Algorithm for Hierarchical Multi-way Divisive Clustering of Document Collections

Kazuaki KISHIDA[1,a)]

Abstract: This paper proposes a novel algorithm of hierarchical divisive clustering, which generates a multi-branch tree, not a binary one, as its output. In order to use the algorithm for clustering large document sets, a spherical k-means clustering algorithm based on a cosine measure is adopted for partitioning recursively the document set from the top to bottom. Also, by selecting automatically the number of clusters in each partitioning according to a criterion, an optimal multi-way branching is determined for each node of the tree. This paper reports an experimental result indicating the effectiveness of the proposed algorithm.

1. Introduction

Tree structures generated by applying a hierarchical clustering algorithm to a document collection (e.g., a set of research articles or web documents) are often useful for applications in information retrieval (IR) and related areas. For instance, if a set of web documents is obtained by entering a query into a search engine, then hierarchical clustering of the set (i.e., a dendrogram) would help the user specify a suitably-sized subset of relevant documents.

However, when the target document set is large, the computational complexity of hierarchical agglomerative clustering (HAC), which is widely used in various areas, becomes very high. In such cases, an algorithm for hierarchical divisive clustering (HDC) may be suitable because its complexity is expected to be lower if the resulting dendrogram is well balanced.

Typically, the entire set is partitioned at first into two parts by a k-means algorithm, and recursively, each part is split by a similar procedure, which is usually called 'bisecting k-means' clustering (Steinbach et al., 2000 [87]; Zhao & Karypis, 2002 [111]). Also, the 'principal direction divisive partitioning (PDDP)' (Boley et al., 1999 [13], [14]) is a well-known hierarchical divisive clustering algorithm, in which each document set is split based on the result of principal component analysis (PCA).

This paper attempts to explore a hierarchical divisive clustering algorithm allowing each document set (i.e., node of a tree) to be partitioned into two or more parts. Since the algorithm for hierarchical multi-way divisive clustering (HMDC) is more flexible than those that divide each node always into just two parts, more valid results are expected to be obtained by the HMDC algorithm. Particularly, as its component, the spherical k-means (spk-means) algorithm (Dhillon & Modha, 2001 [31]) based on a cosine value of two vectors for measuring similarity between two documents is used for the partitioning operation, and the optimal number of clusters in each partitioning is determined by using the ratio of within-cluster dispersion to total dispersion computed from the result of the cosine-based spk-means clustering.

In the next section, the HMDC algorithm is explained. Section 3 reports the results of an experiment confirming the effectiveness of the HMDC algorithm. A set of 6,374 articles extracted from the RCV1 test collection (Lewis, et al., 2004 [61]) was used in the experiment. After discussing the experimental results, some related papers are reviewed.

2. Hierarchical Multi-way Divisive Clustering

2.1 Outline of the algorithm

The basic procedure of the HMDC is to divide each set of documents into two or more parts, which is repeated recursively from the entire set until a full dendrogram whose leaf node at the bottom corresponds to a document (i.e., singleton) is generated. Otherwise, the recursive partitioning can be terminated in a node according to a stopping rule when a sufficiently homogeneous cluster is obtained. In the experiment described below, the stopping rule was used to assess directly the validity of clustering results by external evaluation metrics.

For the partitioning, the spk-means clustering algorithm is used as described above, and the number of parts in each partitioning is determined based on the ratio of within-cluster dispersion to total dispersion (see below).

2.2 Executing k-means clustering

In this paper, term frequency is adopted as the element of document vectors, each of which is always normalized into a unit vector such that $\mathbf{v}_i = \mathbf{d}_i/\|\mathbf{d}_i\|$ where $\mathbf{d}_i = [x_{i1}, \ldots, x_{ij}, \ldots, x_{iM}]^T$ and x_{ij} denotes the occurrence frequency of term t_j in document d_i ($i = 1, \ldots, N$; $j = 1, \ldots, M$). Also, a vector of cluster C_k is computed as

1 School of Library and Information Science, Keio University, Minato-ku, Tokyo 108–8345, Japan
a) kz_kishida@z8.keio.jp

$$\mathbf{c}_k = \sum_{i:d_i \in C_k} \mathbf{v}_i \ . \quad (1)$$

According to the standard IR theory, similarity between a document and a cluster is measured by the cosine coefficient such that $\cos(\mathbf{d}_i, \mathbf{c}_k) = \mathbf{d}_i^T \mathbf{c}_k / (\|\mathbf{d}_i\| \cdot \|\mathbf{c}_k\|) = \mathbf{v}_i^T \mathbf{c}_k / \|\mathbf{c}_k\|$.

In order to execute the spk-means clustering based on the vectors, the present experiment employed a modified version of the Hartigan-Wong algorithm in which the Euclidean distance is replaced with the cosine similarity (see the appendix in Kishida (2014) [57]).

2.3 Determining the number of parts

For partitioning each node into two or more parts, the number of parts (i.e., clusters) has to be automatically determined within the procedures of the HMDC algorithm. Generally, this is a problem of estimating the optimal number of clusters or segments for a given data set, which is not easy to solve. So, many techniques or methods have been proposed (see Section 5.2).

In the case of partitioning medium or large document sets, a computationally efficient method is desirable. Also, it is difficult to assume a probabilistic model such as the Gaussian mixture model for large-scale document clustering. So, rather than using resampling-based methods or model selection techniques, this paper attempts to determine the number of clusters based on a clustering validity indicator like Caliński and Harabasz's index (CH index) [17] which is the ratio of the total within-cluster sum of squared distances about the centroids to the total between-cluster sum of squared distances (Gordon, 1999 [43]).

Since the CH index is defined based on the Euclidean distance, it is necessary to modify it slightly for the cosine similarity. When documents are partitioned into some clusters, the total sum of similarities between all pairs of documents, which is denoted by T_0, can be computed as

$$T_0 = \sum_{i=1}^{N} \sum_{h:d_h \in C[d_i]} \mathbf{v}_i^T \mathbf{v}_h + \sum_{i=1}^{N} \sum_{h:d_h \notin C[d_i]} \mathbf{v}_i^T \mathbf{v}_h \equiv T_1 + T_2, \quad (2)$$

where $C[d_i]$ denotes a cluster including d_i. Namely, the first part T_1 is the sum of similarities between two documents in a same cluster, and the second part T_2 is the sum of similarities between two documents which belong to different clusters.

So, the proportion of T_1 explained by clusters inherent in the set (i.e., $=T_1/(T_1 + T_2)$) can be reasonably employed as an indicator of 'goodness' of the clustering operation because a cluster should be "a set of entities which are alike, and entities from different clusters are not alike" (Xu & Wunsch II, 2009 [105] p.4). One serious problem preventing its actual use is the high complexity of computing T_1 and T_2, for which the inner product of $O(N^2)$ pairs has to be calculated as explicitly suggested by Equation (2).

In order to overcome this problem, this paper computes approximately T_1 and T_2 as

$$W(L) = \sum_{k=1}^{L} \sum_{i:d_i \in C_k} \mathbf{v}_i^T \mathbf{c}_k / \|\mathbf{c}_k\| , \quad \text{and} \quad (3)$$

$$B(L) = \sum_{k=1}^{L} \sum_{i:d_i \notin C_k} \mathbf{v}_i^T \mathbf{c}_k / \|\mathbf{c}_k\| , \quad (4)$$

respectively where L indicates the number of clusters. Therefore, a criterion for selecting the optimal number of clusters is naturally derived as

$$H(L) = \frac{W(L)}{W(L) + B(L)}. \quad (5)$$

More precisely, for a particular document set, the spk-means clustering is repeated with various values of L (e.g., $L = 2, \ldots, 10$), and a partition with

$$L' = \arg\max_L H(L) = \arg\max_L W(L)/[W(L) + B(L)] \quad (6)$$

can be selected as the final result, and L' is considered to be the optimal number of clusters for the set. Namely, in the HMDC algorithm, each document set corresponding to a node of the tree is divided into L' parts defined in Equation (6) after $L_{max} - 1$ executions of the spk-means clustering by varying L such that $L = 2, \ldots, L_{max}$ (note that $L_{max} = 10$ in the experiment described below).

2.4 Terminating recursive partitioning

Automatic termination of recursive partitioning in HDC is also a difficult problem, to which some techniques for estimating the optimal number of clusters reviewed in Section 5.2 may be applied. However, this paper does not explore this research issue deeply, and the experiment adopted the simple stopping rule that "if $H(L') > \theta$, then the document set is treated as a final cluster (i.e., a leaf node in the tree), and the recursive partitioning in the branch is stopped" where θ is a threshold, which means that a value of θ has to be provided a priori before executing the HMDC algorithm.

Generally, when stopping the recursive partitioning based on a predetermined threshold, the number of objects in each cluster, the within-cluster dispersion or the diameter of each cluster can be used (e.g., see Guenóche et al., 1991 [44], Savaresi et al., 2002 [85] and so on). This paper assumes that the document set is sufficiently homogeneous when the value of $H(L')$ is high, which is naturally derived from discussions of this section.

3. Experiment

3.1 Purpose

In the experiment, the effectiveness of the HMDC algorithm was compared to that of bisecting k-means clustering and non-hierarchical (standard) k-means clustering. The algorithm for bisecting k-means clustering in this experiment was the same as that of the HMDC except that L' was always assumed to be two (the same stopping rule was applied). For the non-hierarchical k-means clustering, the spk-means algorithm was used with the predetermined number of clusters (see below).

3.2 Document dataset

The Reuter corpus RCV1 [61] created as a test collection for text categorization was used to measure effectiveness of each algorithm. Since one or more topic codes are assigned to each record of the corpus, which can be considered as 'answers' of clustering, the validity of clusters generated by the algorithms can be assessed based on the topic codes (note that the topic codes

Table 1 Effectiveness of clustering algorithms

Methods	θ	# of clusters	nMI	ARI	BCubed-F
HMDC	0.60	4	0.163	0.118	0.246
(L_{max} = 10)	0.65	102	0.518	0.399	0.464
	0.70	414	0.410	0.101	0.312
	0.75	859	0.377	0.074	0.240
	0.80	1692	0.353	0.041	0.206
Bisecting	0.60	8	0.064	0.029	0.130
(L' = 2)	0.65	18	0.100	0.028	0.119
	0.70	36	0.139	0.032	0.104
	0.75	81	0.161	0.030	0.099
	0.80	184	0.182	0.031	0.086
	0.85	411	0.213	0.033	0.077
	0.90	1031	0.254	0.028	0.063
K-means		68	0.439	0.189	0.268
(non-hierarchical)		102	0.417	0.141	0.221

were used only for evaluation). Particularly, as a test dataset for this experiment, a set of 6,374 records to which just a single topic code is assigned was extracted from news articles published during August 1996 (i.e., $N = 6374$) because evaluation of clustering results including multi-topic documents becomes too complicated. In total, 68 different topic codes appear in the 6,374 records (see Kishida, 2011 [55] for the topic codes).

3.3 Indexing

By standard text processing which consists of tokenization, removing stopwords and stemming by Porter's algorithm, document vectors for clustering were generated from the records. As described above, term frequency was simply used as the element of document vectors, and instead of incorporating the idf factor into the element, non-specific terms appearing in more than 10% of all documents (i.e., over 647 documents) were removed from all document vectors. Also, terms appearing in only one document were not adopted as features for clustering. As a result, in total, 22,503 different terms were included in the set of document vectors and the average document length amounted to 112.10.

3.4 Evaluation metrics

According to a suggestion by Kishida (2014) [57], the experiment employed three external evaluation metrics: nMI (normalized mutual information), ARI (adjusted Rand Index) and BCubed-F. Note that normalization of MI was based on the maximum of entropy scores of two marginal distributions.

3.5 Results

Table 1 indicates values of the three evaluation metrics for clustering results, and the number of remaining nodes (i.e., final clusters) when the recursive partitioning stopped in all nodes, which is referred as "# of clusters". Clearly, it was empirically shown that the HMDC outperformed the bisecting. For example, when $\theta = 0.65$, a very good result, the three metrics of which were 0.518, 0.399 and 0.464 respectively, was obtained by the HMDC. In contrast, values of metrics for results by the bisecting k-means clustering were relatively lower as Table 1 indicates.

Also, the effectiveness of the HMDC would be higher than that of non-hierarchical k-means clustering. In the experiment, when the non-hierarchical k-means clustering was executed with $L = 68$ (the number of 'true' clusters) and $L = 102$ (the number of clusters in the best case of the HMDC within the experiment), their values of the evaluation metrics did not exceed those of the HMDC with $\theta = 0.65$.

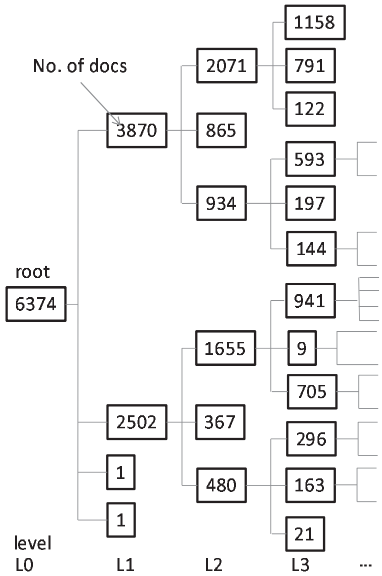

Fig. 1 Dendrogram by HMDC (at the best case, $\theta = 0.65$)

Figure 1 is a portion of a dendrogram obtained by the HMDC algorithm with $\theta = 0.65$, which shows a tree structure in top four levels while the total number of levels was 16. The number in each node indicates the number of documents included in its subset.

4. Discussion

The experiments showed that the HMDC was more effective than top-down bisecting, which is not surprising because the HMDC is more flexible due to multi-way branching. Rather, the results in Table 1 should be interpreted as indicating the success of estimating automatically the number of subsets inherent in each node. However, whether the indicator $H(L)$ in Equation (6) is the best or not is unclear, and further research is needed.

On the other hand, it is clearly difficult to select an appropriate threshold in the stopping rule. Namely, clustering results varied largely with different values of the threshold in the experiment (see Table 1). Although further improvement of it may be necessary for obtaining a good result from the HMDC algorithm, the stopping rule is not needed actually when a 'full' dendrogram whose leaf node is a single document has to be generated.

5. Related Work

5.1 Divisive partitioning

When the target set including n data points (e.g., documents) is partitioned into two nonempty subsets, there are many possible divisions of them, which is computationally expensive [105]. Therefore, without examining every possible division, a partitioning algorithm with less computational complexity is usually used to obtain two approximately valid subsets. Although it may appear that an algorithm for k-means clustering is usually employed for it, namely 'bisecting k-means' clustering (e.g., [87], [111]), actually other algorithms for flat partitioning such as a finite mixture model, nonnegative matrix factorization (NMF) and so on are also available.

Among them, principal component analysis (PCA) has often been applied to hierarchical divisive clustering of document collections, which is called 'principal direction divisive partitioning (PDDP)' (Boley et al., 1999 [13], [14]). In each step of the PDDP, documents contained in a set are classified into two parts depending on whether the first component score is positive or negative. Also, in the NGPDDP (non-greedy version of PDDP) algorithm (Nilsson, 2002 [73]), components other than the first one can be selected for the partitioning according to a criterion on the variance of a set of clusters, and the PDDP(l) algorithm (Zeimpekis & Gallopoulos, 2003 [108]) tries to classify the target set into 2^l parts in each stage where $l \geq 1$. Another extension of the PDDP algorithm is to use kernel PCA, which is a nonlinear version of PCA; the algorithm is called KPDDP(l) (Zeimpekis & Gallopoulos, 2008 [109]). More recently, Tasoulis et al.(2010) [91] explored intensively criteria for selecting a cluster to be split, methods for splitting it, and stopping rules in the PDDP algorithm.

Other than k-means clustering and the PDDP, Cheng et al.(2006) [23] used a spectral clustering algorithm for dividing the target cluster in their procedure combining top-down partitioning and bottom-up merging, while Feng et al.(2010) [35] used an improved discrete particle swarm optimizer, which is a genetic algorithm for clustering, to divide the target node.

5.2 Estimating the optimal number of clusters

5.2.1 Types of estimation

The optimal number of clusters, which is denoted by L', can be selected from several values based on a criterion or rule, or can be determined based on an objective function built into a clustering algorithm. Otherwise, the number of clusters may be posteriorly defined as an output from a clustering algorithm dependent on a threshold, or resampling-based methods providing the number of clusters inherent in a dataset have also been applied. Mirkin (2011) [70] reviewed exhaustively algorithms or methods for estimating the optimal number of clusters, and Mirkin (2013) [71] provided another overview of them.

5.2.2 Criterion for selection

When selecting L' from several values (i.e., $L = 2, \ldots, L_{max}$) based on a criterion, a clustering operation is repeated with individual values of L and the value that provides the clustering result with the minimum (or maximum) score of the criterion is chosen as L' (see Equation (6)). Since the minimum score means the optimal one in the case of Euclidean distance, L' corresponds to an 'elbow' in a curve which is obtained by plotting the criterion scores (on the y-axis) against the values of L (on the x-axis).

Because the within-cluster dispersion, which is often used as an evaluation metric of clustering, decreases monotonically as L increases, the criteria are often computed from a combination of within- and between-cluster dispersion like Caliński and Harabasz's index (CH index) [17]. Actually, Milligan & Cooper (1985) [68] reported a result of empirical comparison between 30 classical criteria proposed before the mid-1980s, most of which are based on between- and within-cluster dispersion measured in the Euclidean space such as the CH index, Hartigan's statistic (Hartigan, 1975 [47]) and so on. After that, Hardy(1996) [46] compared experimentally seven techniques for identifying the number of clusters such as a classical geometric method, a likelihood ratio test for clusters, and so on.

One of the well-known criteria is the Silhouette width (Rousseeuw, 1987 [82]) of a data point, which is basically computed based on dissimilarities between a given data point and other data points in the same cluster and dissimilarities between it and other data points in a different cluster. Pollard & van der Laan (2002) [79] applied the average Silhouette for identifying clusters in gene expression data.

Mirkin (2013) [71] discussed the Gap statistic (Tibshirani et al, 2001 [94]) and Jump statistic (Sugar & James, 2002 [89]) as other criteria based on cluster dispersion. Yan & Ye (2007) [106] modified the Gap statistic by changing slightly the definition of within-cluster homogeneity. While the original Gap statistic uses the logarithm of the within-cluster homogeneity, Mohajer et al. (2010) [72] suggested not applying the logarithm to it.

Pham et al. (2005) [78] proposed another criterion for selecting L', which was computed as the ratio of two cluster distortion values at L and $L-1$. When the curve of criterion scores is smooth with no explicit minimum point (i.e., 'elbow'), it is not possible to determine the optimal number. In order to solve this problem, Salvador & Chan (2004) [84] developed a method for selecting an optimal number as the intersection of two straight lines approximating the left and right sides of the curve, respectively.

5.2.3 Mixture model

A finite mixture model consisting of L components can be used for partitioning a data set, in which the number of components is usually assumed to be the number of clusters. McLachlan (1987) [65] tried to estimate the number of components in a Gaussian mixture model (GMM) from the observed data by using the likelihood ratio test statistic (LRTS) computed in a framework of Bootstrap sampling. A similar technique was also explored by McLachlan & Khan (2004) [66] (see also Lo et al., 2001 [64] for another statistical test).

Another typical strategy for determining the number of components in a mixture model is to apply a model selection technique based on information criteria such as BIC (Bayesian information criterion), AIC (Akaike information criterion) and so on. For instance, a penalty for complexity of the model (i.e., for the number of parameters in it) is incorporated into the BIC, which can be useful for selecting an optimal mixture model. An actual procedure of identifying the optimal number of components based on the BIC in a clustering application was provided by Fraley & Raftery (1998) [38]. Also, various other criteria were explored by Bozdogan(1992) [16], Banfield & Raftery (1993) [7] and so on. Roberts et al.(1998) [80] applied the Bayesian approach to computation of the probability distribution in the GMM, which led to the likelihood including explicitly the number of parameters in the model. Similarly, Biernacki et al.(2003) [12] proposed the 'integrated complete likelihood (ICL)' approximating BIC as a criterion for determining L'. Because the BIC tends to overestimate the value of L', Chiu et al. (2001) [27] attempted to merge clusters based on a distance defined by a log-likelihood function after estimating the 'coarse' number of clusters from the BIC. More recently, Pan & Shen (2007) [74] tried to modify the BIC for estimating L' in 'penalized' model-based clustering.

Also, Xu (1997) [102] explored the method of estimating L' in a Bayesian Ying-Yang (BYY) machine, in which a term including L was incorporated into its objective function for computing the maximum likelihood of a GMM (see also Hu & Xu, 2004, [50] for model selection based on the BYY machine).

Basically, there are many measures for assessing the number of components in mixture models (see Chapter 6 in McLachlan & Peel, 2000 [67]). Such measures can be applied to the problem of determining L' according to the model selection procedure. For example, Bouguila & Ziou (2007) [15] employed MML (minimum message length) for estimating L' in a mixture of general Dirichlet distributions.

Another approach for estimating L' in the framework of GMM is to keep 'rivals' away from the 'winner' to which a data point is allocated in the EM algorithm, which can be considered as a technique of 'rival penalized competitive learning (RPCL)', which was used for estimating L' by Xu et al.(1993) [104]. Xu (1998) [103] extended the algorithm for clusters with more complicated shapes. More recently, Cheung (2003) [24] and Cheung (2005) [25] proposed techniques for fading out redundant densities from a density mixture based on a similar mechanism.

Welling & Kurihara (2009) [100] proposed clustering algorithms that have a stopping rule based on a cost function including L for a GMM, which yields L' automatically. Also, the hierarchical Dirichlet process (HDP) model (Teh, et al., 2006 [92]) allows the number of latent topics to be estimated from a given document set. If the latent topics inherent in the set are used for producing clusters of words or documents, then L' can be considered to be automatically given by the HDP model (see Kishida, 2013 [56]).

Rather than assuming a Gaussian distribution, Herbin et al. (2001) [48] employed a nonparametric Parzen-Rosenblatt window method for kernel density estimation and applied the estimated probabilistic distribution function for segmenting the dataset into some areas. Cuevas et al. (2000) [28] provided an algorithm for estimating L' based on density obtained from a kernel function, and Girolami(2002) [42] explored an unsupervised clustering based on a kernel function and suggested that L' may be determined by examining the distribution of eigenvalues of the kernel matrix.

5.2.4 Stability-based approach

Jain & Moreau(1987) [52] made one of the earliest attempts at applying a 'stability' concept for determining L' under the assumption that partitioning with L' is stable whereas partitioning with other numbers of clusters is not stable. Actually, the stability is measured by an index computed from clustering results for a set of subsamples extracted from the target dataset. In [52], an index based on within-cluster dispersion was calculated from the results of k-means clustering for Bootstrap samples.

As the index, Bel Mufti et al.(2005) [8] examined experimentally a stability measure developed by Bertrand & Bel Mufti (2006) [11], which is based on Loevinger's measure. Also, Pascual et al.(2008) [75] used mutual information (MI) for measuring stability between two clustering results, and similarly, Volkovich et al.(2008) [96] and Volkovich et al.(2011) [97] employed distance measures between two probabilistic distributions for it.

In Levine & Domany (2001) [60], a cluster validity measure was computed from an $N \times N$ matrix, each element of which indicates whether the ith data point and jth data point belong to the same cluster or not (i.e., 'membership'). Similar membership matrices were used in Ben-Hur et al. (2002) [10] and Ben-Hur & Guyon (2003) [9] for determining L'.

There have been many attempts at measuring the stability in a framework of cross-validation which is a standard technique in supervised learning. For example, Roth et al.(2002) [81] divided the entire dataset randomly into two parts and executed a clustering algorithm for them. After that, the result from the second part was used for predicting cluster membership in the first part, and the stability was measured based on the accuracy of the prediction. Similar cross-validation frameworks were adopted by Dudoit & Fridlyand (2002) [33] (in which Fowlkes and Mallows coefficient was used as one of the stability indices), Tibshirani & Walther (2005) [93] (their technical report published in 2001 proposed a metric 'prediction strength'), and Lange et al. (2004) [59] (in which a modified misclassification error was used). Also, Wang(2010) [99] and Fang & Wang(2012) [34] explored intensively the cross-validation approach for determining L'.

By executing repeatedly a k-means algorithm with changing random initialization, it is possible to obtain a set of multiple clustering results, which leads to so-called 'consensus clustering'. If the consensus clustering is also repeated with different values of L, then L' can be determined similarly. Based on the strategy, Kuncheva & Vetrov (2006) [58] tried to estimate L' on data with various cluster shapes (e.g., spiral or half rings), and Steinley (2008) [88] also proposed a procedure for selecting L' from the result of consensus clustering.

Chaea et al.(2006) [22] applied five agglomerative clustering algorithms to subsamples under assumptions of different values of L, and selected L' based on similarity between clustering results of 10 possible pairs of the algorithms.

5.2.5 X-means and related approaches

The k-means clustering can be interpreted as a special case of model-based clustering, and it is possible to combine a criterion like BIC with standard k-means algorithms. Actually, Pelleg & Moore (2000) [77] developed an x-means clustering algorithm with estimating L' based on BIC, and also Ishioka (2005) [51] extended it by adding a post-processing after executing the x-means algorithm in order to merge some over-fragmented clusters.

Several extensions of the k-means algorithm with a function of estimating L' have been developed. For example, the g-means algorithm (Hamerly & Elkan, 2003, [45]) applies a statistical test for determining whether recursive partitioning is stopped or not, the ik-means algorithm (Mirkin, 2005, [69]; Chiang & Mirkin (2010) [26]; Mirkin, 2013, [71]) tries to find desirable initial seeds based on 'anomalous pattern (AP)' method, and the pg-means algorithm (Feng & Hamerly, 2007 [36]) employs the Kolmogorov-Smirnov test for the model selection. Also, Fischer (2011) [37] explored another penalty function.

5.2.6 Fuzzy clustering

In the fuzzy clustering algorithm, automatic estimation of L' has been attempted by using validity measures such as 'fuzzy hypervolume' and 'partition density' (Gath & Geva, 1989 [41]), or 'robust cluster similarity' (Frigui & Krishnapuram, 1996, [39]), which are computed in the framework of fuzzy clustering. Especially, algorithms for estimating the number of objects with complicated shapes in image data have been developed in fuzzy clustering. For example, the 'robust competitive agglomeration (RCA)' algorithm (Frigui & Krishnapuram, 1999, [40]) has a step for removing a cluster with low degree of fuzzy membership based on a threshold. Also, Kaymak & Setnes (2002) [54] estimated L' by merging two clusters between which similarity is higher than a threshold based on a 'volume prototype' representing a complicated shape. Devillez et al.(2002) [29] developed a complicated procedure including hierarchical clustering in order to apply fuzzy clustering to identification of clusters with complicated shapes. In this procedure, 'real' clusters with complicated shapes are automatically identified from the dendrogram.

On the other hand, Sun et al.(2004) [90] applied a standard procedure for finding L' to the fuzzy clustering algorithm, in which a new index based on a linear combination of compactness and separation of clusters was used for measuring the validity of each cluster. Also, Li & Shen (2010) [63] introduced a simple stopping rule based on a threshold for the particular purpose of estimating segmentation of an image by fuzzy clustering.

5.2.7 Genetic algorithm

When applying a non-parametric approach such as a genetic algorithm (GA), some researchers attempted to determine concurrently L' and optimal partitioning of a dataset according to an objective criterion related to the validity of the resulting clusters. In the case of GA, the clustering task is sometimes called 'GCUK-clustering' (e.g., see Bandyopadhyay & Maulik, 2002, [2]) where 'GCUK' is an abbreviation of 'genetic clustering for unknown k' and 'k' denotes the number of clusters. For instance, Bandyopadhyay & Maulik(2001) [1] used a variable string length genetic algorithm (VGA) for it based on the Davies-Bouldin index and Dunn's index (see also Bandyopadhyay & Maulik, 2002, [2]). Also, Kärkkäinen & Fränti (2002) [53] tried to estimate L' in executing the randomized local search (RLS) by employing Davies-Bouldin index and variance-ratio F-test as criteria.

On the other hand, in the case of Hruschka & Ebecken(2003) [49], the 'classic' Silhouette criterion was used for selecting L' in executing a GA algorithm. Especially, Sheng et al.(2005) [86] proposed to use a weighted sum of several normalized cluster validity functions for determining L'.

Bandyopadhyay & Saha (2008) [3] introduced a new cluster validity function incorporating directly L in the framework of GA. The function was called 'Sym', which was also used by combining it with the well-known Xie-Beni index in Saha & Bandyopadhyay (2010) [83]. Such 'multi-objective' GA algorithms were explored also by other researchers (e.g., Banerjee, 2009 [4]; 2010 [5]; 2012 [6]).

Actually, Casillas et al.(2003) [21] applied the GA and a stopping rule by Caliński & Harabasz (1974) [17] to the problem of partitioning a small set of documents (up to 100 documents).

5.2.8 Others

In developing techniques of spectral clustering, automatic determination of L' has been explored. Because spectral clustering tries to find approximately an optimal cut of a graph (its nodes are data points and an edge implies similarity between two nodes) by solving an eigenvalue problem, elements of the eigenvectors can be a clue for selecting L' (see Zelnik-Manor & Perona (2005) [110] or Xiang & Gong, 2008, [101] for actual algorithms). Also, Costa and Netto (1999) [30] tried to incorporate automatic estimation of L' into a SOM (self-organizing map)-based clustering algorithm. Note that some algorithms posteriorly determine the number of clusters as an output from the execution under a predetermined parameter other than L' (e.g., the leader-follower clustering algorithm or the BIRCH algorithm).

There have been some attempts at estimating L' for special-type data such as remote-sensing data (Cao et al., 2007, [20]), time series data (Vasko & Toivonen, 2002 [95]), mathematical function or curves (Li & Chiou, 2011, [62]), and so on. Also, some methods tailored to image data were proposed (e.g., Wang et al., 2009, [98] or Patil & Jondhale, 2010, [76]). Especially, C^3M (cover-coefficient-based concept clustering methodology) (e.g., Can & Ozkarahan, 1984 [18]; 1990 [19]) is a special algorithm for document clustering, which can predict L' from the 'cover coefficient' measuring the degree to which a given document is 'covered' by other documents.

6. Conclusion

This paper tried to develop an algorithm for hierarchical multiway divisive clustering (HMDC) in which the number of parts inherent in each node of a tree is automatically estimated by a criterion based on similarities within and between clusters. The experiment showed that the HMDC algorithm generated good clustering results.

References

[1] Bandyopadhyay, S. and Maulik, U.: Nonparametric genetic clustering: comparison of validity indices, *IEEE Transactions on Systems, Man, and Cybernetics, PART C: Applications and Reviews*, Vol. 31, No. 1, pp.120–125 (2001).

[2] Bandyopadhyay, S. and Maulik, U.: Genetic clustering for automatic evolution of clusters and application to image classification, *Pattern Recognition*, Vol. 35, No. 6, pp.1197–1208 (2002).

[3] Bandyopadhyay, S. and Saha, S.: A point symmetry-based clustering technique for automatic evolution of clusters, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 20, No. 11, pp.1441–1457 (2008).

[4] Banerjee, A.: Robust fuzzy clustering as a multi-objective optimization procedure, *Annual Meeting of the North American Fuzzy Information Processing Society, 2009 (NAFIPS 2009)*, pp. 1 – 6 (2009).

[5] Banerjee, A.: An improved genetic algorithm for robust fuzzy clustering with unknown number of clusters, *Annual Meeting of the North American Fuzzy Information Processing Society (NAFIPS), 2010*, pp. 1– 6 (2010).

[6] Banerjee, A.: Multi-objective genetic algorithm for robust clustering with unknown number of clusters, *International Journal of Applied Evolutionary Computation*, Vol. 3, No. 1, 20 pages (2012).

[7] Banfield, J. D. and Raftery, A. E.: Model-based Gaussian and non-Gaussian clustering, *Biometrics*, Vol. 49, No. 3, pp.803 – 821 (1993).

[8] Bel Mufti, G., Bertrand, P., and El Moubarki, L.: Determining the number of groups from measures of cluster stability, *Proceedings of Applied Stochastic Models and Data Analysis (ASMDA 2005)*, (2005).

[9] Ben-Hur, A. and Guyon, I.: Detecting stable clusters using principal component analysis, *Functional Genomics: Methods in Molecular Biology*, Vol. 224, pp.159–182 (2003).

[10] Ben-Hur, A., Elisseeff, A., and Guyon, I.: A stability based method for discovering structure in clustered data, *Pacific Symposium on Biocomputing*, Vol. 7, pp.6 –17 (2002).

[11] Bertrand, P. and Bel Mufti, G.: Loevinger's measures of rule quality for assessing cluster stability, *Computational Statistics & Data Analysis*, Vol. 50, No. 4, pp. 992 – 1015 (2006).

[12] Biernacki, C., Celeux, G., and Govaert, G.: Assessing a mixture model for clustering with the integrated completed likelihood, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 22, No. 7, pp. 719 – 725 (2000).

[13] Boley, D., Gini, M., Gross, R., Han, E.-H., Hastings, K., Karypis, G., Kumar, V., Mobasher, B., and Moore. J.: Document categorization and query generation on the World Wide Web using WebACE, *Artificial Intelligence Review*, Vol. 13, pp. 365 – 391 (1999).

[14] Boley, D., Gini, M., Gross, R., Han, E.-H., Hastings, K., Karypis, G., Kumar, V., Mobasher, B., and Moore, J.: Partition-based clustering for web document categorization, *Decision Support Systems*, Vol. 27, No. 3, pp. 329 – 341 (1999).

[15] Bouguila, N. and Ziou, D.: High-dimensional unsupervised selection and estimation of a finite generalized Dirichlet mixture model based on Minimum Message Length, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 29, No. 10, pp. 1716 – 1731 (2007).

[16] Bozdogan, H.: Choosing the number of component clusters in the mixture-model using a new informational complexity criterion of the inverse-Fisher information matrix, *Information and Classification: Concepts, Methods and Applications, Proceedings of the 16th Annual Conference of the "Gesellschaft fur Klassifikation e.V."*, pp.40–54 (1992).

[17] Caliński, T. and Harabasz, J.: A dendrite method for cluster analysis, *Communications in Statistics*, Vol.3, No.1, 1–27 (1974).

[18] Can, F. and Ozkarahan, E. A.: Two partitioning type clustering algorithms, *Journal of the American Society for Information Science*, Vol. 35, No. 5, pp. 268 –276 (1984).

[19] Can, F. and Ozkarahan, E. A.: Concepts and effectiveness of the cover-coefficient-based clustering methodology for text databases, *ACM Transactions on Database Systems*, Vol. 15, No. 4, pp.483–517 (1990).

[20] Cao, F., Hong, W., Wu, Y., and Pottier, E.: An unsupervised segmentation with an adaptive number of clusters using the $SPAN/H/\alpha/A$ space and the complex Wishart clustering for fully polarimetric SAR data analysis, *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 45, No. 11, pp. 3454 – 3467 (2007).

[21] Casillas, A., Gonzalez de Lena, M. T., and Martinez, R.: Document clustering into an unknown number of clusters using a genetic algorithm, *Text, Speech and Dialogue: 6th International Conference, TSD 2003*, pp. 43–49 (2003).

[22] Chaea, S. S., DuBienb, J. L., and Wardec, W. D.: A method of predicting the number of clusters using Rand's statistic, *Computational Statistics & Data Analysis*, Vol. 50, No. 12, pp. 3531–3546 (2006).

[23] Cheng, D., Kannan, R., Vempala, S., and Wang, G.: A divide-and-merge methodology for clustering, *ACM Transaction on Database Systems*, Vol. 31, No. 4, pp. 1499–1525 (2006).

[24] Cheung, Y.-M.: k*-Means: a new generalized k-means clustering algorithm, *Pattern Recognition Letters*, Vol. 24, pp.2883 – 2893 (2003).

[25] Cheung, Y.-M.: Maximum weighted likelihood via rival penalized EM for density mixture clustering with automatic model selection, *IEEE Transaction on Knowledge and Data Engineering*, Vol. 17, No. 6, pp. 750–761 (2005).

[26] Chiang, M. M-T. and Mirkin, B.: Intelligent choice of the number of clusters in k-means clustering: an experimental study with different cluster spreads, *Journal of Classification*, Vol. 27, No. 1, pp. 3 – 40 (2010).

[27] Chiu, T., Fang, D. P., Chen, J., Wang, Y., and Jeris, C.: A robust and scalable clustering algorithm for mixed type attributes in large database environment, *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '01*, pp. 263 – 268 (2001).

[28] Cuevas, A., Febrero, M., and Fraiman, R.: Estimating the number of clusters, *Canadian Journal of Statistics*, Vol. 28, pp. 367 – 382 (2000).

[29] Devillez, A., Billaudel, P., and Lecolier, G. V.: A fuzzy hybrid hierarchical clustering method with a new criterion able to find the optimal partition, *Fuzzy Sets and Systems*, Vol. 128, No. 3, pp.323 – 338 (2002).

[30] Costa, J. A. F. and de Andrade Netto, M. L.: Estimating the number of clusters in multivariate data by Self-organizing maps, *International Journal of Neural Systems*, Vol. 9, No. 3, pp.195–202 (1999).

[31] Dhillon, I. S. and Modha, D. S.: Concept decompositions for large sparse text data using clustering, *Machine Learning*, Vol. 42, No. 1-2, pp. 143 – 175 (2001).

[32] Ding, C. and He, X.: Cluster merging and splitting in hierarchical clustering algorithms, *Proceedings of 2002 IEEE International Conference on Data Mining. ICDM 2002*, pp. 139–146 (2002).

[33] Dudoit, S. and Fridlyand, J.: A prediction-based resampling method for estimating the number of clusters in a dataset, *Genome Biology*, Vol. 3, No. 7, pp.research0036.1–research0036.21 (2002).

[34] Fang, Y. and Wang, J.: Selection of the number of clusters via the bootstrap method, *Computational Statistics & Data Analysis*, Vol. 56, No. 3, pp.468–477 (2012).

[35] Feng, L., Qiu, M.-H., Wang, Y.-X., Xiang, Q.-L., Yang, Y.-F., and Liu, K.: A fast divisive clustering algorithm using an improved discrete particle swarm optimizer, *Pattern Recognition Letters*, Vol. 31, No. 11, pp. 1216–1225 (2010).

[36] Feng, Y. and Hamerly, G.: PG-means: learning the number of clusters in data, *Advances in Neural Information Processing Systems 19*, pp. 393–400, MIT Press, 2007.

[37] Fischer, A.: On the number of groups in clustering, *Statistics & Probability Letters*, Vol. 81, No. 12, pp. 1771–1781 (2011).

[38] Fraley, C. and Raftery, E.: How many clusters? Which clustering method? Answers via model-based cluster analysis, *Computer Journal*, Vol. 41, No. 8, pp.578 – 588 (1998).

[39] Frigui, H. and Krishnapuram, R.: A robust algorithm for automatic extraction of an unknown number of clusters from noisy data, *Pattern Recognition Letters*, Vol. 17, No. 12, pp. 1223–1232 (1996).

[40] Frigui, H. and Krishnapuram, R.: A robust competitive clustering algorithm with applications in computer vision, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 21, No. 5, pp. 450 – 465 (1999).

[41] Gath, I. and Geva, A. B.: Unsupervised optimal fuzzy clustering, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 11, No. 7, pp. 773 – 780 (1989).

[42] Girolami, M.: Mercer kernel-based clustering in feature space, *IEEE Transactions on Neural Networks*, Vol. 13, No.3, pp. 780–784 (2002).

[43] Gordon, A. D.: *Classification*, 2nd ed., Boca Raton, FL., Chapman & Hall (1999).

[44] Guenóche, A., Hansen, P., and Jaumard, B.: Efficient algorithms for divisive hierarchical clustering with the diameter criterion, *Journal of Classification*, Vol. 8, No. 1, pp. 5–30 (1991).

[45] Hamerly, G. and Elkan, C.: Learning the k in k-means, *Advances in Neural Information Processing Systems 16 (NIPS 2003)*.

[46] Hardy, A.: On the number of clusters, *Computational Statistics & Data Analysis*, Vol. 23, pp. 83-96 (1996).

[47] Hartigan, J.A.: *Clustering Algorithms*, John Wiley and Sons, New York (1975).

[48] Herbin, M., Bonnet, N., and Vautrot, P.: Estimation of the number of clusters and influence zones, *Pattern Recognition Letters*, Vol. 22, No. 14, pp. 1557 – 1568 (2001).

[49] Hruschka, E. R. and Ebecken, N. F. F.: A genetic algorithm for cluster analysis, *Intelligent Data Analysis*, Vol. 7, No. 1, pp.15-25 (2003).

[50] Hu, X. and Xu, L.: Investigation on several model selection criteria for determining the number of cluster, *Neural Information Processing - Letters and Reviews*, Vol. 4, No. 1, pp. 1 – 10 (2004).

[51] Ishioka, T.: An expansion of x-means for automatically determining the optimal number of clusters: progressive iterations of k-means and merging of the clusters, *Proceedings of the Fourth IASTED International Conference on Computational Intelligence*, pp. 91 – 96 (2005).

[52] Jain, A. K. and Moreau, J. V.: Bootstrap technique in cluster analysis, *Pattern Recognition*, Vol. 20, No. 5, pp. 547–568 (1987).

[53] Kärkkäinen, I. and Fränti, P.: Stepwise algorithm for finding unknown number of clusters, *Proceedings of ACIVS 2002 (Advanced Concepts for Intelligent Vision Systems)*, pp. 136 – 143 (2002).

[54] Kaymak, U. and Setnes, M.: Fuzzy clustering with volume prototypes and adaptive cluster merging, *IEEE Transactions on Fuzzy Systems*, Vol. 10, No. 6, pp. 705 – 712 (2002).

[55] Kishida, K.: Double-pass clustering technique for multilingual document collections, *Journal of Information Science*, Vol.37, No.3, pp.304-321 (2011).

[56] Kishida, K.: Experiment of document clustering by triple-pass leader-follower algorithm without any information on threshold of similarity, *IPSJ SIG Technical Report*, Vol. 2013-IFAT-111, No. 23, pp.1–6 (2013).

[57] Kishida, K.: Empirical comparison of external evaluation measures for document clustering by using synthetic data, *IPSJ SIG Technical Report*, Vol.2014-IFAT-113, No.1, p.1–7 (2014).

[58] Kuncheva, L. I. and Vetrov, D. P.: Evaluation of stability of k-means cluster ensembles with respect to random initialization, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 28, No. 11, (2006).

[59] Lange, T., Roth, V., Braun, M. L., and Buhmann, J. M.: Stability-based validation of clustering solutions, *Neural Computation*, Vol. 16, pp.1299–1323 (2004).

[60] Levine, E. and Domany, E.: Resampling method for unsupervised estimation of cluster validity, *Neural Computation*, Vol. 13, pp. 2573 – 2593 (2001).

[61] Lewis, D. D., Yang, Y., Rose, T. G., and Li, F.: RCV1: a new benchmark collection for text categorization research, *Journal of Machine Learning Research*, Vol. 5, pp. 361–397 (2004).

[62] Li, P.-L. and Chiou, J.-M.: Identifying cluster number for subspace projected functional data clustering, *Computational Statistics & Data*

Analysis, Vol. 55, No. 6, pp.2090–2103 (2011).
[63] Li, Y.-L. and Shen, Y.: An automatic fuzzy c-means algorithm for image segmentation, *Soft Computing*, Vol. 14, No. 2, pp.123 – 128 (2010).
[64] Lo, Y., Mendell, N. R., and Rubin, D. B.: Testing the number of components in a normal mixture. *Biometrika*, Vol. 88, No. 3, pp. 767-778 (2001).
[65] McLachlan, G. J.: On Bootstrapping the likelihood ratio test statistic for the number of components in a normal mixture, *Journal of the Royal Statistical Society. Series C*, Vol. 36, No. 3, pp.318–324 (1987).
[66] McLachlan, G. J. and Khana, N.: On a resampling approach for tests on the number of clusters with mixture model-based clustering of tissue samples, *Journal of Multivariate Analysis*, Vol. 90, pp.90–105 (2004).
[67] McLachlan, G. and Peel, D.: *Finite Mixture Models*, John Wiley & Sons, New York (2000).
[68] Milligan, G. W. and Cooper, M. C.: An examination of procedures for determining the number of clusters in a data set, *Psychometrika*, Vol. 50, No. 2, pp. 159–179 (1985).
[69] Mirkin, B.: *Clustering for Data Mining: A Data Recovery Approach*, CRC Press, Boca Raton, Florida (2005).
[70] Mirkin, B.: Choosing the number of clusters, *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, Vol. 1, No. 3, pp. 252 – 260 (2011).
[71] Mirkin, B.: *Clustering: A Data Recovery Approach*, CRC Press, Boca Raton, Florida, 2nd edition (2013).
[72] Mohajer, M., Englmeier, K.-H., and Schmid, V. J.: A comparison of Gap statistic definitions with and without logarithm function, *Technical Report Number 096*, Department of Statistics, University of Munich (2010).
[73] Nilsson, M.: Hierarchical clustering using non-greedy principal direction divisive partitioning, *Information Retrieval*, Vol. 5, No. 4, pp. 311 – 321 (2002).
[74] Pan, W. and Shen, X.: Penalized model-based clustering with application to variable selection, *Journal of Machine Learning Research*, Vol. 8, pp.1145–1164 (2007).
[75] Pascual, D., Pla, F., and Sánchez, J. S.: Cluster stability assessment based on theoretic information measures, *Progress in Pattern Recognition, Image Analysis and Applications: 13th Iberoamerican Congress on Pattern Recognition, CIARP 2008*, (Ruiz-Shulclopern, J. and Kropatsch, W. G. eds.), Springer, Berlin, pp. 219–226 (2008).
[76] Patil, R. V. and Jondhale, K. C.: Edge based technique to estimate number of clusters in k-means color image segmentation, *3rd IEEE International Conference on Computer Science and Information Technology (ICCSIT)*, Vol. 2, pp. 117 – 121 (2010).
[77] Pelleg, D. and Moore, W.: X-means: extending k-means with efficient estimation of the number of clusters, *Proceedings of the Seventeenth International Conference on Machine Learning (ICML '00)*, pp. 727–734 (2000).
[78] Pham, D. T., Dimov, S. S., and Nguyen, C. D.: Selection of k in k-means clustering, *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, Vol. 219, pp. 103 – 119 (2005).
[79] Pollard, K. S. and van der Laan, M. J.: A method to identify significant clusters in gene expression data, *Proceedings, SCI (World Multiconference on Systemics, Cybernetics and Informatics), V. II*, pp. 318–325 (2002).
[80] Roberts, S. J., Husmeier, D., Rezek, I., and Penny, W.: Bayesian approaches to Gaussian mixture modeling, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 20, No. 11, pp. 1133 – 1142 (1998).
[81] Roth, V., Lange, T., Braun, M., and Buhmann, J.: A resampling approach to cluster validation, *Compstat: Proceedings in Computational Statistics*, pp. 123 – 128 (2002).
[82] Rousseeuw, P. J.: Silhouettes: a graphical aid to the interpretation and validation of cluster analysis, *Journal of Computational and Applied Mathematics*, Vol. 20, pp.53–65 (1987).
[83] Saha, S. and Bandyopadhyay, S.: A symmetry based multiobjective clustering technique for automatic evolution of clusters, *Pattern Recognition*, Vol. 43, pp. 738–751 (2010).
[84] Salvador, S. and Chan, P.: Determining the number of clusters / segments in hierarchical clustering / segmentation algorithms, *16th IEEE International Conference on Tools with Artificial Intelligence (ICTAI), 2004*, pp. 576 – 584 (2004).
[85] Savaresi, S.M., Boley, D.L., Bittanti, S., and Gazzaniga, G.: Cluster selection in divisive clustering algorithms, *SIAM International Conference on Data Mining*, pp. 299–314 (2002).
[86] Sheng, W., Swift, W. S., Zhang, L., and Liu, X.: A weighted sum validity function for clustering with a hybrid niching genetic algorithm, *IEEE Transactions on Systems, Man, and Cybernetics - PART B: Cybernetics*, Vol. 35, No. 6, pp.1156–1167 (2005).
[87] Steinbach, M., Karypis, G., and Kumar, V.: A comparison of document clustering techniques, *KDD Workshop on Text Mining*, (2000).
[88] Steinley, D.: Stability analysis in k-means clustering, *British Journal of Mathematical and Statistical Psychology*, Vol. 61, pp. 255–273 (2008).
[89] Sugar, C. A. and James, G. M.: Finding the number of clusters in a data set: an information theoretic approach, *Journal of the American Statistical Association*, Vol. 98, pp. 750–763 (2002).
[90] Sun, H., Wang, S., and Jiang, Q.: FCM-based model selection algorithms for determining the number of clusters, *Pattern Recognition*, Vol. 37, No. 10, pp. 2027 – 2037 (2004).
[91] Tasoulis, S. K., Tasoulis, D. K., and Plagianakos, V. P.: Enhancing principal direction divisive clustering, *Pattern Recognition*, Vol. 43, No. 10, pp. 3391–3411 (2010).
[92] Teh, Y. W., Jordan, I., Beal, M. J., and Blei, D. M.: Hierarchical Dirichlet processes, *Journal of the American Statistical Association*, Vol. 101, No. 476, pp. 1566–1581 (2006).
[93] Tibshirani, R. and Walther, G.: Cluster validation by prediction strength, *Journal of Computational and Graphical Statistics*, Vol. 14, No. 3, pp. 511–528 (2005).
[94] Tibshirani, R., Walther, G., and Hastie, T.: Estimating the number of clusters in a dataset via the gap statistic, *Journal of the Royal Statistical Society: Series B*, Vol. 63, No.2, pp. 411 –423 (2001).
[95] Vasko, K. T. and Toivonen, H. T. T.: Estimating the number of segments in time series data using permutation tests, *Proceedings of IEEE International Conference on Data Mining (ICDM), 2002*, pp. 466 – 473 (2002).
[96] Volkovich, Z., Barzily, Z., and Morozensky, L.: A statistical model of cluster stability, *Pattern Recognition*, Vol. 41, No. 7, pp. 2174 – 2188 (2008).
[97] Volkovich, Z., Barzily, Z., Weber, G.-W., Toledano-Kitai, D, and Avros, R.: Resampling approach for cluster model selection, *Machine Learning*, Vol. 85, pp. 209–248 (2011).
[98] Wang, L., Leckie, C., Ramamohanarao, K., and Bezdek J.: Automatically determining the number of clusters in unlabeled data sets, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 21, No. 3, pp. 335 – 350 (2009).
[99] Wang, J.: Consistent selection of the number of clusters via crossvalidation, *Biometrika*, Vol. 97, No. 4, pp.893–904 (2010).
[100] Welling, M. and Kurihara, K.: Bayesian k-means as a "maximization-expectation" algorithm, *Neural Computation*, Vol. 21, No. 4, pp. 1145–1172 (2009).
[101] Xiang, T. and Gong, S.: Spectral clustering with eigenvector selection, *Pattern Recognition*, Vol. 41, No. 3, pp. 1012-1029 (2008).
[102] Xu, L.: Bayesian Ying-Yang machine, clustering and number of clusters, *Pattern Recognition Letters*, Vol. 18, No. 11-13, pp. 1167–1178 (1997).
[103] Xu, L.: Rival penalized competitive learning, finite mixture, and multisets clustering, *The 1998 IEEE International Joint Conference on Neural Networks Proceedings, 1998*. Vol. 3, pp. 2525 – 2530 (1998).
[104] Xu, L., Krzyżak, A., and Oja, E.: Rival penalized competitive learning for clustering analysis, RBF Net, and curve detection, *IEEE Transactions on Neural Networks*, Vol. 4, No. 4, pp. 636 – 649 (1993).
[105] Xu, R. and Wunsch II, D. C.: *Clustering* John Wiley and Sons, Hoboken, New Jersey (2009).
[106] Yan, M. and Ye, K.: Determining the number of clusters using the weighted Gap statistic, *Biometrics*, Vol.63, pp. 1031 – 1037 (2007).
[107] Yeung, K. Y., Fraley, C., Murua, A., Raftery, A. E., and Ruzzo, W. L.: Model-based clustering and data transformations for gene expression data, *Bioinformatics*, Vol. 17, No. 10, pp. 977 – 987 (2001).
[108] Zeimpekis, D. and Gallopoulos, E.: PDDP(l): towards a flexible principal direction divisive partitioning clustering algorithm, *Proceedings of IEEE ICDM' 03 Workshop on Clustering Large Data Sets*, pp. 26 – 35 (2003).
[109] Zeimpekis, D. and Gallopoulos, E.: Principal direction divisive partitioning with kernels and k-means steering, *Survey of Text Mining II: Clustering, Classification, and Retrieval*, (Berry, M. W. and Castellanos, M., eds), pp. 45 – 64, New York, Springer (2008).
[110] Zelnik-Manor, L. and Perona, P.: Self-tuning spectral clustering, *Advances in Neural Information Processing Systems 17*, pp. 1601–1608 (2005).
[111] Zhao, Y. and Karypis, G.: Evaluation of hierarchical clustering algorithms for document datasets, *Proceedings of the Eleventh International Conference on Information and Knowledge Management (CIKM '02)*, pp. 515–524 (2002).

