II Krajowa Konferencja
Metody i systemy komputerowe
w badaniach naukowych i projektowaniu inżynierskim
Kraków, 25-27 października 1999

Clustering and classification of elements in multi-dimensional metric spaces

Grzegorz Urbanek
Faculty of Fundamental Machine Design
Silesian Technical University
ul. Konarskiego 18a, 44-100 Gliwice
Grzegorz.Urbanek@polsl.pl

Summary. *A new MATLAB toolbox for clustering and classification is worked out.*

First section contains theoretical introduction to problems connected with clustering and classification. It also contains information about measures of distance and similarity, algorithms of assembling, classifiers, algorithms of classification, estimations of quality of partition and classification. Second section introduces possibility of usage worked out toolbox.

This article is based on M. Sc. Thesis [4].

1 Theoretical introduction. Clustering and classification are problems closely connected with pattern recognition.

Pattern recognition is an area of researches, which goes about working and projecting of systems recognizing patterns in data. It contains such domain as discriminate analysis, features selection, error estimation, cluster analysis (sometimes called statistical pattern recognition), grammatical inference (sometimes called syntactical pattern recognition). Important areas of uses are: images analysis, writing recognizing, speech analysis, people and machines diagnosis, identification of persons and industrial research.

Clustering [1] is calculation of a partition of set of elements on subsets with respect to some criterion.

Classification [5], often confused with clustering, relies on assignment of a given element to pre-defined classes.

1.1 Distance measure and similarity measure [1]. Distance of elements $x, y \in V$ is value of function $d(x,y)$ called function of distance, such that:

$$d : V \times V \to \{r \in R^1 : r \geq 0\}, \qquad (1)$$

realizing conditions:

$$\underset{x,y \in V}{\forall} [d(x,x) \leq d(x,y)], \qquad (2)$$

$$\underset{x,y \in V}{\forall} [d(x,y) = d(y,x)], \qquad (3)$$

$$\underset{x,y,z \in V}{\forall} [d(x,z) \leq d(x,y) + d(y,z)], \qquad (4)$$

where: R^1 – set of real numbers.

Similarity of elements $x, y \in V$ is value of function $h(x,y)$ called function of similarity, such that:

$$h : V \times V \to \{r \in R^1 : 0 \leq r \leq 1\}, \qquad (5)$$

realizing conditions:

$$\underset{x,y \in V}{\forall} [h(x,y) \leq h(x,x)], \qquad (6)$$

$$\underset{x,y \in V}{\forall} [h(x,y) = h(y,x)]. \qquad (7)$$

Most popular measures of distance and similarity:

1) Euclidean distance measure

$$d^2(x,y) \overset{d}{=} (x-y)(x-y)^T, \quad \forall x, y \in V \qquad (8)$$

2) Distance measure of Sebestyen

$$d^2(x,y) \overset{d}{=} (x-y)\, W (x-y)^T, \forall x, y \in V, \qquad (9)$$

where: W is diagonal weighted matrix

$$W = \begin{bmatrix} w_1 & 0 & \dots & 0 \\ 0 & w_2 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & w_m \end{bmatrix}. \qquad (10)$$

3) Distance measure of Mahalanobis

$$d^2(x,y) \overset{d}{=} (x-y) C^{-1} (x-y)^T \quad \forall x, y \in V \qquad (11)$$

where:

C^{-1} – inverse covariance matrix

$$C = \frac{1}{n}\sum_{j=1}^{n}\left(v_j - \bar{v}\right)^T\left(v_j - \bar{v}\right), \quad v_j \in V, \quad (12)$$

n – number of elements of consider set in space V,

$\bar{v}$ – average element of considers set

$$\bar{v} = \frac{1}{n}\sum_{j=1}^{n} v_j. \quad (13)$$

4) Distance measure of Hamming

$$d(x,y) \overset{d}{=} \frac{1}{m}\sum_{i=1}^{m}|x[i]-y[i]|, \quad \forall x,y \in V^m \quad (14)$$

5) Canberra distance measure

$$d(x,y) \overset{d}{=} \sum_{i=1}^{m}\frac{|x[i]-y[i]|}{x[i]+y[i]}, \ \forall x,y \in V^m \quad (15)$$

6) Similarity measure based on distance

$$h_1(x,y) \overset{d}{=} \frac{1}{1+\alpha\, d(x,y)^{\beta}}, \quad (16)$$

$$h_2(x,y) \overset{d}{=} \exp\left(-\alpha\, d(x,y)\right), \quad (17)$$

where: α, β – constants conditioning properties of function h.

7) "Cosine" similarity measure

$$h(x,y) \overset{d}{=} \frac{x\,y^T}{\sqrt{x\,x^T\,y\,y^T}}. \quad (18)$$

1.2 Clustering criterion [1]. Clustering is a partition of finite set V of elements v into L subsets V_k of similar elements.

This partition can be written in a form of family $Q(V)$ subsets V_k of set V

$$Q(V) = \{V_k \subset V : k \in [1:L]\}. \quad (19)$$

The set of all possible (or all acceptable) partitions $Q(V)$ of set V into L subsets is marked as $Q^L(V)$. Partition Q will be regarded as optimal Q_{opt}, if criterion function $e(Q)$ resulting from accepted criterion achieves extreme for this partition. Described further criterion functions are written in a way that for optimum partition they achieve minimum values

$$Q = Q_{opt} \Leftrightarrow e(Q) = \min_{Q_j \in Q^L}\left(e\left(Q_j\right)\right). \quad (20)$$

Examined criteria make optimization of partition of set V possible on given number subsets and/or optimization of choice of representatives of subsets. They are no efficient in range of optimization of subsets number, on which is divided set V.

Criteria using distance function. The simplest criterion function is

$$e(Q) \overset{d}{=} \frac{1}{V}\sum_{k=1}^{L}\left(\overline{V_k}\, d_k\right), \quad (21)$$

where:

Q – family of subsets V_k,

d_k – average distance of elements k subset V_k from representative of this subset.

Duda and Hart [3] propose criterion of sum of square errors as simplest criterion function

$$e(Q) \overset{d}{=} \sum_{k=1}^{L}\sum_{v \in V_k}\left\|\overline{v_k} - v_k\right\|^2, \quad (22)$$

where: $\overline{v_k}$ – average element

$$\overline{v_k} = \frac{1}{V_k}\sum_{v \in V_k} v. \quad (23)$$

Criteria using similarity function. Similarity function makes it possible to qualify of criteria maximizing "cohesion" of subsets. One of the simplest criteria function is

$$e(Q) \overset{d}{=} -\sum_{k=1}^{L}\sum_{(x,y)\in V_k \times V_k} h(x,y). \quad (24)$$

Threshold criteria. A special group of criteria determines threshold criteria. These criteria can be defined as help of distance function or similarity function of elements. From many criteria of this type we can distinguish two types:

– criterion of „furthest neighbour"

$$\forall_{V_k \in Q_{opt}}\ \forall_{x,y \in V_k}\left[d(x,y) < d_{\max}\right] \quad (25)$$

– criterion of „nearest neighbour"

$$\forall_{V_k \in Q_{opt}}\ \forall_{x \in V_k}\ \exists_{\substack{y \neq x \\ y \in V_k}}\left[d(x,y) \le d_{\max}\right] \quad (26)$$

or similarly for $h(x,y) \ge h_{\min}$.

Advantages of these criteria are their simplicity and great compatibility of optimal clustering results with results in clustering realized arbitrarily by man. Criterion of "nearest neighbour" stands out similarity of pairs of elements. Criterion of "furthest neighbour" focuses on the influence of "isolated elements" of space V. It determines inconvenience from the part of view of existence of deviations of elements of space V resulting from qualifications of position of these elements in space V on the ground of quantities marked experimental.

The main difficulty connected with practical usage of the described threshold criteria is

the necessity of accepting threshold values $h_{\min}$ or $d_{\max}$. Acceptance of these values is equivalent to acceptance of number of sets, to which space V is divided.

1.3 Clustering algorithms. The applied algorithms can be divided into several groups according to their nature:

- algorithms searching general extreme of criterion function,
- iterative clustering algorithm,
- hierarchical grouping algorithm,
- graph-theoretical algorithms,
- algorithms using fuzzy sets.

Algorithms searching general extreme of criterion function ([1], [3]). If criterion was selected, clustering is reduced to a well-defined problem: to find such partition of set of elements, which extremalizes criterion function. If a set of elements is finite then a finite number of possible partitions exists. Theoretically clustering problem can be dissolved through exhaustive searching. Realization of algorithm is simple. In practice this algorithm is not used due to the necessity of realization of a large numbers of operations resulting from the fact, that for space V of power n can be calculated

$$\frac{1}{L!}\sum_{k=1}^{L}\binom{L}{k}(-1)^{L-k}k^{n} \qquad (27)$$

partitions on L sets.

Iterative algorithm ([1]). The essence of iterative algorithm can be described as follows

1. elements, which are considered on approximation of representatives, are chosen $q(V_1),...,q(V_L)$,
2. classification of elements of space V (every element is assigned to a set appointed by representative, for which maximum of similarity function) is obtained,
3. for sets received in this way new representatives are marked,
4. if new appointed representatives of sets differ from previous representatives, then we come back to point 2, which is realized respecting new representatives.

Hierarchical grouping algorithm ([1], [3]). The essence of hierarchical grouping algorithm is acceptance that space V of elements v of power n is divided to n separable single-element sets, of which each element of space belongs to other set. In family of n sets, two of "most similar" sets are looked for. Such sets are join and as a result we receive a family composed of n-1 sets. In family of n-1 sets, two of most similar sets are sought, which are joint and as a result we receive a family of n-2 sets. Finally we will reach required number of sets.

Graph-theoretical algorithms ([3], [6]). Three basic kinds of graph-theoretical algorithms can be distinguished: nearest neighbour, k-nearest neighbours, MMD (mean minimum distance), minimum spanning tree.

The essence of *nearest neighbour algorithm* is calculation of distance of every element of space V to his nearest neighbour. Then on the ground of received results thresholds value $d_{\max}$ is calculated (eg $d_{\max}=\mu+3\sigma$) and every pair of elements, of which distance is less than threshold value is joint.

K-nearest neighbours algorithm relies on connection of every element with his k-nearest neighbours, no matter what absolute value of distance is.

Minimum spanning tree algorithm relies on creation of space V minimum spanning tree from elements and then removes "inadequate" edge. Minimum spanning tree is the connection graph, in which every element is join with other, where closed circuits do not appear, and the sum of length of edges is minimum. Ways of choosing "inadequate" edges can be different; eg removed of longest edges to obtain a given number of groups.

MMD algorithm (mean minimum distance) is modification of the nearest neighbour algorithm. A first arithmetic average distance of elements to nearest neighbour is calculated. Then rejects (as noise) all elements, of which distance to nearest neighbour is over k-times larger from this average. For such set of elements, again the average is calculated. Groups are obtained through connection of every element with its nearest neighbour, if their distance is less or equal then k-times of average. If other premises do not occur it is recommended to accept k=2.

Algorithms using fuzzy sets ([1]). Iterative algorithm can be described for optimization of partition of space V, which results the family pseudoseparated fuzzy sets.

Family of fuzzy sets

$$\left\{\widetilde{A}\right\}=\left\{\widetilde{A}_k=\left\{\left(v,a_k(v)\right):v\in V\right\}:k\in\left[1:L\right]\right\} \quad (28)$$

in space V is family of pseudoseparated fuzzy sets then and only then, when

$$\underset{v\in V}{\forall}\left\lceil\sum_{k=1}^{L}a_k(v)\le 1\right\rceil. \quad (29)$$

Partition of space V will be regarded as optimum, if minimum of criterion function is reached. Let ${}^{(t)}\widetilde{A}_k$ represent wanted k fuzzy set after t iteration step. Transformation of fuzzy sets characteristic function value is

$$\omega_m:{}^{(t)}a_k(v)\quad {}^{(t+1)}a_k(v)=$$
$$=\begin{cases}\alpha+(1-\alpha)\,{}^{(t)}a_k(v) & dla\quad k=m\\ (1-\alpha)\,{}^{(t)}a_k(v) & dla\quad k\neq m\end{cases} \quad (30)$$

where: $0\le\alpha\le 1$ – is assumed parameter.

Described algorithm consists of the following phases:

1) – initial solution is accepted

$${}^{(0)}a_k(v)=\frac{1}{L}, \quad (31)$$

- value of criterion function is calculated $e\left({}^{(0)}\widetilde{A}\right)$,

2) - in turn for all elements and in turn for all appointed fuzzy sets $\widetilde{A}_k$:
- characteristic functions of fuzzy sets is transformed

$$a_k^*(v_j)=\omega_m\left({}^{(t)}a_k(v_j)\right), \quad (32)$$

- value of criterion function is counted $e\left(\left\{\widetilde{A}^*\right\}\right)$,
- if $e\left(\left\{\widetilde{A}^*\right\}\right)<e\left(\left\{{}^{(t)}\widetilde{A}\right\}\right)$, to ${}^{(t+1)}a_k:=a_k^*$ for $k=1,...,L$ we come back to the beginning of point 2,

3) if $e\left(\left\{\widetilde{A}^*\right\}\right)\ge e\left(\left\{{}^{(t)}\widetilde{A}\right\}\right)$, for all $m=1,...,L$, then family $\left\{{}^{(t)}\widetilde{A}\right\}$ appoints optimum partition of space V.

1.4 Estimation of partition quality. Partition quality can be defined by two manners:

$$e_1=\frac{\frac{1}{n}\sum_{k=1}^{L}mean(d(v_i,v_j))}{mean(d(V_p,V_q))}, \quad (33)$$

$$e_2=\frac{\sum_{k=1}^{L}I_k}{I}. \quad (34)$$

1.5 Classifier. Classifier contains information about "location" of classes in features space. Manner leadership of classification is affected by form of classifier.

Binary classifiers ([2]). Most problems will demand the use of classifiers permitting recognition of many classes. Multi-class classifiers are often considered as generalization of two-class classifiers, called binary classifiers.

When solving problems concerning of constructing multi-class classifiers it is justify to conduct simplifying, method relying on decomposition of K-class classifier to suitable family of binary classifiers. In particular it is possible to decompose of classifier:

- to K binary classifiers, where every classifier permits distinction of objects belonging to k class from objects, which do not belong to this class,
- to $K(K-1)/2$ binary classifiers, where every classifier permits consider of pair of classes.

A conduct often is applied relying on direct acceptance of form of function qualifying binary classifier. General rules of optimum choice of form of such function are not well known. We can only recommend the following forms of function qualifying classifier:

- potential functions, when classification problem on K classes became transformed in K two-class problems,
- linear function, when classification problem on K classes becomes transformed in $K(K-1)/2$ two-class problems.

Multi-classes classifiers. The simplest multi-classes classifier is a set of centers and diameters of classes. For such given classifier, classification relies on assigning classified element to classes, to which distance (similarity) of element is smallest (biggest).

More complicated classifier contains the following information: centers of classes and covariance matrix, defining weight of coordinates in each direction.

Estimation of quality of classifier ([2]). General algorithm of constructing and testing of classifier can be written as follows:

1) qualification of family M of sets teaching

and families M of sets testing $(M \geq 1)$,

2) qualification of measure of classifier efficiency,
3) for all $m = 1,...,M$: constructing classifier on the basis of data of concerned elements contained in teaching set and testing of classifier on the basis of data of elements contained in testing set,
4) constructing of classifier on the basis of data relating to all of elements contained in source set, and calculation of average value of efficiency of this classifier.

1.6 Classification algorithms. Generally classification of element relies on assigning this element to classes, to which fits "most"– on the basis of used classifier.

Fuzzy classification. Fuzzy classification relies on announcement of membership degree of classified element to every classes. Membership degree is real number from section [0,1]; 0 marks lack of membership, 1 – total membership.

In case of binary classifier classes membership degree of element is described directly on the ground of function qualifying this classifier. For multi-classes classifier – indirectly, on the ground of function of distance or similarity.

Classification is one of the most general applications of neural networks. Neural networks can be constructed to receive equivalents of binary classifiers or multi-classes classifier. For problem of classification to *K* classes net is constructed about *K* or *K*+1 exit neurons – level of signal on exit of *k* neuron is equivalent to membership degree of element to *k* class. Last exit (*k*+1 neuron) can be used to showing of lack of classification.

Exact classification is special case of fuzzy classification. It is carried out it in the same manner but only as result does not announce degrees of membership to each classes, only for every element announces a number of class to which this element was classified.

Algorithm of exact classification can be introduced as composition of algorithm of fuzzy classification and of conversion of fuzzy membership to exact membership. This conversion is based on the fact that for every element a number of class is given, for which degree of memberships of element is greatest. In case when this greatest degree of membership is smaller then fixed threshold value, this is stated that given element did not be classified to any classes.

Estimation of classification quality. The simplest manner of estimation of classification quality of elements is calculation of average value greatest for every element membership degrees to classes. This estimation assumes values from section [0, 1], values nearer 1 mean better quality of classification.

2 A new MATLAB toolbox for clustering and classification of elements in multidimensional metric spaces. Within of M. Sc. thesis [4] toolbox "KLAS" of procedures of clustering and classification of elements in multi-dimensional metric spaces was worked out. This toolbox was worked our for MATLAB in version 5.1.

2.1 Data format. Elementary form of variables in MATLAB is matrix. In this form necessary data and results are recorded.

1) Elements are numbered following natural numbers begin from one (1, 2, 3, ...).
2) Classes are numbered following natural numbers begin from one (1, 2, 3, ...). Number 0 marks lack of memberships of element to class, undefined membership to class is marked by number -1.
3) Degree of memberships of element to class is real number from section [0,1], eg 0.1, 0.6.
4) Input data (elements to clustering / classification) are recorded in form of matrix, of which following verses describe following elements, columns instead of values of features being real numbers, eg

$$\begin{bmatrix} 5.1 & 3.5 & 1.4 & 0.2 \\ 4.9 & 3.0 & 1.4 & 0.2 \\ 7.0 & 3.2 & 4.7 & 1.4 \\ 6.4 & 3.2 & 4.5 & 1.5 \\ 6.3 & 3.3 & 6.0 & 2.5 \end{bmatrix}. \quad (35)$$

5) Results of clustering / exact classification are recorded in form of two matrices: first matrix is matrix of input data, second matrix is one-column, in verses – number of class, eg

$$\left[\begin{array}{cccc} 5.1 & 3.5 & 1.4 & 0.2 \\ 4.9 & 3.0 & 1.4 & 0.2 \\ 7.0 & 3.2 & 4.7 & 1.4 \\ 6.4 & 3.2 & 4.5 & 1.5 \\ 6.3 & 3.3 & 6.0 & 2.5 \end{array}\right]\left[\begin{array}{c} 1 \\ 1 \\ 2 \\ 2 \\ 3 \end{array}\right] . \quad (36)$$

6) Results of fuzzy classification are recorded in form of two matrices: first matrix as previously, second – following columns correspond to following classes, in verses are recorded degrees of membership to given classes, eg

$$\left[\begin{array}{cccc} 5.1 & 3.5 & 1.4 & 0.2 \\ 4.9 & 3.0 & 1.4 & 0.2 \\ 7.0 & 3.2 & 4.7 & 1.4 \\ 6.4 & 3.2 & 4.5 & 1.5 \\ 6.3 & 3.3 & 6.0 & 2.5 \end{array}\right]\left[\begin{array}{ccc} 0.8 & 0.1 & 0.1 \\ 0.6 & 0.3 & 0.1 \\ 0.1 & 0.7 & 0.2 \\ 0 & 0.9 & 0.1 \\ 0.3 & 0.2 & 0.5 \end{array}\right] . \quad (37)$$

7) Classifier is recorded in a form of matrix, where following verses describe central points of following classes, columns describe values of features, last column describes values of diameters of classes, eg

$$\left[\begin{array}{ccccc} 5 & 3.5 & 1.5 & 0.25 & 2.6 \\ 5.5 & 2.5 & 4 & 1.25 & 3.1 \\ 7 & 3 & 5.5 & 2 & 2.8 \end{array}\right] . \quad (38)$$

2.2 Possibilities. The toolbox worked out in this way makes it possible to solve problems of clustering according to the following algorithms: iterative, hierarchical, nearest neighbour, *k*-nearest neighbours, MMD (mean minimum distance), minimum spanning tree. After execution of clustering one can execute estimations of quality of partition.

Classifier applied in toolbox contains information about centers and diameters of classes.

Classification can be passed according to exact or fuzzy algorithm. One can execute estimation of quality of classification. Classification is passed on the ground of classifier containing information about centers and diameters of classes. Such classifier can be generated on the ground of results of clustering, can also to be given.

Results of clustering and of classification are presented in graphic form, can also be saved in text file.

Toolbox makes it possible to choose one from distance measure: Euclidean, of Hamming, Canberra, of Mahalanobis, or similarity measure: counted on the ground of distances or directly in form of cosine.

3 Reference.

[1] Cholewa W.: Metoda diagnozowania maszyn z zastosowaniem zbiorów rozmytych. Gliwice: Pol. Śl., 1983. Zeszyty Naukowe - Mechanika z. 79.

[2] Cholewa W., Kaźmierczak J.: Diagnostyka techniczna maszyn. Przetwarzanie cech sygnałów. Gliwice: Pol. Śl., 1995. Skrypty uczelniane nr 1904.

[3] Duda R. O., Hart P. E.: Pattern Classification and Scene Analysis. John Wiley & Sons Inc., 1973.

[4] Urbanek G.: Biblioteka procedur grupowania i klasyfikacji elementów wielowymiarowych przestrzeni metrycznych, Praca Dyplomowa Magisterska, Katedra Podstaw Konstrukcji Maszyn, Wydział Mechaniczny Technologiczny, Politechni-ka Śląska, Gliwice 1999.

[5] Wright W. E.: A Formalization of Cluster Analysis.

[6] Zahn C.T.: Graph-Theoretical Methods for Detecting and Describing Gestalt Clusters. IEEE Transactions on Computers, 1971, nr 1, s.68-86.