nearest tenth for $\hat{k}$ and the nearest hundredth for ARI_k and $ARI_{\hat{k}}$. Overall, A-HBC and D-HBC performed well, having the most top performances totalled over all metrics. Across all simulations, A-HBC and/or D-HBC were among the top performers for ARI_k. With smaller sample sizes, A-HBC and D-HBC had the strongest performances for smaller numbers of clusters (i.e. 2 or 5). With larger sample sizes, A-HBC tended to overestimate the number of clusters when many dichotomous variables were included. A-HBC demonstrated more precision in estimating the number of clusters than D-HBC. Across simulation studies, the other methods tested had similar total top performance counts with the exception of DIANA which had the worst performance. Other methods tested demonstrated a variety of strengths and weaknesses. For example, the combination of spectral clustering and the gap statistic seemed inadequate for determining the number of clusters of the types of datasets generated in these simulations when $k = 10$, often identifying only a single cluster. However, given k, spectral clustering performs similarly to other methods. When $k = 2$, spectral clustering performed particularly well. MBHAC, a method based on a Gaussian model, performed surprisingly well when provided with Bernoulli data, especially for known k. While providing k improved the median ARI for k-means and spectral clustering, occasionally median $ARI_k <$ median $ARI_{\hat{k}}$ for the hierarchical methods.

Table 2
Performance of clustering methods on simulated multivariate normal datasets, median (IQR) of 50 simulations each. Results presented as $\hat{k}$, $ARI_{\hat{k}}$, and ARI_k. For each metric, the median(s) representing the best performance are listed in bold.

Simulation	Method	$\hat{k}$	$ARI_{\hat{k}}$	ARI_k
10 MVN	A-HBC	**2.0** (2.0, 2.0)	0.94 (0.88, 0.98)	0.94 (0.88, 0.98)
$N = 200$	D-HBC	**2.0** (2.0, 2.0)	**1.0** (0.97, 1.0)	**1.0** (0.98, 1.0)
$k = 2$	Hclust	1.0 (1.0, 1.0)	0.0 (0.0, 0.0)	0.0 (0.0, 0.02)
	DIANA	**2.0** (2.0, 3.0)	0.72 (0.62, 0.83)	0.83 (0.77, 0.86)
	MBHAC	**2.0** (2.0, 2.0)	0.98 (0.89, 1.0)	0.98 (0.89, 1.0)
	k-means	**2.0** (2.0, 3.0)	0.76 (0.58, 0.83)	0.82 (0.77, 0.85)
	Spectral	**2.0** (2.0, 2.0)	0.96 (0.81, 0.98)	0.98 (0.96, 1.0)
10 MVN	A-HBC	3.0 (3.0, 3.0)	0.63 (0.61, 0.66)	0.97 (0.95, 0.99)
$N = 200$	D-HBC	**5.0** (5.0, 5.0)	**0.98** (0.95, 1.0)	**0.98** (0.96, 1.0)
$k = 5$	Hclust	**5.0** (4.0, 5.0)	0.83 (0.76, 0.98)	0.87 (0.78, 0.99)
	DIANA	4.0 (4.0, 6.0)	0.76 (0.68, 0.86)	0.70 (0.65, 0.92)
	MBHAC	8.0 (7.0, 9.0)	0.84 (0.80, 0.86)	0.96 (0.90, 0.99)
	k-means	7.0 (6.0, 7.0)	0.85 (0.81, 0.89)	0.93 (0.89, 0.95)
	Spectral	1.0 (1.0, 4.75)	0.0 (0.0, 0.67)	0.78 (0.73, 0.97)
10 MVN	A-HBC	4.0 (4.0, 4.0)	0.46 (0.45, 0.48)	**0.97** (0.95, 0.99)
$N = 200$	D-HBC	5.0 (4.0, 6.0)	0.57 (0.48, 0.64)	0.91 (0.89, 0.95)
$k = 10$	Hclust	**10.0** (9.0, 11.0)	**0.93** (0.87, 0.96)	0.89 (0.85, 0.95)
	DIANA	**10.0** (8.0, 11.0)	0.85 (0.75, 0.92)	0.84 (0.80, 0.88)
	MBHAC	9.0 (9.0, 9.0)	0.89 (0.85, 0.92)	**0.97** (0.94, 0.99)
	k-means	12.0 (11.0, 12.0)	**0.93** (0.90, 0.95)	0.95 (0.94, 0.97)
	Spectral	1.0 (1.0, 1.0)	0.0 (0.0, 0.0)	0.83 (0.76, 0.87)
10 MVN	A-HBC	**2.0** (2.0, 2.0)	0.95 (0.93, 0.97)	0.95 (0.93, 0.97)
$N = 500$	D-HBC	**2.0** (2.0, 2.0)	**0.98** (0.76, 0.99)	**0.99** (0.98, 0.99)
$k = 2$	Hclust	1.0 (1.0, 1.0)	0.0 (0.0, 0.0)	0.0 (0.0, 0.02)
	DIANA	3.0 (3.0, 3.0)	0.64 (0.60, 0.68)	0.82 (0.80, 0.86)
	MBHAC	**2.0** (2.0, 2.0)	**0.98** (0.97, 0.99)	0.98 (0.97, 0.99)
	k-means	3.0 (2.0, 3.0)	0.65 (0.62, 0.81)	0.82 (0.80, 0.84)
	Spectral	**2.0** (2.0, 2.0)	**0.98** (0.75, 0.99)	0.98 (0.98, 0.99)
10 MVN	A-HBC	**5.0** (5.0, 5.0)	**0.99** (0.98, 1.0)	**0.99** (0.98, 1.0)
$N = 500$	D-HBC	**5.0** (5.0, 5.0)	**0.99** (0.98, 1.0)	**0.99** (0.98, 1.0)
$k = 5$	Hclust	4.0 (3.0, 5.0)	0.78 (0.61, 0.93)	0.80 (0.77, 0.97)
	DIANA	6.0 (4.0, 7.0)	0.82 (0.69, 0.85)	0.68 (0.66, 0.71)
	MBHAC	**5.0** (5.0, 5.0)	0.96 (0.91, 0.99)	0.96 (0.91, 0.99)
	k-means	7.0 (7.0, 8.0)	0.84 (0.81, 0.85)	0.93 (0.91, 0.94)
	Spectral	1.0 (1.0, 4.0)	0.0 (0.0, 0.74)	0.81 (0.76, 0.97)
10 MVN	A-HBC	8.0 (8.0, 9.0)	0.82 (0.80, 0.87)	**0.99** (0.98, 1.0)
$N = 500$	D-HBC	8.0 (2.0, 10.0)	0.79 (0.16, 0.93)	0.97 (0.93, 0.98)
$k = 10$	Hclust	**9.0** (8.0, 10.0)	0.88 (0.82, 0.96)	0.88 (0.86, 0.96)
	DIANA	8.5 (8.0, 11.0)	0.81 (0.74, 0.90)	0.83 (0.81, 0.85)
	MBHAC	8.0 (8.0, 9.0)	0.83 (0.81, 0.88)	0.97 (0.96, 0.99)
	k-means	12.0 (11.0, 12.0)	**0.93** (0.91, 0.94)	0.96 (0.95, 0.97)
	Spectral	1.0 (1.0, 1.0)	0.0 (0.0, 0.0)	0.84 (0.83, 0.86)

5. Identifying Parkinson's disease subtypes

Parkinson's disease (PD) is the second most common neurodegenerative disease in the world, affecting 1% of the population above 60 years (Tysnes and Storstein, 2017). PD patients suffer from progressively disabling symptoms, and the development of new and disease-modifying treatments is limited by variable disease progression and the lack of a robust biomarker (Marek et al., 2011). There is marked heterogeneity in idiopathic PD that is not yet understood. Identifying subpopulations within PD can lead to better understanding of the disease course, rate of progression, and treatment susceptibility. In what follows, we present an implementation of our D-HBC algorithm, discovering subgroups in PD defined by potential biomarkers and a simple assessment at baseline. We have validated our clustering results on additional biomarkers and clinical measures recorded longitudinally.

Longitudinal data from recently diagnosed PD subjects were downloaded from the Parkinson's Progression Markers Initiative (PPMI) database on March 15, 2021. Variables to be used for clustering were the following measurements taken at baseline: $\leq$ 6 months vs. $>$ 6 months since diagnosis (duration), University of Pennsylvania Smell Identification Test (UPSIT) score, cerebrospinal fluid (CSF) amyloid beta (abeta), CSF total tau (tau), and serum neurofilament light chain (NfL). UPSIT assesses olfactory function, CSF abeta forms neurotoxic plaques, CSF tau is an axonal microtubule protein, and serum NfL is a biomarker of axonal damage. For UPSIT, higher scores are indicative of normal olfaction (normosmia) and lower scores represent reduced (microsmia) or even no (anosmia) olfaction (Doty et al., 1984). Studies have demonstrated an association

Table 3
Performance of clustering methods on simulated multivariate normal and Bernoulli datasets, median (IQR) of 50 simulations each. Results presented as $\hat{k}$, $ARI_{\hat{k}}$, and ARI_k. For each metric, the median(s) representing the best performance are listed in bold.

Simulation	Method	$\hat{k}$	$ARI_{\hat{k}}$	ARI_k
5 MVN 15 Bin	A-HBC	**2.0** (2.0, 3.0)	0.95 (0.74, 0.98)	**0.98** (0.96, 1.0)
$N = 200$	D-HBC	3.0 (2.0, 3.0)	0.75 (0.71, 0.96)	0.96 (0.94, 0.98)
$k = 2$	Hclust	**2.0** (2.0, 2.0)	0.87 (0.66, 0.96)	0.87 (0.66, 0.96)
	DIANA	4.0 (3.0, 4.0)	0.43 (0.39, 0.60)	0.79 (0.76, 0.83)
	MBHAC	6.0 (4.0, 7.0)	0.34 (0.30, 0.50)	0.92 (0.55, 0.98)
	k-means	5.0 (4.0, 7.0)	0.40 (0.32, 0.46)	0.77 (0.73, 0.81)
	Spectral	**2.0** (2.0, 2.75)	**0.97** (0.90, 1.0)	**0.98** (0.96, 1.0)
5 MVN 15 Bin	A-HBC	**5.0** (5.0, 5.0)	**0.99** (0.98, 1.0)	**0.99**(0.98, 1.0)
$N = 200$	D-HBC	**5.0** (5.0, 5.0)	**0.99** (0.97, 1.0)	**0.99** (0.97, 1.0)
$k = 5$	Hclust	**5.0** (4.0, 5.0)	0.96 (0.79, 0.99)	0.96 (0.79, 0.99)
	DIANA	**5.0** (4.0, 6.0)	0.86 (0.73, 0.92)	0.75 (0.70, 0.94)
	MBHAC	9.0 (9.0, 9.0)	0.74 (0.71, 0.77)	**0.99** (0.96, 1.0)
	k-means	6.0 (5.35, 6.75)	0.91 (0.87, 0.94)	0.97 (0.96, 0.99)
	Spectral	4.0 (3.0, 5.0)	0.75 (0.57, 0.80)	0.81 (0.76, 0.98)
5 MVN 15 Bin	A-HBC	**10.0** (10.0, 10.0)	**0.98** (0.96, 0.99)	**0.98** (0.96, 0.99)
$N = 200$	D-HBC	**10.0** (2.25, 10.0)	0.92 (0.22, 0.96)	0.90 (0.75, 0.95)
$k = 10$	Hclust	9.0 (8.0, 9.0)	0.87 (0.86, 0.90)	0.87 (0.86, 0.89)
	DIANA	7.0 (6.0, 9.0)	0.69 (0.60, 0.82)	0.83 (0.79, 0.86)
	MBHAC	9.0 (9.0, 9.0)	0.88 (0.85, 0.91)	0.92 (0.86, 0.97)
	k-means	**10.0** (10.0, 11.0)	0.96 (0.94, 0.99)	**0.98** (0.96, 0.99)
	Spectral	1.0 (1.0, 1.0)	0.0 (0.0, 0.0)	0.86 (0.78, 0.91)
5 MVN 15 Bin	A-HBC	8.0 (8.0, 9.0)	0.28 (0.26, 0.30)	**0.98** (0.97, 0.98)
$N = 500$	D-HBC	3.0 (2.0, 3.75)	0.75 (0.71, 0.97)	0.97 (0.94, 0.98)
$k = 2$	Hclust	**2.0** (1.0, 2.0)	0.88 (0.0, 0.93)	0.88 (0.0, 0.93)
	DIANA	4.0 (4.0, 5.0)	0.43 (0.32, 0.53)	0.95 (0.87, 0.98)
	MBHAC	5.0 (4.0, 7.0)	0.42 (0.32, 0.53)	0.95 (0.87, 0.98)
	k-means	6.0 (4.0, 7.0)	0.34 (0.29, 0.47)	0.76 (0.74, 0.79)
	Spectral	**2.0** (2.0, 4.0)	**0.98** (0.72, 0.98)	**0.98** (0.98, 0.99)
5 MVN 15 Bin	A-HBC	5.5 (5.0, 6.0)	0.95 (0.91, 0.99)	**0.99**(0.98, 1.0)
$N = 500$	D-HBC	**5.0** (5.0, 5.0)	**0.99** (0.98, 1.0)	**0.99** (0.99, 1.0)
$k = 5$	Hclust	**5.0** (4.0, 5.0)	0.97 (0.79, 0.98)	0.97 (0.78, 0.98)
	DIANA	6.0 (4.0, 6.0)	0.86 (0.73, 0.90)	0.70 (0.67, 0.71)
	MBHAC	9.0 (9.0, 9.0)	0.74 (0.72, 0.76)	**0.99** (0.98, 0.99)
	k-means	7.0 (6.0, 7.0)	0.87 (0.85, 0.92)	0.98 (0.97, 0.99)
	Spectral	4.0 (3.0, 5.0)	0.73 (0.51, 0.77)	0.79 (0.76, 0.99)
5 MVN 15 Bin	A-HBC	**10.0** (10.0, 10.0)	**0.98** (0.97, 0.99)	**0.98** (0.97, 0.99)
$N = 500$	D-HBC	**10.0** (4.0, 11.0)	0.94 (0.41, 0.96)	0.92 (0.78, 0.94)
$k = 10$	Hclust	8.0 (7.0, 9.0)	0.80 (0.71, 0.87)	0.87 (0.85, 0.88)
	DIANA	6.0 (6.0, 8.0)	0.60 (0.55, 0.76)	0.80 (0.76, 0.82)
	MBHAC	9.0 (9.0, 9.0)	0.87 (0.85, 0.90)	0.88 (0.85, 0.97)
	k-means	11.0 (10.0, 11.0)	0.96 (0.95, 0.97)	**0.98** (0.97, 0.98)
	Spectral	1.0 (1.0, 1.0)	0.0 (0.0, 0.0)	0.86 (0.78, 0.89)

between lower CSF abeta and Alzheimer disease and memory impairment (Alves et al., 2010). Higher levels of CSF tau, which is correlated with alpha synuclein (aggregates of alpha synuclein are a hallmark of PD and other synucleinopathies), have been shown to be associated with motor progression (Hall et al., 2015). NfL demonstrates similar correlations with alpha synuclein (Hall et al., 2015). The biomarkers described are correlated with age (de Wolf et al., 2020).

Duration was a Bernoulli variable, dichotomized based on the preceding cut-point. Skewed variables (abeta, tau, and NfL) were log transformed. To avoid age-driven clusters, all variables except for duration were transformed prior to clustering into their ordinary least squares residuals from a simple regression using age at baseline as the covariate; residuals were then standardized. That is, to avoid having clusters be nothing more than a reflection of age, we preprocessed the data to remove the effect of age on the clinical measures and biomarkers used to perform clustering. Only subjects with complete data of the clustering variables at baseline and without high hemoglobin in CSF (indicative of blood contamination) were included. The D-HBC algorithm was used on dichotomous disease duration and standardized OLS residuals of UPSIT, abeta, tau, and NfL to generate hierarchical clusters, and the existence of α_{root} was used to determine the number of clusters.

Starting with 243 subjects with completed data on clustering variables, D-HBC identified 3 clusters, summarized for the characteristics used for clustering and age in Table 4. All subjects in the purple cluster were diagnosed more than 6 months before enrollment in PPMI while all subjects in the green and yellow clusters were diagnosed within 6 months prior to enrollment. The green cluster had the lowest UPSIT scores and lowest amounts of CSF abeta and tau of any group.

Table 4
Baseline Features used for Clustering.

	Green (N=57)	Purple (N=84)	Yellow (N=102)	Total (N=243)	p-value[a]
Duration					< 0.001
≤ 6 months	57 (100%)	0 (0%)	102 (100%)	159 (65.4%)	
> 6 months	0 (0%)	84 (100%)	0 (0%)	84 (34.6%)	
UPSIT					< 0.001
Mean (SD)	18.70 (6.07)	22.18 (8.30)	24.43 (8.59)	22.31 (8.24)	
Range	9 - 33	1 - 40	6 - 39	1 - 40	
Abeta					< 0.001
Mean (SD)	554.1 (145.7)	876.6 (344.1)	1098.6 (332.6)	894.1 (369.7)	
Range	238.8 - 880.1	318.4 - 1887	572.6 - 2572.0	238.8 - 2572	
Total Tau					< 0.001
Mean (SD)	145.14 (62.25)	169.47 (54.27)	178.53 (47.61)	167.57 (54.99)	
Range	83.9 - 339.2	85.0 - 321.1	108.8 - 345.3	83.9 - 345.3	
NfL					0.424
Mean (SD)	12.51 (5.38)	12.46 (5.61)	11.74 (5.84)	12.17 (5.64)	
Range	4.02 - 26.2	2.76 - 28.0	1.80 - 41.2	1.80 - 41.2	
Age					0.090
Mean (SD)	63.22 (10.10)	61.62 (10.63)	59.71 (9.06)	61.19 (9.93)	
Range	40.83 - 79.89	33.72 - 82.98	33.50 - 84.88	33.50 - 84.88	

[a] P-values are based on one-way ANOVA for non-skewed continuous variables (UPSIT, age), rank-based one-way ANOVA for continuous skewed variables (abeta, tau, NfL), and chi-square test for categorical variables (duration).

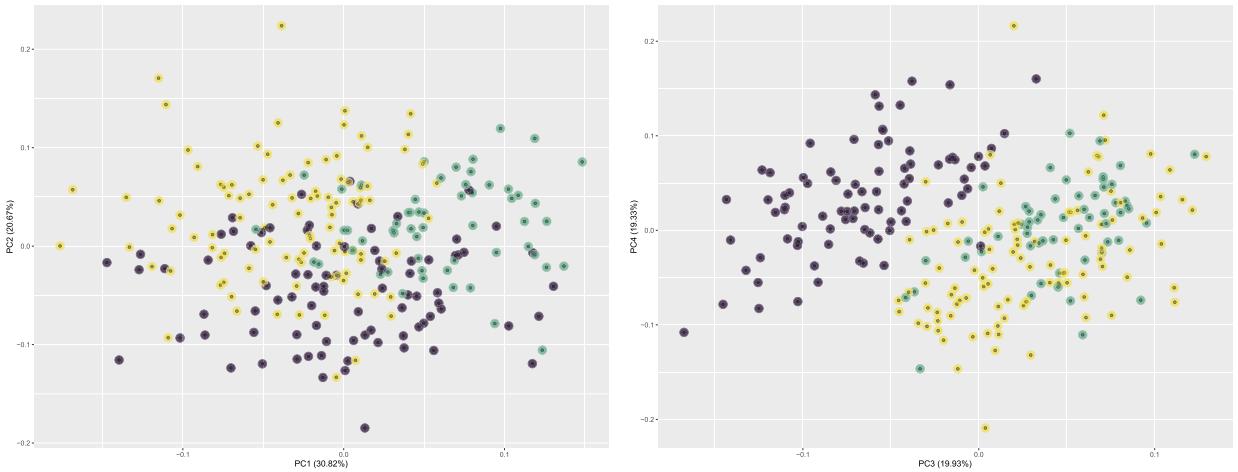

Fig. 3. First and second (left) and third and fourth (right) principal components of baseline features for clustering demonstrating separation of clusters. (Color online.)

As desired, there was no strong evidence of age differences between groups. Clusters are shown with observations plotted along principal components in Fig. 3.

Longitudinal outcomes representative of progression were used to validate clusters. Lowest putamen specific binding ratio (SBR) from DaTSCAN imaging is a measure of dopamine transporter levels in the brain. Montreal Cognitive Assessment (MoCA) is a screening test for mild cognitive impairment (<26) and dementia (<19). Letter number sequencing (LNS) is a measure of executive function and working memory, another indicator of cognitive function. The Movement Disorder Society Unified Parkinson's Disease Rating Scale (MDS-UPDRS) is used to monitor the motor and non-motor function of Parkinson's disease patients over time. MDS-UPDRS 1 is named Non-Motor Aspects of Experiences of Daily Living, MDS-UPDRS 2 is Motor Aspects of Experiences of Daily Living, and MDS-UPDRS 3 is Motor Examination performed by a clinician. These variables are distinct from the baseline variables used to perform clustering and their domains. Baseline variables used to perform clustering measured biological components in the CSF and serum (abeta, tau, NfL), time from diagnosis to baseline (duration), and olfactory function (UPSIT) whereas longitudinal outcomes assessed motor and non-motor function (MDS-UPDRS 1, 2, and 3), cognitive function (MoCA and LNS), and dopamine transporter levels in the brain (low putamen).

Linear mixed effects models with random intercepts were constructed for each longitudinal outcome of interest for the first 3 years since diagnosis per subject. To quantify the differences in progression, a likelihood ratio test was conducted for the interaction between cluster membership and time since diagnosis (i.e. difference in slope between clusters) for each progression outcome. We completed sensitivity analyses by additionally considering models that contained age at baseline.

In Fig. 4, plots of estimated mean and 95% confidence bands are featured for progression outcomes from time of diagnosis to year 3 by cluster. Note that the purple cluster does not have data until at least 6 months due to disease duration at enrollment. Generally, the trajectory of the green cluster appears to demonstrate a more rapid progression over time through a steeper slope. The green cluster appears to have a greater decrease in lowest putamen SBR relative to the other clusters, with weak evidence against no overall cluster by time interaction effect (χ^2 = 4.072, df = 2, p = 0.1305). Cognitive function worsens at a greater rate in the green cluster with strong evidence of an overall cluster effect over time (χ^2 = 8.848, df = 2, p = 0.0170). The green cluster appears to experience a greater decline in executive function and working memory than the other clusters with moderate evidence against no overall cluster by time interaction effect (χ^2 = 5.923, df = 2, p = 0.0517). Similarly the green cluster appears to have steeper worsening in each MDS-UPDRS score, with strong evidence against a null overall cluster effect over time in MDS-UPDRS 1 (χ^2 = 10.476, df = 2, p =0.0053) and weaker evidence in MDS-UPDRS 2 (χ^2 = 3.605, df = 2, p = 0.1649) or 3 (χ^2 = 3.2413, df = 2, p = 0.1978). Sensitivity analyses based on the inclusion of an age effect did not strongly alter results.

We also considered time to initiation of symptomatic therapy (i.e. a PD medication for functional disability, levodopa or dopamine agonist) as a measure of PD progression. A Kaplan-Meier plot of time to symptomatic therapy initiation by cluster is shown in Fig. 5. As a criterion of enrollment, no subjects had initiated symptomatic therapy. Within the first 3 years after enrollment, a smaller proportion of the purple cluster initiated symptomatic therapy than the other clusters. For the first year of enrollment, the green and yellow clusters demonstrate a similar rate of therapy initiation. From year 1 to year 3, a greater proportion of the green cluster initiated symptomatic therapy than the other clusters. The percent of subjects who have not started symptomatic therapy is consistently largest in the purple cluster and smallest in the green cluster at each 6 month interval from baseline to 3 years after enrollment.

To summarize, using D-HBC and only five baseline measurements, three clusters were identified: one (green) with more rapid progression than the others, one (purple) with a longer duration at baseline that appears to defer PD medication initiation for longer than the other groups, and one (yellow) following a more moderate course.

6. Conclusion

We have presented two hierarchical Bayesian clustering algorithms which exhibit strong performance when applied to benchmark datasets. Unlike non-hierarchical approaches, hierarchical clustering provides a data-driven approach to finding latent subgroups in data, yet importantly leaves space for scientific expertise to enter into the determination of the final partition studied. Unlike most other clustering algorithms, our algorithms are well equipped to deal with multiple and varied data types, represent the MAP at each merge/split given previous steps, and feature an integrated local method for determining a plausible number of clusters.

In addition to the performance on benchmark datasets, the D-HBC method provides additional strengths over agglomerative methods. Divisive methods require fewer splits than agglomerative methods would need merges to attain reasonable configurations (often the number of clusters is much closer to 1 than to N). As hierarchical methods are nested and greedy, fewer splits (or merges) lead to fewer chances to go wrong and carry forward an incorrectly sorted individual or individuals. Additionally, divisive methods can be much faster when a stopping rule is implemented. For instance, when we choose to harness a cutoff for α_{root} (e.g. existence) or perhaps $k << N$ (e.g. $k = 15$) as our stopping rule, then the algorithm would only proceed through a small proportion of splits, saving a large amount of time, while still obtaining the cluster configurations of interest. An agglomerative method cannot implement a similar stopping method because merges are completed beginning with the number of clusters equal to the number of subjects.

There are several limitations to our proposed approach. One limitation is computational complexity of our algorithms and computational speed of our code, especially the agglomerative algorithm for which early stopping rules cannot be applied. While we avoid the complexity involved with MCMC, determining the best merge or split for a full dendrogram is $\mathcal{O}(N^2)$ for A-HBC. We believe that future work could render the A-HBC more computationally efficient, and we have completed a step toward this goal. We note that calculating the change in posterior for all possible merges may be unnecessary and slow because unfavorable merges of singletons (i.e. cross cluster merges) are unlikely and such merges are more likely to occur after at least one of those singletons has joined a cluster (i.e. merge of at least three observations). All potential merges involving at least three observations will be calculated in the algorithm even when all possible singleton merges are not. We can initialize the change in posterior for merging singletons for what appear to be reasonable singleton merges by considering the k-nearest neighbors (kNN) of each observation given a reasonably selected distance metric. This reduces the number of possible merges to be calculated considerably; however, the computation complexity of the algorithm remains $\mathcal{O}(N^2)$.

As with all model-based methods, we expect performance to deteriorate when the model selected does not adequately fit the data. While our methods are flexible, allowing for posterior distribution to be specified, the methods proposed are parametric and may perform poorly if the method is misspecified. After clustering, it is recommended to visualize the distributions of variables within clusters to confirm that the distributions specified seem reasonable. Additionally, we

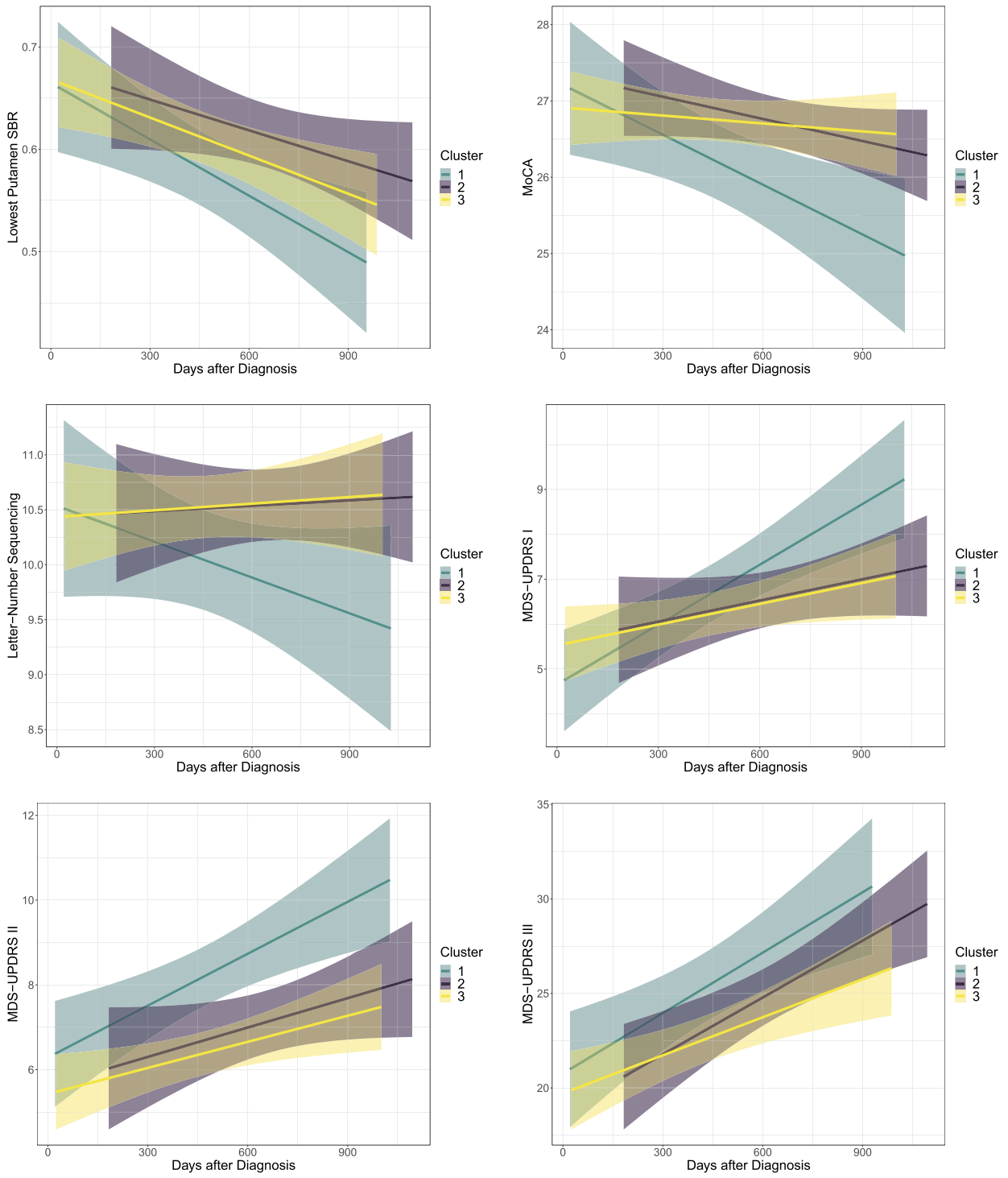

Fig. 4. Plots of estimated means and 95% confidence bands for progression outcomes from time of diagnosis to year 3 by cluster for lowest putamen SBR (top left box), MoCA (top right), LNS (middle left box), MDS-UPDRS 1 score (middle right box), MDS-UPDRS 2 score (bottom left box), and MDS-UPDRS 3 score (bottom right box). (Color online.)

often assume independence of data columns given cluster assignments. When variables for clustering are dependent on one another within clusters, this assumption is not met and our methods may not perform as well. Finally, if data are all categorical and can be perfectly separated, the α_{root} rule of thumb is likely to overestimate k in favor of a "perfectly" separated solution.

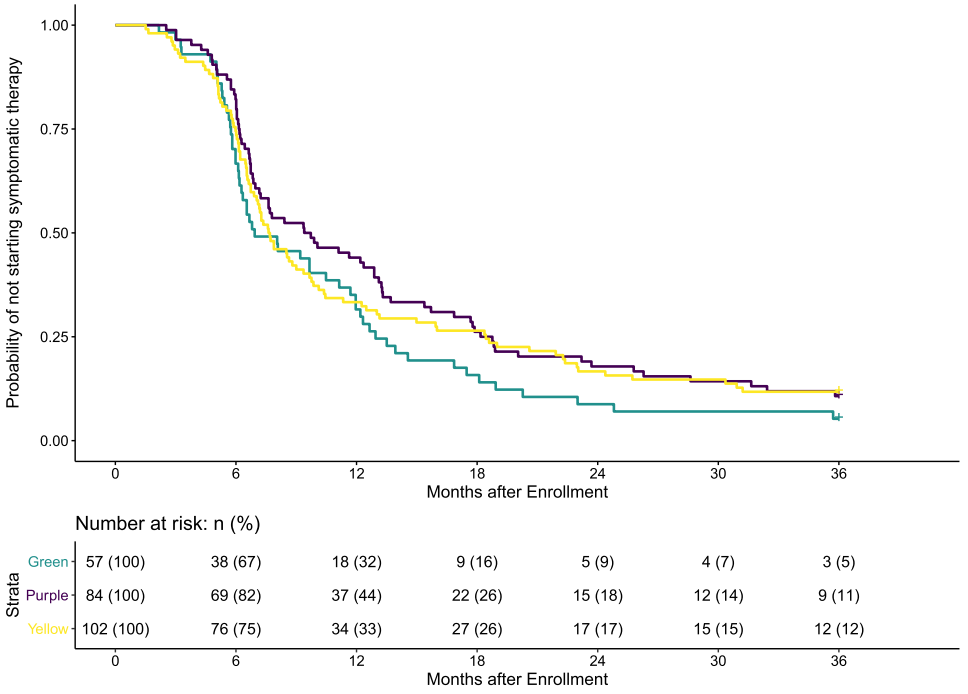

Fig. 5. Kaplan-Meier plot of time to symptomatic therapy initiation from enrollment by cluster. (Color online.)

Acknowledgements

Authors would like to thank the Clinical Trials Statistical and Data Management Center at the University of Iowa for their essential input about PPMI data and analyses. We would also like to thank an Associate Editor and two referees for their recommendations that led to several meaningful improvements that served to strengthen this paper.

Many of the benchmark datasets were from the UCI Machine Learning Repository [http://archive.ics.uci.edu/ml].

Data used in the preparation of this article were obtained from the Parkinson's Progression Markers Initiative (PPMI) database (www.ppmiinfo.org/data). For up-to-date information on the study, visit www.ppmiinfo.org. PPMI – a public-private partnership – is funded by the Michael J. Fox Foundation for Parkinson's Research and funding partners, including PPMI funding partners found at www.ppmiinfo.org/fundingpartners.

This work was supported in part by Iowa MSTP Training Grant (NIH T32GM007337).

Appendix A. Supplementary material

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.csda.2022.107566.

References

Alves, G., Brønnick, K., Aarsland, D., Blennow, K., Zetterberg, H., Ballard, C., Kurz, M.W., Andreasson, U., Tysnes, O.-B., Larsen, J.P., et al., 2010. Csf amyloid-β and tau proteins, and cognitive performance, in early and untreated Parkinson's disease: the Norwegian parkwest study. J. Neurol. Neurosurg. Psychiatry 81 (10), 1080–1086.

Anderson, E., 1936. The species problem in Iris. Ann. Missouri Bot. Garden 23 (3), 457–509.

Banfield, J.D., Raftery, A.E., 1993. Model-based Gaussian and non-Gaussian clustering. Biometrics, 803–821.

Bouveyron, C., Celeux, G., Murphy, T.B., Raftery, A.E., 2019. Model-Based Clustering and Classification for Data Science: with Applications in R, vol. 50. Cambridge University Press.

Celeux, G., Govaert, G., 1993. Comparison of the mixture and the classification maximum likelihood in cluster analysis. J. Stat. Comput. Simul. 47 (3–4), 127–146.

Celeux, G., Frühwirth-Schnatter, S., Robert, C.P., 2019. Model selection for mixture models–perspectives and strategies. In: Handbook of Mixture Analysis. Chapman and Hall/CRC, pp. 117–154.

Chen, X., Wang, H., Yan, D., 2018. Clustering of transcriptomic data for identification of cancer subtypes. In: FSDM, pp. 387–394.

de Wolf, F., Ghanbari, M., Licher, S., McRae-McKee, K., Gras, L., Weverling, G.J., Wermeling, P., Sedaghat, S., Ikram, M.K., Waziry, R., et al., 2020. Plasma tau, neurofilament light chain and amyloid-β levels and risk of dementia; a population-based cohort study. Brain 143 (4), 1220–1232.

DiMartini, A., Dew, M.A., Fitzgerald, M.G., Fontes, P., 2008. Clusters of alcohol use disorders diagnostic criteria and predictors of alcohol use after liver transplantation for alcoholic liver disease. Psychosomatics 49 (4), 332–340.

Doty, R.L., Shaman, P., Applebaum, S.L., Giberson, R., Siksorski, L., Rosenberg, L., 1984. Smell identification ability: changes with age. Science 226 (4681), 1441–1443.

Dua, D., Graff, C., 2017. UCI machine learning repository. http://archive.ics.uci.edu/ml.

Everitt, B.S., Landau, S., Leese, M., Stahl, D., 2011. Cluster Analysis, 5th edition. Wiley.
Filsinger, E.E., Faulkner, J.E., Warland, R.H., 1979. Empirical taxonomy of religious individuals: an investigation among college students. Sociol. Anal. 40 (2), 136–146.
Fisher, R.A., 1936. The use of multiple measurements in taxonomic problems. Ann. Eugen. 7 (2), 179–188.
Fraley, C., Raftery, A.E., 1998. How many clusters? Which clustering method? Answers via model-based cluster analysis. Comput. J. 41 (8), 578–588.
Fraley, C., Raftery, A.E., 2002. Model-based clustering, discriminant analysis, and density estimation. J. Am. Stat. Assoc. 97 (458), 611–631.
Frühwirth-Schnatter, S., Malsiner-Walli, G., Grün, B., 2020. Generalized mixtures of finite mixtures and telescoping sampling. arXiv preprint. arXiv:2005.09918.
Fuentes-García, R., Mena, R.H., Walker, S.G., 2019. Modal posterior clustering motivated by Hopfield's network. Comput. Stat. Data Anal. 137, 92–100.
Gallet, G.A., Pietrucci, F., 2013. Structural cluster analysis of chemical reactions in solution. J. Chem. Phys. 139 (7), 074101.
Hall, S., Surova, Y., Öhrfelt, A., Zetterberg, H., Lindqvist, D., Hansson, O., 2015. Csf biomarkers and clinical progression of Parkinson disease. Neurology 84 (1), 57–63.
Heard, N.A., Holmes, C.C., Stephens, D.A., 2006. A quantitative study of gene regulation involved in the immune response of anopheline mosquitoes: an application of Bayesian hierarchical clustering of curves. J. Am. Stat. Assoc. 101 (473), 18–29.
Heller, K.A., Ghahramani, Z., 2005. Bayesian hierarchical clustering. In: Proceedings of the 22nd International Conference on Machine Learning, pp. 297–304.
Huth, R., Beck, C., Philipp, A., Demuzere, M., Ustrnul, Z., Cahynová, M., Kyselỳ, J., Tveito, O.E., 2008. Classifications of atmospheric circulation patterns: recent advances and applications. Ann. N.Y. Acad. Sci. 1146 (1), 105–152.
Iwayama, M., Tokunaga, T., 1995. Hierarchical Bayesian clustering for automatic text classification. In: Proceedings of the 14th International Joint Conference on Artificial Intelligence, vol. 2, pp. 1322–1327.
Kaufman, L., Rousseeuw, P.J., 2009. Finding Groups in Data: An Introduction to Cluster Analysis, vol. 344. John Wiley & Sons.
Knox, D.B., Lanspa, M.J., Kuttler, K.G., Brewer, S.C., Brown, S.M., 2015. Phenotypic clusters within sepsis-associated multiple organ dysfunction syndrome. Intensive Care Med. 41 (5), 814–822.
Lo, K., Hahne, F., Brinkman, R.R., Gottardo, R., 2009. flowclust: a bioconductor package for automated gating of flow cytometry data. BMC Bioinform. 10 (1), 1–8.
Malsiner-Walli, G., Frühwirth-Schnatter, S., Grün, B., 2017. Identifying mixtures of mixtures using Bayesian estimation. J. Comput. Graph. Stat. 26 (2), 285–295.
Marek, K., Jennings, D., Lasch, S., Siderowf, A., Tanner, C., Simuni, T., Coffey, C., Kieburtz, K., Flagg, E., Chowdhury, S., et al., 2011. The Parkinson progression marker initiative (ppmi). Prog. Neurobiol. 95 (4), 629–635.
Medvedovic, M., Yeung, K.Y., Bumgarner, R.E., 2004. Bayesian mixture model based clustering of replicated microarray data. Bioinformatics 20 (8), 1222–1232.
Meilă, M., 2007. Comparing clusterings—an information based distance. J. Multivar. Anal. 98 (5), 873–895.
Miller, J.W., Harrison, M.T., 2018. Mixture models with a prior on the number of components. J. Am. Stat. Assoc. 113 (521), 340–356.
Milligan, G.W., Cooper, M.C., 1985. An examination of procedures for determining the number of clusters in a data set. Psychometrika 50 (2), 159–179.
Müllner, D., 2013. fastcluster: fast hierarchical, agglomerative clustering routines for r and python. J. Stat. Softw. 53 (1), 1–18.
Ng, A., Jordan, M., Weiss, Y., 2001. On spectral clustering: analysis and an algorithm. Adv. Neural Inf. Process. Syst. 14.
Pikoula, M., Quint, J.K., Nissen, F., Hemingway, H., Smeeth, L., Denaxas, S., 2019. Identifying clinically important copd sub-types using data-driven approaches in primary care population based electronic health records. BMC Med. Inform. Decis. Mak. 19 (1), 1–14.
Qian, Y., Wei, C., Eun-Hyung Lee, F., Campbell, J., Halliley, J., Lee, J.A., Cai, J., Kong, Y.M., Sadat, E., Thomson, E., et al., 2010. Elucidation of seventeen human peripheral blood b-cell subsets and quantification of the tetanus response using a density-based method for the automated identification of cell populations in multidimensional flow cytometry data. Cytometry. Part B, Clin. Cytometry 78 (S1), S69–S82.
R Core Team, 2020. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria. https://www.R-project.org/.
Reaven, G., Miller, R., 1979. An attempt to define the nature of chemical diabetes using a multidimensional analysis. Diabetologia 16 (1), 17–24.
Rossi, P., 2014. Bayesian Non- and Semi-Parametric Methods and Applications. Princeton University Press.
Rousseau, J., Mengersen, K., 2011. Asymptotic behaviour of the posterior distribution in overfitted mixture models. J. R. Stat. Soc., Ser. B, Stat. Methodol. 73 (5), 689–710.
Selvan, A.N., Cole, L.M., Spackman, L., Naylor, S., Wright, C., 2017. Hierarchical cluster analysis to aid diagnostic image data visualization of ms and other medical imaging modalities. In: Imaging Mass Spectrometry. Springer, pp. 95–123.
Sharma, A., López, Y., Tsunoda, T., 2017. Divisive hierarchical maximum likelihood clustering. BMC Bioinform. 18 (16), 139–147.
Shi, J., Malik, J., 2000. Normalized cuts and image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 22 (8), 888–905.
Steele, R.J., Raftery, A.E., 2010. Performance of Bayesian model selection criteria for Gaussian mixture models. Front. Stat. Decis. Mak. Bayesian Anal. 2, 113–130.
Sweeney, T.E., Azad, T.D., Donato, M., Haynes, W.A., Perumal, T.M., Henao, R., Bermejo-Martin, J.F., Almansa, R., Tamayo, E., Howrylak, J.A., et al., 2018. Unsupervised analysis of transcriptomics in bacterial sepsis across multiple datasets reveals three robust clusters. Crit. Care Med. 46 (6), 915.
Tibshirani, R., Walther, G., Hastie, T., 2001. Estimating the number of clusters in a data set via the gap statistic. J. R. Stat. Soc., Ser. B, Stat. Methodol. 63 (2), 411–423.
Tysnes, O.-B., Storstein, A., 2017. Epidemiology of Parkinson's disease. J. Neural Transm. 124 (8), 901–905.
Ultsch, A., 2005. Clustering with som: U*c. In: Proceedings of the Workshop on Self-Organizing Maps.
Vaithyanathan, S., Dom, B.E., 2013. Model-based hierarchical clustering. arXiv preprint. arXiv:1301.3899.
Van Havre, Z., White, N., Rousseau, J., Mengersen, K., 2015. Overfitting Bayesian mixture models with an unknown number of components. PLoS ONE 10 (7), e0131739.
Vianney Kinani, J.M., Rosales Silva, A.J., Gallegos Funes, F., Mújica Vargas, D., Ramos Díaz, E., Arellano, A., 2017. Medical imaging lesion detection based on unified gravitational fuzzy clustering. J. Healthc. Eng. 2017.
Wang, X., Markowetz, F., Felipe De Sousa, E.M., Medema, J.P., Vermeulen, L., 2013. Dissecting cancer heterogeneity–an unsupervised classification approach. Int. J. Biochem. Cell Biol. 45 (11), 2574–2579.