UNIV. BEOGRAD. PUBL. ELEKTROTEHN. FAK.
Ser. Mat. 3 (1992), 21–26

# GRAPH THEORETICAL PROCEDURES IN CLUSTERING DISCRETE DATA

*Dragoš Cvetković*

**We report on difficulties in applying traditional clustering procedures to discrete data. We describe a graph theoretical approach in clustering binary vectors. New clustering procedures are combined from several algorithms and heuristics from graph theory and combinatorial optimizations.**

## 1. INTRODUCTION

We consider the problem of clustering data (see, e.g., [**1**], [**2**]). The data are usually represented by vectors from $\mathbb{R}^n$. Euclidean or other kind of distance function $d(x, y)$ is assumed to be defined for any $x, y \in \mathbb{R}^n$. Given a set of vectors from $\mathbb{R}^n$, the problem is to partition it into subsets called *clusters* under various conditions. Clustering methods are supposed to produce clusters which have the property that vectors from the same cluster in some sense are "closer" one to the other than the vectors from different clusters. The number of clusters may but need not to be given in advance. Sometimes cardinalities of clusters are given or limited by additional conditions.

In this paper we consider clustering of discrete data. A typical example of discrete data are binary vectors, i.e. elements of $B^n$ where $B = \{0, 1\}$. When standard clustering procedures (see, e.g., [**1**], [**2**]) are applied to binary vectors, the resulting clustering has usually a low quality. Among other things, the clustering is highly dependent of the ordering of vectors.

To avoid these difficulties it seems reasonable to use specific properties of discrete data and to apply combinatorial, including graph theoretical, tools in handling the problem. We have developed a number of complex graph theoretical procedures for clustering binary vectors [**4**]. In this paper we describe the procedure for clustering into a given number of clusters, which stems from [**4**]. Other results from [**4**] will be published in future papers. The material from this paper has

1991 AMS Subject Classification: Primary 62 H 30, Secondary 05 C 85

been presented at the 10th Yugoslav seminar on graph theory, Sarajevo—Jahorina, 20–21 April 1990 and the 3rd ECCO meeting, Barcelona, 2–4 May 1990.

## 2. SOME DEFINITIONS

A *hypercube* $H_n$ of dimension $n$ is the graph whose vertex set is $B^n$ and two $n$-tuples are adjacent if they differ in exactly one coordinate. The number of coordinates in which $n$-tuples $x, y \in B^n$ differ is called the HAMMING *distance* between $x$ and $y$.

For a graph $G$ we define its *k-th power* $G^k$. The graph $G^k$ has the same vertex set as $G$ and vertices $x$ and $y$ are adjacent in $G^k$ if they are at (graph theoretical) distance at most $k$ in $G$. For $k = 0$ the graph $G^k$ consists of isolated vertices. For $k = 1$ we have $G^k = G$. If $X$ is a subset of the vertex set of a graph $G$ then $G(X)$ denotes the subgraph of $G$ induced by $X$.

Let $X \subset B^n$ be a set of binary vectors ($n$-tuples) which is to be clustered. Our procedures for clustering makes use of the graph sequence

$$H_n^0(X), H_n^1(X), H_n^2(X), \dots, H_n^n(X) \tag{1}$$

which is called the *basic graph sequence*.

Note that two vectors $x, y \in X$ are at the HAMMING distance $k$ if they are not adjacent in $H_n^{k-1}(X)$ and are adjacent in $H_n^k(X)$. For $i = 1, \dots, n$ the graph $H_n^i(X)$ has all edges from $H_n^{i-1}(X)$ plus those ones connecting vectors at HAMMING distance $i$. $H_n^0(X)$ has only isolated vertices while $H_n^n(X)$ is a complete graph.

Let the vertex set $X$ of a graph $G$ be partitioned into subsets $X_1, X_2, \dots, X_m$. A *condensation* of $G$ is a weighted graph on vertices $x_1, x_2, \dots, x_m$ (called *supervertices*) in which $x_i$ and $x_j$ are connected by an edge if there is at least one edge between $X_i$ and $X_j$ in $G$. Both supervertices and edges in the condensation carry weights. The weight of the supervertex $x_i$ is equal to $|X_i|$ while the weight of the edge between $x_i$ and $x_j$ is equal to the number of edges between $X_i$ and $X_j$. We consider a condensation as a multigraph where edge weights are interpreted as edge multiplicities while supervertices as vertices and supervertex weights are ignored.

Let $A$ be the adjacency matrix of a (multi-)graph $G$ and let $D$ be a diagonal matrix with vertex degrees on the diagonal. The matrix $C = D - A$ is called the KIRCHHOFF (or *Laplacian* or *admittance*) *matrix* of $G$. Let $\mu_1, \mu_2, \dots, \mu_n$ ($\mu_1 \geq \mu_2 \geq \dots \geq \mu_n$) be eigenvalues of $C$. We have $\mu_n = 0$ and the quantity $a(G) = \mu_{n_1}$ is called the *algebraic connectivity* of $G$ (see **[8]** or **[6]**, pp. 265–266).

In the clustering procedure, which will be described in the next section, some algorithms and heuristics described in the literature will be used. We shall quote them in the sequel.

**Algorithm CP.** This is an algorithms for finding components of a graph (**[12]**, pp. 398–405). One starts from a graph without edges when each vertex represents a component. Gradually, we introduce edges of the actual graph thus uniting two components if the edge added links them.

**Algorithm JM.** This is an algorithm for partitioning a connected (multi-)graph into two parts (see [**9**] or [**5**], p. 78). The partition is determined by the eigenvector belonging to the largest eigenvalue of the matrix $PAP$ where $A$ is the adjacency matrix with ones on the diagonal and $P = \|p_{ij}\|_1^n$ with $p_{ij} = \delta_{ij} - 1/n$, $\delta_{ij}$ being the KRONECKER $\delta$-symbol. Vertices with positive coordinates in this eigenvector form one cluster; those with negative coordinates form the other one.

**Heuristic KL.** This is a heuristic for partitioning the vertex set of a (multi-) graph into two parts of given cardinalities with a minimum number of edges between vertices from different parts [**10**]. One starts from a randomly generated partition into two parts on which a local optimization is applied making use of exchanging vertices from different parts.

## 3. A CLUSTERING PROCEDURE

Let $X$ be a set of binary vectors of dimension $n$ and suppose we have to cluster it into $k$ ($k > 1$) clusters. For $k = 2$ we consider the problem in two variants: 1° Cluster cardinalities are not given, 2° Cluster cardinalities are given.

Our procedure consists of two phases.

**Phase 1.** We form the basic graph sequence (1). Let $c_i$ be the number of components of $H_n^i(X)$. Components are sequentially determined in graphs from the basic sequence by algorithm CP. We have $c_0 = |X| \geq c_1 \geq c_2 \geq \ldots \geq c_n = 1$.

There is a non-negative integer $s$ such that $c_s \geq k > c_{s+1}$, the components of $H_n^s(X)$ are clusters and the procedure is finished. If $c_s > k > c_{s+1}$ we proceed to Phase 2.

**Phase 2.** We distinguish cases: 1° $k = 2$ and 2° $k > 2$.

**Case $k = 2$.** Now $H_n^{s+1}(X)$ is connected and we consider the condensation of the graph $H_n^{s+1}(X)$ in which components of $H_n^s(X)$ play role of supervertices.

We consider two subcases:

1° Cluster cardinalities are not given;

2° Cluster cardinalities are given.

**Subcases 1°.** If $c_s > 10$ any of the following two procedures can be applied to the condensation of $H_n^{s+1}(X)$:

a) algorithm JM;

b) heuristic KL.

In any of these cases we get two clusters and the whole procedure is finished.

In variant b) the user can select the range of cluster cardinalities and the number of random generated staring clusterings. The result in variant a) can serve as a hint for the range of cluster cardinalities in variant b).

If $c_s \leq 10$, we form all partitions of the vertex set of the condensation of $H_n^{s+1}(X)$ into two parts since there are only $2^{c_s} - 2$ such partitions. We find the best partition with respect to a selected quality criterion (e.g. minimizing the edge number between two parts). The whole procedure is thus finished.

**Subcase 2°.** We apply algorithm JM to the condensation of $H_n^{s+1}(X)$. If the partition thus obtained shows cluster cardinalities required, we have done. Otherwise we apply heuristic KL to the graph $H_n^{s+1}$ where the starting partitions are formed on the basis of information obtained by the working of the algorithm JM. Let $p, q$ $(p \geq q)$ be the required cluster cardinalities. Let algorithm JM have given a solution with cluster cardinalities $r, s$ $(r \geq s)$. If $p < r$, from the cluster of cardinality $r$ we chose those $p$ vertices for which moduli of the coordinates of the eigenvector from algorithm JM are as great as possible. If $p > r$, then $q < s$, and from the cluster of cardinality $s$ we chose $q$ vertices as above. The result of the working of heuristic KL for the starting partition so formed is compared with result for other, randomly generated, starting partitions.

**Case** $k > 2$**.** Now we have $c_s > k > c_{s+1}$ and we get a clustering into $k$ clusters in one of the following two ways

1° by splitting some of $c_{s+1}$ components of the graph $H_n^{s+1}(X)$ into parts;

2° by uniting some of $c_s$ components of $H_n^s(X)$.

We use first way if $k$ is closer to $c_{s+1}$ than to $c_s$ and the second one otherwise.

Splitting components we perform by partitioning a component into two parts and by iterating this procedure. First we partition components of $H_n^{s+1}(X)$ which do not exist in $H_n^s(X)$ and if there are not sufficiently many such components we treat sequentially those which exist in $H_n^i(X)$ and not in $H_n^{i-1}(X)$ for $i = s, s-1, \dots$. For components of $H_n^i(X)$ $(i = s+1, s, \dots)$ we form condensations with supervertices corresponding to components of $H_n^{i-1}(X)$ and for each condensation we determine the ratio of the algebraic connectivity and the number of vertices. Condensations are ordered by this ratio and partitioned sequentially into two parts starting from those with a smallest ratio. In each step of partitioning the newly generated components are treated as above. For partitioning components into two parts we apply the procedure from the case $k = 2$ above.

When uniting components we consider all possibilities of uniting if $c_s - k < 4$. Otherwise we apply the WARD method, which is one of the best hierarchical clustering methods (see, e.g., [**1**], [**2**]).

## 4. COMMENTS

Both theoretical considerations and experiments on a computer have indicated the inadequacy of standard clustering methods to handle binary vectors. For example, in hierarchical methods (e.g. single or complete linkage) at each step there are usually very many pairs of clusters which are equally good candidates to be united. Hence we can get very different clusterings depending on the choice at each step or depending of the original ordering of vectors if this ordering determines the choice. More details will be given in a forthcoming paper.

Concerning computer experiments, we used the system PARIS [**3**] for standard clustering techniques and the system GRAPH [**7**] as well as some newly developed software [**11**] for graph theoretical techniques.

If the procedure finishes in the first phase, the clusters obtained are components of $H_n^s(X)$. This means that two vectors from the same cluster are at the Hamming distance at most $s$, while two vectors from different clusters are at the Hamming distance greater than $s$. If the procedure finishes in the second phase, this nice property does not hold any more. Now some vectors from different clusters can be at distance $s$ or less than $s$ but the procedure tends to minimize the number of such cases.

When splitting a component $C$ of $H_n^i(X)$ in the second phase, we actually split its condensation where supervertices are components of $H_n^{i-1}(X)$ from which $C$ has been created. In this way we ensure that two vertices from the same component of $H_n^{i-1}(X)$ cannot appear in different clusters.

The algebraic connectivity is known to be a very useful parameter for describing the "shape" of a graph (see, e.g., [**6**, p. 266]). Indeed, low algebraic connectivity shows small connectivity and girth and high diameter, although such a statement lacks a precise formulation. In the context of clustering, low algebraic connectivity indicates that the graph has good clustering properties.

Algorithm JM and the calculation of the algebraic connectivity have complexity $O(|X|^3)$ while other parts of the procedure have lower complexities. Therefore the whole procedure has complexity $O(|X|^3)$ and this is the same as in many standard clustering procedures. However, theoretical reasons and numerical experiments show that the graph theoretical procedure is superior to standard procedures in clustering binary vectors.

## REFERENCES

1. : *Selected topics in pattern recognition with applications.* (Serbian), Mathematical Institute, Novi Sad, 1986.
2. : *Cluster analysis for applications.* Academic Press, New York, 1973.
3. : *PARIS—an interactive system for the analysis and recognition of patterns.* (Serbian), University of Belgrade, Faculty of Electrical Engineering, Belgrade, 1988.
4. : *Combinatorial algorithms and heuristics for clustering points of a hypercube, I, II, III, IV.* (Serbian), University of Belgrade, Faculty of Electrical Engineering, Belgrade, pp. 32 + 39 + 53 + 34, unpublished report, 1988.
5. : *Recent results in the theory of graph spectra.* North-Holland, Amsterdam, 1988.
6. : *Spectra of graphs — Theory and application.* Academic Press, New York, 1980.

7. : *Man-machine theorem proving in graph theory.* Artificial Intelligence, **35** (1988), 1, 1–23.

8. : *Algebraic connectivity of graphs.* Czechoslovak Math. J., **23** (**9**) (1973), 298–305.

9. : *Problems of cluster analysis from the viewpoint of numerical analysis.* Proc. Conf. Numerical Methods, Keszthely, 1977, 405–415.

10. : *An efficient procedure for partitioning graphs.* Bell Sys. Techn. J., **49** (1970), 291–307.

11. : *Algorithms and heuristics for clustering vertices of a graph with applications to pattern recognition.* (Serbian), Master Thesis, University of Belgrade, Faculty of Electrical Engineering, Belgrade, 1989.

12. : *Algorithms.* Addison-Wesley, Reading, 1983.

Faculty of Electrical Engineering,
University of Belgrade,
P.O. Box 816, 11001 Belgrade,
Yugoslavia

(Received December 1, 1991)