Published in final edited form as:
Psychol Methods. 2017 September ; 22(3): 563–580. doi:10.1037/met0000095.

A comparison of latent class, *K*-means, and *K*-median methods for clustering dichotomous data

Michael J. Brusco,
Florida State University

Emilie Shireman, and
University of Missouri

Douglas Steinley[*]
University of Missouri

Abstract

The problem of partitioning a collection of objects based on their measurements on a set of dichotomous variables is a well-established problem in psychological research, with applications including clinical diagnosis, educational testing, cognitive categorization, and choice analysis. Latent class analysis and *K*-means clustering are popular methods for partitioning objects based on dichotomous measures in the psychological literature. The *K*-median clustering method has recently been touted as a potentially useful tool for psychological data and might be preferable to its close neighbor, *K*-means, when the variable measures are dichotomous. We conducted simulation-based comparisons of the latent class, *K*-means, and *K*-median approaches for partitioning dichotomous data. Although all three methods proved capable of recovering cluster structure, *K*-median clustering yielded the best average performance, followed closely by latent class analysis. We also report results for the three methods within the context of an application to transitive reasoning data, where it was found that the three approaches can exhibit profound differences when applied to real data.

Introduction

The problem of partitioning N objects into K clusters based on measurements of the objects on V dichotomous variables is a well-studied problem in the psychological sciences (Brusco, 2004; Dimitriadou, Dolnicar, & Weingessel, 2002). Dichotomous items can arise in the form of correct and incorrect answers on cognitive tests (Chiu, Douglas, & Li, 2009; Ellis, 2014; Van der Ark, Croon, & Sijtsma, 2008), responses to true or false questions on an examination (Ohan, Cormier, Hepp, Visser, & Strain, 2008), the presence or absence of symptoms in a psychiatric evaluation (Williams, Barton, White, & Hosik, 1976), the cognitive categorization of competitive relationships among retailers (Porac & Thomas, 1994), pick-any-subset tasks (Coombs, 1964; Hubert, 1974), and the presence or absence of ties in a social network (Brusco & Steinley, 2007a).

[*]Douglas Steinley's effort was partially supported by Grant 1R01AA023248 from the National Institutes of Health, National Institute on Alcohol and Alcoholism.

The measurement of *N* objects on *V* dichotomous variables results in what is known as *two-mode*, *two-way* dichotomous data. The data are two-way because they are assembled in a two-dimensional array, and are two-mode because the dimensions correspond to two distinct types of entities. One dimension of the array corresponds to the *N* objects and the other to the *V* variables. There is a variety of possible approaches to the problem of classifying the objects based on the dichotomous variable measures. There are also different bases that can be used to distinguish among these approaches. The first formal basis pertains to the important distinction between partitioning and non-partitioning approaches. Partitioning methods produce clusters that are non-empty, mutually exclusive, and exhaustive. Effectively, this means that each cluster has at least one object, each object is assigned to one (and only one) cluster, and all objects are assigned. By contrast, overlapping clustering (Chaturvedi, Carroll, Greeen, & Rotondo, 1997) methods allow objects to be members of more than one cluster. Likewise, fuzzy clustering methods (Bezdek, Coray, Gunderson, & Watson, 1981; Wedel & Steenkamp, 1989) permit objects to have partial (fractional) memberships in multiple clusters. Although overlapping and fuzzy clustering methods and principles have some history in the psychological literature (Arabie & Carroll, 1980; Chapman & Goldberg, 2011; Hedges & Olkin, 1983; Shepard & Arabie, 1979), their usage is negligible in comparison to partitioning methods and, accordingly, our focus is restricted to partitioning throughout the remainder of this paper.

A second formal basis for distinguishing among clustering methods is whether or not they are based on a underlying statistical model (Andrews, Brusco, Currim, & Davis, 2010; Wedel & Kamakura. 2000). Methods based on a statistical model are referred to by terms such as latent class clustering (Magidson & Vermunt, 2002), latent class analysis (Eshghi, Haughton, Legrand, Skaletsky, & Woolford, 2011), mixture model clustering (McLachlan & Peel, 2000; Steinley & Brusco, 2011a), or model-based clustering (Banfield & Raftery, 1993; Fraley & Raftery, 1999). Succinctly, in model-based clustering, the data are assumed to be generated from a mixture of distributions with different parameters, and the relevant objective criterion for the classification process is one of maximizing likelihood. Contrastingly, non-model-based methods are not grounded by an underlying statistical model and typically correspond to discrete optimization algorithms that may seek to optimize a diverse range of objective criteria.

A third, less formal basis for distinguishing among clustering approaches is whether they are *special-purpose* or *general-purpose*. Special-purpose methods are those that are customized in their design for specific applications in the analysis of dichotomous data. The distinguishing aspect of special-purpose methods is the incorporation of salient objective criteria or constraints that are unique to the particular problem under study. In the case of dichotomous data, one family of special-purpose methods includes those procedures that are grounded in set theory (Curry, 1976; Restle, 1959), such as methods found in the diverse family of hierarchical classes (HICLAS) models that are designed for structural analysis of multi-mode, multi-way dichotomous data (Ceulemans & Van Mechelen, 2005, 2008; Ceulemans, Van Mechelen, & Leenen, 2007; DeBoeck & Rosenberg, 1988; Vande Gaer, Ceulemans, Van Mechelen, & Kuppens, 2012; Wilderjans, Ceulemans, & Van Mechelen, 2008, 2012). A second category corresponds to the literature stream pertaining to cognitive diagnosis methods to assess mastery (or non-mastery) of a collection of items in educational testing (Chiu et al., 2009; Macready & Dayton, 1977; Templin & Henson, 2006; Templin, Henson, & Douglas, 2007). Blockmodeling methods for social network analysis represent yet another class of special-purpose methods for dichotomous data (see Doreian, Batagelj, & Ferligoj, 2005 for an extensive review).

Although they may require some parametric assumptions, general-purpose approaches for clustering dichotomous data consist of those methods that are broadly applicable to both dichotomous and non-dichotomous data, and include methods such as *K*-means partitioning, *K*-median partitioning, and latent class analysis. Since its development in the 1950's and 1960's (Forgy, 1965; Jancey, 1966; Lloyd, 1957; MacQueen, 1967; Steinhaus, 1956; Thorndike, 1953), *K*-means partitioning has arguably been the most prominent clustering method in scientific research (see Bock, 2007; Kogan, 2007; and Steinley, 2006 for reviews). Although not originally designed for clustering dichotomous data, there is computational evidence supporting the efficacy of *K*-means for clustering such data (Brusco, 2004; Chiu et al., 2009; Dimitriadou et al., 2002; Köhn, Chiu, & Brusco, 2015). *K*-median partitioning has its origins in graph theory, specifically with respect to the location of switching centers in networks (Hakimi, 1964, 1965). In light of the fact that network ties correspond to dichotomous measurements, it is logical to posit that *K*-median methods might prove particularly effective for dichotomous data applications (Ruiz, Chebat, & Hansen, 2004).

Unlike *K*-means and *K*-median approaches, the latent class model (LCM) is formulated as a probabilistic, finite mixture modeling approach for classifying objects (Goodman, 1974; Lazarsfeld, 1950). The LCM is the most commonly chosen method for clustering dichotomous data in psychological applications. Recent examples published in APA outlets include the use of the LCM in the following contexts: (i) as a diagnostic tool in conjunction with an internet gaming disorder scale (Lemmens, Valkenberg, & Gentile, 2015), (ii) to establish subgroups of smokers with distinct patterns on dichotomous risk behavior measures (Prochaska et al., 2014), (iii) to establish classes based on the presence of different gambling activities (Savage, Slutske, & Martin, 2014), (iv) to obtain subgroups based on dichotomous ratings of social skill (De Los Reyes, Bunnell, & Beidel, 2013), (v) to classify students based on the presence or absence of various forms of bullying (Waasdorp & Bradshaw, 2011) or peer victimization (Bradshaw, Waasdorp, & O'Brennan, 2013), and (vi) to assess knowledge of attention-deficit disorder (Ohan et al., 2008). By contrast, a search of the PsycArticles database for the 2008-2015 time period did not uncover any applications of *K*-means cluster analysis to dichotomous data in APA outlets.

A comparison of special-purpose approaches is difficult because they are designed to accomplish different goals. However, it is possible to undertake a comparison of general-purpose procedures for clustering dichotomous data with respect to their ability to recover underlying known cluster structure. In this paper, we present comparisons of three general methods for the partitioning of dichotomous data: (i) LCM, (ii) *K*-means clustering, and (iii) *K*-median clustering.[1] Although all three of these methods have been available for decades,

[1] Given that LCM is a mixture-model approach to clustering, whereas *K*-means and *K*-median are non-model-based methods, our comparative analyses are comparable to the recent recovery-based comparisons of finite mixture models and *K*-means within the context of metric variables (Steinley & Brusco, 2011a).

little is known about their relative performances for classifying dichotomous data. In light of this gap in the literature, comparisons are conducted using two simulation experiments that control for variation in the number of objects (*N*), the number of clusters (*K*), the number of clustering variables (*V*), the relative sizes of the clusters, and the level of error perturbation applied to the underlying cluster structure. Comparative analyses are performed under the assumption that the correct number of clusters is known, as well as under conditions where *K* is unknown and must be determined as part of the model selection process.

In addition to the simulation-based comparisons, LCM, *K*-means, and *K*-median clustering are applied to dichotomous data associated with the study of transitive reasoning (Verweij, Sijtsma, & Koops, 1996). The results of this application are particularly noteworthy, as the three methods yield profoundly different two-cluster partitions of the transitive reasoning data. This finding suggests that, although the three methods might perform comparably in a controlled simulation experiment, they can lead to different interpretations when applied to real data. Succinctly, *K*-means produced a two-cluster solution that was difficult to interpret, LCM yielded one large cluster and one small cluster of poor performers, and *K*-median clustering provided a solution whereby the two clusters were differentiated primarily based on the three most discriminating test items. In light of its performance in both the simulation experiments and the application to real data, our key recommendation is that researchers give serious consideration to the use of *K*-median clustering when analyzing dichotomous data.

Formal descriptions of the LCM, *K*-means, and *K*-median clustering methods are provided in the next section. Subsequent sections report the results for the implementation of these methods in two simulation studies, as well as for the transitive reasoning data. The paper concludes with a brief summary and suggestions for future research.

Methods

The Latent Class Model (LCM)

The latent class model (LCM) is a finite mixture model originally developed to explain the structure of a set of multivariate dichotomous data (Lazarsfeld, 1950; Lazarsfeld & Henry, 1968). As noted in the introduction, the term ‘latent class clustering’ is now sometimes used more broadly to refer to other types of model-based clustering (Eshghi et al., 2011; Magidson & Vermunt, 2002). However, our focus here is limited to the original formulation in the case of dichotomous measures. There are a number of thorough treatments of the original formulation of the LCM and its extensions (Bartholomew & Knott, 1999; McCutcheon, 1987; McLachlan & Peel, 2000; Vermunt & Magidson, 2005a).

To facilitate the description of the LCM, we define $\mathbf{x}_i = [x_{ij}]$ as the vector of observed measurements for object *i* on the *V* dichotomous variables ($1 \le i \le N$ and $1 \le j \le V$), where x_{ij} may take values from the set {0, 1}. The parameters estimated by the model are contained in the set $\Theta = \{\boldsymbol{\lambda}, \boldsymbol{\Pi}\}$. The vector $\boldsymbol{\lambda} = [\lambda_k]$ consists of the *K* class membership probabilities, where λ_k is the probability that any given object belongs to class *k* (for $1 \le k \le K$). The class probabilities are constrained to sum to unity (i.e., $\sum_{k=1}^{K} \lambda_k = 1$). The matrix $\boldsymbol{\Pi} = [\boldsymbol{\pi}_{jk}]$ contains the probabilities of a positive measurement (i.e., a value of 1) for each variable *j* from class *k*. Given these definitions, the LCM is:

$$f(\mathrm{x}_i | \Theta) = \sum_{k=1}^{K} \lambda_k \prod_{j=1}^{V} \pi_{jk}^{x_{ij}} (1 - \pi_{jk})^{1 - x_{ij}}, \quad (1)$$

The posterior probability that an object *i* with measurements $\mathbf{x}_i$ belongs to class *k* is computed as follows:

$$f(k | \mathrm{x}_i, \Theta) = \frac{\lambda_k \prod_{j=1}^{V} \pi_{jk}^{x_{ij}} (1 - \pi_{jk})^{1 - x_{ij}}}{f(\mathrm{x}_i | \Theta)}, \text{ for} 1 \leq k \leq K. \quad (2)$$

The likelihood function for the LCM is computed across all observations as follows:

$$L = \prod_{i=1}^{N} \left[\sum_{k=1}^{K} \lambda_k \prod_{j=1}^{V} \pi_{jk}^{x_{ij}} (1 - \pi_{jk})^{1 - x_{ij}} \right]; \quad (3)$$

however, for model estimation purposes, it is common to work with the log-likelihood function:

$$\log(L) = \sum_{i=1}^{N} \log \left[\sum_{k=1}^{K} \lambda_k \prod_{j=1}^{V} \pi_{jk}^{x_{ij}} (1 - \pi_{jk})^{1 - x_{ij}} \right]. \quad (4)$$

The estimation of the model parameters for Equation (4), subject to the constraint on the sum of the cluster membership probabilities, is accomplished via maximum likelihood methods available in software packages such as Mplus (Muthén & Muthén, 1998-2012), Latent Gold (Vermunt & Magidson, 2005b), and the R programming libraries *polka* (Linzer & Lewis, 2011), *e1071* (Dimitriadou, Hornik, Leisch, Meyer, & Weingessel, 2014), and *BayesLCA* (White & Murphy, 2014). Herein, estimation was completed using a Matlab (MathWorks, Inc., 2005) implementation of the expectation-maximization (EM) algorithm (Dempster, Laird, & Rubin, 1977). This facilitated the simulation-based comparison with competing methods, which were also implemented in Matlab.

The EM algorithm for LCM assumes that a value of *K* is supplied as input. In our first simulation experiment, the LCM was implemented using the correct (or true) value of *K* for each underlying test problem. The algorithm was restarted 20 times using different random partitions to initialize the algorithm for each restart. The restart producing the maximum value of log(*L*) was stored as the best solution. Cluster assignments were established by assigning each case to the cluster for which its class posterior probability (see Equation 2) was largest, which is the typical practice in mixture model clustering (Steinley & Brusco, 2011a).

In the second experiment, the value of K was assumed to be unknown and was determined as part of the clustering process. Specifically, five restarts of the EM-algorithm were applied for each number of clusters on the interval $2 \leq K \leq 8$. The restart producing the maximum value of the log-likelihood function was identified and this value (and its corresponding solution) was stored for each value of K on the interval $2 \leq K \leq 8$. The value of the log-likelihood function, log(L), generally increases as K increases, but at the expense of a greater number of model parameters. Therefore, the selection of the best value of K requires the use of *information criteria* that incorporate both log(L) and the number of estimated model parameters in their computation. The total number of parameters in Θ is $KV + K$, where KV is the number of parameters in Π and K is the number of cluster membership proportions. However, in light of the constraint that the proportions must sum to one, there are only K-1 proportions that actually require estimation.

Three distinct information criteria were evaluated for selecting the best value of K. The first two criteria were based on Akaike’s (1973) information criterion (AIC). For a given penalty parameter (ρ) and number of clusters (K), the AIC is computed as follows:

$$AIC(\rho, K) = -2\log(L) + \rho(KV + (K-1))\,. \quad (5)$$

The first version of the AIC (AIC2) corresponds to the original penalty of parameter of ρ = 2. However, in light of the excellent performance of ρ = 3 in other clustering contexts (Andrews & Currim, 2003a, 2003b), we consider the AIC with this penalty parameter (AIC3) for the second version. The third criterion that we considered was Schwartz’ (1978) Bayesian information criterion (BIC), which is computed as follows:

$$BIC(K) = -2\log(L) + \log(N)(KV + (K-1))\,. \quad (6)$$

For each of the three criteria (AIC2, AIC3, or BIC), the goal of the model selection process is to identify the value of K that minimizes Equation (5) or (6) as appropriate. To understand the tradeoffs associated with seeking the value of K that produces the minimum value for each of the information criteria, it is helpful to observe that the first term, −2log(L), in Equations 5 and 6 will generally decrease as K increases because log(L) generally increases as K increases. However, the second terms in Equations 5 and 6 clearly increase as a function of K and are, effectively, the penalties incurred for using more parameters.

K-means Partitioning

The K-means algorithm seeks to find a partition that minimizes the sum, across all objects, of the squared Euclidean distances of each object to the centroid of the cluster to which it is assigned. Hereafter, this measure is referred to as SSE. Denoting $P = \{C_1, \ldots, C_K\}$ as a partition of the N objects into K clusters, where C_k contains the set of objects assigned to cluster k, the SSE is computed as follows:

$$SSE = \sum_{k=1}^{K} \sum_{i \in C_k} \sum_{j=1}^{V} \left(x_{ij} - \bar{x}_{jk}\right)^2, \quad (7)$$

where $\bar{x}_{jk}$ is the mean for variable j of the objects assigned to cluster k.

As noted by Brusco and Steinley (2007b), there is a variety of different algorithmic implementations of *K*-means clustering (Forgy, 1965; Hartigan & Wong, 1979; Jancey, 1966; MacQueen, 1967; Steinhaus, 1956). In this paper, we use an implementation known as *HK*-means (Hansen & Mladenović, 2001), which was particularly effective in the comparative study performed by Brusco and Steinley (2007b). The steps of the *HK*-means algorithm are as follows:

1. Randomly assign the objects to obtain a *K*-cluster partition.
2. Compute the cluster centroids.
3. Assign each object to the cluster corresponding to its nearest centroid.
4. Repeat steps 2 and 3 until convergence (i.e., no objects change membership at Step 3).
5. Consider each object in turn with respect to relocation from its current cluster to one of the other *K*-1 clusters. Any relocation that reduces *SSE* should be accepted.
6. Repeat Step 5 until no relocation of an object will further reduce *SSE*.

Steps 1 through 4 are often used to characterize the *K*-means algorithm (Chiu et al., 2009; Steinley, 2003), although others refer to these steps as *H*-means (Hansen & Mladenović, 2001; Späth, 1980).[2] It is important to recognize that completion of Step 4 does not guarantee a solution that is locally-optimal with respect to all possible relocations of an object from its current cluster to one of the other clusters. Completion of Steps 5 and 6 does afford such a guarantee of local optimality and commonly improves the *SSE* measure produced after Step 4. Nevertheless, a global optimum is not assured. Multiple restarts of the *K*-means algorithm using different random initial partitions can help mitigate the chances of a poor local optimum (Steinley, 2003; Steinley & Brusco, 2007). Alternatively, metaheuristics such as tabu search, genetic algorithms, variable neighborhood search, and simulated annealing can be used in lieu of multiple restarts of the *K*-means heuristic. However, simulation results reported by Brusco and Steinley (2007b) indicate that *HK*-means generally performed as well or better than most metaheuristics when there were 10 or fewer clusters.

[2] Steps 1-4 are typically what is used to define the *K*-means algorithm (Bock, 2007; Steinley, 2003, 2006); however, not all authors adopt this terminology. Hartigan (1975), Späth (1980), Hansen and Mladenović (2001) and others use *K*-means to refer to the process of Steps 5-6. The latter two of these sources refer to Steps 1-4 as *H*-means. Following Hansen and Mladenović, we integrate Steps 1-4 with 5-6 and refer to the entire procedure as *HK*-means.

Like the EM algorithm for the LCM, *K*-means procedures require a pre-specified value of *K*. For the first simulation study, the *K*-means algorithm is supplied with the correct value of *K*. For the second experiment, we select the value of *K* based on the Calinski-Harabasz (CH: 1974) pseudo-*F* statistic.[3] Although many other methods for selection of *K* are possible (see Dimitriadou et al., 2002; Steinley & Brusco, 2011b), the CH measure has performed well in previous simulation experiments (Milligan & Cooper, 1985; Steinley & Brusco, 2011a). The *CH* index is computed as follows:

$$CH(K) = \frac{[(SST - SSE(K))/(K-1)]}{[SSE(K)/(N-K)]}, \quad (8)$$

where $SSE(K)$ is the value of Equation 7 for *K* clusters and *SST* is computed as:

$$SST = \sum_{i=1}^{N} \sum_{j=1}^{V} \left(x_{ij} - \bar{x}_j\right)^2, \quad (9)$$

and $\bar{x}_j$ is the mean for variable *j* across all *N* objects. The selected value of *K* is the one for which $CH(K)$ is maximized.

K-median Partitioning

The *K*-median partitioning problem requires the selection of exactly *K* objects (often called exemplars) to serve as cluster centers and the assignment of each object to its nearest exemplar, with the goal of minimizing the sum of the distances of the objects to the exemplars. Denoting $E(k)$ as the exemplar object for cluster *k*, the criterion function for the *K*-median problem is as follows:

$$SSKmed = \sum_{k=1}^{K} \sum_{i \in C_k} \sum_{j=1}^{V} \left(x_{ij} - x_{E(k)j}\right)^2, \quad (10)$$

A comparison of *SSE* in Equation (7) and *SSKmed* (10) helps to clarify the distinction between *K*-means and *K*-median clustering. The *SSE* measures, for each variable, distances between each object and the cluster means (i.e., $\bar{x}_{jk}$). By contrast, *SSKmed* measures, for each variable, distances between each object and its cluster exemplar (i.e., $x_{E(k)j}$).

In this paper, we use the multistart fast interchange procedure (see Brusco & Köhn, 2008a, 2009; Köhn, Steinley, & Brusco, 2010) that seeks to obtain solutions that minimize *SSKmed*. This method is based on the pioneering work of Teitz and Bart (1968), Whitaker (1983), and Hansen and Mladenović (1997). The procedure consists of the following steps:

1. Randomly select *K* objects to serve as exemplars.

[3] We also tested the method of Ratkowsky and Lance (1978), which performed best in the Dimitriadou et al. (2002) study, however, it was less effective than CH at choosing the correct number of clusters.

2. Assign each object to the cluster corresponding to its nearest exemplar.
3. Consider each exemplar in turn with respect to its replacement with one of the objects not selected as an exemplar. Any replacement that reduces the sum of the distances of the objects to their exemplars should be accepted.
4. Repeat Step 3 until no replacement of an exemplar will further reduce *SSKmed*.

Completion of Step 4 guarantees a solution that is locally-optimal with respect to all possible replacements of an exemplar with an object not currently selected as an exemplar. However, like the K-means heuristic, the resulting solution is not guaranteed to be globally-optimal. Therefore, multiple restarts of the algorithm are advised. Hansen and Mladenović (1997) indicated that the multiple restart approach was generally effective for problems where $K < 50$ (see also Brusco & Köhn. 2009). Given the relatively small number of clusters associated with our analyses ($K \leq 6$), we confidently used the multiple restart approach and followed the practice of Brusco and Köhn (2008a) by using 20 restarts for the results obtained herein. As is the case for K-means clustering, when K exceeds 10, we recommend the use of metaheuristics for K-median clustering (see Mladenović, Brimberg, Hansen, & Moreno-Pérez, 2007 for a review).

For instances where the number of clusters was not pre-specified, we used the following rule based on maximum ratio of percentage changes (MRPC) in *SSKmed* on the left and right, which is computed for K clusters as follows:

$$MRPC(K) = \frac{[(SSKmed(K-1) - SSKmed(K))/SSKmed(K-1)]}{[(SSKmed(K) - SSKmed(K+1))/SSKmed(K)]}, \quad (11)$$

where $SSKmed(K)$ is the value of Equation (10) for K clusters.[4] If MRPC(K) is large, that means that the percentage change going from $K - 1$ clusters to K clusters is large relative to the percentage change going from K to $K + 1$ clusters and, accordingly, K is judged to be a good stopping point for the number of clusters. The motivation for the MRPC index dates back (at least) to the work of Hansen and Delattre (1978), who used a similar index within the context of minimum diameter partitioning (see also Brusco & Cradit, 2004). More recently, Ceulemans and Van Mechelen (2005) found that ratio rules such as MRPC generally performed well within the context of hierarchical classes models, and their use in the context of multi-mode clustering is particularly common (Schepers, Ceulemans, & Van Mechelen, 2008; Schepers & Van Mechelen, 2011; Wilderjans, Ceulemans, & Meers, 2013).

Methods Not Included in the Comparisons

It should be recognized that LCM, K-means, and K-median clustering methods are not the only possible methods that can be used for clustering dichotomous data. Among the other possible methods are clique partitioning (Brusco & Köhn, 2009b), SEGWAY (Krieger & Green, 1999), and artificial neural networks (ANNs: McCulloch & Pitts, 1943). Andrews, Brusco, and Currim (2010) compared the recovery performances of latent class modeling

[4] We also tested the silhouette index (Kaufman & Rousseeuw, 2005), however it was far less effective than MRPC at choosing the correct number of clusters.

approach, clique partitioning, and SEGWAY within the context of finding consensus partitions. The latent class approach provided the best recovery in their study. This fact, in conjunction with greater scalability and accessibility of latent class clustering software, contributed to our decision to exclude clique partitioning and SEGWAY from consideration.

Du (2010) provides a thorough review of ANN approaches to clustering. There are several categories of ANNs, such as the self-organizing map (SOM: Kohonen, 1982), learning vector quantization (LVQ: Kohonen, 1990), and adaptive resonance theory (ART; Grossberg, 1976), as well as multiple versions of methods within each of these categories. One reason that we excluded ANNs from our simulation analyses is that several previous comparisons in the context of continuous clustering variables have shown that *K*-means and *K*-median clustering generally provide better recovery of cluster structure than ANNs. For example, a simulation comparison reported by Balakrishnan, Cooper, Jacob, and Lewis (1994) found that *K*-means clustering outperformed the SOM with respect to cluster recovery. A subsequent study comparing the SOM and *K*-means was conducted by Mingoti and Lima (2006) and supported the Balakrishnan et al. (1994) findings. Balakrishnan, Cooper, Jacob, and Lewis (1996) also report the results of a study showing that *K*-means provided better recovery than an alternative type of ANN that used frequency-sensitive competitive learning. One of the newest and most popular clustering methods is affinity propagation (Frey & Dueck, 2007); however, results reported by Brusco and Köhn (2008a, 2009a) suggest that the *K*-median method typically provides better values of the criterion function that affinity propagation seeks to optimize.

Another significant factor in our decision not to include ANNs in the comparative study is the lack of a definitive model that would be best for neural clustering based on dichotomous measures. Kohonen's (1982) SOM was originally designed for clustering based on continuous variables; however, some efforts have been made to adapt the method for dichotomous data (Lebbah, Bennani, & Rogovschi, 2008; Lebbah, Thiria, & Badran, 2000; Leisch, Weingessel, & Dimitriadou, 1998). The ART1 neural clustering procedure (Carpenter & Grossberg, 1987) is also designed explicitly for dichotomous data, and Du (2010, p. 93) provides a list of several ART1-type clustering algorithms. Unfortunately, there do not appear to be any thorough, simulation-based comparisons of the variety of neural models available for dichotomous data. Therefore, unlike *K*-means, *K*-median, and LCM, where the appropriate clustering criterion is definitive, the appropriate neural modeling criterion is unknown.

To summarize, we deemed it appropriate to focus our comparison on the two most accessible and popular methods (LCM and *K*-means) for clustering dichotomous data, plus one additional method that is closely related to *K*-means and has been touted as effective for dichotomous data. This focus also enabled us to use a larger number of design feature levels, a larger number of cell replicates, and a more generous implementation of the three methods than would have been possible if a large number of methods had been included.

Simulation I

Experimental Design Features

Five experimental design features were manipulated to generate the datasets for the first simulation experiment. These *between dataset* features were:

1. The sample size (levels of $N = 100$, $N = 200$, $N = 400$)

2. The number of latent classes/clusters (levels of $K = 2$, $K = 3$, $K = 4$, $K = 5$, and $K = 6$).

3. The number of clustering variables (levels of $V = 6$, $V = 9$, and $V = 12$)

4. The cluster membership probabilities (levels of $\lambda_k = 1/K$ for $1 \leq k \leq K$ (equal), $\lambda_1 = .6$ with $\lambda_k = .4 / (K - 1)$ for $2 \leq k \leq K$, and $\lambda_1 = .1$ with $\lambda_k = .9 / (K - 1)$ for $2 \leq k \leq K$)

5. The level of error in the cluster structure ($\varepsilon = 5\%$, $\varepsilon = 10\%$, and $\varepsilon = 15\%$).

The selection of the design feature levels is based on an amalgamation of simulation studies from the literature. For example, Steinley (2003) used $N = 300$ is his simulation study, and two of our selected levels ($N = 200$ and $N = 400$) surround this value. The other level of $N = 100$ was added to capture the important instance of small sample sizes. Dimitriadou et al. (2002) used settings of $4 \leq K \leq 6$ in their simulation study; however, we augmented these settings with the levels of $K = 2$ and $K = 3$ because they are fairly common in practice. The levels for the number of variables are comparable to those used in recent simulation studies (Dimitriadou et al., 2002; Steinley & Brusco, 2011a). The cluster membership probabilities are based on the three classic settings for relative cluster sizes (often referred to as cluster *density*) originally devised by Milligan (1980). The most challenging parameter settings to establish are those for the perturbation parameter, ε. In a *perfectly structured* dataset, all objects in any given cluster, k, have exactly the same patterning of zeros and ones across the V variables. For each element in the data matrix, $\mathbf{X}$, the value of x_{ij} is changed from 0 to 1 (or 1 to 0) with probability ε. Accordingly, the structure in the dataset is degraded as ε is increased. The challenge is to choose levels for ε that are small enough to preserve structure in the data, but large enough to differentiate performance among the methods.[5] A full-factorial design associated with each of these six design features produced $5 \times 3^4 = 405$ cells. Twenty datasets were generated for each cell (i.e., 20 replicates per cell), resulting in 8100 unique datasets.

The data generation process was based largely on the methods described by Brusco (2004) and, especially, Dimitriadou et al. (2002). The set of baseline patterns are shown in Table 1. The pattern corresponding to $V = 12$ and $K = 6$ was taken from Dimitriadou et al. (2002, p. 138) and provided the foundation for the development of the other patterns in the table. The *within dataset* design feature was the clustering method, which consisted of levels corresponding to the LCM, K-means, and K-median procedures.

[5] Although phrased in a slightly different way, the study by Dimitriadou et al. (2002) used error levels of 10%, 20%, and 30%. We experimented with these levels but found them to be far too deleterious to cluster recovery on average. Our original experiments were with levels of 2%, 4%, and 6%, but these yielded very high ARI’s for all methods.

Implementation Issues

The LCM, *K*-means, and *K*-median procedures were written as Matlab (MathWorks, Inc., 2005) m-files. All computational results were obtained by implementing these programs on a microcomputer using an Intel 3.4 GHz processor and 8GB of RAM. In the first simulation study, the EM-algorithm for the LCM, *K*-means heuristic, and *K*-median heuristic were implemented assuming the correct number of clusters for each dataset. The EM algorithm and *K*-median heuristic were limited to 20 restarts. The *K*-means heuristic was allotted 5000 restarts based on the recommendation of Steinley (2003).

The key performance measure for Simulation I is the adjusted Rand index (ARI: Hubert & Arabie, 1985; Steinley, 2004), which quantifies the agreement between two partitions. The ARI between two partitions (1 and 2) is computed as follows (see Brusco, 2004):

$$\mathrm{ARI} = \frac{H(\tau_1 + \tau_2) - [(\tau_1 + \tau_3)(\tau_1 + \tau_4) + (\tau_2 + \tau_3)(\tau_2 + \tau_4)]}{H^2 - [(\tau_1 + \tau_3)(\tau_1 + \tau_4) + (\tau_2 + \tau_3)(\tau_2 + \tau_4)]}. \qquad (12)$$

where, $H = N(N\text{-}1)/2$, $\boldsymbol{\tau}_1$ is the number of object pairs in the same cluster in both partitions, $\boldsymbol{\tau}_2$ is the number of object pairs in different clusters in partition 1 and partition 2, $\boldsymbol{\tau}_3$ is the number of object pairs in the same cluster in partition 1 but different clusters in partition 2, and $\boldsymbol{\tau}_4$ is the number of object pairs in the same cluster in partition 2 but different clusters in partition 1. The ARI assumes a value of 1 for perfect agreement, and is 0 for chance agreement.

For each of the 8100 test problems and each of the three clustering methods, the ARI is computed between the partition obtained by the method and the correct (or true) cluster memberships associated with the test problem. Steinley (2004, p. 392) has provided some guidelines for interpreting ARI values in simulation experiments, with thresholds of 0.90, 0.80, and 0.65 provided for excellent, good, and fair recovery, respectively. Values of the ARI below 0.65 are judged to be poor.

Results

Across the 8100 test problems, the average ARI values for the *K*-median, LCM, and *K*-means methods were .8367, .8317, and .8173, respectively. This order of performance was consistent for the criterion corresponding to the number of times each method produced the best ARI value. Defining, for each test problem, ARI* as the best ARI across the three methods, the *K*-median, LCM, and *K*-means procedures matched ARI* for, respectively, 57%, 51%, and 42% of the 8100 problems.

A repeated measures analysis-of-variance (ANOVA) was used to analyze the ARI results (see Brusco, 2004; Steinley, 2006b; Steinley & Brusco, 2008). The within datasets design feature was *clustering method*, with the three levels corresponding to the LCM, *K*-means, and *K*-median methods. The between datasets features were N, K, V, λ_k, and ε. Given the large number of datasets used in the simulation, it is expected that most main effects and almost all interactions will be significant; consequently, Table 2 reports effect sizes ($\hat{\eta}^2$) for each of the design features. Although all higher-order interactions in the full-factorial design were estimated, consistent with Steinley and Brusco (2008), specific interactions are displayed in Table 2 only if they account for at least 1% of the variance (i.e., $\hat{\eta}^2 \geq .01$).

The largest effect sizes were observed for the between datasets main effects of V ($\hat{\eta}^2 = .3308$) and ε ($\hat{\eta}^2 = .3307$). The main effect of K ($\hat{\eta}^2 = .1228$) also accounted for a large portion of the variance. The main effects for the sample size (N) and cluster membership probability (λ_k) did not exceed the 1% cut-off that is commonly used in these types of studies. Table 3 reports the means at each level of all of the design features, which correspond to the main effects. As can be seen, the biggest changes in mean levels align with the largest effects seen in the repeated measures ANOVA (in terms of $\hat{\eta}^2$). The general takeaway being that, regardless of method, it is more difficult to accurately recover groups when there are a large number of clusters and a higher degree of error; however, having more variables helps with cluster recovery.

In addition to the between datasets main effects, there were three two-way interactions that had $\hat{\eta}^2 \geq .01$. The number of clusters by number of variables interaction reflected a sharper decrease in recovery when moving from $K = 2$ to $K = 6$ for fewer variables than in the presence of more variables (e.g., the change in ARI from $K = 2$ to $K = 6$ was .28 for $V = 6$; however, it was only .05 when $V = 12$). The second largest interaction in terms of effect size was the interaction between number of clusters and the probability of cluster membership, with the interaction being driven by a greater difficulty in accurately recovering the true cluster structure when there was one large cluster and several smaller clusters. The final two-way interaction with $\hat{\eta}^2 \geq .01$ was the number of clusters by the error level, where the interaction is driven by the greater drop in recovery experienced when $\varepsilon = .15$ (change of .22 when moving from $K = 2$ to $K = 6$) versus $\varepsilon = .05$ (change of .09 when moving from $K = 2$ to $K = 6$).

For the within datasets features, the cluster method was significant and accounted for approximately three percent of the variation. All of the two way interactions involving the clustering method design had modest effect sizes; however, the three-way interaction corresponding to $method \times K \times \lambda_k$ ($\hat{\eta}^2 = .0801$) had the strongest effect of all three-way interactions. This interaction exhibited the same properties as described above for the two-way between-datasets interaction ($K \times \lambda_k$), with the addition that K-means clustering was the most resilient method when the number of clusters increased and probability of cluster membership was equal, while also being the most vulnerable with increasing numbers of clusters and one large cluster and several smaller clusters – a finding consistent with other studies investigating K-means clustering (see, for instance, Steinley, 2006b).

Further inspection of Table 3, reveals perhaps the most striking aspect of the table involves the comparison of the two non-model-based methods: K-means and K-median. The K-median clustering method yielded a better average ARI than K-means for all levels of all design features. Most notably, the degree of superiority of the K-median method generally increased as the number of clusters increased and as the error level increased. Moreover, there was a marked superiority of the *K*-median method at the 60% and 10% levels for relative cluster size.

The LCM and *K*-median approaches were competitive, and relative performance tended to be influenced by the levels of the design features. The *K*-median method yielded better recovery for $N = 100$ observations ($ARI^{LCM} = .8133$, $ARI^{Kmed} = .8309$) and $N = 200$ observations ($ARI^{LCM} = .8339$, $ARI^{Kmed} = .8366$); however, LCM was superior for $N = 400$ observations ($ARI^{LCM} = .8480$, $ARI^{Kmed} = .8426$). Thus, whereas LCM recovery improves as sample size increases, *K*-median recovery is not as strongly influenced by the increase in *N*.

The results also show that the *K*-median method systematically improved relative to LCM over the range of the number of clusters ($2 \leq K \leq 6$). Although LCM yielded better average recovery for $2 \leq K \leq 3$, *K*-median clustering was superior for the three largest levels for the number of clusters ($K = 4$, $K = 5$ and $K = 6$). Moreover, by $K = 6$, *K*-median clustering was appreciably better ($ARI^{LCM} = .7373$, $ARI^{Kmed} = .7631$). In the presence of only $V = 6$ clustering variables, LCM provided better average recovery than *K*-median clustering ($ARI^{LCM} = .7146$, $ARI^{Kmed} = .6989$), whereas the *K*-median approach was superior at $K = 9$ ($ARI^{LCM} = .8559$, $ARI^{Kmed} = .8807$) and K =12 ($ARI^{LCM} = .9247$, $ARI^{Kmed} = .9305$). The *K*-median clustering method generated a larger average ARI when cluster sizes were equal ($ARI^{LCM} = .8319$, $ARI^{Kmed} = .8441$) and at the 60% cluster density condition (ARI^{LCM} = . 8332, $ARI^{Kmed} = .8381$); however, LCM was better at the 10% condition ($ARI^{LCM} = .8300$, $ARI^{Kmed} = .8280$). Finally, *K*-median clustering provided better average recovery than LCM at all three error levels. The strongest advantage of *K*-median clustering was observed at the lowest level corresponding to 5% error ($ARI^{LCM} = .9347$, $ARI^{Kmed} = .9416$), whereas the performances were closer at the 10% ($ARI^{LCM} = .8485$, $ARI^{Kmed} = .8512$), and 15% ($ARI^{LCM} = .7119$, $ARI^{Kmed} = .7173$) levels of error.

Table 4 provides a comparison of the three methods on two secondary measures of performance: (i) attraction rate, and (ii) computation time. The attraction rate is the percentage of restarts for which each algorithm obtained its best-found solution. For example, if the *K*-median heuristic obtained the same minimum value of Equation (10) for 15 of its 20 restarts for a given dataset, then its attraction rate for that dataset would be 75%. Table 4 shows that, across all 8100 test problems, the average attraction rates for the *K*-median, LCM, and *K*-means methods were 89%, 57%, and 34%, respectively. It is also evident from Table 4 that the attraction rate of the *K*-median method is somewhat less sensitive to parameter settings than the other methods. Moreover, the average attraction rate for the *K*-median method is larger than that of its competitors for all levels of all design features, with one exception: the average attraction rate of 96% for LCM is slightly greater than the 93% attraction rate for the *K*-median method at $K = 2$.

We acknowledge that computation times are affected by hardware and software considerations, as well as our decisions for the number of restarts; however, some examination of the differences among the methods is noteworthy. Across all 8100 test problems, the average computation times for the *K*-median, LCM, and *K*-means methods were .31, .89, and 80.89 seconds, respectively. The average *K*-means time is appreciably greater because of its much greater number of restarts. Although the difference between the average *K*-median and LCM computation times is small in an absolute sense, this should not obscure the fact that the average computation time for LCM is nearly three times greater than that of the *K*-median method.

It is especially interesting to observe that the rank order of the three methods from best-to-worst in terms of ARI and attraction rate is identical to the rank ordering of the methods from least-to-greatest in terms of computation time. For example, the *K*-median method yielded the largest average ARI, the largest average attraction rate, and the smallest average computation time. The LCM is second on all three of these measures, and the *K*-means method is third.

Simulation II

Motivation for the Second Simulation Study

Simulation I compared the relative performance of the three methods under the assumption that the correct number of clusters is known. However, in most applications, the correct number of clusters is unknown. For this reason, following the work of Steinley and Brusco (2011a), we conducted a second simulation study (Simulation II) to compare the methods when the number of clusters is assumed unknown. Accordingly, Simulation II was designed to assess the cluster recovery performances of the three methods when used in conjunction with an appropriate rule for choosing the number of clusters.

Experimental Design Features

The between dataset experimental design features and test problems for Simulation II were identical to those from Simulation I. The within dataset design feature was expanded to include different combinations of methods with different rules for choosing the number of clusters.[6] The seven combinations were: (i) LCM using AIC with penalty parameter $\rho = 2$; (ii) LCM using AIC with penalty parameter $\rho = 3$; (iii) LCM using BIC; (iv) *K*-means using CH; (v) *K*-means using AIC3; (vi) *K*-median using MRPC; and (vii) *K*-median using AIC3.

Implementation Issues

Simulation II was conducted using the same hardware and software platform as Simulation I. The LCM algorithm was implemented for $2 \leq K \leq 8$ clusters, using five restarts for each value of *K*. The partition maximizing the log-likelihood across all restarts was obtained for each value of *K*, and the selection of *K* using AIC2, AIC3, or BIC was made accordingly. The *K*-means algorithm was also run for $2 \leq K \leq 8$ clusters, using 100 restarts for each value of *K*. The CH measure was used to choose the number of clusters for the *K*-means_CH combination. Because the CH measure is undefined for $K = 1$, all methods in the comparison were required to select two or more clusters and, therefore, underestimation of the number

[6] It is important to acknowledge that Simulation I provides a purer test of the LCM, *K*-means, and *K*-median approaches because they are each supplied with the true number of clusters. Contrastingly, Simulation II is *simultaneously* evaluating the efficacy of the clustering method and the criterion used for selecting the number of clusters. To help disentangle whether it is the method or the criterion for choosing *K* that leads to superior performance, we also report results for *K*-means and *K*-median clustering using AIC3. Admittedly, these results must be viewed as hybrid approaches because LCM was used to produce the AIC3.

of clusters at the $K = 2$ design feature level was not possible in the simulation. For the *K*-means_AIC3 combination, the value of *K* for the *K*-means algorithm was made based on the LCM_AIC3 result. Similarly, the LCM_AIC3 results were used to select *K* for the *K*-median_AIC3 combination. Finally, for the *K*-median_MRPC combination, 10 restarts of the *K*-median heuristic were run for $2 \leq K \leq 9$ clusters and the MRPC criterion was used to choose *K*. The nine-cluster solution was obtained for the *K*-median method because it is needed to compute the MRPC criterion for $K = 8$. Thus, to ensure that all three methods could select anywhere from $2 \leq K \leq 8$ clusters, it was necessary to have the $K = 9$ result for the *K*-median method.

The LCM_AIC2, LCM_AIC3, and LCM_BIC methods use model-based clustering procedures and an information criterion to choose the number of clusters. The *K*-means_CH and *K*-median_MRPC use a non-model-based clustering procedure and a heuristic to choose the number of clusters. The *K*-means_AIC3 and *K*-median_AIC3 combinations, however, are hybrid methods. They use the model-based clustering procedure (LCM) with AIC3 to choose *K*, but then use a non-model-based clustering method with the selected *K* to obtain the partition.

Like Simulation I, the primary performance measure for Simulation II is the adjusted Rand index. For each of the 8100 test problems and each of the seven combinations of clustering method and criterion for choosing *K*, the ARI is computed between for the partition obtained by the method and the correct (or true) cluster memberships associated with the test problem. A secondary performance measure pertains to the *precision* of recovery of the true number of clusters in the dataset. For each of the LCM_AIC2, LCM_AIC3, LCM_BIC, *K*-means_CH, and *K*-median_MRPC methods, precision is actually defined based on three related sub-measures: (i) the *hit-ratio*, which is the percentage of test problems for which the correct number of clusters was obtained, (ii) the *bias*, which is the average of the raw differences between the correct (or true) number of clusters and the number of clusters selected by the method, and (iii) the mean-absolute-deviation (*MAD*), which is the average of the absolute differences between the correct(or true) number of clusters and the number of clusters selected by the method. The *MAD* and *bias* are measures, respectively, of the average magnitude and direction of error.

Results

A repeated measures ANOVA was conducted using the seven combinations of clustering method and *K* selection rule as the levels of the within datasets design feature. The between datasets design features were the same as those in Simulation I. The patterns of between datasets effect sizes were identical to those displayed in Table 2; furthermore, the magnitudes of the effect sizes were almost identical as well (for instance, $\hat{\eta}^2$for the number of clusters was .1542 versus the .1228 found in Simulation I). Similar results are seen for the within datasets effect sizes, where both the patterning and magnitude correspond closely to what was seen in Simulation I. The sole exception is the much larger effect size for clustering method ($\hat{\eta}^2 = .1603$), an effect that can be attributed to the across the board ARI performance of *K*-means_CH.

Table 5 reports the average ARI values for each level of each of the design features for all combinations of clustering methods and criteria for choosing *K*. We begin with a comparison of the different information criteria used in conjunction with LCM. Across all 8100 datasets, the average recovery measures associated with the AIC3, BIC, and AIC2 criteria were $\text{ARI}^{\text{LCM\_AIC3}}$ = .8260, $\text{ARI}^{\text{LCM\_BIC}}$ = .8115, and $\text{ARI}^{\text{LCM\_AIC2}}$ = .8107, respectively. The AIC3 yielded better average recovery than AIC2 for all levels of all design features with the exception of $V = 6$. The AIC3 also yielded better average recovery than BIC for all levels of all design features with the exception of $N = 400$ (where the two criterion were tied) and $K = 2$.

Relative to the LCM approaches, the overall average recovery performance of *K*-means_CH ($\text{ARI}^{\text{Kmeans\_CH}}$ = .7188) was poor; however, the performance of *K*-means_AIC3 ($\text{ARI}^{\text{Kmeans\_AIC3}}$ = .8147) was much more competitive with the LCM versions. The *K*-median procedure performed reasonably well using MRPC ($\text{ARI}^{\text{Kmedian\_MRPC}}$ = .7952). However, the *K*-median approach was the top performing method overall with respect to average ARI when using LCM in conjunction with AIC3 to select its number of clusters ($\text{ARI}^{\text{Kmedian\_AIC3}}$ = .8288). It is also encouraging to recognize that this level of average recovery differs by less than .01 from the average ARI of .8367 that was realized for the *K*-median method in Simulation I, where the number of clusters was assumed known.

Tables 6, 7, and 8 summarize the precision performance for all combinations of clustering methods and criteria for choosing *K*. Table 6 reports, both overall and for each level of each of the design features, the *hit-ratio* measures, which correspond to the percentage of datasets for which the correct number of datasets was identified. Table 7 reports the average differences between the correct number of clusters and the number selected by the method, which is a measure of the *bias* of the method (i.e., the propensity for too many or too few clusters). Table 8 is similar to Table 7, except that it corresponds to the *MAD* measures, which are averages of absolute differences.

Collectively, Tables 6, 7, and 8 reveal that the use of LCM with the AIC3 criterion yielded the greatest level of precision. Table 6 shows that, across all 8100 datasets, the LCM_AIC3, LCM_AIC2, and LCM_BIC methods yielded the correct number of clusters for 80%, 74%, and 68% of the test problems, respectively. The *K*-median_MRPC method also performed well on this measure, providing the correct number of clusters 76% of the time, while the *K*-means_CH method performed the worst, only estimating the correct number of clusters 49% of the time – a performance so much worse than the competing methods that it can be dropped from consideration for the remainder of the discussion regarding Simulation II. The LCM using AIC3 also yielded the smallest average deviation between the correct number of clusters and the number selected by the method. Table 8 reveals that the average *MAD* values for LCM_AIC3, LCM_AIC2, and LCM_BIC were 0.31, 0.33, and 0.58, respectively. The *K*-median_MRPC method yielded an average *MAD* of 0.65. Table 7 shows that, across all 8100 datasets, the *bias* of 0.55 and 0.61 for LCM_BIC and *K*-median_MRPC, respectively, were very similar to their respective *MAD* values of 0.58 and 0.65. This suggests that the *bias* of these two methods is toward underestimation of the correct number of clusters. The LCM_AIC3 approach also has a propensity for underestimation, but to a lesser degree, with an average *bias* of 0.23. By contrast, LCM_AIC2 has a near zero average *bias* of −0.07, which suggests that it overestimates the number of clusters on average.

Final Caveat Regarding Design Features—The design feature settings for the number of objects (*N*), number of clusters (*K*), number of variables (*V*), and the relative densities/ sizes of the clusters are easily controllable and facilitate the assessment of methods over a range of conditions that might be encountered in practice. By contrast, the design feature corresponding to error is tricky. If not enough error is introduced, then there is an inability to differentiate among methods because recovery of the correct underlying cluster structure is too easy. However, if too much error is introduced, then the issue of recovery becomes thorny as there might actually no longer be a 'correct' cluster structure preserved in the data and, therefore, the relative ARI's of the methods effectively become meaningless.[7]

In Simulation I, the average ARI (across all three methods) at the 5% error condition was .9380, whereas the average at the 15% error condition was .7050. It could be argued that, at the 15% error level condition, the measurement of cluster recovery is becoming tenuous because of the severe degradation in the structure of the clusters. By similar logic, it might be appropriate to consider the results at the 5% error condition as the most reflective of relative performance because of the greatest degree of preservation of a *correct* cluster structure to be recovered. From this perspective, the clear 'winner' in Simulation I is the *K*-median method, and the methods using AIC3 were the top performers in Simulation II. At the 5% error level in Simulation I, the average ARI's for the *K*-median, *K*-means, and LCM methods were .9416, .9364, and .9347, respectively. Likewise, in Simulation II, the average ARI's for *K*-median_AIC3, *K*-means_AIC3, LCM_AIC3, LCM_BIC, LCM_AIC2, *K*-median_MRPC, and *K*-means_CH, were .9317, .9287, .9270, .9243, .9211, .9191, and .9084, respectively.

An Example: Transitive Reasoning

The Data

We consider an example corresponding to measurements for $N = 425$ schoolchildren on $V = 12$ dichotomous variables. The data were originally collected and studied by Verweij et al. (1996) and are available in the mokken package in R (Van der Ark, 2007, 2012). The 12 dichotomous items pertain to transitive reasoning problems that were presented to the children. Specifically, the children were posed with transitivity problems associated with a stimulus set consisting of either three {A, B, C} or four {A, B, C, D} stimuli, and then asked to deduce the relationship for a different subset. For example, the crux of transitive reasoning within the context of a triple of stimuli {A, B, and C} is that, when presented with relationships between two pairs (e.g., {A, B} and {B, C}), children should be able to identify the relationship between the third pair {A, C}. The binary measurement for each child on each item corresponds to the incorrectness ($x_{ij} = 0$) or correctness ($x_{ij} = 1$) of their deduction. The $V = 12$ test items are summarized in Table 9.

[7] This problem is not unique to our study, but pervades most cluster-recovery based simulation experiments in the literature during the past several decades.

Subsequent to the original study by Verweij et al. (1996), the transitive reasoning data have been analyzed by Van der Ark (2012) and Brusco, Köhn, and Steinley (2015). In each of these studies, however, the focus was principally on Mokken (1971) scaling of the 12 dichotomous items rather than the clustering of the 425 schoolchildren. Van der Ark et al. (2008) have identified two components of Mokken scaling: (1) evaluating a collection of ordered items to detect the presence of one or more scales, and (2) follow-up analyses to discover relevant psychometric characteristics of those scales. Thus, the first part of this process can be viewed as a problem of partitioning the scale items, and that was the problem tackled by Van der Ark (2012) and Brusco et al. (2015). Contrastingly, we do not seek to partition the 12 scale items here. Instead, our goal is to compare and contrast the results obtained by LCM, *K*-means, and *K*-median clustering when used to partition the 425 schoolchildren based on their measurements on the 12 dichotomous variables.

Results

All three methods were applied to the transitive reasoning data for $2 \le K \le 8$ clusters. A two-cluster solution for LCM was selected based on the BIC. Likewise, two-cluster solutions for the *K*-means and *K*-median approaches were selected based on the CH and MRPC measures, respectively. Although a two-cluster solution was supported for each of the three methods, the resulting partitions were profoundly different from one another. The ARI between the *K*-means and *K*-median partitions was only .2590. The agreement between the *K*-means and LCM partitions was .0503, whereas the agreement between the LCM and *K*-median partitions was .0657.[8] Table 10 reports the number and percentage of schoolchildren in the complete sample who correctly answered each of the 12 questions. The table also includes, for each of the three methods, the cluster sizes and the percentage of children in each cluster who correctly answered each question.

The LCM partition consisted of two clusters that were substantially different in size (n_1 = 375 and n_2 = 50). Effectively, cluster two of the LCM partition corresponds to a small cluster of children who performed poorly on the test items. The percentage of correct responses for the larger cluster in the LCM partition was 77.2%; however, the corresponding percentage for the second cluster was only 54.8%. Moreover, the results for the individual items shows that the second cluster consists of children that even performed poorly on some of the easiest items (item 7 and item 3, for example).

The *K*-means partition was arguably the least interpretable of the three solutions. The method made a strong separation of the children based on item #10, but was appreciably less differentiating with respect to many of the other variables. By contrast, the *K*-median partitions makes a sharp differentiation between the clusters on the three most differentiating items (i.e., items 10, 11, and 12 which have percentage correct responses closest to 50%).

There is, of course, some degree of subjectivity in the assessment of which of the three partitions is the most appropriate for the transitive reasoning data. Interpretability is an

[8] The ARI's among the three methods improved slightly for larger *K*, but remained well below the threshold of .65 for adequate agreement recommended by Steinley (2004). For example,, at *K* = 4, the ARI's among the pairs (LCM, *K*-means), (LCM, *K*-median), and (*K*-means, *K*-median) were .2400, .2298, and .4480, respectively.

important factor, but that alone is not sufficient to select a model. Both the LCM and *K*-median partitions are quite interpretable, yet they are very different. Another criterion that is sometimes of relevance in cluster analysis is substantiality (see Wedel & Kamakura, 2000). This criterion pertains to the relative sizes of the clusters. For the LCM partition, the smaller cluster (n_2 = 50) contains only 12% of the students in the overall sample. Effectively, the LCM has peeled off a very small subset of poor performers. By contrast, the smaller cluster (n_2 = 178) in the *K*-median partition is far more substantial, containing 42% of the students in the overall sample. This factor, in conjunction with the ability of the *K*-median partition to differentiate with respect to the most challenging questions, could be purported as a basis for choosing the *K*-median solution as the most appropriate for the application.

Conclusions

Summary of Major Findings

Two simulation comparisons of latent class, *K*-means, and *K*-median methods for clustering dichotomous data were completed: (i) Simulation I applied the methods using the correct number of clusters, and (ii) Simulation II required selection of the number of clusters as part of the model-fitting process. The three methods were also compared on a real dataset. A summary of findings is as follows:

1. All three methods are capable of providing good cluster recovery when the number of clusters is assumed to be known. The overall average ARI values ranged from .8173 for *K*-means to .8367 for *K*-median clustering when this assumption was made.

2. When the number of clusters was assumed to be known, *K*-median clustering outperformed *K*-means clustering with respect to average recovery of the underlying cluster structure across all design feature levels.

3. When the number of clusters was assumed to be known, LCM and *K*-median clustering were competitive, with the latter method holding a slight advantage on average. However, LCM tended to outperform *K*-median clustering when *K* was 2 or 3, but *K*-median clustering was superior when *K* was 4 or greater.

4. When the number of clusters was assumed to be unknown, LCM provided its best recovery when using AIC3 (ARI = .8260), whereas average recovery dipped slightly for BIC (ARI = .8115) and AIC2 (ARI = .8107). The use of AIC3 with LCM also resulted in the selection of the correct number of clusters 80% of the time, which was better than any other index evaluated.

5. The recovery performance of *K*-means clustering deteriorated substantially when using the CH heuristic to select the number of clusters (ARI = .7188). When using the number of clusters obtained using LCM with AIC3, the recovery performance of *K*-means clustering improved markedly to ARI = .8147.

6. The recovery performance of *K*-median clustering remained quite strong (ARI = .7952) when using the MRPC measure to choose the number of clusters. Moreover, when supplied with the number of clusters selected using the AIC3,

K-median clustering yielded the best average recovery of ARI = .8288. A particularly encouraging finding is that this average recovery differs by less than .01 from the average *K*-median recovery of .8367 in Simulation I, where the correct number of clusters was assumed to be known.

7. When applied to the transitive reasoning data, the three methods yielded profoundly different two-cluster solutions. The *K*-means solution was difficult to interpret. The LCM solution consisted of one large cluster and a small cluster of poor performers. The *K*-median clustering method produced two clusters that were differentiated by performance on some of the most discriminating items. The key takeaway from this application, in conjunction with the simulation results, is that psychological researchers cannot make the assumption that the three methods will generally produce comparable partitions for dichotomous data.

Limitations and Extensions

Some of the findings of our simulation experiment differ from those obtained in a comparison of *K*-means and mixture-model clustering using data that were not dichotomous, but rather continuous measures generated from mixtures of normal distributions (Steinley & Brusco, 2011a). For example, the BIC measure performed poorly in the Steinley and Brusco study, but performed much better in our simulation experiments. Nevertheless, the BIC was outperformed by the AIC3 measure, which is consistent with the findings of Andrews and Currim (2003a). Another difference between our results and those from the Steinley and Brusco (2011a) study pertains to the CH measure. Whereas this measure generally yielded good results when used with *K*-means in the Steinley and Brusco (2011a) study, it was ineffective in our experiments. These findings are concordant, however, with the results of Dimitriadou et al. (2002), who reported mediocre performance for CH in their experiments with dichotomous data. Accordingly, we recommend caution regarding the use of CH with dichotomous data.

One of the advantages of LCM is that it is flexible and can be adapted easily to accommodate other objectives, such as those that include both metric and nonmetric variables. The LCM can also be extended to allow for repeated measurements and the estimation of trajectories over time (see, for example, Martin-Storey & Crosnoe, 2015). The *K*-median procedure can also be adapted and customized for problem-specific situations. For example, Blanchard, Aloise, and DeSarbo (2012) developed an extension known as the heterogeneous *p*-median model that is particularly well-suited for applications pertaining to categorization tasks. Moreover, *K*-median clustering is broadly applicable to very general proximity data, including asymmetric and rectangular dissimilarity matrices (see Köhn et al., 2010).

Another advantage of LCM (and *K*-means) relative to *K*-median clustering is software accessibility. As noted previously, programs such as Mplus and Latent Gold are generally accessible for latent class analysis, and *K*-means clustering is available in nearly all of the major commercial software packages (see Steinley, 2003). By contrast, *K*-median software is often not commercially available and, instead, must be obtained from other sources. For example, Kaufman and Rousseeuw (2005) offer an R implementation of a *K*-median heuristic, and Köhn et al. (2010) have described a suite of Matlab programs for *K*-median clustering.

Although the accessibility and generalizability of *K*-median clustering is undoubtedly less than that of LCM, the results from our experiments clearly reveal that *K*-median clustering yields comparable and, in some cases, better performance than LCM when applied to dichotomous data. When the true number of clusters was assumed known, the simulation results showed that *K*-median clustering provided, on average, slightly better cluster recovery than LCM. Moreover, the results showed that *K*-median clustering was superior when there were four or more clusters in the dataset.

Given the number of studies in the psychological literature that have recently used LCM to cluster dichotomous data, the results reported herein suggest that *K*-median clustering should at least be considered as an alternative approach. Perhaps one strategy is to use LCM to obtain a partition and select the number of clusters (*K*). Subsequently, a *K*-median clustering solution could be obtained for that same number of clusters. If the LCM and *K*-median solutions are concordant, then additional confidence in the clustering process is established.

In cases of discordant LCM and *K*-median partitions, great care is necessary in the selection of a solution. In a purely exploratory context, a researcher might opt for the solution that has the more salient interpretation. However, even here the researcher is exerting a degree of subjective bias in the selection process. This problem is even more severe if the cluster analysis is seeking to confirm a hypothesized structure in the data. Although there is no definitive solution to this problem, one approach that is commonly used is to profile the clusters using variables that are external to the clustering process. For example, in the case of the transitive reasoning data, a researcher could profile the clustering variables on external variables such as course grades in mathematics, IQ measures, or any other measures or constructs that might be deemed to be theoretically related to transitive reasoning. These profiles could potentially uncover differences between the partitions that might make the selection process less arbitrary.

Acknowledgments

The authors are grateful for the helpful comments of three reviewers and the Action Editor, which led to substantial improvements in this article.

References

Akaike, H. Information theory and an extension of the maximum likelihood principle. In: Petrov, BN., Csaki, BF., editors. Second international symposium on information theory. Budapest: Academiai Kiado; 1973. p. 267-281.

Andrews RL, Brusco MJ, Currim IS. Amalgamation of partitions from multiple segmentation bases: A comparison of model-based and non-model based procedures. European Journal of Operational Research. 2010; 201:608–618.

Andrews RL, Brusco MJ, Currim IS, Davis B. An empirical comparison of methods for clustering problems: Are there benefits from having a statistical model? Review of Marketing Science. 2010; 8:1–32.

Andrews RL, Currim IS. Retention of latent segments in regression-based marketing models. International Journal of Research in Marketing. 2003a; 20:315–321.

Andrews RL, Currim IS. A comparison of segment retention criteria for finite mixture logit models. Journal of Marketing Research. 2003b; 40:235–243.

Arabie P, Carroll JD. MAPCLUS: A mathematical programming approach to fitting the ADCLUS model. Psychometrika. 1980; 45:211–235.

Balakrishnan PV, Cooper MC, Jacob VS, Lewis PA. A study of the classification capabilities of neural networks using unsupervised learning: A comparison with K-means clustering. Psychometrika. 1994; 59:505–525.

Balakrishnan PV, Cooper MC, Jacob VS, Lewis PA. Comparative performance of the FSCL neural net and K-means algorithm for market segmentation. European Journal of Operational Research. 1996; 93:346–367.

Banfield JD, Raftery AE. Model-based Gaussian and non-Gaussian clustering. Biometrics. 1993; 49:803–821.

Bartholomew, DJ., Knott, M. Latent variable models and factor analysis. London: Arnold; 1999.

Blanchard SJ, Aloise D, DeSarbo WS. The heterogeneous *p*-median problem for categorization based clustering. Psychometrika. 2012; 77:741–762.

Bock, HH. Clustering methods: a history of K-means algorithms. In: Brito, P.Bertrand, P.Cucumel, C., DeCarvalho, F., editors. Selected contributions in data analysis and classification. Heidelberg, Germany: Springer; 2007. p. 161-172.

Bradshaw CP, Waasdorp TE, O’Brennan LM. A latent class approach to examining forms of peer victimization. Journal of Educational Psychology. 2013; 105:839–849. [PubMed: 25414522]

Brusco MJ. Clustering binary data in the presence of masking variables. Psychological Methods. 2004; 9:510–523. [PubMed: 15598102]

Brusco MJ, Cradit JD. Graph coloring, minimum-diameter partitioning, and the analysis of confusion matrices. Journal of Mathematical Psychology. 2004; 48:301–309.

Brusco MJ, Köhn HF. Comment on ‘Clustering by passing messages between data points’. Science. 2008a; 319:726c.

Brusco MJ, Köhn HF. Optimal partitioning of a data set based on the *p*-median model. Psychometrika. 2008b; 73:89–105.

Brusco MJ, Köhn HF. Exemplar-based clustering via simulated annealing. Psychometrika. 2009a; 74:457–475.

Brusco MJ, Köhn HF. Clustering qualitative data based on binary equivalence relations: A neighborhood search heuristic for the clique partitioning problem. Psychometrika. 2009b; 74:685–703.

Brusco MJ, Köhn HF, Steinley D. An exact algorithm for item selection within the framework of the monotone homogeneity model. Psychometrika. 2015; 80:949–967. [PubMed: 25850618]

Brusco M, Steinley D. A variable neighborhood search method for generalized blockmodeling of two-mode binary matrices. Journal of Mathematical Psychology. 2007a; 51:325–338.

Brusco MJ, Steinley D. A comparison of heuristic procedures for minimum within-cluster sums of squares partitioning. Psychometrika. 2007b; 72:583–600.

Calinski T, Harabasz J. A dendrite method for cluster analysis. Communications in Statistics. 1974; 3:1–27.

Carpenter GA, Grossberg S. A massively parallel architecture for a self-organizing neural pattern recognition machine. Computer Vision, Graphics, and Image Processing. 1987; 37:54–115.

Ceulemans E, Van Mechelen I. Hierarchical classes models for three-way three-mode binary data: interrelations and model selection. Psychometrika. 2005; 70:461–480.

Ceulemans E, Van Mechelen I. CLASSI: A classification model for the study of sequential processes and individual differences therein. Psychometrika. 2008; 73:107–124.

Ceulemans E, Van Mechelen I, Leenen I. The local minima problem in hierarchical classes analysis: An evaluation of a simulated annealing algorithm and various multistart procedures. Psychometrika. 2007; 72:377–391.

Chapman BP, Goldberg LR. Replicability and 40-year predictive power of childhood ARC types. Journal of Personality and Social Psychology. 2011; 101:593–606. [PubMed: 21744975]

Chiu CY, Douglas JA, Li X. Cluster analysis for cognitive diagnosis: theory and applications. Psychometrika. 2009; 74:633–665.

Coombs, CH. A theory of data. New York: Wiley; 1964.

Curry DJ. Some statistical considerations in clustering with binary data. Multivariate Behavioral Research. 1976; 11:175–188. [PubMed: 26821670]

De Boeck P, Rosenberg S. Hierarchical classes: Model and data analysis. Psychometrika. 1988; 53:361–381.

De Los Reyes A, Bunnell BE, Beidel DC. Informant discrepancies in adult social anxiety disorder assessments: links with contextual variations in observed behavior. Journal of Abnormal Psychology. 2013; 122:376–386. [PubMed: 23421526]

Dempster AP, Laird NM, Rubin DB. Maximum likelihood from incomplete data via the EM algorithm (with discussion). Journal of the Royal Statistical Society B. 1977; 39:1–38.

Dimitriadou E, Dolničar S, Weingessel A. An examination of indices for determining the number of clusters in binary data sets. Psychometrika. 2002; 67:137–160.

Dimitriadou, E., Hornik, K., Leisch, F., Meyer, D., Weingessel, A. Misc functions of the department of statistics (e1071) TU Wien. 2014. p. e1071R package version 1.6-2: URL http://CRAN.R-project.org/package=e1071

Doreian, P., Batagelj, V., Ferligoj, A. Generalized blockmodeling. Cambridge, UK: Cambridge University Press; 2005.

Du KL. Clustering: a neural network approach. Neural Networks. 2010; 23:89–107. [PubMed: 19758784]

Ellis J. An inequality for correlations in unidimensional monotone latent variable models for binary variables. Psychometrika. 2014; 79:303–316. [PubMed: 24659373]

Eshghi A, Haughton D, Legrand P, Skaletsky M, Woolford S. Identifying groups: a comparison of methodologies. Journal of Data Science. 2011; 9:271–291.

Forgy, EW. Biometric Society Meeting. Vol. 21. Riverside, CA: 1965. Cluster analyses of multivariate data: Efficiency versus interpretability of classifications; p. 7681965Abstract in Biometrics

Goodman L. The analysis of systems of qualitative variables when some of the variables are unobservable. Part I: A modified latent structure approach. American Journal of Sociology. 1974; 79:1179–1259.

Grossberg S. Adaptive pattern classification and universal recording: I. Parallel development and coding of neural feature detectors. Biological Cybernetics. 1976a; 23:121–134. [PubMed: 974165]

Grossberg S. Adaptive pattern classification and universal recording: II Feedback, expectation, olfaction, and illusions. Biological Cybernetics. 1976b; 23:187–202. [PubMed: 963125]

Hakimi SL. Optimum locations of switching centers and the absolute centers and medians of a graph. Operations Research. 1964; 12:450–459.

Hakimi SL. Optimum distribution of switching centers in a communication network and some related graph theory problems. Operations Research. 1965; 13:462–475.

Hansen P, Delattre M. Complete-link cluster analysis by graph coloring. Journal of the American Statistical Association. 1978; 73:397–403.

Hansen P, Mladenović N. Variable neighborhood search for the *p*-median. Location Science. 1997; 5:207–226.

Hansen P, Mladenović N. J-Means: A new local search heuristic for minimum sum of squares clustering. Pattern Recognition. 2001; 34:405–413.

Hartigan, JA. Clustering algorithms. New York: Wiley; 1975.

Hartigan JA, Wong MA. Algorithm AS136: A *k*-means clustering program. Applied Statistics. 1979; 28:100–128.

Hedges LV, Olkin I. Clustering estimates of effect magnitude from independent studies. Psychological Bulletin. 1983; 93:563–573.

Hubert L. Problems of seriation using a subject by item response matrix. Psychological Bulletin. 1974; 81:976–983.

Hubert LJ, Arabie P. Comparing partitions. Journal of Classification. 1985; 2:193–218.

Jancey RC. Multidimensional group analysis. Australian Journal of Botany. 1966; 14:127–130.

Kaufman, L., Rousseeuw, PJ. Finding groups in data: An introduction to cluster analysis. 2nd. New York: Wiley; 2005.

Kogan, J. Introduction to clustering large and high-dimensional data. New York, NY: Cambridge University Press; 2007.

Kohonen T. Self-organized formation of topologically correct feature maps. Biological Cybernetics. 1982; 43:59–69.

Kohonen T. The self-organizing map. Proceedings of the IEEE. 1990; 78:1464–1480.

Köhn HF, Chiu CY, Brusco MJ. Heuristic cognitive diagnosis when the Q matrix is unknown. British Journal of Mathematical and Statistical Psychology. 2015; 68:268–291. [PubMed: 25496248]

Köhn HF, Steinley D, Brusco MJ. The *p*-median model as a tool for clustering psychological data. Psychological Methods. 2010; 15:87–95. [PubMed: 20230105]

Krieger AM, Green PE. A generalized Rand-index method for consensus clustering of separate partitions of the same data base. Journal of Classification. 1999; 16:63–89.

Lazarsfeld, PF. The logical and mathematical foundations of latent structure analysis. In: Stouffer, SA., editor. Measurement and prediction. Princeton, NJ: Princeton University Press; 1950. p. 362-412.

Lazarsfeld, PF., Henry, N. Latent structure analysis. Boston: Houghton-Mifflin; 1968.

Lebbah M, Bennani Y, Rogovschi N. A probabilistic self-organizing map for binary data topographic clustering. International Journal of Computational Intelligence and Applications. 2008; 7:363–383.

Lebbah, M., Thiria, S., Badran, F. Topological map for binary data. In: Verleysen, M., editor. Proceedings of the 8th European symposium for artificial neural networks – ESANN 2000. Bruges, Belgium: D-facto publications; 2000. p. 267-272.

Leisch, F., Weingessel, A., Dimitriadou, E. Competitive learning for binary data. In: Niklasson, L.Boden, M., Ziemke, T., editors. Proceedings of the 8th international conference on artificial neural networks - ICANN ’98. London: Springer-Verlag; 1998. p. 779-784.

Lemmens JS, Valkenburg PM, Gentile DA. The Internet gaming disorder scale. Psychological Assessment. 2015 in press.

Linzer DA, Lewis JB. polka: An R package for polytomous variable latent class analysis. Journal of Statistical Software. 2011; 42:1–29.

Lloyd, SP. Least squares quanitization in PCM. Vol. 2. Bell Telephone Labs Memorandum; Murray Hill, NJ: 1957. p. 129-137.Reprinted in IEEE Transactions on Information Theory IT-28 (1982)

MacQueen, JB. Some methods for classification and analysis of multivariate observations. In: Le Cam, LM., Neyman, J., editors. Proceedings of the fifth Berkeley symposium on mathematical statistics and probability. Vol. 1. Berkeley, CA: University of California Press; 1967. p. 281-297.

Macready GB, Dayton CM. The use of probabilistic models in the assessment of mastery. Journal of Educational Statistics. 1977; 33:379–416.

Magidson J, Vermunt JK. Latent class models for clustering: A comparison with K-means. Canadian Journal of Marketing Research. 2002; 20:37–44.

Martin-Storey A, Crosnoe R. Trajectories of overweight and their association with adolescent depressive symptoms. Health Psychology. 2015 in press.

MathWorks, Inc. Using MATLAB. Natick, MA: The MathWorks, Inc; 2005. (Version 7)

McCullough WS, Pitts W. A logical calculus of the ideas immanent in nervous activity. Bulletin of Mathematical Biophysics. 1943; 5:115–133.

McCutcheon, AL. Latent class analysis. Newbury Park, CA: Sage; 1987.

McLachlan, G., Peel, D. Finite mixture models. New York: Wiley; 2000.

Milligan GW. An examination of the effects of six types of error perturbation on fifteen clustering algorithms. Psychometrika. 1980; 45:325–342.

Milligan GW, Cooper MC. An examination of procedures for determining the number of clusters in a data set. Psychometrika. 1985; 50:159–179.

Milligan GW, Cooper MC. A study of the standardization of variables in cluster analysis. Journal of Classification. 1988; 5:181–204.

Mingoti SA, Lima JO. Comparing SOM neural network with fuzzy c-means, K-means, and traditional clustering algorithms. European Journal of Operational Research. 2006; 174:1742–1759.

Mladenović N, Brimberg J, Hansen P, Moreno-Pérez JA. The *p*-median problem: A survey of metaheuristic approaches. European Journal of Operational Research. 2007; 179:927–939.

Mokken, RJ. A theory and procedure of scale analysis. The Hauge/Berlin: Mouton/DeGruyter; 1971.

Muthén, LK., Muthén, BO. Mplus user's guide. 7th. Los Angeles: Author; 1998-2012.

Ohan JL, Cormier N, Hepp SL, Visser TAW, Strain MC. Does knowledge about attention-deficit/hyperactivity disorder impact teachers' reported behaviors and perceptions? School Leadership Quarterly. 2008; 23:436–449.

Porac JF, Thomas H. Cognitive categorization and subjective rivalry among retailers in a small city. Journal of Applied Psychology. 1994; 79:54–66.

Prochaska JJ, Fromont SC, Delucchi K, Young-Wolff KC, Benowitz NL, Hall S, Bonas T, Hall SM. Multiple risk-behavior profiles of smokers with serious mental illness and motivation for change. Health Psychology. 2014; 33:1518–1529. [PubMed: 24467257]

Ratkowsky DA, Lance GN. A criterion for determining the number of groups in a classification. Australian Computer Journal. 1978; 10:115–117.

Restle F. A metric and an ordering on sets. Psychometrika. 1959; 24:207–220.

Ruiz JP, Chebat JC, Hansen P. Another trip to the mall: a segmentation study of customers based on their activities. Journal of Retailing and Consumer Services. 2004; 11:333–350.

Savage JE, Slutske WS. Personality and gambling involvement: a person-centered approach. Psychology of Addictive Behaviors. 2014; 28:1198–1211. [PubMed: 25134059]

Schepers J, Ceulemans E, Van Mechelen I. Selection among multi-mode partitioning models of different complexities. Journal of Classification. 2008; 25:67–85.

Schepers J, Van Mechelen I. A two-mode clustering method to capture the nature of the dominant interaction pattern in large profile data matrices. Psychological Methods. 2011; 16:361–371. DOI: 10.1037/a0024446 [PubMed: 21744969]

Schwartz G. Estimating the dimension of a model. Annals of Statistics. 1978; 6:461–464.

Späth, H. Cluster analysis algorithms for data reduction and classification of objects. Chichester, England: Ellis Horwood; 1980.

Steinhaus H. Sur la division des corps matériels en parties. Bulletin de l'Académie Polonaise des Sciences, Classe III Mathématique, Astronomie, Physique, Chimie, Géologie, et Géographie. 1956; IV(12):801–804.

Steinley D. Local optima in *K*-means clustering: What you don't know may hurt you. Psychological Methods. 2003; 8:294–304. [PubMed: 14596492]

Steinley D. Properties of the Hubert-Arabie adjusted Rand index. Psychological Methods. 2004; 9:386–396. [PubMed: 15355155]

Steinley D. *K*-means clustering: A half-century synthesis. British Journal of Mathematical and Statistical Psychology. 2006a; 59:1–34. [PubMed: 16709277]

Steinley D. Profiling local optima in *K*-means clustering: Developing a diagnostic technique. Psychological Methods. 2006b; 11:178–192. [PubMed: 16784337]

Steinley D, Brusco MJ. Initializing *K*-means batch clustering: A critical analysis of several techniques. Journal of Classification. 2007; 24:99–121.

Steinley D, Brusco MJ. A new variable weighting and selection procedure for K-means cluster analysis. Multivariate Behavioral Research. 2008:77–108. [PubMed: 26788973]

Steinley D, Brusco MJ. Evaluating mixture-modeling for clustering: Recommendations and cautions. Psychological Methods. 2011a; 16:63–79. [PubMed: 21319900]

Steinley D, Brusco MJ. Choosing the number of clusters in *K*-means clustering. Psychological Methods. 2011b; 16:271–285.

Templin JL, Henson RA. Measurement of psychological disorders using cognitive diagnosis models. Psychological Methods. 2006; 11:287–305. [PubMed: 16953706]

Templin J, Henson R, Douglas J. General theory and estimation of cognitive diagnosis models: Using Mplus to retrieve model estimates. 2007 Unpublished manuscript.

Thorndike RL. Who belongs in the family? Psychometrika. 1953; 18:267–276.

Vande Gaer E, Ceulemans E, Van Mechelen I, Kuppens P. The CLASSI-N method for the study of sequential processes. Psychometrika. 2012; 77:85–105.

Van der Ark LA. Mokken scale analysis in R (version 2.4). Journal of Statistical Software. 2007; 20:1–19.

Van der Ark LA. New developments in Mokken scale analysis in R. Journal of Statistical Software. 2012; 48:1–27.

Van der Ark LA, Croon MA, Sijtsma K. Mokken scale analysis for dichotomous items using marginal models. Psychometrika. 2008; 73:183–208. [PubMed: 20046851]

Vermunt, JK., Magidson, J. Latent class cluster analysis. In: Hagenaars, JA., McCutcheon, AL., editors. Applied latent class analysis. Cambridge, England: Cambridge University Press; 2005a. p. 89-106.

Vermunt, JK., Magidson, J. Technical guide for Latent Gold 4.0: Basic and advanced. Statistical Innovations Inc; Belmont, Massachusetts: 2005b.

Verweij AC, Sijtsma K, Koops W. A Mokken scale for transitive reasoning suited for longitudinal research. International Journal of Behavioral Development. 1996; 19:219–238.

Waasdorp TE, Bradshaw CP. Examining student responses to frequent bullying: a latent class approach. Journal of Educational Psychology. 2011; 103:336–352.

Wedel, M., Kamakura, WA. Market segmentation: Conceptual and methodological foundations. Boston, MA: Kluwer; 2000.

Wedel M, Steenkamp JBEM. Fuzzy clusterwise regression approach to benefit segmentation. International Journal of Research in Marketing. 1989; 6:241–258.

White A, Murphy TB. BayesLCA: An R package for Bayesian latent class analysis. Journal of Statistical Software. 2014; 61:1–28.

Wilderjans TF, Ceulemans E, Van Mechelen I. The CHIC model: A global model for coupled binary data. Psychometrika. 2008; 73:729–751.

Wilderjans TF, Ceulemans E, Meers K. CHull: A generic convex hull based model selection method. Behavior Research Methods. 2013; 45:1–15. [PubMed: 23055156]

Wilderjans TF, Ceulemans E, Van Mechelen I. The *SIMCLAS* model: Simultaneous analysis of coupled binary data matrices with noise and heterogeneity between and within data blocks. Psychometrika. 2012; 77:724–740.

Williams GW, Barton GM, White AA, Hosik W. Cluster analysis applied to symptom ratings of psychiatric patients: An evaluation of its predictive ability. British Journal of Psychiatry. 1976; 129:178–185. [PubMed: 9175]

Table 1

Patterns for combinations of *K* and *V*. The pattern for *K* = 6 and *V* = 12 is from Dimitriadou et al., 2002, p. 138) and served as the foundation for construction of the other patterns.

	***V* = 6**	***V* = 9**	***V* = 12**
K = 2	1 1 1 1 0 0	1 1 1 1 1 1 0 0 0	1 1 1 1 1 1 1 1 1 0 0 0
	0 0 1 1 1 1	0 0 0 1 1 1 1 1 1	0 0 0 1 1 1 1 1 1 1 1 1
K = 3	1 1 1 1 0 0	1 1 1 1 1 1 0 0 0	1 1 1 1 1 1 0 0 0 0 0 0
	0 0 0 0 1 1	0 0 0 0 0 0 1 1 1	0 0 0 0 0 0 1 1 1 1 1 1
	0 0 1 1 1 1	0 0 0 1 1 1 1 1 1	0 0 0 1 1 1 1 1 1 0 0 0
K = 4	1 1 1 1 0 0	1 1 1 1 1 1 0 0 0	1 1 1 1 1 1 0 0 0 0 0 0
	0 0 0 0 1 1	0 0 0 0 0 0 1 1 1	0 0 0 0 0 0 1 1 1 1 1 1
	0 0 1 1 1 1	0 0 0 1 1 1 1 1 1	0 0 0 1 1 1 1 1 1 0 0 0
	1 1 0 0 0 0	1 1 1 0 0 0 0 0 0	1 1 1 0 0 0 0 0 0 1 1 1
K = 5	1 1 1 1 0 0	1 1 1 1 1 1 0 0 0	1 1 1 1 1 1 0 0 0 0 0 0
	0 0 0 0 1 1	0 0 0 0 0 0 1 1 1	0 0 0 0 0 0 1 1 1 1 1 1
	0 0 1 1 1 1	0 0 0 1 1 1 1 1 1	0 0 0 1 1 1 1 1 1 0 0 0
	1 1 0 0 0 0	1 1 1 0 0 0 0 0 0	1 1 1 0 0 0 0 0 0 1 1 1
	0 0 1 1 0 0	0 0 0 1 1 1 0 0 0	0 0 0 1 1 1 0 0 0 1 1 1
K = 6	1 1 1 1 0 0	1 1 1 1 1 1 0 0 0	1 1 1 1 1 1 0 0 0 0 0 0
	0 0 0 0 1 1	0 0 0 0 0 0 1 1 1	0 0 0 0 0 0 1 1 1 1 1 1
	0 0 1 1 1 1	0 0 0 1 1 1 1 1 1	0 0 0 1 1 1 1 1 1 0 0 0
	1 1 0 0 0 0	1 1 1 0 0 0 0 0 0	1 1 1 0 0 0 0 0 0 1 1 1
	0 0 1 1 0 0	0 0 0 1 1 1 0 0 0	0 0 0 1 1 1 0 0 0 1 1 1
	1 1 0 0 1 1	1 1 1 0 0 0 1 1 1	1 1 1 0 0 0 1 1 1 0 0 0

Table 2

Repeated measures ANOVA and effect sizes for Simulation I.

Effect	Source	*DF*	*SS*	*F*	$\hat{\eta}^2$
Between data sets effects	Sample Size (N)	2	1.90	141.07	*
	Number of Clusters (K)	4	83.36	3088.03	.1228
	Number of Variables (V)	2	224.60	16639.30	.3308
	Cluster Size (λ_k)	2	1.55	114.92	*
	Error Level (ε)	2	224.50	16631.90	.3307
	$K \times V$	8	27.42	507.90	.0404
	$K \times \lambda_k$	8	15.73	291.25	.0232
	$K \times \varepsilon$	8	11.19	207.19	.0165
	Model Error	7695	51.93		
Within data sets effects	Clustering Method (Method)	2	1.64	433.35	.0323
	Method × N	4	0.39	51.72	*
	Method × K	8	1.51	99.21	.0296
	Method × V	4	1.50	197.81	.0295
	Method × λ_k	4	1.68	221.34	.0330
	Method × ε	4	0.87	114.79	.0171
	Method × $K \times \lambda_k$	16	4.08	134.36	.0801
	Method × $K \times V \times \lambda_k$	32	1.20	19.74	.0235
	Method × $\lambda_k \times \varepsilon$	8	1.31	86.38	.0258
	Method × $K \times \lambda_k \times \varepsilon$	32	2.81	46.22	.0551
	Method × $K \times V \times \lambda_k \times \varepsilon$	64	0.56	4.59	.0110
	Error (Method)	15390	29.19		

Note – all between datasets interactions for which $\hat{\eta}^2 < .01$ are suppressed from the table, as are all within datasets interactions involving two or more between datasets design features. An asterisk (*) in the $\hat{\eta}^2$ column indicates a between datasets main effect or within datasets two-way interaction for which $\hat{\eta}^2 < .01$.

Table 3

Summary of results for Simulation I: The table reports the average of the ARI values for the LCM, *K*-means, and *K*-median methods for each category level of each design feature.

Design feature	Level	LCM	*K*-means	*K*-median
Sample size	$N = 100$	.8133	.8071	.8309
	$N = 200$	.8339	.8193	.8366
	$N = 400$	.8480	.8255	.8426
Number of clusters	$K = 2$	.9290	.8830	.9109
	$K = 3$	.8758	.8668	.8740
	$K = 4$	.8390	.8252	.8419
	$K = 5$	.7775	.7710	.7937
	$K = 6$	.7373	.7406	.7631
Number of variables	$V = 6$	.7146	.6778	.6989
	$V = 9$	.8559	.8566	.8807
	$V = 12$	.9247	.9176	.9305
Cluster sizes	Equal	.8319	.8435	.8441
	60%	.8332	.8003	.8381
	10%	.8300	.8081	.8280
Error level	5%	.9347	.9364	.9416
	10%	.8485	.8336	.8512
	15%	.7119	.6820	.7173
Overall		.8317	.8173	.8367

Table 4

Summary of results for Simulation I: The table reports the average attraction rates (percentage of restarts yielding the best-found solution) and average computation times for the LCM, *K*-means, and *K*-median methods for each category level of each design feature.

		Average attraction rates			Average computation times		
Design feature	**Level**	**LCM**	***K*-means**	***K*-median**	**LCM**	***K*-means**	***K*-median**
Sample size	$N = 100$	48%	35%	87%	0.28	39.13	0.08
	$N = 200$	59%	34%	89%	0.70	71.05	0.22
	$N = 400$	65%	32%	91%	1.69	132.49	0.64
Number of clusters	$K = 2$	96%	62%	93%	0.08	33.17	0.18
	$K = 3$	72%	40%	88%	0.44	52.93	0.25
	$K = 4$	50%	28%	88%	0.83	77.60	0.31
	$K = 5$	40%	21%	88%	1.27	105.64	0.38
	$K = 6$	28%	16%	89%	1.81	135.11	0.45
Number of variables	$V = 6$	47%	11%	85%	1.33	77.34	0.29
	$V = 9$	57%	37%	91%	0.78	81.34	0.32
	$V = 12$	67%	53%	90%	0.55	83.99	0.34
Cluster sizes	Equal	67%	41%	89%	0.65	81.14	0.31
	60%	42%	27%	88%	1.29	80.65	0.33
	10%	62%	32%	90%	0.72	80.88	0.31
Error level	5%	51%	35%	79%	0.66	74.53	0.28
	10%	58%	36%	93%	0.86	80.57	0.32
	15%	61%	30%	95%	1.15	87.57	0.35
Overall		57%	34%	89%	0.89	80.89	0.31

Table 5

Summary of results for Simulation II: The table reports, for each category level of each design feature, the average of the ARI values for the all tested combinations of method (LCM, *K*-means, *K*-median) and criterion for choosing *K* (AIC2, AIC3, BIC, CH, MRPC).

Design feature	Level	LCM (AIC2)	LCM (AIC3)	LCM (BIC)	*K*-means (CH)	*K*-means (LCM_AIC3)	*K*-median (MRPC)	*K*-median (LCM_AIC3)
Sample size	$N = 100$	.8043	.8069	.7744	.7159	.7809	.8038	.8166
	$N = 200$	.8104	.8287	.8178	.7165	.7966	.8175	.8315
	$N = 400$	.8175	.8424	.8424	.7239	.8081	.8229	.8382
Number of clusters	$K = 2$	.8942	.9273	.9285	.8450	.9026	.8819	.9100
	$K = 3$	.8522	.8693	.8593	.7883	.8624	.8633	.8684
	$K = 4$	.8186	.8282	.8128	.6836	.8015	.8231	.8334
	$K = 5$	.7610	.7693	.7445	.6571	.7295	.7669	.7822
	$K = 6$	.7276	.7359	.7126	.6199	.6801	.7386	.7499
Number of variables	$V = 6$	.7095	.7055	.6810	.6128	.6469	.6799	.6952
	$V = 9$	.8301	.8519	.8341	.7178	.8346	.8523	.8681
	$V = 12$	.8926	.9206	.9195	.8257	.9041	.9121	.9229
Cluster sizes	Equal	.8197	.8240	.8035	.7257	.8238	.8342	.8339
	60%	.8043	.8311	.8262	.7436	.7675	.8087	.8333
	10%	.8081	.8229	.8049	.6870	.7943	.8013	.8190
Error level	5%	.9211	.9270	.9243	.9084	.9191	.9287	.9317
	10%	.8247	.8419	.8275	.7107	.8056	.8307	.8432
	15%	.6864	.7091	.6828	.5373	.6610	.6848	.7114
Overall		.8107	.8260	.8115	.7188	.7952	.8147	.8288

Table 6

Summary of results for Simulation II: Mean cluster recovery precision for each method and each level of each design feature: Part 1 – the average percentage of datasets for which the correct number of clusters was selected.

Design feature	**Level**	**LCM (AIC2)**	**LCM (AIC3)**	**LCM (BIC)**	***K*-means (CH)**	***K*-median (MRPC)**
Sample size	$N = 100$	72%	67%	52%	46%	69%
	$N = 200$	75%	81%	68%	49%	77%
	$N = 400$	75%	91%	82%	51%	82%
Number of clusters	$K = 2$	86%	99%	100%	92%	98%
	$K = 3$	82%	94%	86%	44%	87%
	$K = 4$	79%	83%	68%	33%	72%
	$K = 5$	66%	67%	48%	39%	64%
	$K = 6$	56%	54%	36%	37%	59%
Number of variables	$V = 6$	75%	67%	50%	41%	59%
	$V = 9$	71%	80%	66%	44%	80%
	$V = 12$	75%	91%	87%	61%	89%
Cluster sizes	Equal	81%	89%	78%	62%	92%
	60%	61%	65%	53%	34%	55%
	10%	79%	85%	72%	51%	81%
Error level	5%	84%	86%	80%	84%	92%
	10%	76%	82%	70%	41%	77%
	15%	62%	70%	53%	22%	59%
Overall		74%	80%	68%	49%	76%

Table 7

Summary of results for Simulation II: Mean cluster recovery precision for each method and each level of each design feature: Part 2 – the average of the true number of clusters minus the algorithmically selected number of clusters.

Design feature	Level	LCM (AIC2)	LCM (AIC3)	LCM (BIC)	*K*-means (CH)	*K*-median (MRPC)
Sample size	*N* = 100	0.16	0.52	0.92	0.91	0.72
	N = 200	−0.11	0.18	0.51	1.05	0.61
	N = 400	−0.27	0.00	0.22	1.04	0.50
Number of clusters	*K* = 2	−0.16	−0.01	0.00	−0.43	−0.07
	K = 3	−0.18	0.02	0.12	0.52	0.12
	K = 4	−0.15	0.13	0.41	1.23	0.48
	K = 5	−0.03	0.36	0.84	1.52	0.95
	K = 6	0.15	0.66	1.38	2.17	1.55
Number of variables	*V* = 6	0.22	0.51	0.96	0.83	1.01
	V = 9	−0.15	0.20	0.55	1.31	0.53
	V = 12	−0.29	−0.01	0.14	0.88	0.29
Cluster sizes	Equal	−0.10	0.17	0.41	0.89	0.26
	60%	−0.06	0.34	0.78	1.47	1.24
	10%	−0.06	0.20	0.46	0.65	0.32
Error level	5%	−0.11	0.03	0.21	0.13	0.25
	10%	−0.08	0.20	0.51	1.13	0.60
	15%	−0.04	0.47	0.93	1.75	0.98
Overall		−0.07	0.23	0.55	1.00	0.61

Table 8

Summary of results for Simulation II: Mean cluster recovery precision for each method and each level of each design feature: Part 3 – the average of the absolute deviation of the true number of clusters minus the algorithmically selected number of clusters.

Design feature	**Level**	**LCM (AIC2)**	**LCM (AIC3)**	**LCM (BIC)**	***K*-means (CH)**	***K*-median (MRPC)**
Sample size	$N = 100$	0.38	0.55	0.92	1.28	0.79
	$N = 200$	0.31	0.27	0.55	1.23	0.63
	$N = 400$	0.31	0.11	0.27	1.18	0.53
Number of clusters	$K = 2$	0.16	0.01	0.00	0.43	0.07
	$K = 3$	0.20	0.06	0.14	0.59	0.14
	$K = 4$	0.26	0.20	0.43	1.27	0.50
	$K = 5$	0.41	0.47	0.89	1.60	0.96
	$K = 6$	0.63	0.83	1.44	2.26	1.56
Number of variables	$V = 6$	0.34	0.54	0.97	1.49	1.11
	$V = 9$	0.34	0.29	0.57	1.32	0.54
	$V = 12$	0.31	0.11	0.20	0.88	0.29
Cluster sizes	Equal	0.23	0.17	0.41	0.91	0.26
	60%	0.51	0.55	0.87	1.57	1.25
	10%	0.25	0.21	0.46	1.22	0.43
Error level	5%	0.19	0.18	0.28	0.37	0.26
	10%	0.29	0.26	0.53	1.36	0.63
	15%	0.52	0.50	0.93	1.96	1.06
Overall		0.33	0.31	0.58	1.23	0.65

Table 9

Properties of the 12 variables for the transitive reasoning data from Verweij et al. (1996).

Variable	Variable labels	Stimulus items	Measures	Nature of relationship
1	T01L	3 sticks	length (in cm.): 12, 11.5, 11	A > B > C
2	T02L	4 tubes	length (in cm.): 12, 12, 12, 12	A = B = C = D
3	T03W	3 tubes	weight (in grams): 45, 25, 18	A > B > C
4	T04W	4 cubes	weight (in grams): 65, 65, 65, 65	A = B = C = D
5	T05W	3 balls	weight (in grams): 40, 50, 70	A < B < C
6	T06A	3 discs	area (diameter in cm.): 7.5, 7, 6.5	A > B > C
7	T07L	3 sticks	length (in cm.): 28.5, 27.5, 27.5	A > B = C
8	T08W	3 balls	weight (in grams): 65, 40, 40	A > B = C
9	T09L	4 sticks	length (in cm.): 12.5, 12.5, 13, 13	A = B < C = D
10	T10W	4 balls	weight (in grams): 60, 60, 100, 100	A = B < C = D
11	T11P	pseudo		
12	T12P	pseudo		

Table 10

Two-cluster partitions for the LCM, *K*-means, and *K*-median methods for the transitive reasoning data from Verweij et al. (1996). The 12 variables have been placed in order of ‘hardest to easiest’.

				Percent correct for the LCM clusters		**Percent correct for the *K*-means clusters**		**Percent correct for the *K*-median clusters**	
Variable	**Variable label**	**Overall number correct**	**Overall percent correct**	**cluster 1 (n = 375)**	**cluster 2 (n = 50)**	**cluster 1 (n = 204)**	**cluster 2 (n = 221)**	**cluster 1 (n = 247)**	**cluster 2 (n = 178)**
9	T09L	128	.301	.333	.060	.162	.430	.372	.202
12	T12P	202	.475	.445	.700	.554	.403	.202	.854
10	T10W	221	.520	.576	.100	.000	1.000	.737	.219
11	T11P	273	.642	.632	.720	.642	.643	.838	.371
4	T04W	333	.784	.771	.880	.828	.742	.761	.815
5	T05W	341	.802	.835	.560	.745	.855	.842	.747
2	T02L	344	.809	.808	.820	.809	.810	.814	.803
7	T07L	359	.845	.907	.380	.755	.928	.903	.764
3	T03W	376	.885	.973	.220	.814	.950	.927	.826
1	T01L	400	.941	.987	.600	.907	.973	.964	.910
8	T08W	411	.967	.995	.760	.946	.986	.980	.949
6	T06A	414	.974	1.000	.780	.956	.991	.980	.966
Average				.772	.548	.676	.809	.777	.702

Note – for each of the 12 variables, the table reports: (i) ‘overall number correct’ as the number of students in the full sample of 425 students who got the correct answer, (ii) ‘overall percent correct’ as the percentage of students in the full sample who got the correct answer, and (iii) the percentage of students who got the correct answer in each of the two clusters obtained by each of the three methods (LCM, *K*-means, and *K*-median). The number of students in each cluster is also reported.