A semiparametric method for clustering mixed data

Alex Foss¹ · Marianthi Markatou¹ · Bonnie Ray² ·
Aliza Heching²

Received: 9 March 2015 / Accepted: 27 June 2016 / Published online: 15 July 2016
© The Author(s) 2016

Abstract Despite the existence of a large number of clustering algorithms, clustering remains a challenging problem. As large datasets become increasingly common in a number of different domains, it is often the case that clustering algorithms must be applied to heterogeneous sets of variables, creating an acute need for robust and scalable clustering methods for mixed continuous and categorical scale data. We show that current clustering methods for mixed-type data are generally unable to equitably balance the contribution of continuous and categorical variables without strong parametric assumptions. We develop KAMILA (KAy-means for MIxed LARge data), a clustering method that addresses this fundamental problem directly. We study theoretical aspects of our method and demonstrate its effectiveness in a series of Monte Carlo simulation studies and a set of real-world applications.

Keywords Clustering · Unsupervised learning · Mixed data · k -means · Finite mixture models · Big data

1 Introduction

Data sets analyzed in real applications are often comprised of mixed continuous and categorical variables, particularly when they consist of data merged from different sources. This is the case across a diverse range of areas including health care (electronic health records containing continuous blood chemistry measurements and categorical diagnostic codes), technical service centers (call center records containing service time and one or more problem categories), and marketing (customer records including gender, race, income, age, etc.). Additionally, the

Editor: Xiaoli Fern.

✉ Marianthi Markatou
markatou@buffalo.edu

¹ Department of Biostatistics, University at Buffalo, Buffalo, NY, USA

² IBM TJ Watson Research Center, Yorktown Heights, NY, USA

increasing prevalence of so called “big data” exacerbates the issue, as large data sets are commonly characterized by a mix of continuous and categorical variables (Fan et al. 2014).

A common approach to analyzing large data sets is to begin with clustering. Clustering identifies both the number of groups in the data as well as the attributes of such groups. The primary focus in the literature has been on clustering data sets that are comprised of a single type, that is, either all variables are continuous or all variables are categorical. As such, analysts working with data sets containing a mix of continuous and categorical-valued data will typically often convert the data set to a single data type by either coding the categorical variables as numbers and applying methods designed for continuous variables to achieve their clustering objective or converting the continuous variables into categorical variables via interval-based bucketing. See Dougherty et al. (1995), Ichino and Yaguchi (1994) for examples. Clustering methods that are explicitly designed to address mixed data types have received less attention in the literature and are reviewed in Sect. 2.1.

In this paper we first investigate the performance of existing clustering methods for mixed-type data. We find that existing methods are generally unable to equitably balance the contribution of continuous and categorical variables without strong parametric assumptions, a problem which we address through the development of a novel method, KAMILA.

The KAMILA (KAY-means for MIXed LARge data sets) algorithm is an advance over existing methods in four ways: first, variables (i.e. interval scale, and nominal or categorical scale) are used in their original measurement scale and hence are not transformed to either all interval or all categorical, avoiding a loss of information. Second, it ensures an equitable impact of continuous and categorical variables. Third, it avoids overly restrictive parametric assumptions, generalizing the form of the clusters to a broad class of elliptical distributions. Finally, it does not require the user to specify variable weights or use coding schemes.

The remainder of the paper is organized as follows. Section 2 provides additional background on the clustering problem of interest, and describes prior work in this area. Section 3 describes our novel KAMILA clustering method, and Sect. 4 presents the results of simulation studies investigating the performance of the new approach in clustering problems of varying difficulty, and in the presence of non-normal data. Section 5 presents analyses of real-world benchmark data sets comparing KAMILA to similar clustering methods. Section 6 illustrates the new approach for a real-world application, that of clustering client records associated with IT service requests. Section 7 concludes and discusses potential avenues for further research.

2 Background and literature review

2.1 Existing techniques for clustering mixed data

One of the more common approaches for clustering mixed-type data involves converting the data set to a single data type, and applying standard distance measures to the transformed data. Dummy coding all categorical variables is one example of such an approach. Dummy coding increases the dimensionality of the data set, which can be problematic when the number of categorical variables and associated categorical levels increase with the size of the data. Further, any semantic similarity that may have been observable in the original data set is lost in the transformed data set. Perhaps most importantly, coding strategies involve a non-trivial choice of numbers or weights that must be used to represent categorical levels. The difficulty of choosing these numbers is illustrated via a small simulation study, discussed in Sect. 2.2.

The coding strategy introduced by Hennig and Liao (2013, Section 6.2) is one example of such dummy coding strategies. It involves selecting values that control the expected contribution of categorical variables in relation to the quantity $E(X_1 - X_2)^2 = 2$, where X_i denotes independent and identically distributed observations of a continuous variable standardized to unit variance. Rather than set the expected contribution of the categorical variables equal to this quantity, however, Hennig and Liao (2013) set the expected contribution to half of this quantity, based on a concern that the gaps between the coded categorical dummy variables would unduly influence the resulting clusters. Hennig and Liao (2013) justify this claim based on results of a Monte Carlo study with 50 simulations per condition, in which they inspect the performance of the k -medoids algorithm CLARA (Kaufman and Rousseeuw 1990) using two weighting schemes. We have found that k -medoids algorithms can be unduly influenced by dummy coded categorical variables in this way, but also by hybrid distance metrics such as Gower's distance (1971). We have not observed the same phenomena with k -means; compare the performance of k -means and PAM in Fig. 1 for higher weights on the categorical variables. This discrepancy appears to be, at least in part, due to the fact that k -medoids requires the centroid to be an observed data point, while the means in k -means can smoothly interpolate between the observed categorical levels in the space defined by their associated dummy variables.

Fig. 1 Performance of KAMILA versus weighted k -means with dummy coding (solid line) versus PAM with Gower's distance (dashed line); one continuous and one categorical variable. Continuous and categorical overlap are either 1 or 30%. Various dummy coding strategies 0– c are used for the k -means algorithm, where c is depicted as a plotting character. For Gower's distance, the continuous variable is assigned a weight of 1.0, and the categorical variable is assigned the weight depicted as a plotting character. The performance of KAMILA is depicted using the letter K as plotting character. The x-axis indicates performance (Monte Carlo mean ARI; higher is better) when the continuous variable has 1% overlap and the categorical variable has 30%; the y-axis indicates performance when the overlap levels are reversed

As we demonstrate in the simulations and data analysis examples below, the method of Hennig and Liao (2013) performs well in many situations. One potential drawback to Hennig & Liao weighting arises due to the fact that it amounts to choosing fixed weights based only on the number of categorical levels in each variable in question. We show in Sect. 2.2 an example in which performance of any fixed weighting scheme (including but not limited to Hennig & Liao weighting) will fluctuate based on the number and/or quality of variables in the data set. We see examples of this in simulation A below (Table 4; Fig. 5), in which k -means with Hennig & Liao weighting is outperformed by KAMILA due to the fact that the latter method is able to adaptively adjust the relative contribution of continuous vs categorical variables based on the number and quality of variables. The difficulty inherent in choosing a fixed weighting scheme is further illustrated in Sect. 5, in which a naive weighting scheme is directly compared to Hennig & Liao weighting. In one example (Table 10) Hennig & Liao weighting improves the performance of k -means relative to the naive strategy, in the second it yields identical performance (Table 11), and in the third it decreases performance (Table 12).

An alternative approach is to use distance measures developed specifically for mixed data sets, e.g., Gower's distance (1971) defined as follows. Consider measuring the distance between p -dimensional vectors $\mathbf{x}$ and $\mathbf{y}$. If x_j and y_j are continuous, let $f_j(x_j, y_j)$ denote $|x_j - y_j|/r_j$, where r_j is the sample range across all observations of the j th variable. If x_j and y_j are categorical, let $f_j(x_j, y_j)$ denote the matching distance, that is, 1 if the observed levels are different and 0 if the levels match. Gower's distance between $\mathbf{x}$ and $\mathbf{y}$ is then given by

$$d_G(\mathbf{x}, \mathbf{y}) = \frac{\sum_{j=1}^p w_j f_j(x_j, y_j)}{\sum_{j=1}^p w_j} \quad (1)$$

where w_j is a user specified weight applied to data element j . Selection of the weights in Gower's distance suffers from the same problem described for numerical coding techniques, i.e. there is no clear way to choose the weights; usually, the default choice of setting all weights to one is used, although this default is often less effective than other weighting choices. For example, in Sect. 2.2 Gower's distance with weights of one achieves reasonable balance between continuous and categorical variables in a synthetic data set; however, simply including one additional variable in the data set changes this balance entirely, yielding a scenario in which weights of approximately 1.0 and 0.4 achieve balance between continuous and categorical variables, respectively (see Tables 1, 2, and associated discussion).

One of the most common clustering methods, used in applications across many different fields, is the k -means method (Forgy 1965; Hartigan and Wong 1979; Lloyd 1982; MacQueen 1967), which is based on a squared Euclidean distance measure between data points and the centroids of each cluster. With appropriate initialization, the k -means algorithm has been found to be robust to various forms of continuous error perturbation (Milligan 1980). However, the fact that it is designed for continuous variables introduces difficulties in selecting an appropriate coding strategy for categorical variables.

Huang (1998) proposed the k -prototypes algorithm, a variant of the k -means algorithm that is based on the weighted combination of squared Euclidean distance for continuous variables and matching distance for categorical variables. The k -prototypes algorithm relies on a user-specified weighting factor that determines the relative contribution of continuous and categorical variables, not unlike what is required to use Gower's distance, and thus suffers from the same limitation. Although various weighting schemes have been proposed in the literature (e.g., DeSarbo et al. 1984; Gnanadesikan et al. 1995; Huang et al. 2005), most do not address the problem of mixed data. If they do, they fail to address the central challenge of

Table 1 Results of a simulation study illustrating weaknesses in three weighting strategies for mixed-type data

Clustering method	Handling of categorical vars	Categorical weights	ARI: Con 1 %, Cat 30 %	ARI: Con 30 %, Cat 1 %
KAMILA	KAMILA	NA	0.985	0.906
<i>k</i> -means	Dummy coding	0–0.50	0.984	0.538
<i>k</i> -means	Dummy coding	0–1.00	0.984	0.701
<i>k</i> -means	Dummy coding	0–1.25	0.974	0.812
<i>k</i> -means	Dummy coding	0–1.50	0.946	0.909
<i>k</i> -means	Dummy coding	0–1.75	0.866	0.963
<i>k</i> -means	Dummy coding	0–2.00	0.570	0.979
<i>k</i> -means	Dummy coding	0–2.50	0.490	0.981
<i>k</i> -means	Dummy coding	0–3.00	0.486	0.979
PAM	Dummy coding	0–0.50	0.983	0.556
PAM	Dummy coding	0–1.00	0.949	0.680
PAM	Dummy coding	0–1.25	0.867	0.711
PAM	Dummy coding	0–1.50	0.752	0.718
PAM	Dummy coding	0–1.75	0.710	0.714
PAM	Dummy coding	0–2.00	0.706	0.709
PAM	Dummy coding	0–2.50	0.698	0.699
PAM	Dummy coding	0–3.00	0.687	0.670
PAM	Dummy coding	0–6.00	0.595	0.572
PAM	Gower	0.10	0.982	0.594
PAM	Gower	0.20	0.970	0.663
PAM	Gower	0.30	0.939	0.714
PAM	Gower	0.40	0.889	0.720
PAM	Gower	0.50	0.723	0.721
PAM	Gower	1.00	0.703	0.701
PAM	Gower	1.25	0.703	0.688
PAM	Gower	1.50	0.701	0.670
PAM	Gower	6.00	0.617	0.549

All results shown are Monte Carlo mean Adjusted Rand index scores (ARI; higher is better) between the identified clusters and the true membership. The *k*-means Hartigan–Wong algorithm with dummy coding, PAM with dummy coding or Gower’s distance, and the KAMILA algorithm were used. The dummy coding schemes ranged from (0–0.5) to (0–3.0) for the categorical variable. Gower’s distance used the weight shown for the categorical variable, and a weight of 1.0 for the continuous. This table shows results for data sets with one continuous variable and one categorical variable. The overlap between clusters was manipulated separately for the continuous and categorical variables: either the continuous variable was more closely related to the true cluster structure (second to last column; 1 % overlap in the continuous variable, 30 % overlap in the categorical) or the categorical variable was more closely related to the true cluster structure (last column; overlap levels reversed)

effectively balancing the contribution of continuous and categorical variables (Ahmad and Dey 2007; Burnaby 1970; Chae et al. 2006; Friedman and Meulman 2004; Goodall 1966).

An exception is the method of Modha and Spangler (2003), which defines a weighted combination of squared Euclidean distance and cosine distance very similar to that of Huang (1998), with a single weight determining the relative contribution of continuous and cate-

Table 2 Results of a simulation study illustrating weaknesses in three weighting strategies for mixed-type data

Clustering method	Handling of categorical vars	Categorical weights	ARI: Con 1 %, Cat 30 %	ARI: Con 30 %, Cat 1 %
KAMILA	KAMILA	NA	0.989	0.988
<i>k</i> -means	Dummy coding	0–0.50	0.987	0.591
<i>k</i> -means	Dummy coding	0–1.00	0.988	0.884
<i>k</i> -means	Dummy coding	0–1.25	0.978	0.972
<i>k</i> -means	Dummy coding	0–1.50	0.955	0.991
<i>k</i> -means	Dummy coding	0–1.75	0.924	0.994
<i>k</i> -means	Dummy coding	0–2.00	0.907	0.994
<i>k</i> -means	Dummy coding	0–2.50	0.899	0.993
<i>k</i> -means	Dummy coding	0–3.00	0.851	0.993
PAM	Dummy coding	0–0.50	0.986	0.619
PAM	Dummy coding	0–1.00	0.937	0.813
PAM	Dummy coding	0–1.25	0.849	0.846
PAM	Dummy coding	0–1.50	0.741	0.853
PAM	Dummy coding	0–1.75	0.701	0.853
PAM	Dummy coding	0–2.00	0.697	0.853
PAM	Dummy coding	0–2.50	0.695	0.854
PAM	Dummy coding	0–3.00	0.694	0.854
PAM	Dummy coding	0–6.00	0.692	0.853
PAM	Gower	0.10	0.984	0.688
PAM	Gower	0.20	0.965	0.795
PAM	Gower	0.30	0.920	0.848
PAM	Gower	0.40	0.872	0.849
PAM	Gower	0.50	0.719	0.848
PAM	Gower	1.00	0.674	0.840
PAM	Gower	1.25	0.603	0.839
PAM	Gower	1.50	0.569	0.838
PAM	Gower	6.00	0.474	0.826

All results shown are Monte Carlo mean Adjusted Rand index scores (ARI; higher is better). The *k*-means Hartigan–Wong algorithm with dummy coding, PAM with dummy coding or Gower’s distance, and the KAMILA algorithm were used. The dummy coding schemes ranged from (0–0.5) to (0–3.0) for the categorical variables. Gower’s distance used the weight shown for the categorical variables, and a weight of 1.0 for the continuous. This table shows results for data sets with one continuous variable and two categorical variables

gorical variables. In contrast to Huang (1998), however, the method adaptively selects, in an unsupervised fashion, the relative weight that simultaneously minimizes the within-cluster dispersion and maximizes the between-cluster dispersion for both the continuous and categorical variables. The weight is identified through a brute-force search over the range of possible scalar values it could take, and the within- to between-cluster dispersion ratio is calculated separately for continuous and categorical variables for each value tested; the weight that minimizes the product of the continuous and categorical dispersion ratio is selected. In theory, if all of the continuous variables show no cluster structure, their dispersion ratio will not change in the course of the brute-force search, and they will contribute little to the

weight search. If in addition to this the categorical variables show a strongly detectable cluster structure, then upweighting the categorical variables will result in overall clusters that are more cleanly separated as the deleterious influence of the continuous variables decreases. In this case, the Modha–Spangler procedure will upweight the categorical variables, since this leads to a smaller categorical dispersion ratio and unchanged continuous dispersion ratio (and vice-versa if the cluster structure is uniquely contained in the continuous variables).

The Modha–Spangler weighting reduces to a single weight that determines the relative contribution of continuous and categorical variables, and by design, it cannot downweight individual variables within the collection of categorical (or continuous) variables. For example, in a data set containing categorical variables with varying strength of association with the underlying cluster structure, it must up- or down-weight all categorical variables uniformly. A further drawback involves cases in which the number of combinations of categorical levels (e.g. two ternary variables have 3×3 level combinations) is equal to the number of specified clusters; in this case, the degenerate solution of assigning each combination of categorical levels to its own cluster results in “perfect” separation between the clusters and thus a dispersion ratio of zero. In this case, Modha–Spangler will always select this degenerate solution and ignore the continuous variables. An example of this limitation is given in simulation B.

Model-based or statistical approaches to clustering mixed-type data typically assume the observations follow a normal-multinomial finite mixture model (Browne and McNicholas 2012; Everitt 1988; Fraley and Raftery 2002; Hunt and Jorgensen 2011; Lawrence and Krzanowski 1996). When parametric assumptions are met, model-based methods generally perform quite well and are able to effectively use both continuous and categorical variables, while avoiding undue vulnerability to variables with weak association with the identified clusters. Normal-multinomial mixture models can be extended using the location model (Krzanowski 1993; Olkin and Tate 1961), which allows a distinct distribution for the continuous variables for each unique combination of categorical levels. While this accounts for any possible dependence structure between continuous and categorical variables, it becomes infeasible when the number of categorical variables or number of levels within each categorical variable is large. As shown below in simulation A, when parametric assumptions are violated, performance of model-based methods often suffer. For exclusively continuous data, kernel density (KD) methods allow these parametric assumptions to be relaxed (Azzalini and Torelli 2007; Comaniciu and Meer 2002; Esther et al. 1996; Li et al. 2007), however KD methods incur a prohibitively large computational cost with a large number of continuous variables, along with other well-documented problems associated with high-dimensional KD estimation (Scott 1992, Chapter 7). Additionally, with the possible exception of one proposal based on Gower’s distance (Azzalini and Torelli 2007; Azzalini and Menardi 2014), KD based clustering methods have not been developed for categorical or mixed-type data. The method of Azzalini and Torelli (2007) was designed for continuous data, although the authors suggest a technique for adapting it to mixed-type data using a mixed-type distance metric in Azzalini and Menardi (2014). As Azzalini and Menardi (2014) point out, their method is agnostic with regard to the particular distance metric used, thus leaving the central problem of constructing distance metrics for mixed-type data unsolved. The authors suggest using Gower’s distance (see Eq. 1 in the current paper and associated discussion), which introduces the difficult problem of selecting weights for mixed-type data; the problem of weight selection is discussed in the current section and in Sect. 2.2. Solving the weighting problem is beyond the scope of Azzalini and Menardi (2014).

In the current paper, we develop a semi-parametric generalization of k -means clustering that balances the contribution of the continuous and categorical variables without any need

to specify weights. We refer to this method as KAMILA (KAY-means for MIXed LARge data sets).

2.2 Example: problems constructing a distance measure for mixed-type data

In this section we illustrate the challenge in constructing a distance measure that appropriately combines continuous and categorical data. We consider a method to appropriately combine continuous and categorical data if its performance is approximately equal or superior to existing methods in a given set of conditions. Our intent is to draw attention to methods with obvious deficiencies relative to alternative clustering techniques.

Most distance measures involve some choice of weights that dictate the relative contribution of each data type (either explicitly as in Gower's distance, or implicitly as in selecting the numbers to use for dummy coding). Here we illustrate the challenge in selecting appropriate weights in the context of Euclidean distance and Gower's distance. These challenges motivated the development of the KAMILA clustering algorithm. First, we present theoretical calculations that show that even in very simple cases, using z-normalized continuous variables and dummy-coded categorical variables with the Euclidean distance metric results in a procedure that is dominated by the continuous variables at the expense of the categorical variables. Second, we present simulation results suggesting that no weighting scheme can overcome this imbalance between continuous and categorical variables in any general way.

Consider sampling from a mixed-type bivariate data vector (V, W) with two underlying populations such that

$$V = \begin{cases} Y_1 & \text{with probability } \pi \in [0, 1] \\ Y_2 & \text{with probability } 1 - \pi \end{cases}$$

where Y_1, Y_2 follow arbitrary continuous distributions with means μ and 0 respectively, and corresponding variances σ_1^2 and σ_2^2 without loss of generality. We assume that the standard practice of z-normalizing the continuous variables is used. Specifically, instead of V , the transformed variable $(V - \eta)/\nu$ is used, where η and ν are assumed to be known constants, $\eta = E[V]$ and $\nu^2 = \text{Var}[V] = \pi\sigma_1^2 + (1 - \pi)\sigma_2^2 + \pi(1 - \pi)\mu^2$.

Let W represent a 0–1 dummy coded mixture of two Bernoulli random variables:

$$W = \begin{cases} B_1 & \text{with probability } \pi \\ B_2 & \text{with probability } 1 - \pi \end{cases}$$

with $B_1 \sim \text{Bern}(p_1)$ and $B_2 \sim \text{Bern}(p_2)$. The squared Euclidean distance between populations 1 and 2 is

$$\left(\frac{Y_1 - \eta}{\nu} - \frac{Y_2 - \eta}{\nu} \right)^2 + (B_1 - B_2)^2,$$

where the first term corresponds to the continuous contribution and the second term is the categorical contribution.

We will make use of the following proposition, with proof provided in the Appendix.

Proposition 1 *Let (V, W) be a mixed-type bivariate data vector defined as above. Then the expectation of the continuous contribution to the distance between population 1 and 2 is*

$$E \left[\left(\frac{Y_1 - Y_2}{\nu} \right)^2 \right] = \phi, \quad \text{with } \begin{cases} \phi > 1, & \sigma_1 \neq \sigma_2 \\ \phi > 2, & \sigma_1 = \sigma_2. \end{cases}$$

Also, since $|B_1 - B_2|$ can only be 0 or 1, the expectation of the categorical contribution to the distance between population 1 and 2 is

$$0 < E[(B_1 - B_2)^2] < 1 \quad \forall p_1, p_2 \in (0, 1).$$

Proof See Appendix.

Thus, even this seemingly straightforward approach, based on the Euclidean distance, leads to unbalanced treatment of the continuous and categorical variables. The increased contribution of continuous variables may be ideal in certain restricted cases (e.g., when the continuous variables are more useful than the categorical variables for purposes of identifying cluster structure), but this is not a generally valid assumption.

To adjust for this unbalanced treatment of continuous and categorical variables, one may consider choosing a set of variable weights to modify the contribution of continuous and categorical variables (e.g. [Gower 1971](#); [Hennig and Liao 2013](#); [Huang 1998](#)). However, choosing an appropriate set of weights is a difficult task, and in many cases impossibly difficult. Consider the ubiquitous scenario in which the observed variables do not each have equally strong relationships with the underlying cluster structure (and it would be rare indeed for all variables to be identical in this regard); for example, consider a data set in which the clusters have little overlap along variable 1 (i.e. clusters are well separated), but show large overlap along variable 2 (poor separation). If the overlap of each variable is known, we can simply choose variable weights that upweight those with less overlap. However, overlap is rarely known ahead of time in a cluster analysis; while in a simple two variable data set this might be inspected manually, it is not realistic or even possible to do this in larger multivariate data sets. If weights are chosen incorrectly, the problem of differential overlaps can remain unaddressed or even exacerbated. Fortunately, there exist techniques, such as mixture models and our currently proposed KAMILA, that can handle the differential overlap problem without requiring user-specified weights; we illustrate this in the following example and in simulation A.

To illustrate the challenge of differential overlap, we conducted a small scale simulation, the results of which are presented in Tables 1 and 2 and Figs. 1 and 2. The tables and figures show the adjusted Rand index (ARI; [Hubert and Arabie 1985](#)) score for the k -means algorithm with dummy-coded categorical variables, the k -medoids algorithm PAM (Partitioning Around Medoids; [Kaufman and Rousseeuw 1990](#)) with dummy-coded categorical variables, and PAM with Gower's distance. A broad range of dummy coding schemes 0– c were used for the categorical variables in the k -means algorithm, where c ranges from 0.5 to 3.0. For Gower's distance, the continuous variable was assigned weight 1.0, and the categorical variables were assigned weights ranging from 0.1 to 6.0. ARI measures the agreement between two partitions of the same set (in this case the true and estimated cluster memberships), with 0 indicating chance-level performance and 1 indicating perfect agreement. Two- and three-variable mixed-type datasets were generated with two underlying clusters of varying separation. One or two categorical variables were generated from the same distribution, and one continuous variable was used. We used a Gaussian mixture for the generation of continuous variables and a multinomial mixture (four categorical levels) for the generation of categorical variables. We quantified the separation between clusters using the concept of overlap in distributions, which consists of calculating the area of overlap between the cluster densities in a mixture distribution (for details, see Sect. 4.2). We used continuous and categorical overlap levels of 1 and 30 %. For all cases considered, sample size was 500, with 500 Monte Carlo samples drawn for each cell. Monte Carlo error is less than 0.01 in all cases, and was calculated as $\sigma_{mc}/\sqrt{M}$, where σ_{mc} is the standard deviation of the M Monte Carlo samples.

Fig. 2 Performance of KAMILA versus weighted k -means with dummy coding (*solid line*) versus PAM with Gower's distance (*dashed line*); one continuous and two categorical variables. Continuous and categorical overlap are either 1 or 30 %. Various dummy coding strategies 0– c are used for the k -means algorithm, where c is depicted as a plotting character. For Gower's distance, the continuous variable is assigned a weight of 1.0, and the categorical variables are assigned the weight depicted as a plotting character. The performance of KAMILA is depicted using the letter K as plotting character. The x-axis indicates performance (Monte Carlo mean ARI; higher is better) when the continuous variable has 1 % overlap and the categorical variables have 30 %; the y-axis indicates performance when the overlap levels are reversed

There is a clear trade-off in the choice of weights used for dummy coding with k -means. In both the two and three variable data sets, larger weights (e.g. 0–3 coding) perform poorly in conditions with high categorical overlap and low continuous overlap, and perform best in conditions with lower categorical overlap and high continuous overlap. This pattern is reversed for smaller weights. This is due to the fact that higher categorical weighting emphasizes the contribution of the categorical variables. The KAMILA algorithm, on the other hand, does not require any weights to be specified, and can adaptively adjust to the overlap levels, achieving a favorable performance regardless of the overlap level. For example, in the two variable data set, k -means with the smallest weights (0–0.5 and 0–1) perform comparably to KAMILA in the higher categorical overlap condition; however, these same weighting schemes perform very poorly in the higher continuous overlap condition relative to KAMILA, achieving ARI of 0.54 and 0.70 compared to KAMILA's 0.91. If the overlap levels in a data set are not known, KAMILA clearly appears to be a superior choice compared to a weighted k -means clustering. There is a similar trade-off in the performance of the PAM clustering method, although it appears to perform uniformly worse than k -means in this context, regardless of whether dummy coding or Gower's distance is used.

In addition to being dependent on the overlap levels, performance of the weighting strategies varies with the number of variables. For example, consider the weighting scheme that achieves the most balanced performance, in the sense that ARI in both overlap conditions are as close as possible to each other. In the three variable condition, k -means with 0–1.25

weighting performs equally well in both overlap conditions (ARI of 0.98 and 0.97), whereas in the two variable condition k -means with 0–1.25 weighting is quite unbalanced (ARI of 0.97 and 0.81); in the two variable condition weights between 0–1.5 and 0–1.75 appear to give the most balanced performance of the k -means algorithm. The KAMILA algorithm, on the other hand, achieves balanced performance in both the two and three variable conditions.

Even in this very simple example, we show that a weight selection strategy that does not depend on overlap levels and the number of variables (e.g. Hennig and Liao 2013, Section 6.2) does not recover the clusters as well as more sophisticated strategies that do make use of this information, such as KAMILA.

3 KAMILA clustering algorithm

The KAMILA clustering algorithm is a scalable version of k -means well suited to handle mixed-type data sets. It overcomes the challenges inherent in the various extant methods for clustering mixed continuous and categorical data, i.e., either they require strong parametric assumptions (e.g., the normal–multinomial mixture model), they are unable to minimize the contribution of individual variables (e.g. Modha–Spangler weighting), or they require an arbitrary choice of weights determining the relative contribution of continuous and categorical variables (e.g., dummy/simplex coding and Gower’s distance).

The KAMILA algorithm combines the best features of two of the most popular clustering algorithms, the k -means algorithm (Forgy 1965; Lloyd 1982) and Gaussian-multinomial mixture models (Hunt and Jorgensen 2011), both of which have been adapted successfully to very large data sets (Chu et al. 2006; Wolfe et al. 2008). Like k -means, KAMILA does not make strong parametric assumptions about the continuous variables, and yet KAMILA avoids the limitations of k -means described in Sect. 2.2. Like Gaussian-multinomial mixture models, KAMILA can successfully balance the contribution of continuous and categorical variables without specifying weights, but KAMILA is based on an appropriate density estimator computed from the data, effectively relaxing the Gaussian assumption.

3.1 Notation and definitions

Here, we denote random variables with capital letters, and manifestations of random variables with lower case letters. We denote vectors with boldfont, and scalars in plaintext.

Let $\mathbf{V}_1, \dots, \mathbf{V}_i, \dots, \mathbf{V}_N$ denote an independent and identically distributed (i.i.d.) sample of $P \times 1$ continuous random vectors following a mixture distribution with arbitrary spherical clusters with density h such that $\mathbf{V}_i = (V_{i1}, \dots, V_{ip}, \dots, V_{iP})^T$, with $\mathbf{V}_i \sim f_{\mathbf{V}}(\mathbf{v}) = \sum_{g=1}^G \pi_g h(\mathbf{v}; \boldsymbol{\mu}_g)$, where G is the number of clusters in the mixture, $\boldsymbol{\mu}_g$ is the $P \times 1$ centroid of the g th cluster of $\mathbf{V}_i$ and π_g is the prior probability of drawing an observation from the g th population. Let $\mathbf{W}_1, \dots, \mathbf{W}_i, \dots, \mathbf{W}_N$ denote an i.i.d. sample of $Q \times 1$ discrete random vectors, where each element is a mixture of multinomial random variables such that $\mathbf{W}_i = (W_{i1}, \dots, W_{iq}, \dots, W_{iQ})^T$, with $W_{iq} \in \{1, \dots, \ell, \dots, L_q\}$, and $\mathbf{W}_i \sim f_{\mathbf{W}}(\mathbf{w}) = \sum_{g=1}^G \pi_g \prod_{q=1}^Q m(w_q; \boldsymbol{\theta}_{gq})$, where $m(w; \boldsymbol{\theta}) = \prod_{\ell=1}^{L_q} \theta_{\ell}^{I\{w=\ell\}}$ denotes the multinomial probability mass function, $I\{\cdot\}$ denotes the indicator function, and $\boldsymbol{\theta}_{gq}$ denotes the $L_q \times 1$ parameter vector of the multinomial mass function corresponding to the q th random variable drawn from the g th cluster. We assume W_{iq} and $W_{iq'}$ are conditionally independent given population membership $\forall q \neq q'$ (a common assumption in finite mixture models Hunt and

Jorgensen 2011). Let $\mathbf{X}_1, \dots, \mathbf{X}_i, \dots, \mathbf{X}_N$ denote an i.i.d. sample from $\mathbf{X}_i = (\mathbf{V}_i^T, \mathbf{W}_i^T)^T$ with $\mathbf{V}_i$ conditionally independent of $\mathbf{W}_i$, given population membership.

In the general case, where categorical variables are not independent, we model them by supplanting them with a new categorical variable with a categorical level for every combination of levels in the dependent variables. For example, if W_{i1} and W_{i2} are not conditionally independent and have L_1 and L_2 categorical levels respectively, then they would be replaced by the variable W_i^* with $L_1 \times L_2$ levels, one for each combination of levels in the original variables. If categorical and continuous variables are not conditionally independent, then the location model (Krzanowski 1993; Olkin and Tate 1961) can be used, although see the discussion of the location model in Sect. 2.1. KAMILA can be modified to accommodate elliptical clusters; we discuss at the end of Sect. 3.3 below methods for extending KAMILA in this way, and illustrate one such implementation in simulation C. The decision to use KAMILA to identify spherical or elliptical clusters must be specified before the algorithm is run. As in other mixture modeling problems, this decision must be made based on a priori knowledge of the data and clustering goals, or through comparing the performance of the different models using, for example, measures of internal cluster validity. We avoid endorsing any particular measure of cluster validity as their appropriateness is entirely dependent on the particular problem at hand.

At iteration t of the algorithm, let $\hat{\mu}_g^{(t)}$ denote the estimator for the centroid of population g , and let $\hat{\theta}_{gq}^{(t)}$ denote the estimator for the parameters of the multinomial distribution corresponding to the q th discrete random variable drawn from population g .

3.2 Kernel density estimation

We seek a computationally efficient way to evaluate joint densities of multivariate spherical distributions. We proceed using kernel density (KD) estimates. However, for multivariate data, KD estimates suffer from the problems of unreasonable computation times for high-dimensional data and overfitting the observed sample, yielding density estimates for observed points that are too high and density estimates for points not used in the KD fitting that are too low (Scott 1992, Chapter 7).

The proposed solution is first derived for spherically distributed clusters, and later extended to elliptical clusters. Using special properties of these distributions, we can obtain KD estimates that are more accurate and faster to calculate than the standard multivariate approaches.

Note that we are not referring to data scattered across the surface of a sphere (e.g. Hall et al. 1987): we are interested in data with densities that are radially symmetric about a mean vector (Kelker 1970); that is, densities that only depend on the distance from the sample to the center of the distribution.

KAMILA depends upon a univariate KD estimation step for the continuous clusters. The densities of the continuous clusters are estimated using the transformation method, a general framework for estimating densities of variables that have been transformed by some known function (Bowman and Azzalini 1997, pp. 14–15). Briefly, for KD estimation of a random variable X , this method involves constructing the desired KD estimate as $\hat{f}(x) = \hat{g}(t(x))t'(x)$, where t is some differentiable function (e.g. $\log(x)$ or $\sqrt{x}$; in this case we use a continuous distance measure), g denotes the PDF of $t(X)$ with KD estimate $\hat{g}$, and $t'(x)$ denotes the derivative of t with respect to x .

We now make use of the following proposition.

Proposition 2 Let $\mathbf{V} = (V_1, V_2, \dots, V_p)^T$ be a random vector that follows a spherically symmetric distribution with center at the origin. Then

$$f_{\mathbf{V}}(\mathbf{v}) = \frac{f_R(r) \Gamma(\frac{p}{2} + 1)}{p r^{p-1} \pi^{p/2}},$$

where $r = \sqrt{\mathbf{v}^T \mathbf{v}}$, $r \in [0, \infty)$.

Proof See Appendix.

Under spherical cluster densities, we set the function t to be the Euclidean distance, and we obtain a density estimation technique for $\hat{f}_{\mathbf{V}}$ by replacing f_R with the *univariate* KD estimate $\hat{f}_R$, thus avoiding a potentially difficult multidimensional KD estimation problem.

3.3 Algorithm description

Pseudocode for the KAMILA procedure is provided in Algorithm 1. First, each $\hat{\mu}_{gp}^{(0)}$ is initialized with a random draw from a uniform distribution with bounds equal to the minimum and maximum of the p^{th} continuous variable. Each $\hat{\theta}_{gq}^{(0)}$ is initialized with a draw from a Dirichlet distribution (Kotz et al. 2004) with shape parameters all equal to one, i.e., a uniform draw from the simplex in $\mathbb{R}^{L_q}$.

The algorithm is initialized multiple times. For each initialization, the algorithm runs iteratively until a pre-specified maximum number of iterations is reached or until population membership is unchanged from the previous iteration, whichever occurs first. See the online resource, Section 3, for a discussion on selecting the number of initializations and the maximum number of iterations. Each iteration consists of two broad steps: a partition step and an estimation step.

Given a complete set of $\hat{\mu}_{gp}^{(t)}$ and $\hat{\theta}_{gq}^{(t)}$'s at the t^{th} iteration, the partition step assigns each of N observations to one of G groups. First, the Euclidean distance from observation i to each of the $\hat{\mu}_g^{(t)}$'s is calculated as

$$d_{ig}^{(t)} = \sqrt{\sum_{p=1}^P [\xi_p(v_{ip} - \hat{\mu}_{gp}^{(t)})]^2},$$

where ξ_p is an optional weight corresponding to variable p . Next, the minimum distance is calculated for the i^{th} observation as $r_i^{(t)} = \min_g(d_{ig}^{(t)})$. The KD of the minimum distances is constructed as

$$\hat{f}_R^{(t)}(r) = \frac{1}{N h^{(t)}} \sum_{\ell=1}^N k\left(\frac{r - r_{\ell}^{(t)}}{h^{(t)}}\right), \quad (2)$$

where $k(\cdot)$ is a kernel function and $h^{(t)}$ the corresponding bandwidth parameter at iteration t . The Gaussian kernel is currently used, with bandwidth $h = 0.9 A n^{-1/5}$, where $A = \min(\hat{\sigma}, \hat{q}/1.34)$, $\hat{\sigma}$ is the sample standard deviation, and $\hat{q}$ is the sample interquartile range (Silverman 1986, p. 48, equation 3.31). The function $\hat{f}_R^{(t)}$ is used to construct $\hat{f}_{\mathbf{V}}^{(t)}$ as shown in Sect. 3.2.

Assuming independence between the Q categorical variables within a given cluster g , we calculate the log probability of observing the i^{th} categorical vector given population membership as $\log(c_{ig}^{(t)}) = \sum_{q=1}^Q \xi_q \cdot \log(m(w_{iq}; \hat{\theta}_{gq}^{(t)}))$, where $m(\cdot; \cdot)$ is the multinomial

probability mass function as given above, and ξ_q is an optional weight corresponding to variable q .

Although it is possible to run KAMILA with weights for each variable as described above, these weights are not intended to be used to balance the contribution of continuous and categorical variables; setting all weights equal to 1 will accomplish this. Rather, the weights are intended to allow compatibility with other weighting strategies.

Object assignment is made based on the quantity

$$H_i^{(t)}(g) = \log \left[\hat{f}_{\mathbf{V}}^{(t)}(d_{ig}^{(t)}) \right] + \log \left[c_{ig}^{(t)} \right], \quad (3)$$

with observation i being assigned to the population g that maximizes $H_i^{(t)}(g)$. Note that $\hat{f}_{\mathbf{V}}^{(t)}$ is constructed using just the minimum distances as described above, but it is then evaluated at $d_{ig}^{(t)}$ for all g , not just the minimum distances.

Given a partition of the N observations at iteration (t) , the estimation step calculates $\hat{\mu}_g^{(t+1)}$ and $\hat{\theta}_{gq}^{(t+1)}$ for all g , p , and q . Let $\Omega_g^{(t)}$ denote the set of indices of observations assigned to population g at iteration t . We calculate the parameter estimates

$$\begin{aligned} \hat{\mu}_g^{(t+1)} &= \frac{1}{|\Omega_g^{(t)}|} \sum_{i \in \Omega_g^{(t)}} \mathbf{v}_i \\ \hat{\theta}_{gq\ell}^{(t+1)} &= \frac{1}{|\Omega_g^{(t)}|} \sum_{i \in \Omega_g^{(t)}} I\{w_{iq} = \ell\} \end{aligned}$$

where $I\{\cdot\}$ denotes the indicator function and $|A| = \text{card}(A)$.

The partition and estimation steps are repeated until a stable solution is reached (or the maximum number of iterations is reached) as described above. For each initialization, we calculate the objective function

$$\sum_{i=1}^N \max_g \{H_i^{(final)}(g)\}. \quad (4)$$

The partition that maximizes equation 4 over all initializations is output.

The number of clusters may be obtained using the prediction strength algorithm (Tibshirani and Walther 2005), as illustrated in Sect. 6. We choose the prediction strength algorithm due to the flexibility with which it can be adapted to many clustering algorithms, as well as the logical and interpretable rationale for the solutions obtained. Given an existing clustering of a data set, the prediction strength requires a rule that allocates new points into clusters, where the new points might not have been used to construct the original clusters. Tibshirani and Walther (2005) further discuss a strategy for adapting the prediction strength algorithm to hierarchical clustering techniques. The gap statistic (Tibshirani et al. 2001) might be used, although it would need to be adapted to mixed-type data. Information-based methods such as BIC (Schwarz 1978) might be applicable, although whether the KAMILA objective function behaves as a true log-likelihood should be carefully investigated, particularly with regard to asymptotics. Many internal measures of cluster validity, such as silhouette width (Kaufman and Rousseeuw 1990), require a distance function defined between points. In this case, the distance function as given in equation 3 is only defined between a cluster and a point; standard internal measures of cluster validity are thus not immediately applicable to KAMILA without further study. Popular internal measures for k -means such as pseudo- F (Calinski and Harabasz 1974) and pseudo- T (Duda and Hart 1973) are also not readily applicable since

within- and between-cluster sums of squares do not have any obvious analogue in the current approach.

In certain special cases, if the distribution of $\mathbf{V}$ is specified, the distribution of R is known [e.g. normal, t , Kotz distributions, and others (Fang et al. 1989)]. However, in our case we allow for arbitrary spherical distributions with distinct centers, which we estimate using the KDE in (2) and proposition 2. An investigation of the convergence of $\hat{f}_R(r)$ to the true density $f_R(r)$ requires an examination of the mean squared error and mean integrated squared error (Scott 1992), which is beyond the scope of the current paper.

KAMILA can be generalized to elliptical clusters as follows. For a given data set with $N \times P$ continuous data matrix V , we partition the total sum of squares and cross-products matrix of the continuous variables as described in Art et al. (1982), Gnanadesikan et al. (1993), yielding an estimate of the covariance matrix of the individual clusters, $\hat{\Sigma}$. The continuous variables can then be rescaled so that individual clusters have approximately identity covariance matrix by taking $V^* = V\hat{\Sigma}^{-1/2}$. The KAMILA clustering algorithm can then be used on the transformed data set. In simulation C we show that this strategy gives reasonable results on mixtures with elliptical clusters in the continuous variables.

Algorithm 1 KAMILA Clustering

```

for User-specified number of initializations do
  Initialize  $\hat{\mu}_g^{(0)}, \hat{\theta}_{gq}^{(0)} \forall g, q$ 
  repeat
    PARTITION STEP
     $d_{ig}^{(t)} \leftarrow \text{dist}(\mathbf{v}_i, \hat{\mu}_g^{(t)})$ 
     $r_i^{(t)} \leftarrow \min_g(d_{ig}^{(t)})$ 
     $\hat{f}_{\mathbf{V}}^{(t)} \leftarrow \text{RadialKDE}(\mathbf{r}^{(t)})$ 
     $c_{ig}^{(t)} \leftarrow \widehat{Pr}(\mathbf{w}_i \mid \text{observation } i \in \text{population } g)$ 
     $H_i^{(t)}(g) \leftarrow \log[\hat{f}_{\mathbf{V}}^{(t)}(d_{ig}^{(t)})] + \log[c_{ig}^{(t)}]$ 
    Assign observation  $i$  to population  $\arg\max_g\{H_i^{(t)}(g)\}$ 

    ESTIMATION STEP
    Calculate  $\hat{\mu}_g^{(t+1)}$  and  $\hat{\theta}_{gq}^{(t+1)}$ 
  until Convergence
   $ObjectiveFun \leftarrow \sum_{i=1}^N \max_g\{H_i^{(final)}(g)\}$ 
end for
Output partition that maximizes  $ObjectiveFun$ 

```

3.4 Identifiability considerations

A referee posed the question of identifiability of radially symmetric distributions. Identifiability in finite mixture models is important for inference purposes, and there is an extensive literature discussing this issue in parametric, semi-, and non-parametric contexts. See, for example, Titterton et al. (1985), Lindsay (1995), and McLachlan and Peel (2000). Recent developments in the semi-parametric context include (Hunter et al. 2007), who obtain identifiability for univariate samples by imposing a symmetry restriction on the individual components of the mixture. Further work in the univariate case includes (Cruz-Medina and Hettmansperger 2004), who assume that the component distributions are unimodal and continuous (Ellis 2002; Bordes et al. 2006).

Of relevance to our work is [Holzmann et al. \(2006\)](#). These authors prove identifiability of finite mixtures of elliptical distributions of the form

$$f_{\alpha,p}(\mathbf{x}) = |\Sigma|^{-1/2} f_p \left[(\mathbf{x} - \boldsymbol{\mu})^T \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu}); \boldsymbol{\theta} \right] \quad (5)$$

where $\mathbf{x} \in \mathbb{R}^p$, $\boldsymbol{\alpha} = (\boldsymbol{\theta}, \boldsymbol{\mu}, \Sigma) \in \mathcal{A}^p \subset \mathbb{R}^{k \times p \times p(p+1)/2}$, $f_p : [0, \infty) \rightarrow [0, \infty)$ is a density generator, that is, a nonnegative function such that $\int f(\mathbf{x}^T \mathbf{x}; \boldsymbol{\theta}) d\mathbf{x} = 1$. To prove identifiability of mixtures of spherically and elliptically distributed clusters we use theorem 2 of [Holzmann et al. \(2006\)](#).

Our proposition 2 expresses the density of a spherically symmetric random vector $\mathbf{V}$ as a function of a general density $f_R(r)$, $r = \sqrt{\mathbf{v}^T \mathbf{v}}$. A univariate kernel density estimator of f_R is given in Eq. (2). We first need the following conditions on the kernel $k(\cdot)$:

A1. The kernel $k(\cdot)$ is a positive function such that $\int k(u) du = 1$, $\int u k(u) du = 0$, and $\int u^2 k(u) du > 0$.

A2. The kernel function $k(\cdot)$ is a continuous, monotone decreasing function such that $\lim_{u \rightarrow \infty} k(u) = 0$.

A3. The kernel $k(\cdot)$ is such that

$$\lim_{z \rightarrow \infty} \frac{k(z \gamma_2)}{k(z \gamma_1)} = 0,$$

where γ_1, γ_2 are constants such that $\gamma_2 > \gamma_1$.

A4. The number of clusters G is fixed and known, with different centers $\boldsymbol{\mu}_j$, $j = 1, 2, \dots, G$.

A5. The density functions of the different clusters come from the same family of distributions and differ in terms of their location parameters, i.e. they are $f(\mathbf{v} - \boldsymbol{\mu}_j)$, $j = 1, 2, \dots, G$.

Assumptions A1 and A2 are standard in density estimation ([Scott 1992](#)). Assumption A3 is satisfied by the normal kernel that is widely used in density estimation. Furthermore, this condition is satisfied by kernels of the form $k(z) = A_2 \exp[-A_1 |z|^s]$, where $-\infty < z < \infty$, and A_1, A_2, s are positive constants linked by the requirement that $k(z)$ integrates to unity. This class of kernels is an extension of the normal kernel and examples include the standard normal and double exponential kernels. Generally, for identifiability to hold, it is sufficient that the kernel has tails that decay exponentially. Assumption A4 is similar to declaring the number of clusters in advance, as for example, in the k -means algorithm. Although the problem of identification of the number of clusters is interesting in its own right, the identifiability proof is based on having the number of clusters fixed in advance.

Proposition 3 *Under assumptions A1–A5 the density generator resulting from Proposition 2 with density estimator given in (2) satisfies condition (5) of Theorem 2 of [Holzmann et al. \(2006\)](#).*

Proof See Appendix.

Remark Theorem 2 of Holzmann et al., and hence Proposition 3 above, establish identifiability in the case of elliptical densities. The identifiability of spherical clusters is a special case which follows by setting Σ to the identity matrix.

4 Simulation study

4.1 Aims of the simulations

In this section, we present the results of a comprehensive Monte Carlo simulation study that we conducted to illustrate the effectiveness of our clustering methodologies. Via simulation, we compared the performance of the following methods: KAMILA, Hartigan–Wong k -means algorithm (Hartigan and Wong 1979) with a weighting scheme of Hennig and Liao (2013) as described above in Sect. 2.1, Hartigan–Wong k -means with Modha–Spangler weighting (Modha and Spangler 2003), and two finite mixture models. The first mixture model restricts the within-cluster covariance matrices to be diagonal, as implemented by the `flexmixedruns` function in the `fpc` package version 2.1-7 in R (Hennig 2014). The second, suggested by one of the reviewers, specifies the covariance matrices to be equal to cI , where $c > 0$ is equal across all clusters, and I is the identity matrix. We use the ARI (Hubert and Arabie 1985) as implemented in the `mclust` package version 4.3 (Fraley et al. 2012) to compare algorithm performance.

Simulation A aims to study the performance of the clustering algorithms on non-normal data sets with varying levels of continuous and categorical overlap. Simulation B compares the clustering algorithms in a setting that leads to the failure of the Modha–Spangler weighting method. Simulation C shows an example generalizing the KAMILA method to elliptical clusters. Finally, simulation D illustrates the performance of the KAMILA algorithm as sample size increases, revealing that it scales approximately linearly and does not suffer a decrease in accuracy for larger data sets.

4.2 Design

In all simulations, we generate mixed-type data with an underlying cluster structure. In order to vary the difficulty of the clustering problem in a consistent and comparable way across various continuous and categorical distributions, we use overlap (see e.g., Maitra and Melnykov 2010). The overlap between two random variables is defined as the area under each PDF over the region where that PDF is smaller than the other, that is, the overlap between clusters with densities f_1 and f_2 in equal proportion is given by

$$\int_{A_1} f_1(t)dt + \int_{A_2} f_2(t)dt$$

where $A_1 = \{x : f_1(x) < f_2(x)\}$ and $A_2 = \{x : f_2(x) < f_1(x)\}$. This generalizes in a straightforward manner to categorical random variables, using sums instead of integrals. An overlap of zero indicates complete separation between clusters, whereas an overlap of 1.0 or 100 % corresponds to identical component distributions. We report univariate overlap for each variable in each simulated data set. Table 3 summarizes the various simulation conditions used, and further details are described in this section.

In simulation A, we generate data from a two-cluster mixture distribution with individual clusters following a p -generalized normal–multinomial distribution or the lognormal–multinomial distribution. The p -generalized normal variables were generated by the `pgnorm` package version 1.1 in R (Kalke and Richter 2013), and have density of the form

$$f(t) = \frac{p^{1-1/p}}{2\sigma \Gamma(1/p)} \exp \left[-\frac{\left| \frac{t-\mu}{\sigma} \right|^p}{p} \right], \quad t \in \mathbb{R}$$

Table 3 Description of simulation studies A and B

	Simulation A	Simulation B
Sample size	250, 1000	500, 1000, 10000
# Con vars	2	4
Con dist	PG-normal, lognormal	normal
Con overlap	1, 15, 30, 45 %	1 %
# Cat vars	2	1
# Cat levs	4	2
Cat overlap	1, 15, 30, 45 %	1, 15, 30, 45, 60, 75, 90 %
# of clusters	2	2

where $\mu \in \mathbb{R}$ and $\sigma \in \mathbb{R}^+$ are location and scale parameters, and $p \in \mathbb{R}^+$ is a parameter indexing the kurtosis of the distribution. The p-generalized normal distribution contains as special cases the Laplace (double exponential) distribution when $p = 1$, the normal distribution when $p = 2$, and approaches the uniform distribution as $p \rightarrow \infty$ (Kalke and Richter 2013), that is, allowing for tails that are heavier than normal or lighter than normal. For the p-generalized normal distributions, we used the parameter settings $\sigma = 1$, with p equal to 0.7019, 0.7363, and 0.7785 for kurtosis of 8.0, 7.0, and 6.0, respectively. For the lognormal distributions, we set the log mean to 1 and the log SD to 0.3143, 0.6409, 1.1310 to achieve skewness 1.0, 2.5, and 9.0 respectively.

We generate continuous variables following mixture distributions with p-generalized normal clusters with varying kurtosis and overlap levels 1, 15, 30, and 45 %. Categorical variables follow multinomial mixtures with overlap levels 1, 15, 30, and 45 %.

In simulation B, we generate data from a two-cluster mixture with clusters following a joint normal-Bernoulli distribution. Four continuous variables and one categorical variable are used. The continuous variables are well-separated (1 % overlap), while the categorical variable varies from 1 to 90 % overlap as given in Table 3. This simulation was replicated for sample sizes of 500, 1000, and 10,000. The specific calculations used to compute overlap are described in the online resource, Section 1.

Simulation C investigated the extension of the KAMILA algorithm to accommodate elliptical clusters. The method described at the end of Sect. 3.3 was compared to two formulations of the finite mixture model. First, we used a finite mixture model in which the full covariance matrix was estimated in the parameter estimation stage of the EM algorithm. Second, we used a finite mixture model in conjunction with the rescaled data $V^* = V \hat{\Sigma}^{-1/2}$ as described in Sect. 3.3, in which the covariance matrix is restricted to have zero off-diagonal terms. We did not include k -means, Modha–Spangler, or the equal spherical mixture model in this simulation since they assume spherical clusters.

In simulation C, we generated data consisting of two continuous and three binary variables, with four clusters. The clusters were generated from the p-generalized normal distribution described above. We simulated bivariate p-generalized normal variables with $\sigma_1 = 1$, $\sigma_2 = 0.25$, $p_1 = p_2 = 0.7784$ (corresponding to kurtosis = 6.0), and cluster centers $(-5, -5)$, $(0, 0)$, $(2, -5)$, and $(5, 0)$. We then rotated each cluster about its center by $\pi/4$ radians. To ensure that clustering results primarily depended on the continuous variables, the three binary variables were generated to have little separation as follows: conditionally independent Bernoulli variables were generated having within-cluster probabilities of observing level 1 set to $(0.45, 0.45, 0.45)$, $(0.45, 0.45, 0.5)$, $(0.45, 0.5, 0.5)$, and $(0.5, 0.5, 0.5)$ for clusters 1 through 4 respectively. The entire simulation was then repeated with normal continuous variables.

In simulation D, we investigate the effects of increasing sample size of the data set on the performance and timing of KAMILA, the finite mixture model, and Modha–Spangler technique. In this simulation, the data followed a normal-multinomial mixture with two clusters. Two continuous and two categorical variables with four levels each were used. All variables had 30% overlap. In order to ensure that the timings were measured across the three methods in a fair way, the conditions of simulation D were carefully chosen to yield comparable performance (as measured by ARI). We drew samples of size 250, 500, 1000, 2500, 5000, 7500, and 10000. We then ran KAMILA only on data sets of size 2.5, 5, and 7.5 million.

4.3 Results

Results of Simulation A: Detailed results showing the mean ARI of each method for each condition in simulation A ($N = 1000$) are shown in Table 4 for the p-generalized normal condition and Table 5 for the lognormal condition.

When both the continuous and categorical overlaps are low, the clusters are relatively easy to identify, leading to good performance of all algorithms. In both the p-generalized normal and the lognormal condition, KAMILA outperforms the two parametric mixture models when the continuous and categorical overlap are both above 30%, as shown in Figs. 3 and 4. Additionally, in the lognormal condition, KAMILA outperforms the mixture models when skewness is 9 and continuous overlap is 15% or greater. In these conditions, the poor performance of the mixture models appears to be due to the fact that the normality assumptions of both models are strongly violated, leading to a poor use of the continuous and categorical variables.

In the p-generalized normal condition, KAMILA outperforms Modha–Spangler and the weighted k -means algorithm across a broad set of conditions, but the increased performance of KAMILA is most dramatic when continuous overlap is high (30 and 45%) and categorical overlap is low (1 or 15%), as shown in Fig. 5. In the lognormal condition, KAMILA is equal or superior to Modha–Spangler and the weighted k -means algorithm in most conditions, except when skewness is 9 and continuous overlap is 45%, as shown in Fig. 6.

The results of simulation A remained essentially unchanged when the sample size was 250, as shown in the online resource, Section 1.4.

Results of Simulation B: All methods except for Modha–Spangler perform well (ARI scores all round to 1.00). Modha–Spangler performance, however, was determined by the categorical overlap level: for categorical overlap levels of 1, 15, and 30%, etc. ranging up to 90%, the ARI steadily decreased from 0.98 to 0.01 for all sample sizes. Results for $N = 1000$ are shown in Table 6. Results for the $N = 500$ and $N = 10,000$ conditions were equivalent, and are shown in the online resource, Section 1.5.

Modha–Spangler weighting performs poorly in this condition due to the fact that the categorical component of the within-cluster distortion can be minimized to zero, resulting in a clustering that depends exclusively on the least informative categorical variable, with no regard to the continuous variables. All other methods perform optimally. This problem will arise whenever the number of unique combinations of categorical levels is equal to the number of clusters specified by the analyst when running the algorithm.

Results of Simulation C: Results of simulation C are shown in Table 7. In simulation C we see that KAMILA can be successfully adapted to accommodate elliptical data. All methods perform well, although KAMILA outperforms the other methods in the p-generalized normal

Table 4 Results of simulation A, p-generalized normal data

Kurt.	Categorical Overlap	Continuous Overlap	Mix. Model Diagonal	Mix. Model Eq. Spherical	KAMILA	k-means HL	Modha–Spangler
6	0.01	0.01	1.000	1.000	0.999	1.000	0.999
6	0.01	0.15	0.995	0.995	0.995	0.964	0.948
6	0.01	0.30	0.993	0.992	0.993	0.887	0.839
6	0.01	0.45	0.988	0.991	0.989	0.818	0.719
6	0.15	0.01	0.999	0.999	0.999	0.999	0.999
6	0.15	0.15	0.951	0.957	0.961	0.939	0.933
6	0.15	0.30	0.900	0.906	0.914	0.821	0.873
6	0.15	0.45	0.843	0.852	0.867	0.694	0.664
6	0.30	0.01	0.998	0.999	0.999	0.999	0.999
6	0.30	0.15	0.916	0.920	0.929	0.912	0.919
6	0.30	0.30	0.792	0.797	0.813	0.752	0.793
6	0.30	0.45	0.666	0.633	0.720	0.580	0.680
6	0.45	0.01	0.998	0.999	0.998	0.999	0.999
6	0.45	0.15	0.888	0.891	0.900	0.888	0.888
6	0.45	0.30	0.697	0.698	0.733	0.695	0.707
6	0.45	0.45	0.454	0.365	0.577	0.493	0.551
7	0.01	0.01	0.999	1.000	0.999	1.000	0.999
7	0.01	0.15	0.993	0.993	0.995	0.960	0.944
7	0.01	0.30	0.992	0.992	0.992	0.881	0.831
7	0.01	0.45	0.988	0.991	0.989	0.815	0.711
7	0.15	0.01	0.999	0.999	0.999	0.999	0.999
7	0.15	0.15	0.948	0.954	0.960	0.935	0.929
7	0.15	0.30	0.895	0.902	0.911	0.812	0.863
7	0.15	0.45	0.832	0.843	0.863	0.687	0.645
7	0.30	0.01	0.998	0.999	0.998	0.999	0.999
7	0.30	0.15	0.912	0.916	0.927	0.908	0.916
7	0.30	0.30	0.782	0.792	0.810	0.744	0.787
7	0.30	0.45	0.637	0.605	0.719	0.574	0.679
7	0.45	0.01	0.998	0.999	0.998	0.998	0.999
7	0.45	0.15	0.882	0.886	0.897	0.883	0.884
7	0.45	0.30	0.667	0.685	0.727	0.685	0.698
7	0.45	0.45	0.376	0.342	0.567	0.483	0.540
8	0.01	0.01	0.999	1.000	0.999	1.000	0.999
8	0.01	0.15	0.993	0.993	0.995	0.956	0.940
8	0.01	0.30	0.992	0.991	0.992	0.876	0.825
8	0.01	0.45	0.987	0.991	0.989	0.814	0.711
8	0.15	0.01	0.998	0.999	0.999	0.999	0.999
8	0.15	0.15	0.945	0.951	0.957	0.931	0.924
8	0.15	0.30	0.892	0.902	0.912	0.808	0.858
8	0.15	0.45	0.821	0.838	0.862	0.685	0.633
8	0.30	0.01	0.998	0.999	0.998	0.999	0.999

Table 4 continued

Kurt.	Categorical Overlap	Continuous Overlap	Mix. Model Diagonal	Mix. Model Eq. Spherical	KAMILA	<i>k</i> -means HL	Modha–Spangler
8	0.30	0.15	0.909	0.913	0.926	0.904	0.913
8	0.30	0.30	0.772	0.785	0.803	0.737	0.778
8	0.30	0.45	0.596	0.580	0.716	0.572	0.675
8	0.45	0.01	0.998	0.999	0.998	0.999	0.999
8	0.45	0.15	0.879	0.884	0.896	0.880	0.882
8	0.45	0.30	0.638	0.677	0.725	0.679	0.692
8	0.45	0.45	0.290	0.332	0.563	0.478	0.535

Mean Adjusted Rand Index is shown for each of the five methods compared, arranged by kurtosis of the continuous variables, and by continuous and categorical overlap. Results shown for sample size $N = 1000$. Monte Carlo standard error 0.008 or less in all conditions

condition. In the normal condition, all methods perform approximately equally (ARI of 0.99 or better).

Results of Simulation D: Results of simulation D are shown in Fig. 7, where we see that the computing time taken by the KAMILA algorithm appears to scale linearly with increasing sample size. Figure 8 shows that the number of iterations required by KAMILA to reach convergence increases sublinearly by sample size. Mean ARI scores are shown in Table 8, where we confirm that simulation conditions yield similar performance of the three algorithms included, by design, to ensure that timing differences between the algorithms are unlikely to be due to differences in the identified clusters.

5 Data analysis examples

We analyze three publicly available data sets from the UCI Machine Learning Repository (<http://archive.ics.uci.edu/ml>). The specific data sets are described in detail below and in Table 9. We analyzed each data set with the same five methods described in Sect. 4.1, along with an additional *k*-means algorithm in which categorical dummy variables were constructed such that the Euclidean distance between distinct categories was one.¹ For each algorithm, the number of clusters was specified to be the true number of outcome classes. The performance of each algorithm was compared to the true outcome classes using the measures of purity (Manning et al. 2008), macro-precision, and macro-recall (Modha and Spangler 2003).

5.1 Australian credit

This dataset concerns credit applications to a large bank. All predictor and outcome variable values have been replaced by arbitrary symbols for confidentiality purposes. There are two outcome classes. In contrast to the Cylinder Bands data set, this data set has about an equal proportion of continuous and categorical variables (Table 9). This data set is a canonical data set for evaluating supervised learning algorithms; the challenge it presents for unsupervised

¹ This was achieved using a dummy variable for each factor level with values 0 and $1/\sqrt{2}$ denoting absence and presence of the level, respectively. This yielded Euclidean distance between distinct categories of $\sqrt{(1/\sqrt{2})^2 + (-1/\sqrt{2})^2} = 1$.

Table 5 Results of simulation A, lognormal data

Skew	Categorical Overlap	Continuous Overlap	Mix. Model Diagonal	Mix. Model Eq. Spherical	KAMILA	k-means HL	Modha–Spangler
1.0	0.01	0.01	0.999	1.000	0.999	0.999	0.998
1.0	0.01	0.15	0.997	0.998	0.995	0.981	0.971
1.0	0.01	0.30	0.995	0.995	0.993	0.945	0.920
1.0	0.01	0.45	0.992	0.993	0.991	0.897	0.829
1.0	0.15	0.01	0.997	0.998	0.998	0.998	0.997
1.0	0.15	0.15	0.961	0.968	0.967	0.960	0.956
1.0	0.15	0.30	0.917	0.929	0.928	0.892	0.917
1.0	0.15	0.45	0.873	0.884	0.885	0.795	0.857
1.0	0.30	0.01	0.996	0.997	0.997	0.997	0.997
1.0	0.30	0.15	0.923	0.935	0.940	0.937	0.935
1.0	0.30	0.30	0.825	0.852	0.856	0.837	0.855
1.0	0.30	0.45	0.717	0.755	0.761	0.694	0.757
1.0	0.45	0.01	0.995	0.996	0.996	0.996	0.996
1.0	0.45	0.15	0.888	0.911	0.919	0.918	0.906
1.0	0.45	0.30	0.733	0.781	0.794	0.783	0.786
1.0	0.45	0.45	0.573	0.631	0.651	0.603	0.649
2.5	0.01	0.01	0.993	0.998	0.995	0.996	0.990
2.5	0.01	0.15	0.973	0.990	0.991	0.939	0.919
2.5	0.01	0.30	0.976	0.988	0.988	0.892	0.854
2.5	0.01	0.45	0.980	0.988	0.986	0.861	0.800
2.5	0.15	0.01	0.983	0.993	0.991	0.992	0.989
2.5	0.15	0.15	0.846	0.938	0.945	0.906	0.915
2.5	0.15	0.30	0.765	0.891	0.902	0.823	0.840
2.5	0.15	0.45	0.702	0.834	0.849	0.744	0.699
2.5	0.30	0.01	0.978	0.990	0.988	0.990	0.989
2.5	0.30	0.15	0.751	0.883	0.904	0.878	0.885
2.5	0.30	0.30	0.606	0.772	0.798	0.753	0.777
2.5	0.30	0.45	0.527	0.638	0.697	0.626	0.672
2.5	0.45	0.01	0.973	0.986	0.985	0.986	0.986
2.5	0.45	0.15	0.659	0.839	0.873	0.849	0.840
2.5	0.45	0.30	0.496	0.648	0.714	0.689	0.684
2.5	0.45	0.45	0.364	0.366	0.561	0.501	0.541
9.0	0.01	0.01	0.953	0.986	0.986	0.986	0.979
9.0	0.01	0.15	0.761	0.946	0.972	0.900	0.873
9.0	0.01	0.30	0.652	0.851	0.922	0.888	0.844
9.0	0.01	0.45	0.077	0.729	0.795	0.906	0.855
9.0	0.15	0.01	0.930	0.978	0.980	0.982	0.978
9.0	0.15	0.15	0.614	0.852	0.901	0.851	0.841
9.0	0.15	0.30	0.053	0.635	0.725	0.744	0.718
9.0	0.15	0.45	0.010	0.529	0.555	0.704	0.732
9.0	0.30	0.01	0.911	0.980	0.976	0.979	0.978

Table 5 continued

Skew	Categorical Overlap	Continuous Overlap	Mix. Model Diagonal	Mix. Model Eq. Spherical	KAMILA	k-means HL	Modha–Spangler
9.0	0.30	0.15	0.355	0.686	0.802	0.798	0.799
9.0	0.30	0.30	0.015	0.392	0.534	0.589	0.584
9.0	0.30	0.45	0.008	0.281	0.328	0.260	0.532
9.0	0.45	0.01	0.895	0.970	0.973	0.976	0.976
9.0	0.45	0.15	0.111	0.441	0.724	0.745	0.703
9.0	0.45	0.30	0.009	0.206	0.360	0.361	0.423
9.0	0.45	0.45	0.006	0.134	0.170	0.016	0.308

Mean Adjusted Rand Index is shown for each of the five methods compared, arranged by skewness of the continuous variables, and by continuous and categorical overlap. Results shown for sample size $N = 1000$. Monte Carlo standard error 0.019 or less in all conditions

Fig. 3 Simulation A, p-generalized normal data. Direct comparison of ARI between the finite mixture models and KAMILA by kurtosis of the continuous variable, and by continuous and categorical overlap levels. Results shown for $N = 1000$

Fig. 4 Simulation A, lognormal data. Direct comparison of ARI between the finite mixture models and KAMILA by skewness of the continuous variable, and by continuous and categorical overlap levels. Results shown for $N = 1000$

algorithms is to identify the underlying outcome classes and allocate observations without reference to a training set. As shown in Table 10, the mixture model with equal spherical covariance matrices and Modha–Spangler are the top performers, followed by KAMILA.

5.2 Cylinder bands

This dataset concerns the presence or absence of process delays known as “cylinder banding” in rotogravure printing. Variables describing various aspects of the printing process are available, for example cylinder size and paper type. The outcome variable describes whether or not cylinder banding occurred during the printing process. In contrast to the Australian Credit data set, this data set features about twice as many categorical variables as continuous variables (Table 9).

The raw data set was preprocessed according to the following criteria. If a variable had between one and eight missing values, the corresponding rows (across the entire data set) were removed to create a complete variable. If eight or more values were missing, that variable

Fig. 5 Simulation A, p-generalized normal data. Direct comparison of ARI between Modha–Spangler weighting, Hennig–Liao weighting, and KAMILA by kurtosis of the continuous variable, and by continuous and categorical overlap levels. Results shown for $N = 1000$

was removed from the data set. Variables with only two unique values, one of which only appears once, were removed from the data set.

As shown in Table 11, KAMILA is the top performer. The cluster structure identified by all other methods is nearly the same, and does not correspond as closely to the true outcome variable.

5.3 The insurance company benchmark (COIL 2000)

This data set was used in the 2000 CoIL (Computational Intelligence and Learning) competition, and concerns customers of an insurance company. Variables describe various attributes of current and potential customers, for example marital status, home ownership, and educational achievement. We investigated how well the clustering algorithms could recover customer type, a variable with ten classes, for example “Conservative families” and “Successful hedonists.” This data set is notable due to the large number of variables, the large sample size compared to the previous Australian Credit and Cylinder Bands data sets, and the large number of outcome categories (see Table 9). Since customer segmentation (discovering

Fig. 6 Simulation A, lognormal data. Direct comparison of ARI between Modha–Spangler weighting, Hennig–Liao weighting, and KAMILA by skewness of the continuous variable, and by continuous and categorical overlap levels. Results shown for $N = 1000$

Table 6 Simulation B: Mean ARI for each algorithm in each categorical overlap condition

Categorical Overlap	Mix. Model Eq. Spherical	Mix. Model Diagonal	KAMILA	k -means HL	Modha–Spangler
0.01	1.00	1.00	1.00	1.00	0.98
0.15	1.00	1.00	1.00	1.00	0.72
0.30	1.00	1.00	1.00	1.00	0.49
0.45	1.00	1.00	1.00	1.00	0.30
0.60	1.00	1.00	1.00	1.00	0.16
0.75	1.00	1.00	1.00	1.00	0.06
0.90	1.00	1.00	1.00	1.00	0.01

Results shown for $N = 1000$. Monte Carlo error is less than or equal to 0.0014 in all conditions

Table 7 Simulation C: Monte Carlo mean ARI for each algorithm in each condition

Monte Carlo error less than or equal to 0.01 in all conditions

Distribution	KAMILA	Mix. Model Unrestricted	Mix. Model Diagonal
Normal	0.997	0.991	0.996
PG-Normal	0.986	0.922	0.941

Fig. 7 *Left*: A plot showing the computing time of KAMILA, the finite mixture model (FMR), and Modha–Spangler (M–S) clustering methods for increasing sample size in simulation D. Time is the mean elapsed time in seconds, averaged over 1000 Monte Carlo samples. Note that the computation times of the three algorithms are not directly comparable due to differing proportions of compiled and interpreted code. *Right*: Computing time of KAMILA for larger sample sizes, with 95 % CI

coherent subtypes of customers in a business setting) is a common use of clustering algorithms, this was perhaps a more relevant scenario than the previous two prediction problems.

The sociodemographic variables were used (variables 1–43), omitting the two variables describing customer type (variables 1 and 5). The variables were either continuous or ordinal; since we did not expect the ordinal variables to have a monotonic relationship relative to outcome, we treated them as nominal variables. Since no training set is required in unsupervised learning, the training and test data sets were merged to produce the final data set.

The top performer in this data set depended on the metric used for evaluation against the ground truth. The mixture model with diagonal covariance matrices had the best purity score, while the KAMILA algorithm (unweighted) had the best macro-averaged precision/recall scores.

5.4 Discussion of the data analysis examples

In the analysis of these three data sets, we verify that the findings described in Sects. 2 and 4 above are generalizable in the context of real-world applications. We observe that coding strategies do not appear to offer general solutions to the weighting problem: note that while Hennig–Liao coding improves performance over naive 0–1 weighting in the Australian credit data set (Table 10), it hurts performance relative to naive weighting in the insurance data set (Table 12). We observe that the two mixture models perform well when their specific distributional assumptions hold approximately, but suffer when violated: the equal spherical mixture model is the top performer in the Australian credit data set, while the diagonal

Fig. 8 Boxplots depicting the mean number of iterations taken by the KAMILA algorithm to reach convergence for increasing sample sizes in simulation D. Each data point is averaged over a complete application of the KAMILA algorithm consisting of ten random initializations

Table 8 Simulation D: Mean ARI, for each algorithm for each of the sample sizes used in the simulation

Sample Size	KAMILA	Mix. Model Diagonal	Modha–Spangler
100	0.864	0.860	0.873
500	0.881	0.880	0.881
1000	0.880	0.881	0.881
2500	0.882	0.883	0.882
5000	0.882	0.884	0.882
7500	0.883	0.884	0.883
10000	0.882	0.884	0.882

Monte Carlo error less than or equal to 0.003 in all conditions

mixture model is one of the bottom performers; this is flipped in the insurance data set where the diagonal mixture model is the top performer and the equal spherical is the bottom performer. In either case, the KAMILA algorithm is close in performance to the better of the two mixture models. As in the simulations in Sect. 4, these data analyses illustrate the versatility of KAMILA. KAMILA is the only method to perform well across all conditions, as is indicated by the aforementioned data analyses. Additionally, KAMILA is the only algorithm in the bands data set that is able to uncover clusters with a reasonable relationship with the outcome of interest.

Table 9 Key characteristics of the data sets analyzed

Data set name	# Obs	# Continuous Variables	# Categorical Variables	Outcome Variable
Australian Credit	690	6	8	Acc (44 %) Rej (56 %)
Bands	516	7	13	Band (41 %) No Band (59 %)
The Insurance Company Benchmark (COIL 2000)	9822	3	38	Successful hedonists (9.8 %) Career loners (0.8 %) Retired and religious (9.0 %) Farmers (5.0 %) Driven growers (8.4 %) Living well (9.6 %) Family with grown ups (27.4 %) Average family (15.4 %) Cruising seniors (3.3 %) Conservative families (11.3 %)

Table 10 Results of Australian Credit data set analysis, purity and macro precision/recall

	Purity	Purity % over M–S	Macro precision/recall	Macro P/R % over M–S
<i>k</i> -means 0–1	0.668	80.6%	0.725/0.634	86.6%/ 77.4%
<i>k</i> -means HL	0.746	90.0%	0.785/0.723	93.8%/ 88.3%
Modha–S.	0.829	100.0%	0.837/0.819	100.0%/100.0%
Mix Mod Diag	0.683	82.3%	0.689/0.664	82.3%/ 81.1%
Mix Mod Spher	0.848	102.3%	0.854/0.839	102.0%/102.5%
KAMILA	0.775	93.5%	0.808/0.755	96.5%/ 92.2%

Row shows the method used, and column corresponds to the metric used. Modha–Spangler is taken as the gold standard. Greater than 100 % indicates superior performance over Modha–Spangler

Table 11 Results of cylinder bands data set analysis, purity metric

	Purity	Purity % over M–S	Macro Precision/Recall	Macro P/R % over M–S
<i>k</i> -means 0-1	0.589	100.0%	0.589/0.500	100.0%/100.0%
<i>k</i> -means HL	0.589	100.0%	0.589/0.500	100.0%/100.0%
Modha–S.	0.589	100.0%	0.589/0.500	100.0%/100.0%
Mix Mod Diag	0.589	100.0%	0.589/0.500	100.0%/100.0%
Mix Mod Spher	0.589	100.0%	0.589/0.500	100.0%/100.0%
KAMILA	0.671	113.8%	0.662/0.665	112.4%/132.9%

Row shows the method used, and column corresponds to the metric used. Modha–Spangler is taken as the gold standard. Greater than 100 % indicates superior performance over Modha–Spangler

Table 12 Results of insurance data set analysis, purity metric. Row shows the method used, and column corresponds to the metric used. Modha–Spangler is taken as the gold standard. Greater than 100% indicates superior performance over Modha–Spangler

	Purity	Purity % over M–S	Macro Precision/Recall	Macro P/R % over M–S
<i>k</i> -means 0-1	0.361	108.1%	0.400/0.212	103.0%/ 98.6%
<i>k</i> -means HL	0.329	98.6%	0.308/0.200	79.3%/ 93.0%
Modha–S.	0.334	100.0%	0.388/0.215	100.0%/100.0%
Mix Mod Diag	0.380	113.7%	0.365/0.223	94.0%/103.7%
Mix Mod Spher	0.289	86.5%	0.476/0.129	122.6%/ 59.8%
KAMILA	0.354	106.1%	0.461/0.225	118.8%/104.6%

6 A practical example

6.1 Description of data and study

To illustrate the practical application of the KAMILA algorithm, we analyze data collected from seven call centers from a large real-world service environment supporting globally distributed customers and multiple request types. The data covers the January 2011 through April 2012 time-frame. [Heching and Squillante \(2012\)](#) discuss in detail IT service delivery environments with global delivery locations responding to requests for service from a collection of customers around the world. As opposed to the analyses in Sect. 5, here we do not know the true outcome classes, which is a more realistic setting for the application of cluster analysis.

Agents at each center receive calls from customers with IT service requests of varying severity. Calls are categorized based on the nature and urgency of the service request, time taken to resolve the service request, whether or not the service request was resolved, and whether the service request was completed within a time satisfying the contractual obligations to the client. In order to successfully manage the call centers with regard to staffing requirements, customer satisfaction, and workload assignment, it is necessary to understand the call centers in detail. This includes a study of the types of calls received, including when the call is received, the duration until service request resolution, whether the resolution time was quick enough to meet the service level agreement, and how the call was resolved. In particular, it is necessary to understand the dependencies between these variables. Cluster analysis offers a natural way to identify and describe the most salient of these dependencies, without relying on the analyst to pre-specify outcome or predictor variables.

The data set consisted of 44,518 observations of seven variables: service duration (continuous, measured in minutes), day of week service began (Sunday, Monday, etc.), breach time (contractually mandated upper limit on time taken to resolve the service request, measured in minutes), contract breach (binary yes/no; whether the service time exceeded the contractual limit), service request category (planned system upgrade, planned maintenance, unexpected problem), closure code (successful, reassigned, duplicate, cancelled, failure, false alarm), and job complexity (low, medium, high). Summary statistics are given in Table 13.

Table 13 Summary statistics for call center data set. Observed frequencies of the levels of each categorical variable; mean, median, standard deviation, and IQR for log-transformed service duration (minutes) and log-transformed time to breach of contract (minutes)

Variable	Levels	Frequency
Day of week	Sunday	919
	Monday	11,153
	Tuesday	8811
	Wednesday	7896
	Thursday	7766
	Friday	7051
	Saturday	922
Contract breach	Yes	10,900
	No	33,618
Service request category	Planned system upgrade	145
	Planned maintenance	75
	Unexpected problem	44,298
Closure code	Failure	18
	Cancelled	242
	Successful	40,967
	Duplicate	486
	False alarm	99
	Reassigned	2706
Job complexity	Low	31,230
	Medium	9422
	High	3866
	Mean (SD)	Median (IQR)
Log(Service duration)	5.21 (2.09)	5.00 (3.45)
Log(Breach time)	12.09 (0.47)	12.19 (0.47)

We used a log-transformation for service duration and breach time due to a right skew in both variables characteristic of measurements taken spanning multiple orders of magnitude (times ranged from minutes/hours to days/months); as discussed in [Hennig and Liao \(2013\)](#), a log-transformation in this case allowed for a more sensible “interpretative distance” between variable values. We rescaled log service duration and log breach time variables to mean zero and variance one before entering them into the analysis. Since day of the week of the call is a cyclical variable, we coded levels as 0 = Sunday, 1 = Monday, etc., and mapped them to the unit circle in $\mathbb{R}^2$ by taking the real and imaginary components of $\exp(2j\pi i/7)$, where $i = \sqrt{-1}$ and j denotes coded day of week, with $j \in \{0, 1, \dots, 6\}$ (see, for example, [Zhao et al. 2011](#)).

We clustered the data using the KAMILA algorithm, using ten random initializations and a maximum of twenty iterations per initialization. We chose these parameters as they were sufficiently large to yield stable results over repeated runs with the same number of initializations/iterations specified. We selected the number of clusters using the prediction strength criteria ([Tibshirani and Walther 2005](#)). Prediction strength estimates the greatest number of clusters that can be reliably identified with a given clustering method

Fig. 9 A figure showing the prediction strength of the KAMILA algorithm in the current data set by number of clusters. Error bars show plus/minus one standard error taken over five two-fold cross-validation runs

in a given data set. We estimated prediction strength over five two-fold cross-validation runs.

6.2 Results

6.2.1 Primary clustering

We selected the three-cluster solution based on the prediction strength method (Tibshirani and Walther 2005), as shown in Fig. 9. As suggested in Tibshirani and Walther (2005), the number of clusters is selected to be the largest k such that the prediction strength at k plus the standard error at k is over the chosen threshold, where the standard error is taken across the cross-validation runs. Threshold values of 0.8–0.9 are recommended for well separated clusters; to allow for overlapping clusters, we chose a threshold of 0.6.

Boxplots of the service duration and breach time variable for each cluster are shown in Fig. 10. Cross-tabulations of the variables day, contract breach, and job complexity are shown in Table 14. Results of the same analysis using Modha–Spangler clustering are shown in Fig. 11 and Table 15, and for k -means clustering with Hennig–Liao weights (2013) in Fig. 12 and Table 16.

6.3 Discussion of call center analysis

The results of the prediction strength algorithm (Tibshirani and Walther 2005) suggest that three clusters can be reliably identified in the current data set using KAMILA. These three

Fig. 10 Boxplots of log service duration by cluster and log time to contract breach by cluster in the call center data analysis, KAMILA method

Table 14 Cross-tabulation of cluster membership by day, contract breach status, and job complexity in the call center analysis

		Cluster		
		1	2	3
Day of week	Sun	329	59	531
	Mon	2899	0	8254
	Tue	2157	46	6608
	Wed	2996	2558	2342
	Thu	2493	5273	0
	Fri	1962	5089	0
	Sat	364	558	0
Contract	N	6746	11, 223	15, 649
Breach	Y	6454	2360	2086
Job complexity	Low	9245	8282	13, 703
	Medium	3056	3824	2542
	High	899	1477	1490

The cluster memberships were derived from an application of the KAMILA algorithm

Fig. 11 Boxplots of log service duration by cluster and log time to contract breach by cluster in the call center data analysis, Modha–Spangler method

Table 15 Cross-tabulation of cluster membership by day, contract breach status, and job complexity in the call center analysis

		Cluster		
		1	2	3
Day of week	Sun	202	0	717
	Mon	3056	8097	0
	Tue	2032	6779	0
	Wed	1940	5956	0
	Thu	1751	0	6015
	Fri	1406	0	5645
	Sat	191	0	731
Contract	N	0	20, 832	12, 786
Breach	Y	10, 578	0	322
Job complexity	Low	8016	14, 583	8631
	Medium	1949	4346	3127
	High	613	1903	1350

The cluster memberships were derived from an application of the Modha–Spangler clustering technique

Fig. 12 Boxplots of log service duration by cluster and log time to contract breach by cluster in the call center data analysis, *k*-means with Hennig–Liao coding

clusters identify the most salient features of the data set without specifying predictor or outcome variables, and achieve a clustering solution that equitably balances the contribution from both continuous and categorical variables without manually choosing weights.

One salient feature is the increased volume of calls early in the work-week, captured by cluster 3. Cluster 3 has a higher proportion of calls involving low complexity problems (77 vs. 70 % and 61 % in clusters 1 and 2, resp.), suggesting that the increased volume of calls is not driven by service requests stemming from (for example) catastrophic system failures. This is supported by the fact that cluster 2, which captures a set of calls primarily from the latter half of the work-week (Wednesday–Friday), has a higher proportion of problems with medium and high complexity. Cluster 1 is comprised of calls spanning the entire week, and has a higher median service duration (approximately 26 h higher than the median time for clusters 2 or 3). This increase in time is associated with a larger number of contract breaches: 49 % of the calls in cluster 1 resulted in a duration long enough to violate the contract, compared to 17 and 12 % violations in clusters 2 and 3 respectively.

Table 16 Cross-tabulation of cluster membership by day, contract breach status, and job complexity in the call center analysis

		Cluster		
		1	2	3
Day of week	Sun	541	378	0
	Mon	8131	3022	0
	Tue	6224	2587	0
	Wed	0	2157	5739
	Thu	0	2256	5510
	Fri	0	2352	4699
	Sat	0	450	472
Contract	N	10, 727	10, 565	12, 326
Breach	Y	4169	2637	4094
Job complexity	Low	14, 813	0	16, 417
	Medium	0	9422	0
	High	83	3780	3

The cluster memberships were derived from an application of *k*-means with Hennig–Liao weighting

This large number of contract breaches in cluster 1 is an important and potentially actionable insight. Cluster 1 appears to contain a constant two- to three-thousand service requests per weekday that are time consuming and likely to breach the contractually obligated time allocation. A manager of these call centers would be advised to consider service requests falling within cluster 1 as high risk, and perhaps allocate additional resources towards resolving these requests.

If either Modha–Spangler or *k*-means with Hennig–Liao weighting are used to cluster the same data set, the resulting clusters are dominated by one variable type (categorical variables), with minimal separation in either service duration or time to breach. The clusters identified by the Modha–Spangler technique are not substantially different from a straightforward crosstabulation of contract breach by day of the week: note that 97 % of the contract breach “Yes” observations are all in the same category, and show minimal separation with regard to the continuous variables. This is in comparison to KAMILA cluster 1, which contains a high proportion of contract breach “Yes” observations, but clearly separates out those with long service durations. Similarly, Hennig–Liao coding is dominated by day of the week and job complexity, and as a result shows minimal separation in the continuous variables, as well as offering little information beyond a crosstabulation of “low” versus other job complexity by day of the week.

7 Discussion

KAMILA is a flexible method that performs well across a broad range of data structures, while competing methods tend to perform well in certain specific contexts. This is illustrated by the fact that KAMILA performs at or near the top across all analyses in the current paper, including both the simulations and real data analyses. The radial density estimation technique used in KAMILA allows a general class of distributions to be accommodated by our model, with the sole condition that the contour lines of the individual clusters are elliptical. We recommend that KAMILA be used with mixed-type data when the underlying

data distribution is unknown. It will generally provide a reasonable clustering of the data in terms of precision/recall and purity of the clusters. For example, KAMILA can be used when the clusters are suspected to be non-normal or with skewed data distributions.

If the clusters are known in advance to be normal in the continuous dimension, a normal mixture model can be used, but as we show in simulation A this is problematic when normality is violated. In contrast, we have shown in simulation A that KAMILA performs well when faced with these same violations of normality. Other than the normal distribution, we note that none of the distributions used in the simulations (p-generalized normal, lognormal) are elliptical distributions with respect to the L_2 norm, demonstrating that KAMILA can perform well with non-normal and more generally with non-elliptical clusters.

In Sect. 2.2 we show that methods relying on user-specified variable weights to balance the continuous and categorical variables can perform poorly unless the user resolves the difficult (and perhaps impossible) task of manual weight selection. The method of Modha–Spangler addresses the weight selection challenge, and often improves upon naive weight selection strategies such as Hennig–Liao coding (2013). However, Modha–Spangler does not always achieve balance between continuous and categorical variables as shown in simulation A (e.g. Fig. 5), simulation B (e.g. Table 6), and the call center analysis in Sect. 6. We confirm these general findings in a set of analyses of real-world data sets in Sect. 5, with discussion in Sect. 5.4.

We have generalized our results to accommodate elliptical distributions in Sect. 3.3 and simulation C using the decomposition method described in Art et al. (1982), Gnanadesikan et al. (1993). Future work will investigate a second approach to elliptical clusters, in which a scale matrix for each cluster could be estimated during the iterative estimation steps, as in some formulations of the finite Gaussian mixture model.

As described in Sect. 3.1, we currently assume that the continuous and categorical variables are conditionally independent given cluster membership. This assumption could be relaxed through the use of models such as the location model (Olkin and Tate 1961). The downside to this approach is that the number of unknown parameters increases exponentially with the number of categorical variables, resulting in a method that does not scale well for increasing numbers of variables in the data set. While this approach might be appropriate for small data sets, we did not pursue it further in the current paper.

If the true number of clusters in a data set is unknown, the method of cluster validation using prediction strength can be used to select the appropriate number of clusters (Tibshirani and Walther 2005). We illustrate this in the context of an application of the KAMILA algorithm to an example data set in Sect. 6. A further possibility is the use of information-based methods such as Schwarz’ BIC (Schwarz 1978). Assuming that the quantity calculated in Sect. 3.3, equation 4, is a reasonable approximation of a log-likelihood for the overall model, it may be possible to construct an approximation to the BIC and use it to select the number of clusters as used in finite mixture models (Hennig and Liao 2013).

Acknowledgments We would like to thank the anonymous reviewers for constructive feedback that led to an improved manuscript.

Compliance with ethical standards

Conflict of interest Alex Foss has no conflicts of interest to report. Marianthi Markatou has no conflicts of interest to report. Bonnie Ray has no conflicts of interest to report. Aliza Heching has no conflicts of interest to report.

Appendix

Proof of Proposition 1

Proof of Proposition 1 We have

$$E \left[\left(\frac{Y_1 - Y_2}{v} \right)^2 \right] = \frac{\mu^2 + \sigma_1^2 + \sigma_2^2}{\pi(1-\pi)\mu^2 + \pi\sigma_1^2 + (1-\pi)\sigma_2^2},$$

where $\mu^2 > 0$, $\sigma_1^2 > 0$, $\sigma_2^2 > 0$, $\pi \in (0, 1)$, and $v^2 = \text{Var}[V] = \pi\sigma_1^2 + (1-\pi)\sigma_2^2 + \pi(1-\pi)\mu^2$. Since $\mu^2 + \sigma_1^2 + \sigma_2^2 > \pi(1-\pi)\mu^2 + \pi\sigma_1^2 + (1-\pi)\sigma_2^2$, it follows that the expectation is greater than 1.

When $\sigma_1^2 = \sigma_2^2 = \sigma^2$, we have $\mu^2 + 2\sigma^2 > 2[\pi(1-\pi)\mu^2 + \sigma^2]$, which implies $\mu^2 > 2\pi(1-\pi)\mu^2$, but since $\pi(1-\pi) \in (0, 1/4)$, it follows that the expectation is greater than 2. $\square$

Proof of Proposition 2

Proof of Proposition 2 We define the transformation from p -dimensional Euclidean space to p -spherical space [e.g. see Blumenson (1960)], that is, if $\mathbf{V}$ is a p -dimensional random vector, we define the transformation $\mathbf{V}^T \rightarrow (R, \Theta_1, \Theta_2, \dots, \Theta_{p-1})^T$ with $R = \|\mathbf{V}\|_2 = \sqrt{\mathbf{V}^T \mathbf{V}}$, $\Theta_{p-1} \in [0, 2\pi)$ and all other angles on $[0, \pi)$ with mapping $V_1 = R \cos(\Theta_1)$, $V_j = R \cos(\Theta_j) \prod_{i=1}^{j-1} \sin(\Theta_i)$ for $j = 2, \dots, p-1$, and $V_p = R \prod_{i=1}^{p-1} \sin(\Theta_i)$. The Jacobian of this transformation is

$$|J| = r^{p-1} \sin(\Theta_1)^{p-2} \sin(\Theta_2)^{p-3} \dots \sin(\Theta_{p-2}).$$

Now, the volume of a unit p -sphere, that is, a hypersphere with radius 1 in p -dimensional Euclidean space, is $\text{Vol}_p = \pi^{p/2} / \Gamma(\frac{p}{2} + 1)$ (Huber 1982), where $\Gamma(\cdot)$ is the gamma function. We can also write the formula for the volume as $\text{Vol}_p = \int_{r=0}^1 \int_{\theta} \dots \int r^{p-1} g(\theta) d\theta dr = \frac{1}{p} \int_{\theta} \dots \int g(\theta) d\theta$, where $g(\theta) = |J|/r^{p-1}$. This implies that $\int_{\theta} \dots \int g(\theta) d\theta = \frac{p\pi^{p/2}}{\Gamma(\frac{p}{2} + 1)}$, and thus that the marginal density of R is

$$\begin{aligned} f_R(r) &= \int \dots \int f_{R, \Theta}(r, \theta) r^{p-1} g(\theta) d\theta \\ &= f_{\mathbf{V}}(\mathbf{v}) r^{p-1} \frac{p\pi^{p/2}}{\Gamma(\frac{p}{2} + 1)} \end{aligned}$$

and thus proposition 2 is proven. $\square$

Proof of Proposition 3

Proof of Proposition 3 The density generator of the density function given in Proposition 2 with kernel density estimator for f_R as given in (2) is

$$f_p \left((\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right) = \frac{\hat{f}_R \left(\sqrt{(\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu})} \right) \Gamma(p/2 + 1)}{p\pi^{p/2} \left((\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right)^{\frac{p-1}{2}}}$$

Now, let $\mathbf{x} = u\mathbf{z}$, where $u \in \mathbb{R}$ varies and $\mathbf{z} \in \mathbb{R}^p$ is fixed. Then, $(\mathbf{x} - \boldsymbol{\mu}_i)^T \boldsymbol{\Sigma}_i^{-1} (\mathbf{x} - \boldsymbol{\mu}_i)$ can be written as $a_i u^2 + b_i u + c_i$, where $a_i = \mathbf{z}^T \boldsymbol{\Sigma}_i^{-1} \mathbf{z}$, $b_i = 2\mathbf{z}^T \boldsymbol{\Sigma}_i^{-1} \boldsymbol{\mu}_i$, and $c_i = \boldsymbol{\mu}_i^T \boldsymbol{\Sigma}_i^{-1} \boldsymbol{\mu}_i$. Define a total ordering on $\lambda_i = (a_i, b_i)$ as follows: $\lambda_1 < \lambda_2$ if $a_1 < a_2$, or if $a_1 = a_2$ and $b_1 < b_2$. Consider the ratio

$$\begin{aligned} \frac{f_p(a_2 u^2 + b_2 u + c_2)}{f_p(a_1 u^2 + b_1 u + c_1)} &= \left(\frac{a_1 u^2 + b_1 u + c_1}{a_2 u^2 + b_2 u + c_2} \right)^{\frac{p-1}{2}} \frac{\hat{f}_R \left(\sqrt{a_2 u^2 + b_2 u + c_2} \right)}{\hat{f}_R \left(\sqrt{a_1 u^2 + b_1 u + c_1} \right)} \\ &= \left(\frac{a_1 u^2 + b_1 u + c_1}{a_2 u^2 + b_2 u + c_2} \right)^{\frac{p-1}{2}} \frac{\sum_{\ell=1}^N k \left(\frac{\sqrt{a_2 u^2 + b_2 u + c_2} - r_\ell}{h} \right)}{\sum_{\ell=1}^N k \left(\frac{\sqrt{a_1 u^2 + b_1 u + c_1} - r_\ell}{h} \right)} \\ &< \left(\frac{a_1 u^2 + b_1 u + c_1}{a_2 u^2 + b_2 u + c_2} \right)^{\frac{p-1}{2}} \sum_{\ell=1}^N \frac{k \left(\frac{\sqrt{a_2 u^2 + b_2 u + c_2} - r_\ell}{h} \right)}{k \left(\frac{\sqrt{a_1 u^2 + b_1 u + c_1} - r_\ell}{h} \right)} \quad (6) \end{aligned}$$

with the last inequality holding since $k(u) > 0 \forall u \in \mathbb{R}$ by assumption A1.

For $a_1 < a_2$, we can write $(\sqrt{a_i u^2 + b_i u + c_i} - r)/h = \gamma_i |u| - r/h$, where $\gamma_i = \sqrt{a_i + O(1/u)}/h$. Substituting into the ℓ^{th} term in the summation in (6) we obtain

$$\frac{k(\gamma_2 |u| - r_\ell/h)}{k(\gamma_1 |u| - r_\ell/h)} \rightarrow 0 \quad \text{as } u \rightarrow \infty$$

by property A3.

For $a_1 = a_2 = a$ and $b_1 < b_2$, the terms in the summation in (6) also converge to zero. For example, in the case of the normal kernel $k(t) \propto \exp\{-t^2/2\}$, we obtain

$$k \left(\frac{\sqrt{a u^2 + b_i u + c_i} - r}{h} \right) \propto \exp \left\{ -\frac{1}{2h^2} \left(a u^2 + b_i u + c_i - 2r|u|\sqrt{a + O(1/u)} + r^2 \right) \right\}$$

which we can substitute into the ℓ^{th} term in the summation in (6) yielding $\exp \left\{ -\frac{1}{2h^2} (u(b_2 - b_1) + c_2 - c_1) \right\} \rightarrow 0$ as $u \rightarrow \infty$.

Therefore, identifiability follows by Theorem 2 of [Holzmann et al. \(2006\)](#). $\square$

References

- Ahmad, A., & Dey, L. (2007). A k-means clustering algorithm for mixed numeric and categorical data. *Data and Knowledge Engineering*, 63(2), 503–527.
- Art, D., Gnanadesikan, R., & Kettenring, J. (1982). Data-based metrics for cluster analysis. *Utilitas Mathematica*, 21A, 75–99.
- Azzalini, A., & Menardi, G. (2014). Clustering via nonparametric density estimation: The R package pdfCluster. *Journal of Statistical Software*, 57(11), 1–26.
- Azzalini, A., & Torelli, N. (2007). Clustering via nonparametric density estimation. *Statistics and Computing*, 17(1), 71–80.
- Blumenson, L. (1960). A derivation of n-dimensional spherical coordinates. *The American Mathematical Monthly*, 67(1), 63–66.
- Bordes, L., Mottelet, S., & Vandekerckhove, P. (2006). Semiparametric estimation of a two-component mixture model. *The Annals of Statistics*, 34(3), 1204–1232.
- Bowman, A., & Azzalini, A. (1997). *Applied smoothing techniques for data analysis*. Oxford: Oxford Science Publications.

- Browne, R., & McNicholas, P. (2012). Model-based clustering, classification, and discriminant analysis of data with mixed type. *Journal of Statistical Planning and Inference*, 142(11), 2976–2984.
- Burnaby, T. (1970). On a method for character weighting a similarity coefficient, employing the concept of information. *Journal of the International Association for Mathematical Geology*, 2(1), 25–38.
- Calinski, T., & Harabasz, J. (1974). A dendrite method for cluster analysis. *Communications in Statistics*, 3(1), 1–27.
- Chae, S., Kim, J., & Yang, W. (2006). Cluster analysis with balancing weight on mixed-type data. *The Korean Communications in Statistics*, 13(3), 719–732.
- Chu, C., Kim, S., Lin, Y., Yu, Y., Bradski, G., Ng, A., et al. (2006). Map-reduce for machine learning on multicore. In B. Schölkopf, J. C. Platt, & T. Hoffman (Eds.), *NIPS* (pp. 281–288). Cambridge: MIT Press.
- Comaniciu, D., & Meer, P. (2002). Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5), 603–619.
- Cruz-Medina, I., & Hettmansperger, T. (2004). Nonparametric estimation in semi-parametric univariate mixture models. *Journal of Statistical Computation and Simulation*, 74(7), 513–524.
- DeSarbo, W., Carroll, J., Clark, L., & Green, P. (1984). Synthesized clustering: A method for amalgamating alternative clustering bases with differential weighting of variables. *Psychometrika*, 49(1), 57–78.
- Dougherty, J., Kohavi, R., & Sahami, M. (1995). Supervised and unsupervised discretization of continuous features. In *Machine learning: Proceedings of the twelfth international conference* (pp. 194–202). Morgan Kaufmann.
- Duda, R., & Hart, P. (1973). *Pattern classification and scene analysis*. New York: Wiley.
- Ellis, S. (2002). Blind deconvolution when noise is symmetric: Existence and examples of solutions. *Annals of the Institute of Statistical Mathematics*, 54(4), 758–767.
- Esther, M., Kriegel, H., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of KDD* (pp. 226–231).
- Everitt, B. (1988). A finite mixture model for the clustering of mixed-mode data. *Statistics and Probability Letters*, 6(5), 305–309.
- Fang, K., Kotz, S., & Ng, K. (1989). *Monographs on statistics and applied probability* (Vol. 36). New York: Chapman and Hall.
- Fan, J., Han, F., & Liu, H. (2014). Challenges of big data analysis. *National Science Review*, 1(2), 293–314.
- Forgy, E. (1965). Cluster analysis of multivariate data: Efficiency versus interpretability of classifications. *Biometrics*, 21, 768–769.
- Fraley, C., Raftery, A., Murphy, T., & Scrucca, L. (2012). mclust version 4 for R: Normal mixture modeling for model-based clustering, classification, and density estimation. Technical Report 597, Department of Statistics, University of Washington.
- Fraley, C., & Raftery, A. (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association*, 97(458), 611–631.
- Friedman, J., & Meulman, J. (2004). Clustering objects on subsets of attributes (with discussion). *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 66(4), 815–849.
- Gnanadesikan, R., Harvey, J., & Kettenring, J. (1993). Mahalanobis metrics for cluster analysis. *Sankhya, Series A*, 55(3), 494–505.
- Gnanadesikan, R., Kettenring, J., & Tsao, S. (1995). Weighting and selection of variables for cluster analysis. *Journal of Classification*, 12(1), 113–136.
- Goodall, D. (1966). A new similarity index based on probability. *Biometrics*, 22, 882–907.
- Gower, J. (1971). A general coefficient of similarity and some of its properties. *Biometrics*, 27(4), 857–871.
- Hall, P., Watson, G., & Cabrera, J. (1987). Kernel density estimation with spherical data. *Biometrika*, 74(4), 751–762.
- Hartigan, J., & Wong, M. (1979). A k-means clustering algorithm. *Applied Statistics*, 28, 100–108.
- Heching, A., & Squillante, M. (2012). Stochastic decision making in information technology services delivery. In J. Faulin, A. Juan, S. Grasman, & M. Fry (Eds.), *Decision making in service industries: A practical approach*. Boca Raton: CRC Press.
- Hennig, C. (2014). fpc: Flexible procedures for clustering. <http://CRAN.R-project.org/package=fpc>. R package version 2.1-7.
- Hennig, C., & Liao, T. (2013). How to find an appropriate clustering for mixed-type variables with application to socio-economic stratification. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 62(3), 309–369.
- Holzmann, H., Munk, A., & Gneiting, T. (2006). Identifiability of finite mixtures of elliptical distributions. *Scandinavian Journal of Statistics*, 33(4), 753–763.
- Huang, Z. (1998). Extensions to the k-means algorithm for clustering large data sets with categorical values. *Data Mining and Knowledge Discovery*, 2(3), 283–304.

- Huang, J., Ng, M., Rong, H., & Li, Z. (2005). Automated variable weighting in k-means type clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(5), 657–668.
- Huber, G. (1982). Gamma function derivation of n-sphere volumes. *The American Mathematical Monthly*, 89(5), 301–302.
- Hubert, L., & Arabie, P. (1985). Comparing partitions. *Journal of Classification*, 2(1), 193–218.
- Hunter, D., Wang, S., & Hettmansperger, T. (2007). Inference for mixtures of symmetric distributions. *The Annals of Statistics*, 35(1), 224–251.
- Hunt, L., & Jorgensen, M. (2011). Clustering mixed data. *WIREs Data Mining and Knowledge Discovery*, 1, 352–361.
- Ichino, M., & Yaguchi, H. (1994). Generalized minkowski metrics for mixed feature type data analysis. *IEEE Transactions on Systems, Man and Cybernetics*, 24(4), 698–708.
- Kalke, S., & Richter, W. (2013). Simulation of the p-generalized Gaussian distribution. *Journal of Statistical Computation and Simulation*, 83(4), 641–667.
- Kaufman, L., & Rousseeuw, P. (1990). *Finding groups in data*. New York: Wiley.
- Kelker, D. (1970). Distribution theory of spherical distributions and a location-scale parameter generalization. *Sankhya: The Indian Journal of Statistics, Series A (1961–2002)*, 32(4), 419–430.
- Kotz, S., Balakrishnan, N., & Johnson, N. (2004). *Continuous multivariate distributions, models and applications*. Continuous multivariate distributions. Hoboken: Wiley.
- Krzanowski, W. (1993). The location model for mixtures of categorical and continuous variables. *Journal of Classification*, 10(1), 25–49.
- Lawrence, C., & Krzanowski, W. (1996). Mixture separation for mixed-mode data. *Statistics and Computing*, 6(1), 85–92.
- Lichman, M. UCI machine learning repository. <http://archive.ics.uci.edu/ml>. Accessed Sept 2015.
- Lindsay, B. (1995). *Mixture models: Theory, geometry, and applications*. Hayward: Institute of Mathematical Statistics.
- Li, J., Ray, S., & Lindsay, B. (2007). A nonparametric statistical approach to clustering via mode identification. *Journal of Machine Learning Research*, 8, 1687–1723.
- Lloyd, S. (1982). Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28(2), 129–137.
- MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Vol. 1: Statistics* (pp. 281–297). Berkeley: University of California Press.
- Maitra, R., & Melnykov, V. (2010). Simulating data to study performance of finite mixture modeling and clustering algorithms. *Journal of Computational and Graphical Statistics*, 19(2), 354–376.
- Manning, C., Raghavan, P., & Schütze, H. (2008). *Introduction to information retrieval*. Cambridge: Cambridge University Press.
- McLachlan, G., & Peel, D. (2000). *Finite mixture models*. New York: Wiley.
- Milligan, G. (1980). An examination of the effect of six types of error perturbation on fifteen clustering algorithms. *Psychometrika*, 45(3), 325–342.
- Modha, D., & Spangler, W. (2003). Feature weighting in k-means clustering. *Machine Learning*, 52(3), 217–237.
- Olkin, I., & Tate, R. (1961). Multivariate correlation models with mixed discrete and continuous variables. *The Annals of Mathematical Statistics*, 32(2), 448–465.
- Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics*, 6(2), 461–464.
- Scott, D. (1992). *Multivariate density estimation*. Hoboken: Wiley.
- Silverman, B. (1986). *Density estimation*. London: Chapman and Hall.
- Tibshirani, R., & Walther, G. (2005). Cluster validation by prediction strength. *Journal of Computational and Graphical Statistics*, 14(3), 511–528.
- Tibshirani, R., Walther, G., & Hastie, T. (2001). Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(2), 411–423.
- Titterton, D., Smith, A., & Makov, U. (1985). *Statistical analysis of finite mixture models*. Chichester: Wiley.
- Wolfe, J., Haghighi, A., & Klein, D. (2008). Fully distributed em for very large datasets. In *Proceedings of the 25th international conference on machine learning* (pp. 1184–1191). ICML '08 New York, NY: ACM.
- Zhao, Y., Zeng, D., Herring, A., Ising, A., Waller, A., Richardson, D., et al. (2011). Detecting disease outbreaks using local spatiotemporal methods. *Biometrics*, 67(4), 1508–1517.