Chapter 6
Randomized Algorithm of Finding the True Number of Clusters Based on Chebychev Polynomial Approximation

R. Avros[1], O. Granichin[2], D. Shalymov[2],
Z. Volkovich[1], and G.-W. Weber[3]

[1] Ort Braude College of Engineering, Karmiel 21982, Israel
r_avros@braude.ac.il, vlvolkov@braude.ac.il
[2] Saint Petersburg State University, Russia
Oleg_granichin@mail.ru, shalydim@mail.ru
[3] Institute of Applied Mathematics, Middle East Technical University,
06531 Ankara, Turkey University of Siegen (Germany),
University of Aveiro (Portugal), Universiti Teknologi Malaysia, Skudai
gweber@metu.edu.tr

Abstract. One of the important problems arising in cluster analysis is the estimation of the appropriate number of clusters. In the case when the expected number of clusters is sufficiently large, the majority of the existing methods involve high complexity computations. This difficulty can be avoided by using a suitable confidence interval to estimate the number of clusters. Such a method is proposed in the current chapter.

The main idea is to allocate the jump position of the within-cluster dispersion function using Chebyshev polynomial approximations. The confidence interval for the true number of clusters can be obtained in this way by means of a comparatively small number of the distortion calculations. a significant computational complexity decreasing is proven. Several examples are given to demonstrate the high ability of the proposed methodology.

Keywords: Cluster analysis, Clustering, Cluster stability, Randomized algorithms.

1 Introduction

Cluster analysis methods can be roughly divided into two categories: clustering and validation approaches. In the latter methods, which are intended to estimate the optimal ("true") number of clusters, the obtained partitions are evaluated according to a given rule, and the number of clusters is selected on the basis of the optimal rule value. This crucial problem, known as an "ill posed" problem [23,30], may have several solutions. For example, the answer may depend on the data measurement units. The selection of the particular clustering algorithm used here is another major difficulty since the partitions constructed are

D.E. Holmes, L.C. Jain (Eds.): Data Mining: Found. & Intell. Paradigms, ISRL 23, pp. 131–155.
springerlink.com

intended more or less to reflect the inner hidden data structure. Solutions given by different algorithms can be essentially different and lead to quite different conclusions about the stable cluster configurations. To illustrate this phenomenon, we consider partitions into two clusters created by two algorithms for a dataset simulated on the real line as a mix of three Gaussian components. The partition obtained using the randomly initialized standard k-means algorithm is presented in Figure 1, while Figure 2 demonstrates the result obtained using the $ClassificationEM$ algorithm, a version the EM approach introduced in [8]. It can be seen that the k-means algorithm reveals "false" stable two-cluster construction, even as a more flexible CEM method leads to an unreliable two-cluster structure. Figure 2 demonstrates the result obtained by the CEM algorithm.

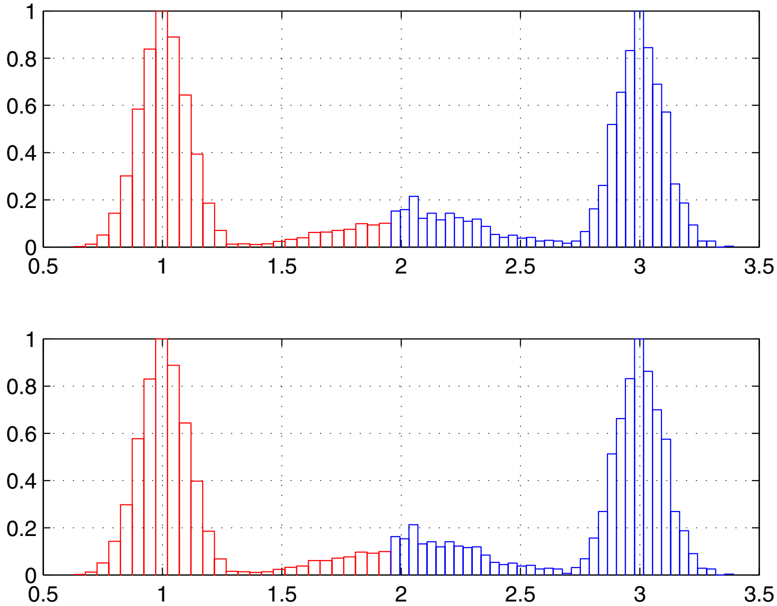

Fig. 1. Outcomes of repeated clusterings by means of the standard k-means algorithm

The current manuscript discusses a new approach to the determination of the true number of clusters. Although a great number of methods have been proposed to tackle this problem, none of them has yet been accepted as a superior one. We review here several approaches because they can be incorporated into the proposed in this paper methodology.

Geometrical approaches were employed by Dunn [17], Hubert and Schultz [29] (C-index), Calinski-Harabasz [7], Hartigan [26], Krzanowski-Lai [35], Sugar-James [45], Gordon [22], Milligan and Cooper [40], and Tibshirani, Walter and Hastie [47] (the Gap Statistic method). Stability models compare the pairs of

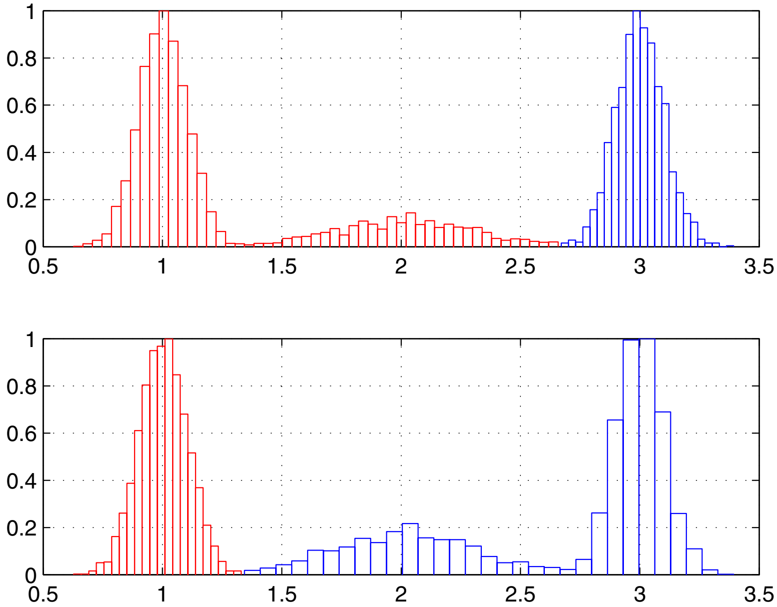

Fig. 2. Outcomes of repeated clusterings by means of the *CEM*

clustered samples obtained by application of a clustering algorithm in which the partition consistency is interpreted as the partition reliability [9], and thus the "true" number of clusters corresponds to the maximal stability score. In the framework of this methodology, Levine and Domany [38], Ben-Hur, Elisseeff and Guyon [3], and Ben-Hur and Guyon [4] represented the stability criteria through the fractions of times that pairs of elements provide the same membership within the clustering algorithm reiterations. Bel Mufti, Bertrand, and El Moubarki [41] used the Loevinger's isolation measure to determine the stability function.

Another group of methods utilizes external correlation indexes as a stability degree. For example, such a method was implemented in the known Clest approach of Dudoit and Fridlyand [16]. A general prediction resampling procedure was proposed by Roth, Lange, Braun and Buhmann [36,37]. Tibshirani and Walther [46] described a process of comparable forecast strength process. Jain and Moreau [31] considered the dispersions of empirical distributions as a stability measure. Nonparametric density estimation methodology relates the clusters to the probability density function peaks. The clustering assigns each item to a "domain of attraction" of the density modes. Evidently, Wishart [51] appears to be the first to propose looking for modes in order to reveal the cluster structure. Apparently, this idea was formulated by Hartigan ([27], Section 11, and [28]) to introduce the notion of high density clusters. The number of clusters is given here as the number of regions where the density is higher than a certain specified level. Thus, clusters are viewed as isolated islands of "high" density in the ocean of "low" densities (see, e.g., [10,11,44]). The goodness-offit-test procedures should be also mentioned. Pelleg and Moore [42] developed an X-means algorithm, where the best score of the so-called Bayesian Information Criterion [32] is found in order to determine the true number of clusters. Hamerly and Elkan [25] applied another scoring criteria to the G-means algorithm, namely, the null hypothesis about the clusters drawn from a Gaussian population was tested by means of a statistical projection procedure. Furthermore, Feng and Hamerly [18] reported the PG-means (PG stands for projected Gaussian) algorithm, which also operates with projections onto the clustering model. The PG-means employs the Gaussian mixture model incorporated into the Expectation-Maximization algorithm. Other applications of the goodness-of-fit test were suggested by Volkovich, Barzily and Morozensky [50], Barzily, Volkovich, Akteke-Ozturk and Weber [2] and Volkovich and Barzily [48]. Here the models of clusters were created based on the model of well-mixed samples within the clusters. Volkovich, Barzily, Avros and Toledano-Kitai [49] used the binomial model of the K-Nearest Neighbors belonging to the own point's sample. Another model suggested by Volkovich, Barzily and Morozensky [50] and Barzily, Volkovich, Akteke-Ozturk and Weber [2] considers the probability distances between clustered samples taken from the cluster cores and the whole population, correspondingly.

The cluster validation procedures mentioned above usually check the cluster quality for all possible numbers of clusters in a given area. The strategy could be in the geometrical approach to generate the distortion curve for the input data by running a standard clustering algorithm, such as the k-means, for all values of k between 1 and $k_{\max}$ and computing the resulting clustering distortions. In the case when the suggested number of clusters is sufficiently large, such a methodology leads to high complexity of the computations.

This difficulty can be avoided by a preliminary estimation of the appropriate number of clusters by means of a suitable confidence interval. Such a method is proposed in the present article. Generally speaking, the presented methodology is based on the employment of the "elbow criterion". This rule recommends selecting of a number of clusters in order that further clusters splitting does not provide more relevant information. It is expressed as a sharp jump point on the graph of the explained by the clusters variance fraction as a function of the number of clusters. The main idea is to compute a small amount of differential distortion function values and to allocate the jump position relaying on its approximations by fixed set of Chebyshev polynomials with uniformly bounded coefficients. A confidence interval for the true number of clusters can be obtained by comparatively small amount of the distortion calculations.

The rest of the paper is organized in the following way. Subsection 2.1 is devoted to description of the known clustering algorithm such k-means and PAM. Several stability based cluster validation methods are presented in Subsection 2.2. Geometrical approaches are discussed in Subsection 2.3. A randomized algorithm for estimation of the true number of clusters is explained in Section 3. Examples of the algorithm application are presented in Section 4.

Notations

- $\mathbf{X}$ is a finite subset of the Euclidean space $\mathbb{R}^d$ to be clustered. The elements of $\boldsymbol{X}$ are represented in the form $\mathbf{x} = (x_1, ..., x_d)$;
- $N_X = |\boldsymbol{X}|$ is the size of the set $\boldsymbol{X}$;
- $\mathbb{R}_+ = [0, +\infty)$;
- $< \cdot, \cdot >$ denotes the inner product of two elements of $\mathbb{R}^d$;
- $tr(A)$ denotes the trace of a matrix A;
- k denotes the number of clusters being considered; $Cl(\boldsymbol{X}, k)$ is a clustering algorithm dividing the set $\boldsymbol{X}$ into k non-overlapping clusters;
- $k^\star$ denotes the true number of clusters.

2 Clustering

2.1 Clustering Methods

Partitioning and hierarchical methods are frequently used in the clustering process. According to hierarchical approaches, a collection of nested partitions is built based on point clusters which include just one data element. On the other hand, the whole set of points, which is, actually, the universal cluster, is found at the end of the process. The traditional representation of the cluster hierarchy is a two-dimensional diagram tree. The true number of clusters is not usually specified; instead, an appropriate solution is obtained by cutting the dendrogram tree at a certain level. Hierarchical procedures are divided into agglomerative ("bottom-up") or divisive ("top-down"). Divisive ("top-down") algorithms start from the whole set and successively separate the items into improved partitions. Agglomerative ("bottom-up") methods produce series of fusions of the data elements into groups.

Partitioning approaches are based on an optimization procedure applied to an objective function which governs the partition quality and can produce a tighter cluster structure than in the case of the hierarchical methods. Additionally, such procedures have lower complexity since a large number of variables are clustered into a small number of groups. However, these approaches often provide non-globular clusters and, consequently, different methods may generate different results. Moreover, a partition may be constructed for almost any number of clusters without its verification.

Let us consider a general clustering model. A partition of the set $\boldsymbol{X}$ is defined as a collection of non-empty disjoint subsets

$$\boldsymbol{\Pi}_k(\boldsymbol{X}) = \{\pi_i(\boldsymbol{X}),\ i = 1, ..., k\},$$

such that

$$\boldsymbol{X} = \bigcup_{i=1}^{k} \pi_i(\boldsymbol{X}).$$

The elements $\pi_i(\boldsymbol{X})$ of the partition are called clusters. The partition quality, intended to be minimized in the partitioning approaches, is defined for a given real-valued function q of X subsets as

$$Q\left(\boldsymbol{\Pi}_k(\boldsymbol{X})\right) = \sum_{i=1}^{k} q(\pi_i(\boldsymbol{X})). \tag{1}$$

Hence, clustering can be interpreted as a particular case of a global optimization problem where the partition $\Pi^{(0)}$, which optimizes the objective function Q, is to be found. Function q can be constructed using a distance-like function $d(x, y)$ and a predefined set of k cluster centroids (medoids) $\boldsymbol{C}$=($\mathbf{c}_1, ..., \mathbf{c}_k$). The partition of $\mathbf{X}$ is built as follows:

$$\pi_i(\boldsymbol{X}) = \{\mathbf{x} \in \boldsymbol{X} : d(\mathbf{c}_i, \mathbf{x}) \leq d(\mathbf{c}_j, \mathbf{x}),\ for\ j \neq i\},\ i = 1, ..., k,$$

(Ties are broken arbitrarily). Alternatively, the centroid is given by a partition in the form:

$$c(\pi_j) = \arg\min_{c}\{\sum_{x \in \pi_j} d(\mathbf{c}, \mathbf{x})\}.$$

Thus

$$q(\pi_j(\boldsymbol{X})) = \sum_{x \in \pi_j} d(c(\pi_j), \mathbf{x})$$

and the mentioned optimization problem is reduced to finding a set of centroids as the solution of the problem being considered:

$$\boldsymbol{C} = \arg\min_{\mathbf{c} \in \boldsymbol{C}}\{\sum_{j=1}^{k} \sum_{x \in \pi_j(\boldsymbol{X})} d(\mathbf{c}_j, \mathbf{x})\}. \tag{2}$$

In the case when $d(\cdot, \cdot)$ is the squared standard Euclidean distance, the objective function is represented as

$$\min_{\boldsymbol{C}} R(\boldsymbol{C}) = \sum_{j=1}^{k} \sum_{\mathbf{x} \in \boldsymbol{X}} \min_{c_j} \|\mathbf{x} - c_j\|^2. \tag{3}$$

The well-known k-means algorithm was proposed in [19] to provide an approximate solution to this optimization task.

Input: $\boldsymbol{X}$ is the set to be clustered; k is the number of clusters.
***k*-means Algorithm:**

1. Randomly place k items as initial cluster centers (centroids) into the space represented by $\boldsymbol{X}$;
2. Assign each point $\mathbf{x} \in \boldsymbol{X}$ to the nearest cluster center;
3. Recompute the new centroids as the group mean values once all elements have been assigned;

4. Repeat Steps 2 and 3 until the convergence criterion is met (e.g. the assignment is no longer changed or the centroids do not move any longer).

The algorithm has the computational complexity $O(kN_X)$ and frequently constructs the so-called "non-optimal stable partitions". To overcome this problem the incremental k-means algorithm can be used (see, e.g. [15,12,13,14,34]). The k-means approach can be viewed as a version of the famous Expectation Maximization (EM) approach which suggests the Gaussian Mixture Model (GMM) of data in the clustering context. (see, e.g. [1,8,21]):

$$f(\mathbf{x}) = \sum_{j=1}^{k} p_j G(\mathbf{x}|\mu_j, \mathit{\Gamma}_j), \tag{4}$$

where $f(\mathbf{x})$ is the underlying data density; $G(\mathbf{x}|\mu, \mathit{\Gamma})$ is the Gaussian density (with the mean value μ and the covariance matrix $\mathit{\Gamma}$).

The EM-method maximizes the log likelihood function:

$$L = \sum_{\mathbf{x}\in \boldsymbol{X}} \log(\sum_{j=1}^{k} p_j G(\mathbf{x}|\mu_j, \mathit{\Gamma}_j)).$$

Celeux and Govaert [8] demonstrated that the k-means approach appears in the case when the cluster proportions are equal to each other:

$$p_1 = p_2 = ... = p_k \tag{5}$$

and the covariance matrix has the form

$$\mathit{\Gamma}_j = \sigma^2 I, \ j = 1, ..., k,$$

where I is the identity matrix and σ^2 is an unknown parameter.

The Partitioning Around Medoids (PAM) clustering procedure ([33], Chapter 2) is the most common realization of k-medoid approach. In contrary to the k-means algorithm, the k-medoid method seeks for the data elements, named medoids, as the cluster centers. The corresponding objective function, similar to (2), has the form:

$$\min_{\boldsymbol{C}\in \boldsymbol{X}} R(\boldsymbol{C}) = \sum_{j=1}^{k} \sum_{\mathbf{x}\in \boldsymbol{X}} \min_{c_j \in \boldsymbol{X}} d(\mathbf{c}_j, \mathbf{x}). \tag{6}$$

The PAM algorithm which gives an approximate solution of this problem consists of two phases:

– **BUILD** - constructing initial clustering;
– **SWAP** - refining the clustering.

Input: $\boldsymbol{Dis}$ is a prepared beforehand $N_X * N_X$ dissimilarity matrix between the items to be clustered; k is the number of clusters.

PAM Algorithm:

```
1. BUILD Phase: Build the set $\boldsymbol{C}$ of medoids which minimizes the objective function (6);
2. SWAP Phase: Until no change, do:
3. Assign each point $\mathbf{x} \in \boldsymbol{X}$ to the nearest cluster center (medoid);
4. For each $\mathbf{c} \in \boldsymbol{C}$ and for each $\mathbf{x} \in \boldsymbol{X} \backslash \boldsymbol{C}$:
    (a) Compute the total cost $S$ of swapping medoid $\mathbf{c}$ with $\mathbf{x}$;
    (b) If $S < 0$, swap $\mathbf{c}$ with $\mathbf{x}$ to create a new set of medoids;
5. end loop until.
```

The PAM algorithm is more robust than the k-means algorithm, especially in the case of noisy data with outliers; however, it has higher computational complexity of $O(k(N_X - k)^2)$ for each iteration. An important property of the PAM algorithm is its ability to construct clusterings based on any distances. Additionally, medoids provide more robust cluster centers as compared to k-means centroids.

2.2 Stability Based Methods

We have already mentioned above that stability-based determination of the true number of clusters is a very common cluster analysis tool. Several approaches of this kind are discussed below.

2.2.1 External Indexes

External indexes are often used in cluster stability approaches. These nominal measures of associations are based on the so-called cross-tabulation or contingency tables. Let us suppose that Π_r and Π_c are two partitions of the same dataset of size n into r and c clusters, respectively. Denote by n_{ij} the number of elements belonging to cluster i of Π_r and to cluster j of Π_c ($i = 1, ..., r$, $j = 1, ..., c$). The Cramer's V statistic measure of the strength of association between two (nominal) categorical variables is defined in the following way:

$$V = \sqrt{\frac{\chi^2}{N_X * \min(r-1, c-1)}},$$

where

$$\chi^2 = \sum_{i=1}^{r} \sum_{j=1}^{c} \frac{(n_{ij} - e_{ij})^2}{e_{ij}}, \ e_{ij} = \frac{n_i^{(r)} * n_j^{(c)}}{N_X}$$

is the chi-square statistic and

$$n_i^{(r)} = \sum_{j=1}^{c} n_{ij}, \ i = 1, ..., r,$$

$$n_j^{(c)} = \sum_{i=1}^{r} n_{ij}, \ j = 1, ..., c.$$

Denote

$$Z = \sum_{j=1}^{c} \sum_{i=1}^{r} n_{ij}^2 .$$

The index of partition association introduced by Rand [43] is

$$R = 1 + \left(\frac{Z - 0.5 * \left(\sum_{j=1}^{c} \left(n_j^{(c)} \right)^2 + \sum_{i=1}^{r} \left(n_i^{(r)} \right)^2 \right)}{\binom{n}{2}} \right) .$$

Another index was proposed by Jain and Dubes [30]:

$$JD = \frac{(Z - N_X)}{\left(\sum_{j=1}^{c} \left(n_j^{(c)} \right)^2 + \sum_{i=1}^{r} \left(n_i^{(r)} \right)^2 - Z - N_X \right)},$$

while Fowlkes and Mallows [20] suggested the coefficient

$$FM = \frac{(Z - N_X)}{2\sqrt{\sum_{j=1}^{c} \binom{n_j^{(c)}}{2} \sum_{i=1}^{r} \binom{n_i^{(r)}}{2}}} .$$

Apparently, the indexes R and FM are linear functions of each other because they are linear functions of Z. The adjusted R index is the corrected-for-chance version of the Rand index, standardized in such a way that its expected value is 0 if the partitions are random and 1 if they correspond to each other perfectly. The standardization is performed as follows:

$$Ind^{'} = \frac{(Ind - E(ind))}{(Ind_{\max} - E(ind))},$$

where $Ind_{\max}$ is the maximal value of the index Ind. The common null hypothesis suggests that the contingency table is built on the assumption of the generalized hyper-geometric distribution and that partitions Π_r and Π_c are mutually independent. In this case, the adjusted R index equals zero. These indexes were used in the Clest method [16] as a measure of clustering stability.

2.2.2 Clest Algorithm

The Clest algorithm splits the clustered data $\boldsymbol{X}$ into two non-overlapping halves $\boldsymbol{L}_b$ and $\boldsymbol{T}_b$, called a learning and a test sets, respectively. The main idea of the method, proposed, apparently, by Breckenridge [5], is constructing two partitions on $\boldsymbol{T}_b$ in such a way that the first partition is obtained by applying the clustering procedure directly, while the second one is obtained as an extension of the $\boldsymbol{L}_b$ partition to $\boldsymbol{T}_b$. The two partitions are compared using one of the described-above external indices. The true number of clusters corresponds to the largest significant evidence against the null hypothesis about the absence of the cluster structure. The algorithm can be presented in the following form:

For each tested number of clusters k, $2 \leq k \leq k_{\max}$, do 1-4.

1. Repeat B times:
   (a) Split the original dataset into two non-overlapping sets $\boldsymbol{L}_b$ and $\boldsymbol{T}_b$;
   (b) Construct $\Pi\left(\boldsymbol{L}_b\right) = Cl(\boldsymbol{L}_b, k)$;
   (c) Construct a classifier $C(\boldsymbol{L}_b)$ based on $\Pi\left(\boldsymbol{L}_b\right)$;
   (d) Apply the classifier $C(\boldsymbol{L}_b)$ to the test set $\boldsymbol{T}_b$ and get $\Pi_1\left(\boldsymbol{T}_b\right)$;
   (e) Construct $\Pi_2\left(\boldsymbol{T}_b\right) = Cl(\boldsymbol{T}_b, k)$;
   (f) Calculate the external index $I_{k,b}$ comparing $\Pi_1\left(\boldsymbol{T}_b\right)$ and $\Pi_2\left(\boldsymbol{T}_b\right)$;
2. Consider the observed median value of the external index

$$t_k = median(I_{k,1}, ..., I_{k,B}).$$

3. Produce B_0 datasets under an appropriate null hypothesis of the absence of the cluster structure and repeat the above steps 1 and 2 until both of them getting B_0 statistics $t_{k,1}, ..., t_{k,Bo}$.
4. Consider the average of the above B_0 statistics:

$$t_k^{(0)} = \frac{1}{B_0} \sum_{b=1}^{B_0} t_k,$$

and denote by p_k the proportion of those $t_{k,b}, 1 \leq b \leq B_0$, that are at least as large as the observed statistic t_k, i.e., the p-value for t_k. Let

$$d_k = t_k - t_k^{(0)}$$

denote the difference between the observed similarity statistic and its estimated expected value under the null hypothesis.

5. Introduce the set A as

$$A = \{2 \leq k \leq k_{\max} : p_k \leq pmax, d_k \geq dmin\},$$

where $pmax$ and $dmin$ are predefined parameters. This set is empty if no cluster structure has been found. Otherwise, the number of clusters k corresponds to the largest significant difference statistic d_k:

$$k = \arg\max_{k \in K} d_k.$$

The authors used the PAM algorithm described in Section 2.1, the naive Bayes classificator, the FM index (see, Section 2.2.1), $B = B_0 = 20$, and $pmax = dmin = 0.05$.

2.3 Geometrical Cluster Validation Criteria

The majority of geometrical cluster validation criteria are based on the total dispersion, or total scatter, matrices T_k as well as on between and within k-cluster sums of squares B_k and W_k defined for a given partition $\mathit{\Pi}_k(\mathbf{X}), k \geq 2$, as (see, e.g. [39]):

$$T_k = \sum_{j=1}^{k} \sum_{z \in \pi_j} (\mathbf{x} - \overline{\boldsymbol{\mu}})(\mathbf{x} - \overline{\boldsymbol{\mu}})^t, \tag{7}$$

$$B_k = \sum_{j=1}^{k} |\pi_j| \, (\overline{\boldsymbol{\mu}}_j - \overline{\boldsymbol{\mu}})(\overline{\boldsymbol{\mu}}_j - \overline{\mu})^t, \; W_k = \sum_{j=1}^{k} \sum_{\mathbf{x} \in \pi_j(\boldsymbol{X})} (\mathbf{x} - \overline{\boldsymbol{\mu}}_j)(\mathbf{z} - \overline{\boldsymbol{\mu}}_j)^t, \tag{8}$$

where $\overline{\mu}$ is the mean point of the set $\mathbf{X}$, and $\overline{\mu}_j$ are the arithmetic means of $\pi_j(\mathbf{X})$, $j = 1, ..., k$. It should be noted that $T_k = W_k + B_k$. The first method proposed for the evaluation of the true number of clusters appears to be the, so called, "elbow criterion", which employs the graph of the within-cluster dispersion W_k as a function of the number of clusters k. As a rule, this characteristic decreases with the increase of the number of clusters. An attempt to divide a cluster into subgroups further decreases the criterion value when well-separated clusters are considered. In this case, the W_k graph has a sharp decline. The number of clusters is determined at this point and this is where the method got its name from. However, the appropriate "elbow" cannot always be explicitly recognized. Attempts to develop approaches for detecting the "elbow" were made in many studies [7,27,40,47], etc.

The "elbow" phenomenon is illustrated in Figure 4, where the graphs of the functions $log(W_k)$ (observed), marked in blue, and $log(W_k^*)$ (reference), marked in red, employed in calculating the Gap statistic [47] are presented for a four-component dataset described in Figure 3. The reference function values are found on the basis of an appropriate null data distribution which appears to be the least favorable from the standpoint of clustering. The uniform distribution is usually used for this purpose.

The inner indexes based on the "elbow" methodology are often employed in the procedures of partitioning, the stopping rules being applied to determine the number of clusters. The stopping-rule (index) value is found, in this case, for a set of cluster solutions and the extreme value, which depends on the particular stopping rule, indicates the most appropriate solutions.

1. The Calinski-Harabasz index (pseudo-F index) [7] is defined as

$$CH_k = \frac{tr(B_k)/(k-1)}{tr\left(W_k\right)/(N_X - k)}.$$

   The estimated true number of clusters is determined as the value of k that corresponds to the maximum of CH_k. The Calinski-Harabasz index was the best of the 30 indices tried on synthetic data by Milligan and Cooper [40].

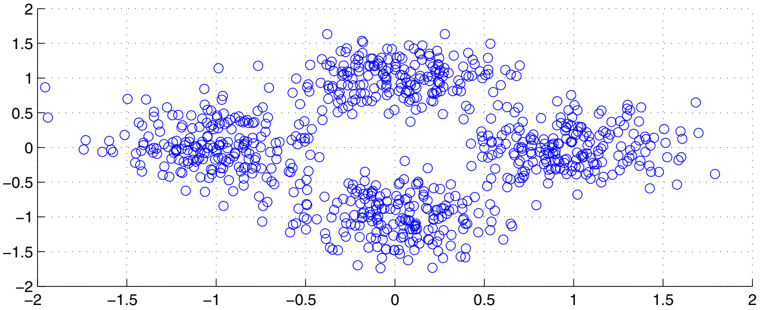

Fig. 3. Scatter plot of a four-component dataset

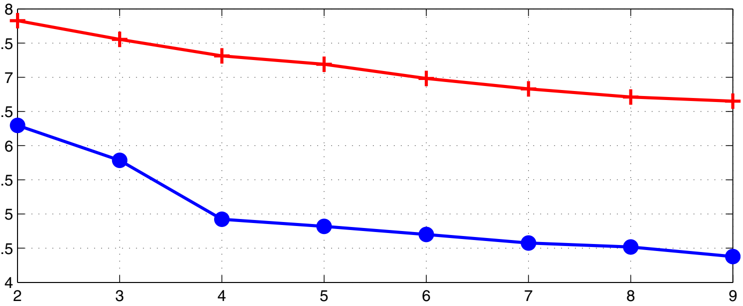

Fig. 4. The graphs of the logarithms of the Gap statistic values

2. The Krzanowski-Lai index [35] is defined by the following relationships:

$$diff_k = (k-1)^{2/d} tr(W_{k-1}) - k^{2/d} tr(W_k),$$

and

$$KL_k = |diff_k| / |diff_{k+1}|.$$

The estimated number of clusters corresponds to the maximal value of the index KL_k.

3. The Hartigan index [26] is defined as

$$h_k = \left(\frac{tr(W_k)}{tr(W_{k+1})} - 1 \right) (N_X - k - 1).$$

The estimated number of clusters is the smallest value of $k \geq 1$ for $h_k \leq 10$.

4. The above-mentioned Gap method [47] also deals with the values of $tr(W_k), k \geq 1$, such that B reference datasets are created under the null hypothesis. Then the reference datasets are clustered and the values of

$tr\left(W_k^1\right), ..., tr\left(W_k^B\right)$ are computed. The estimated value of the Gap statistic is found as

$$gap_k = \frac{1}{B}\sum_b \log(tr(W_k^b)) - \log(tr(W_k)).$$

Let sd_k be the standard deviation of $\log\left(tr\left(W_k^b\right)\right), 1 \le b \le B$, and

$$sd_k = sd_k\sqrt{1+\frac{1}{B}}.$$

The estimated true number of clusters corresponds to the smallest value of $k \ge 1$ that satisfies the inequality

$$gap_k \ge gap_{k^*} - sd_{k^*},$$

where $k^* = argmax_{k \ge 1}(gap_k)$.

5. A modification of the above approach was proposed by Sugar and James in the framework of the rate distortion theory [45]. In this version of a "jump" method, a distortion curve is computed for d-dimensional data. The latter is assumed to have an underling distribution composed of G components with the common covariance matrix $\varGamma$. The distortion value has the form

$$D_k = \frac{1}{d}\min_{c_1,...,c_k} E[(\mathbf{X}-c_x)^t \varGamma(\mathbf{X}-c_x)],$$

where $c_1, ..., c_k$ is a set of k cluster centers obtained by running a standard clustering procedure such as the k-means algorithm; c_x is the center nearest to a given sample of $\mathbf{X}$. Actually, this version of W_k is the average Mahalanobis distance per dimension between the datum and the family of the cluster centers. Next step, a "jumping differential" curve is constructed according to the following rule:

$$J_k = \left(D_k^{-\lambda} - D_{k-1}^{-\lambda}\right),$$

where λ is the transformation power. Its preferred option

$$\lambda = (d/2)$$

is obtained from the asymptotic results of the rate distortion theory. Moreover, for sufficiently high values of d, the differential distortion J_k is approximately zero; if the number of clusters is less than the number of components, then the value jumps and increases linearly. Summarizing the above results, we see that, for sufficiently high values of d, the transformed distortion is approximately zero for $k < G$, then jumps abruptly and increases linearly for $k >= G$. The jump algorithm makes use of this behavior to identify the most likely value of k as the true number of clusters. The estimated number of clusters corresponds to the maximal value of the index J_k. An example of the distortion curve obtained for the data presented in Figure 3 is given in Figure 5.

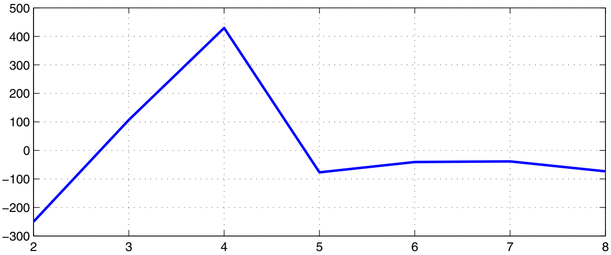

Fig. 5. Graph of the distortion curve

3 Randomized Algorithm

Based on the rate distortion criteria proposed by Sugar and James (see Section 2.3), the task of determining the true number of clusters can be theoretically interpreted as a particular case of a more general problem, namely, the problem of locating the discontinuity points of an implicitly defined function. Let us consider the function of J_k transformed "distortions" mapped into the interval $[0, 1]$ as the index function $I(k)$. This function behaves in a semi-linear way before and after the jump. To determine such jump point, a randomized approach can be used (see [24]). Generally, the problem can be formulated in the following way. Let us take a real-valued function f on the interval $[0, 1]$ having not more than one jump: point of discontinuity $x^* \in [0, 1]$. The considered in [24] problem is: To define the confidence interval for the x^* if the function satisfies conditions:

1. The function $f(\cdot)$ is Lipschitz continuous with a Lipschitz constant C on the intervals $[0, x^*)$ and $(x^*, 1]$;
2. If jump discontinuity exists, then the jump size at this point is above a certain constant value $B > 0$.

The first constant C represents the "smoothness" of the index function on the part of the interval where the function is continuous. The second constant B characterizes a possible "jump" of the index function at the point x^* which corresponds, in our context, to the true number of clusters. Let $k_{\max}$ be the maximal number of clusters tested. Obviously, the case $B >> C$ appears to be the most interesting because the behavior of the index function scaled by $k_{\max}$ near point the x^* should be essentially different from its behavior at other points.

The scenario optimization method discussed in [6] is an effective technique for solving convex optimization problems with large amount of constrains in a probabilistic setting. For any given sufficiently small positive values ϵ and β, the number of random trials N is *a priori* defined for a given sufficiently small positive confidence parameters ϵ and $(1 - \beta)$. Thus, the solution obtained for merely N constraints satisfies all the others with the probability of $(1 - \beta)$ except for a set whose probability does not exceed ϵ.

To implement the above methodology in the framework of the clustering concept, consider the transformed "distortions" I proposed by Sugar and James [45]. For the sake of generality, we assume that

$$I\left(0\right)=I(1)$$

and introduce a continuous piecewise linear function f in the form:

$$f_I\left(\frac{k}{k_{\max}}\right)=I\left(k\right),$$

$$f_I(x)=I(k)+\left(x-\frac{k}{k_{\max}}\right)\left(I(k+1)-I(k)\right)$$

for

$$\frac{k}{k_{\max}}\leq x\leq\frac{k+1}{k_{\max}},k=0,...,k^*-2,k^*,...,k_{\max}-1,$$

$$f_I(x)=I(k^*-1)$$

for

$$\frac{k^*-1}{k_{\max}}\leq x\leq\frac{k^*}{k_{\max}}.$$

In this case, the analogs of the above restrictions 1 and 2 are the following:

1. $C\geq\max_{j=2,\ldots,k^\star-1,k^\star+1,\ldots,k_{\max}}|I(j)-I(j-1)|*k_{\max}$,
2. $B\leq|I(k^\star)-I(k^\star-1)|$.

Thus, it follows that

1. The function f_I has no more than one jump discontinuity $x^*\in[0,1]$.
2. The function $f_I(\cdot)$ is Lipschitz continuous with the Lipschitz constant C on the intervals $[0,x^*)$ and $(x^*,1]$.
3. If there exits jump discontinuity, then the function jump size at the jump point is above some constant value $B>0$.

An algorithm which implements the approach under consideration can be described as follows:

1. Choose the reliability parameters $\beta\in(0,1)$.
2. Choose the parameter M presenting the highest power in the approximation of the function f_I by means of Chebyshev polynomials:

$$p_m(x)=\cos\left(m\arccos x\right),\ m=0,1,2,\ldots,M. \tag{9}$$

3. Choose a number $N\geq M$ and set the number of a group points $T>1$:

$$T=\left[\frac{4C}{\beta BN}-\frac{1}{N}\right]. \tag{10}$$

4. Choose randomly T sets of points having size N in the interval $(0, 1)$:
$$Z_t = \{z_{tn},\ n = 1, ..., N\},\ t = 1, ..., T$$
and denote
$$Z = \bigcup_t Z_t.$$
In the proof of Theorem 1, it will be demonstrated that the largest distance between two sequential points belonging to Z does not exceed $B/4C$ with the probability of $(1 - \beta)$.
5. For each one of the groups Z_t, $t = 1, ..., T$ construct the uniform approximation for $f_I(x)$:
$$g_t(x) = \sum_{m=0}^{M} d_{tm} p_m(x), t = 1, ..., T \tag{11}$$
minimizing the lost
$$\gamma_t = \max_{x \in Z_t} |g_t(x) - f_I(x)|$$
subject to
$$|d_{tm}| \le D,\ m = 0, ..., M,\ t = 1, ..., T,$$
where D is a constant.
Here a convex optimization MathLaB's TOOLBOX (YALMIP, SeDuMi or cvx) can be applied.
If one of the approximation problems is not resolved then return to Step 2 with another parameters M, N, K, D.
6. Define the functions
$$\chi(x) = \max_{t=1,...,T} g_t(x) - \min_{t=1,...,T} g_t(x),\ x \in (0, 1) \tag{12}$$
and
$$h(x) = \max_{z \in [z_l(x), z_r(x)]} \max_{t=1,...,T} |g'_t(z)|, \tag{13}$$
where
$$z_l(x) = \arg\max\{z\ \in Z : z <= x\},\ x \in (0, 1)$$
and
$$z_r(x) = \arg\min\{z\ \in Z : z > x\},\ x \in (0, 1).$$
7. Calculate
$$\gamma = \max_t \gamma_t \tag{14}$$
and introduce the high line (the level of decision-making)
$$L(x) = \frac{3B}{4} - \frac{B}{4C} h(x) - 2\gamma.$$
The interval
$$\mathit{\Delta} = \{\tilde{x} = x k_{max} : x \in (0, 1),\ \chi(x) > L(x)\} \tag{15}$$

is not empty with the probability of

$$P = (1 - \beta)$$

and the true number of clusters is located in Δ.

Theorem 1. *If conditions 1 and 2 formulated above hold, then, with the probability of* $p = (1 - \beta)$ *the set* Δ *is not empty and contains the point* $x^* k_{\max}$ *equal to the true number of clusters.*

Sketch of Proof. By virtue of Markov's inequality, it follows from Condition 10 that there exist two points z_{i_l} and z_{j_r} in Z:

$$z_{i_l} < x^*,$$

$$z_{j_r} >= x^*,$$

and

$$|z_{j_r} - z_{i_l}| \leq \frac{B}{4C}$$

with probability of $(1 - \beta)$. Consider the corresponding functions g_i or g_j. It follows from Definition (14) that

$$|f_I(z_{i_l}) - g_i(z_{i_l})| + |f_I(z_{j_r}) - g_j(z_{j_r})| \leq 2\gamma.$$

Consider the intervals $\bar{\Delta}_l = [z_{i_l}, x^*]$ and $\bar{\Delta}_r = [x^*, z_{j_r}]$. The following relationships can be subsequently derived from the above formulas and conditions of the algorithm:

$$\chi(x^*) \geq |g_j(x^*) - g_i(x^*)| \geq |g_j(z_{j_r}) - g_i(z_{i_l})| - (|\bar{\Delta}_l| + |\bar{\Delta}_r|)H \geq$$

$$\geq |f_I(z_{j_r}) - f_I(z_{i_l})| - 2\gamma - (|\bar{\Delta}_l| + |\bar{\Delta}_r|)H \geq B - 2\gamma - (|\bar{\Delta}_l| + |\bar{\Delta}_r|)(H + C) \geq$$

$$\geq B - 2\gamma - \frac{B}{4C}(H + C),$$

where H is the maximal derivation $g_i(\cdot)'$ on the interval $[z_{i_l}, z_{j_r}]$.

Finally, taking into account the definition (13) we obtain

$$\chi(x^*) \geq \frac{3B}{4} - \frac{B}{4C}h(x^*) - 2\gamma.$$

4 Examples

Example 1. The first dataset is available at http://archive.ics.uci.edu/ml/datasets/Libras+Movement. This datum contains 15 equal sized clusters of 24 instances. Each set refers to one type of hand movements in the official Brazilian signal language LIBRAS.

In the video pre-processing, time normalization is carried out by selecting 45 frames from each video according to a uniform distribution. In each frame, the centroid pixels of the segmented objects (the hand) are found, which compose the discrete version of curve F comprising 45 points. All curves are normalized in the unitary space. In order to make the hand movements suitable for algorithm analysis, a mapping procedure has been carried out, in which each curve F is mapped in a representation with 90 features, with representing the coordinates of movement.

- Number of Instances - 360;
- Number of Attributes - 91.

We consider the interval $[1, 30]$ which, supposedly, contains the true number of clusters. For each point, the transformed Sugar and James distortion function $I(k)$ is calculated using the Partition Around Medoids approach. The curve obtained in this way is presented in Figure 6.

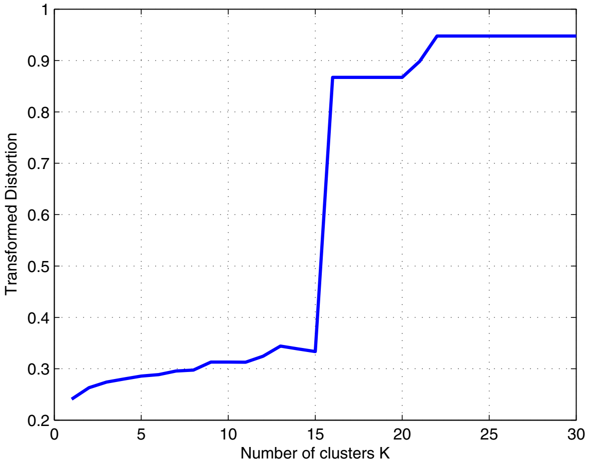

Fig. 6. Sugar and James distortion function $I(k)$ calculated for the official Brazilian signal language LIBRAS dataset

The curve has a "jump" at the point $x^* = 15$ with $B = 0.534$ and

$$c = \max_{k \neq 15}(I(k) - I(k-1)) = 0.049.$$

Thus the true number of clusters is found. The Lipschitz constant of the function f_I is $0.049 * 30 = 1.482$. The total number of the computations of the function $I(k)$ is 30 in this case.

In the framework of the new method proposed here, the number of the points at which the index function values are computed, can be reduced. For example, if we choose $\beta = 0.9$, $M = 4$, $N = 5$, $T = 3$, $D = 0.9$, the function values are

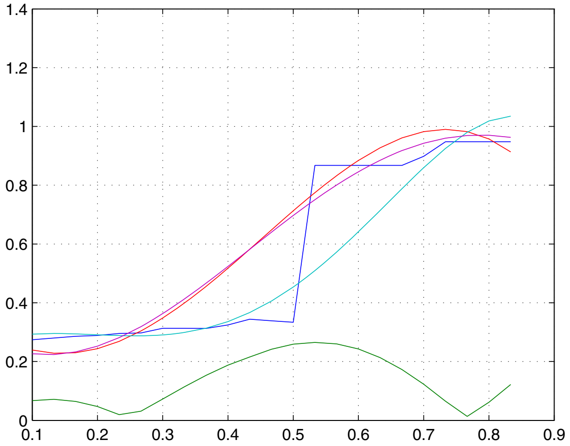

Fig. 7. Approximation curves $g_t(\cdot)$ presented and the resulting function $\chi(\cdot)$

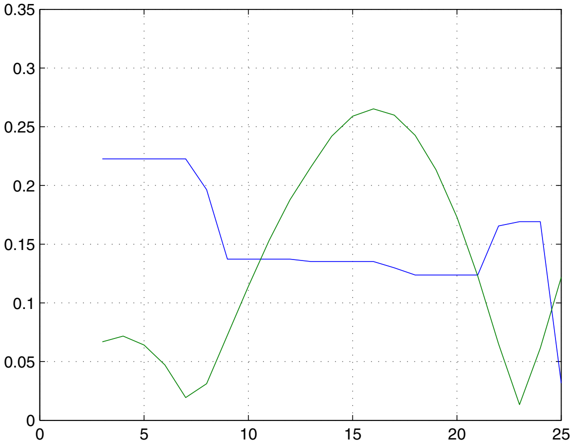

Fig. 8. Level of decision making and the resulting function $\chi(\cdot)$

calculated only at 15 points. As a result, three values of $\{0.034, 0.022, 0.021\}$ of γ_t are obtained, which correspond to the three approximation curves $g_t(\cdot)$ presented in Figure 7 together with the resulting function $\chi(\cdot)$.

We do not know the real values of B and C before the calculation of all values. Thus we need to make some "a priory" assumptions about it. If we suggest that $B > 0.5$ and $k_{\max}B/C \approx 10$ then we get the level of decision making which is exposed on Figure 8 together with the resulting function $\chi(\cdot)$. It can be seen that the curve has a peak located near the point $x^* = 15$.

If we choose the confidential interval $[11, 21]$, then 10 additional calculations of the index function $I(x)$ are required to obtain the final solution of the original problem. As the parameter B values increase, the corresponding confidential interval decreases.

Example 2. To check whether the algorithm proposed here can be applied to a large number of clusters, a synthetic dataset was generated. It contained 1024 clusters, each composed of 8 - 16 instances. Instances in each cluster were generated according to a uniform distribution based on a circle with the radius from 10 to 30 (a random value for each cluster).

- Number of Instances - 11245;
- Number of Attributes - 2.

The scatter plot of the synthetic dataset is presented in Figure 9.

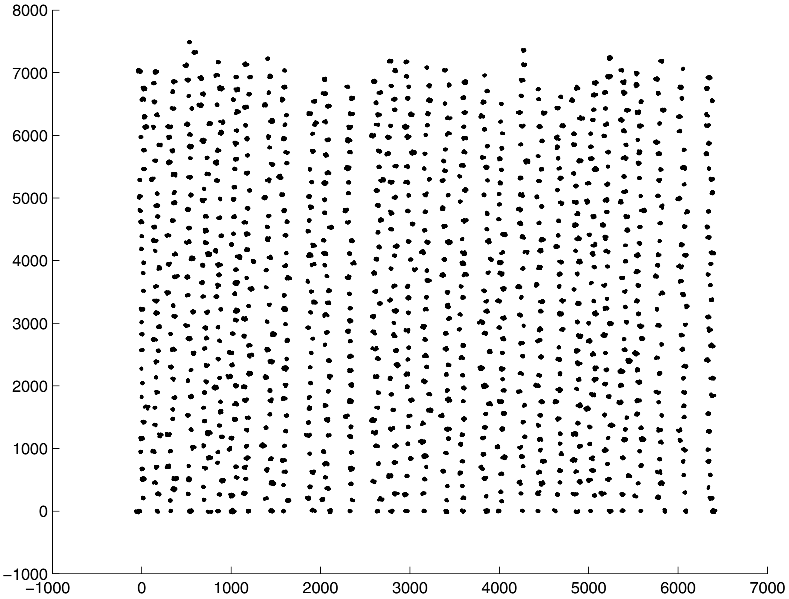

Fig. 9. Synthetic dataset with 1024 clusters

We consider the interval $[1, 3100]$ which contains the real number of clusters. For each point the transformed distortion function $I(k)$ is calculated using the algorithm of Sugar and James. The results are presented in Figure 10.

The scenario approach described above allows us to reduce significantly the number of clustering algorithm rerunning. If we choose $\beta = 0.95$, $M = 8$, $N = 10$, $T = 3$ and $D = 0.7$, then we have to calculate only 30 values of $I(k)$ instead of 3100. Three approximation curves $g_t(\cdot)$ are shown in Figure 11, together with the resulting function $\chi(\cdot)$.

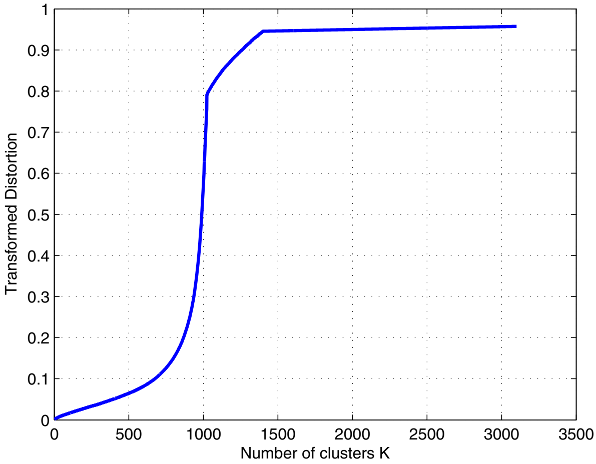

Fig. 10. Sugar and James distortion function $I(k)$

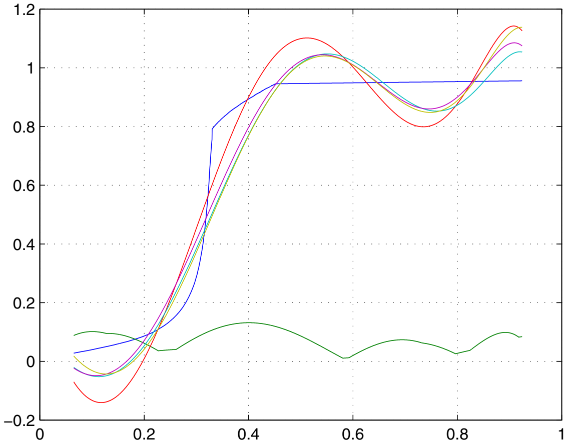

Fig. 11. Level of decision making and the resulting function $\chi(\cdot)$

With the assumption $B > 0.7$ and $k_{\max}B/C \approx 10$ we obtain the level of decision making which is shown in Figure 12 together with the resulting function $\chi(\cdot)$.

A peak near the point $x^* = 1024$ can be observed.

If we choose the confidential interval $[950, 1358]$, then, in order to resolve the original problem, 408 additional calculations of the index function $I(x)$ must be performed. The total number of the computations to be made is 438, which is significantly less than the above number of 3100.

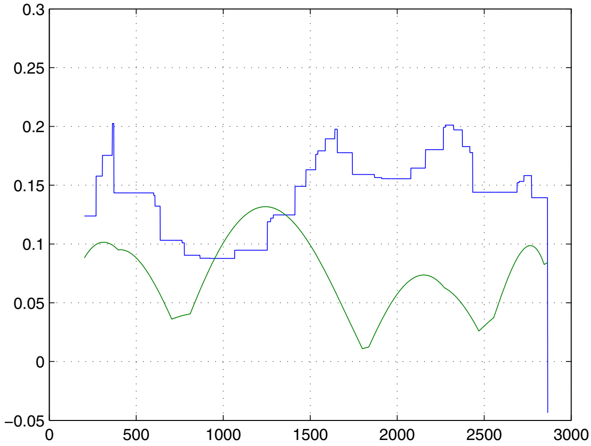

Fig. 12. Level of decision making and the resulting function $\chi(\cdot)$

5 Conclusion

We propose a novel method for the cluster stability assessment based on the ideas of randomized learning theory in the spirit of the well-known "elbow criterion". The main idea is to compute a small amount of differential distortion function values and to allocate the jump position (an elbow) relying on its approximations by a fixed set of Chebyshev polynomials with uniformly bounded coefficients. A confidence interval for the true number of clusters can be obtained by comparatively small amount of the distortion calculations. As a result one can get sufficiently small confidence interval. The significant decreasing of computations is proved under very general conditions.

References

1. Banfield, J.D., Raftery, A.E.: Model-based gaussian and non-gaussian clustering. Biometrics 49, 803–821 (1993)
2. Barzily, Z., Volkovich, Z., Akteke-Ozturk, B., Weber, G.-W.: On a minimal spanning tree approach in the cluster validation problem. Informatica 20(2), 187–202 (2009)
3. Ben-Hur, A., Elisseeff, A., Guyon, I.: A stability based method for discovering structure in clustered data. In: Pacific Symposium on Biocomputing, pp. 6–17 (2002)
4. Ben-Hur, A., Guyon, I.: Detecting stable clusters using principal component analysis. In: Brownstein, M.J., Khodursky, A. (eds.) Methods in Molecular Biology, pp. 159–182. Humana press (2003)

5. Breckenridge, J.: Replicating cluster analysis: Method, consistency and validity. Multivariate Behavioral Research 24, 147–161 (1989)
6. Calafiore, G., Campi, M.C.: The scenario approach to robust control design. IEEE Trans. Automat. Control 51(5), 742–753 (2006)
7. Calinski, R., Harabasz, J.: A dendrite method for cluster analysis. Communications in Statistics 3(1), 1–27 (1974)
8. Celeux, G., Govaert, G.: A classification *em* algorithm and two stochastic versions. Computational Statistics and Data Analysis 14, 315–332 (1992)
9. Cheng, R., Milligan, G.W.: Measuring the influence of individual data points in a cluster analysis. Journal of Classification 13, 315–335 (1996)
10. Cuevas, A., Febrero, M., Fraiman, R.: Estimating the number of clusters. The Canadian Journal of Statistics 28(2), 367–382 (2000)
11. Cuevas, A., Febrero, M., Fraiman, R.: Cluster analysis: A further approach based on density estimation. Computational Statistics and Data Analysis 28, 441–459 (2001)
12. Dhillon, I.S., Kogan, J., Guan, Y.: Refining clusters in high-dimensional text data. In: Dhillon, I.S., Kogan, J. (eds.) Proceedings of the Workshop on Clustering High Dimensional Data and its Applications at the Second SIAM International Conference on Data Mining, pp. 71–82. SIAM, Philadelphia (2002)
13. Dhillon, I.S., Kogan, J., Nicholas, C.: Feature selection and document clustering. In: Berry, M.W. (ed.) A Comprehensive Survey of Text Mining, pp. 73–100. Springer, Heidelberg (2003)
14. Dhillon, I.S., Mallela, S., Kumar, R.: Enhanced word clustering for hierarchical text classification. In: Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining(KDD-2002), pp. 191–200 (2002)
15. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification, 2nd edn. John Wiley and Sons, Chichester (2000)
16. Dudoit, S., Fridly, J.: A prediction-based resampling method for estimating the number of clusters in a dataset. Genome Biol. 3(7) (2002)
17. Dunn, J.C.: Well Separated Clusters and Optimal Fuzzy Partitions. Journal Cybern. 4, 95–104 (1974)
18. Feng, Y., Hamerly, G.: *pg*-means: learning the number of clusters in data. In: Proceedings of the Twentieth Annual Conference on Neural Information Processing Systems (NIPS) (December 2006)
19. Forgy, E.W.: Cluster analysis of multivariate data - efficiency vs interpretability of classifications. Biometrics 21(3), 768 (1965)
20. Fowlkes, E.W., Mallows, C.L.: A method for comparing two hierarchical clusterings. J. Am. Stat. Assoc. 78, 553–584 (1983)
21. Fraley, C., Raftery, A.E.: How many clusters? which clustering method? answers via model-based cluster analysis. The Computer Journal 41(8), 578–588 (1998)
22. Gordon, A.D.: Identifying genuine clusters in a classification. Computational Statistics and Data Analysis 18, 561–581 (1994)
23. Gordon, A.D.: Classification. Chapman and Hall, CRC, Boca Raton, FL (1999)
24. Granichin, O.N., Khalidov, V.I.: Randomized approach to the detection of discontinuity of a function. Stochastic Optimization in Informatics 1(1), 73–80 (2005)
25. Hamerly, G., Elkan, C.: Learning the k in k-means. In: Proceedings of the seventeenth annual conference on neural information processing systems (NIPS), December 2003, pp. 281–288 (2003)

26. Hartigan, J.: Statistical theory in clustering. Journal Classification 2, 63–76 (1985)
27. Hartigan, J.A.: Clustering Algorithms. John Wiley, New York (1975)
28. Hartigan, J.A.: Consistency of single linkage for high-density clusters. Journal of the American Statistical Association 76, 388–394 (1981)
29. Hubert, L., Schultz, J.: Quadratic assignment as a general data-analysis strategy. Br. J. Math. Statist. Psychol. 76, 190–241 (1974)
30. Jain, A., Dubes, R.: Algorithms for Clustering Data. Prentice-Hall, Englewood Cliffs (1988)
31. Jain, A.K., Moreau, J.V.: Bootstrap technique in cluster analysis. Pattern Recognition 20(5), 547–568 (1987)
32. Kass, R.E.: A reference bayesian test for nested hypotheses and its relationship to the schwarz criterion. The Journal of the American Statistical Association 90(431), 928–934 (1995)
33. Kaufman, L., Rousseeuw, P.J.: Finding Groups in Data: An Introduction to Cluster Analysis. Wiley and Sons, New York (1990)
34. Kogan, J., Teboulle, M., Nicholas, C.: The entropic geometric means algorithm: an approach for building small clusters for large text datasets. In: Boley, D., et al.(eds.) Proceedings of the Workshop on Clustering Large Data Sets (held in conjunction with the Third IEEE International Conference on Data Mining), pp. 63–71 (2003)
35. Krzanowski, W., Lai, Y.: A criterion for determining the number of groups in a dataset using sum of squares clustering. Biometrics 44, 23–34 (1985)
36. Lange, T., Braun, M., Roth, V., Buhmann, J.M.: Stability-based model selection (2003)
37. Lange, T., Roth, V., Braun, M.L., Buhmann, J.M.: Stability-based validation of clustering solutions. Neural Computation 16(6), 1299–1323 (2004)
38. Levine, E., Domany, E.: Resampling method for unsupervised estimation of cluster validity. Neural Computation 13, 2573–2593 (2001)
39. Mardia, J., Kent, K., Bibby, J.: Multivariate Analysis. Academic Press, San Diego (1979)
40. Milligan, G., Cooper, M.: An examination of procedures for determining the number of clusters in a data set. Psychometrika 50, 159–179 (1985)
41. Mufti, G.B., Bertrand, P., El Moubarki, L.: Determining the number of groups from measures of cluster validity. In: In Proceedigns of ASMDA 2005, pp. 404–414 (2005)
42. Pelleg, D., Moore, A.: X-means: Extending k-means with efficient estimation of the number of clusters. In: Proceedings of the 17th International Conf. on Machine Learning, pp. 727–734. Morgan Kaufmann, San Francisco (2000)
43. Rand, W.: Objective criteria for the evaluation of clustering methods. Journal Am. Stat. Assoc. 66, 846–850 (1971)
44. Stuetzle, W.: Estimating the cluster tree of a density by analyzing the minimal spanning tree of a sample. J. Classification 20(5), 25–47 (2003)
45. Sugar, C.A., James, G.M.: Finding the number of clusters in a dataset: An information-theoretic approach. J. of the American Statistical Association 98(463), 750–763 (2003)
46. Tibshirani, R., Walther, G.: Cluster validation by prediction strength. Journal of Computational & Graphical Statistics 14(3), 511–528 (2005)
47. Tibshirani, R., Walther, G., Hastie, T.: Estimating the number of clusters via the gap statistic. J. Royal Statist. Soc. B 63(2), 411–423 (2001)

48. Volkovich, Z., Barzily, Z.: On application of probability metrics in the cluster stability problem. In: 1st European Conference on Data Mining (ECDM 2007), Lisbon, Portugal, July 2007, pp. 5–7 (2007)
49. Volkovich, Z., Barzily, Z., Avros, R., Toledano-Kitay, D.: On application of the k-nearest neighbors approach for cluster validation. In: Proceeding of the XIII International Conference Applied Stochastic Models and Data Analysis (ASMDA 2009), Vilnius (2009)
50. Volkovich, Z., Barzily, Z., Morozensky, L.: A statistical model of cluster stability. Pattern Recognition 41(7), 2174–2188 (2008)
51. Wishart, D.: Mode analysis: A generalisation of nearest neighbour which reduces chaining effects. In: Numerical Taxonomy, pp. 282–311 (1969)