8
Data clustering methods

8.1 Introduction

In daily life as well as in different fields of science we encounter big, sometimes enormous volume of information. One look is enough for humans to distinguish the shapes of objects being of interest to us from a specific image. Intelligent machines, however, are still incapable of prompt and unerring distinguishing of objects in the image, due to the lack of universal algorithms which would work in every situation.

The objective of data clustering is a partition of data set into clusters of similar data. Objects in the data set may be e.g. bank customers, figures or things in a photograph, sick and healthy persons. A human being may effectively group only one- and two-dimensional data, while three-dimensional data may cause serious difficulties. The scale of the problem is intensified by the fact that the number of samples in real tasks may amount to thousands and millions. In the light of those facts it would be very useful to have algorithms for automatic data clustering. Operation of those algorithms would result in a fixed structure of data partition, i.e. location and shape of the clusters and membership degrees of each sample to each cluster. Data clustering is a complicated issue as the structures hidden in the data set may have any shapes and sizes. Moreover, the number of clusters is usually unknown. Unfortunately, the literature so far does not provide any algorithm which would work in the case of any shapes of clusters.

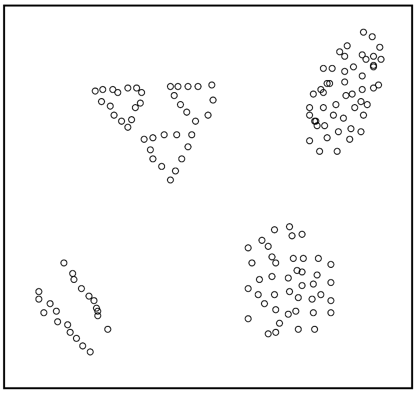

FIGURE 8.1. Various shapes of clusters in two-dimensional space

When choosing the proper clustering algorithm, we should use the knowledge of the problem described by the data set. Generally, data partition should have two features:

- *homogeneity in clusters*, i.e. data within a given cluster should be as similar to each other as possible,
- *heterogeneity between clusters*, i.e. data belonging to different clusters should be as different from each other as possible.

Similarity of data vectors may be defined in different ways, depending on the type of data being clustered. As data most often describe features of objects in a numerical form (as numbers), the most appropriate similarity measure is to measure the distance between objects. We may use e.g. the Euclidean norm which is the most frequently used method of measuring the similarity of objects. Clusters may be represented in different ways. Most frequently the cluster is represented by its central point in the data space. By using various similarity measures we can obtain different shapes of clusters, with the center represented by the central point. Figure 8.1 illustrates examples of various clusters in the two-dimensional space.

In the data clustering tasks we do not have at our disposal the so-called desired output signal, from the teacher. Thus, the process of data clustering may be equated with unsupervised learning. This chapter presents various methods of data partitioning and algorithms for automatic data clustering. Data clustering validity measures are also discussed in this chapter.

8.2 Hard and fuzzy partitions

Data subject to clustering will be represented by n-dimensional vectors $\mathbf{x}_k = [x_{k1}, \ldots, x_{kn}]^T, \mathbf{x}_k \in R^n$, $k = 1, \ldots, M$, which consist of numerical values describing the objects. The set of M vectors creates matrix $\mathbf{X}$ of dimension $n \times M$

$$\mathbf{X} = \begin{bmatrix} x_{11} & x_{21} & \cdots & x_{M1} \\ x_{12} & x_{22} & \cdots & x_{M2} \\ \vdots & \vdots & \vdots & \vdots \\ x_{1n} & x_{2n} & \cdots & x_{Mn} \end{bmatrix}. \tag{8.1}$$

In case of classification, matrix columns (8.1) are objects, and the rows are features (attributes). In the case of medical diagnostics, objects may be identified with patients, while features will be identified with symptoms of a disease or with results of laboratory analysis of those patients.

If clusters are represented by their centers, the objective of clustering algorithms is to obtain c vectors $\mathbf{v}_i = [v_{i1}, \ldots, v_{in}],\ i = 1, \ldots, c$, which are representatives of particular clusters in the data space.

It must be emphasized that from the computational point of view it would be very hard to analyze all possible partitions of M objects into c clusters as their number equals [48]

$$\frac{1}{c!} \sum_{i=1}^{c} \binom{c}{i} (-1)^{(c-i)}\, i^M. \tag{8.2}$$

Example 8.1

Let us consider the problem of partitioning 100 patients ($M = 100$) into 5 different clusters, characterizing particular pathological cases ($c = 5$). It is easy to check that by using formula (8.2), we obtain approximately $6.57 \cdot 10^{67}$ different partitions. Thus it is extremely important to find methods which would perform optimal partition without the necessity to analyze all possible results of clustering.

In the data clustering tasks it is essential to define the type of data partition. The literature distinguishes between hard, fuzzy and possibilistic partitions, where possibilistic partitions are treated as modification of fuzzy partitions.

In the hard data clustering the object entirely belongs or does not belong to a given cluster. The objective of data clustering is data partitioning into c clusters A_i so that

$$\bigcup_{i=1}^{c} A_i = \mathbf{X}, \tag{8.3}$$

$$A_i \cap A_j = \emptyset, \quad 1 \le i \ne j \le c, \tag{8.4}$$

$$\emptyset \subset A_i \subset \mathbf{X}, \quad 1 \le i \le c. \tag{8.5}$$

Assumption (8.3) means that the set of all clusters contains all data vectors, and each object belongs to exactly one cluster. The clusters are disjoint (condition (8.4)), and none of them is empty nor contains the whole data set $\mathbf{X}$ (condition (8.5)). In order to partition data into c clusters, it is comfortable to use the partition matrix $\mathbf{U}$ of the dimension $c \times M$, containing the membership degrees μ_{ik} of the k-th data $\mathbf{x}_k$ to the i-th cluster, $k = 1, ..., M$, $i = 1, ..., c$.

Definition 8.1
Let $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_M\}$ be a finite set. Let c, $2 \leq c < M$, be an integer. *Hard partitioning space* of the set $\mathbf{X}$ is defined in the following way:

$$\begin{aligned} Z_1 = \Bigg\{ \mathbf{U} \in R^{c \times M} \mid \mu_{ik} \in \{0,1\}, \quad \forall i, k; \sum_{i=1}^{c} \mu_{ik} \\ = 1, \quad \forall k; \quad 0 < \sum_{k=1}^{M} \mu_{ik} < M, \quad \forall i \Bigg\}. \end{aligned} \tag{8.6}$$

The partition above assumes that the object belongs to one cluster only and there are no empty clusters or clusters containing all objects.

Example 8.2
Let us consider the data presented in Fig. 8.1. For such data hard partitioning into three clusters ($c = 3$) may be represented by the following matrix $\mathbf{U}$:

$$\mathbf{U} = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix}. \tag{8.7}$$

Let us notice that the object $\mathbf{x}_{10}$ is assigned to cluster 2, although intuitively we would not include it in any of the clusters. However, hard partition makes it necessary for each of the objects to belong to one of the clusters.

The most frequently considered problems do not permit such an unambiguous data partition as in Definition 8.1, as the areas of clusters occurrence may overlap. What is helpful in such a case are algorithms which cause that objects may belong to many clusters with different membership degrees at the same time. It is a natural extension of the hard partition where, like in real problems, a given object may not always be classified unambiguously to one category. For example the boundaries between small, compact and big cars are not strictly defined. There are two types of soft partition: fuzzy and possibilistic. In both partitions the objects may belong to any number of clusters with a membership degree which is a number from the range $[0, 1]$. In the fuzzy partition there is additionally a constraint imposed on membership degrees of a particular object so that the sum of membership degrees of this object to each of c clusters equals 1. This constraint is analogous to the constraint occurring in the probabilistics, therefore this partition is also called *probabilistic partition.*

Definition 8.2
Let $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_M\}$ be a finite set. Let c, $2 \leq c < M$, be an integer. *Fuzzy partition* of the set $\mathbf{X}$ is defined in the following way:

$$Z_2 = \left\{ \mathbf{U} \in R^{c \times M} \mid \mu_{ik} \in [0,1], \quad \forall i,k; \sum_{i=1}^{c} \mu_{ik} \right. \tag{8.8}$$
$$\left. = 1, \quad \forall k; \quad 0 < \sum_{k=1}^{M} \mu_{ik} < M, \quad \forall i \right\}.$$

The partition above assumes that the object may at the same time belong to all clusters with a certain membership degree but the sum of all membership degrees must equal 1. Moreover, there may be no empty clusters or clusters containing all data.

Example 8.3
Let us consider the data presented in Fig. 8.2. For such data the fuzzy partition into three clusters may be represented by the following matrix $\mathbf{U}$:

$$\mathbf{U} = \begin{bmatrix} 0 & 0.06 & 0.02 & 0.98 & 0.98 & 0.99 & 0.01 & 0.01 & 0 & 0.29 \\ 1 & 0.89 & 0.93 & 0.01 & 0.01 & 0.00 & 0.01 & 0.01 & 0 & 0.33 \\ 0 & 0.05 & 0.05 & 0.01 & 0.01 & 0.01 & 0.98 & 0.98 & 1 & 0.38 \end{bmatrix}. \tag{8.9}$$

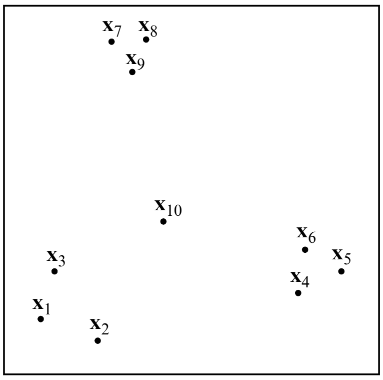

FIGURE 8.2. A dataset in Example 8.3

Let us notice that object $\mathbf{x}_{10}$ is characterized by similar degrees of membership to all three clusters, which corresponds to its almost equal distance from the centers of those clusters. This object may be identified as an outlier (noise). Intuitively we would assign very low membership degrees to noise $\mathbf{x}_{10}$, equal to e.g. 0.1, to all three clusters. However, then the condition that the sum of all membership degrees of a given object must equal 1 would not be met.

Historically in the literature the next partition was a possibilistic partition getting rid of the restriction that the sum of membership degrees is equal one. The only restriction for the object is to belong at least to one cluster. In practice it is not a big inconvenience as the low value of membership degree may be regarded as the lack of membership.

Definition 8.3
Let $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_M\}$ be a finite set. Let c, $2 \leq c < M$, be an integer. *Possibilistic partition* of the set $\mathbf{X}$ is defined in the following way:

$$Z_3 = \left\{ \mathbf{U} \in R^{c \times M} \mid \mu_{ik} \in [0,1],\ \forall i,k;\ \forall k,\ \exists i,\ \mu_{ik} > 0;\ 0 < \sum_{k=1}^{M} \mu_{ik} < M,\ \forall i \right\}. \tag{8.10}$$

Example 8.4
Let us consider the data presented in Fig. 8.2. For such data the possibilistic partition into three clusters may be represented by the following matrix $\mathbf{U}$:

$$\mathbf{U} = \begin{bmatrix} 0.01 & 0.02 & 0.01 & 0.52 & 0.39 & 0.87 & 0.01 & 0.01 & 0.01 & 0.03 \\ 0.87 & 0.44 & 0.79 & 0.04 & 0.03 & 0.03 & 0.05 & 0.04 & 0.05 & 0.12 \\ 0.01 & 0.01 & 0.02 & 0.01 & 0.01 & 0.01 & 0.53 & 0.63 & 0.79 & 0.03 \end{bmatrix} \tag{8.11}$$

Currently the condition that the sum of membership degrees is equal to one does not have to be met. Therefore the noise $\mathbf{x}_{10}$ belongs to all clusters but with a small membership degree.

8.3 Distance measures

An important factor influencing the result of data partition is the method of determining distances between objects. In case of data clustering we measure the distance in the features space in which there are clustered objects and centers (prototypes) of clusters. The most frequently used distance measure is *the Euclidean norm*, interpreted as geometric distance between two points in the space $\mathbf{X}$. Let us consider two points $\mathbf{x}_d = [x_{d1}, ..., x_{dn}]^T$ and $\mathbf{v}_i = [v_{i1}, ..., v_{in}]^T$. *The Euclidean distance* between those points is defined in the following way:

$$D_{id} = \sqrt{\sum_{j=1}^{n} (x_{dj} - v_{ij})^2} = \|\mathbf{x}_d - \mathbf{v}_i\|_2, \tag{8.12}$$

and in the vector notation

$$D_{id} = \left[(\mathbf{v}_i - \mathbf{x}_d)^T (\mathbf{v}_i - \mathbf{x}_d) \right]^{\frac{1}{2}}. \tag{8.13}$$

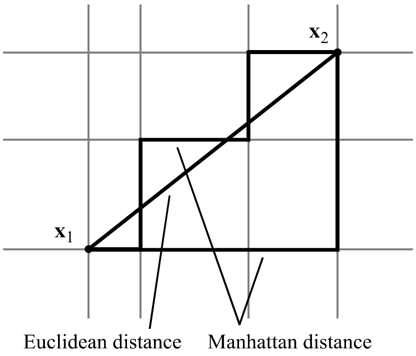

FIGURE 8.3. Illustration of the Euclidean and Manhattan distance

This measure is a generalization of the *Minkowski metric*

$$D_{id} = \left(\sum_{j=1}^{n} \left| x_{dj} - v_{ij} \right|^{r} \right)^{\frac{1}{r}}. \tag{8.14}$$

For different values of the parameter r we may obtain other than the Euclidean norm distance measures. For example for $r = 1$ we obtain *the Manhattan distance* (also called *the city block* measure). The interpretation of this measure may be identified with moving along city streets where we are forced to keep to the network of streets and only 90-degree turns are allowed. Figure 8.3 illustrates the interpretation of the Euclidean norm and Manhattan distance. In case of binary variables the Manhattan distance is called *the Hamming distance*. This measure gives the number of bits by which two bit strings differ. Those strings may represent for example black and white images.

Minkowski measures are susceptible to differences in size (scale) of particular variables. High value variables will dominate low value variables which are for example in a different scale. The method to avoid this problem is variables scaling, which leads to the weighted Euclidean norm

$$D_{id} = \sqrt{\sum_{j=1}^{n} w_j \left(x_{dj} - v_{ij} \right)^2}, \tag{8.15}$$

where w_j is the weight of a given dimension. Assigning the weights to particular variables is useful if we want to obtain the same importance of variables without scaling the data set, or to impose another dimension hierarchy.

If we introduce an additional matrix $\mathbf{A}$ to the Euclidean norm, the clusters may take the shape of ellipses of any orientation. Then we obtain the family of norms induced by a scalar product. In the simplest case the matrix $\mathbf{A}$ is an identity matrix, i.e. $\mathbf{A} = \mathbf{I}$. The measure then becomes the Euclidean distance given by the formula (8.12), and from the geometric point of view the clusters constitute hyperspheres. Figure 8.4 shows how the Euclidean norm operates for $n = 2$. The dotted lines have been used to mark circles characterized by a constant distance between the points lying on those circles from the central point (center). In general the matrix $\mathbf{A}$ is an $n \times n$ diagonal matrix of the form of

$$\mathbf{A} = \begin{bmatrix} c_1 & 0 & \cdots & 0 \\ 0 & c_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & c_n \end{bmatrix}, \tag{8.16}$$

where $c_i > 0$, $i = 1, ..., n$. The clusters generated by the norm with such a matrix are hyperellipses with main diagonals perpendicular to the axis of data space, which is illustrated in Fig. 8.5. The dotted lines have been used to mark ellipses characterized by a constant distance between the points lying on those ellipses from the central point.

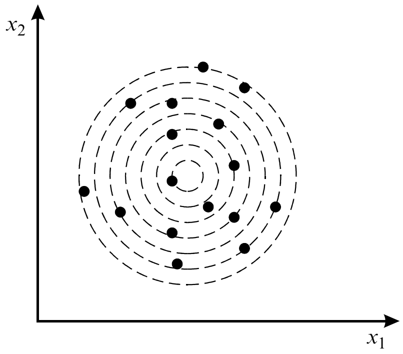

FIGURE 8.4. Euclidean norm

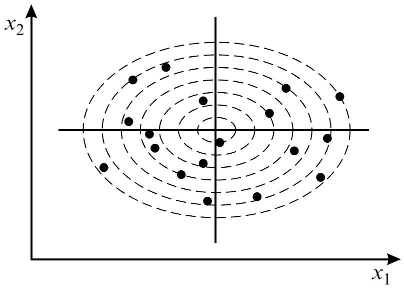

FIGURE 8.5. Diagonal norm

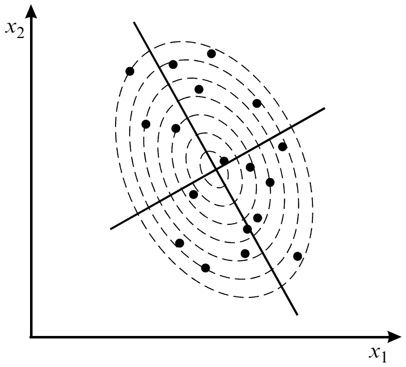

FIGURE 8.6. Mahalanobis norm

Now we will show another method of creation of the matrix **A**. Let us define the covariance matrix of data from the set **X**

$$\mathbf{R} = \frac{1}{M} \sum_{k=1}^{M} (\mathbf{x}_k - \overline{\mathbf{x}}) (\mathbf{x}_k - \overline{\mathbf{x}})^T , \tag{8.17}$$

where $\overline{\mathbf{x}}$ means the average of data $\mathbf{x}_k$, $k = 1, ..., M$. Matrix **A** is defined in the following way:

$$\mathbf{A} = \mathbf{R}^{-1}. \tag{8.18}$$

Matrix **A** created in this way induces *the Mahalanobis norm* in the space R^n, and the clusters are now hyperellipses with any shape and orientation, which is illustrated in Fig. 8.6.

8.4 HCM algorithm

The HCM algorithm (*Hard C-Means*) unambiguously partitions the data contained in the matrix **X** into c clusters. When executing this algorithm, we compute the distance between each vector $\mathbf{x}_k \in R^n$, $k = 1, \ldots, M$ and the cluster center $\mathbf{v}_i$, $i = 1, ..., c$. *The cluster center* is the average of the location of all objects belonging to this cluster. It is convenient to describe the membership in a cluster by means of matrix $\mathbf{U} = [\mu_{ik}] \in Z_1$ (see Definition 8.1). Elements of this matrix are zeros and ones saying that the object $\mathbf{x}_k$ belongs to the i-th cluster. The algorithm is performed in the following stages:

1. Algorithm initialization.
2. Determining the membership of objects on the basis of their distance from the cluster centers.

3. Determining new cluster centers by computing the average of the location of the objects belonging to a given cluster.

4. Checking the algorithm stopping criterion. If the condition is not met, then we proceed to step 2.

The algorithm initialization consists in the choice of the number of clusters c and determining the initial location of their centers. This location may be chosen at random. Alternatively, the initial location of the centers may be identical with c vectors $\mathbf{x}_k$ chosen at random or with first c objects in the data set. A detailed flowchart of the algorithm is illustrated in Fig. 8.7.

The algorithm stopping criterion is the most frequently an appropriately small change of the value of elements of the matrix $\mathbf{U}$, that is $\left\|\mathbf{U}^{(t+1)} - \mathbf{U}^{(t)}\right\| < \varepsilon$, where ε is a fixed constant. Alternatively, we may check the change

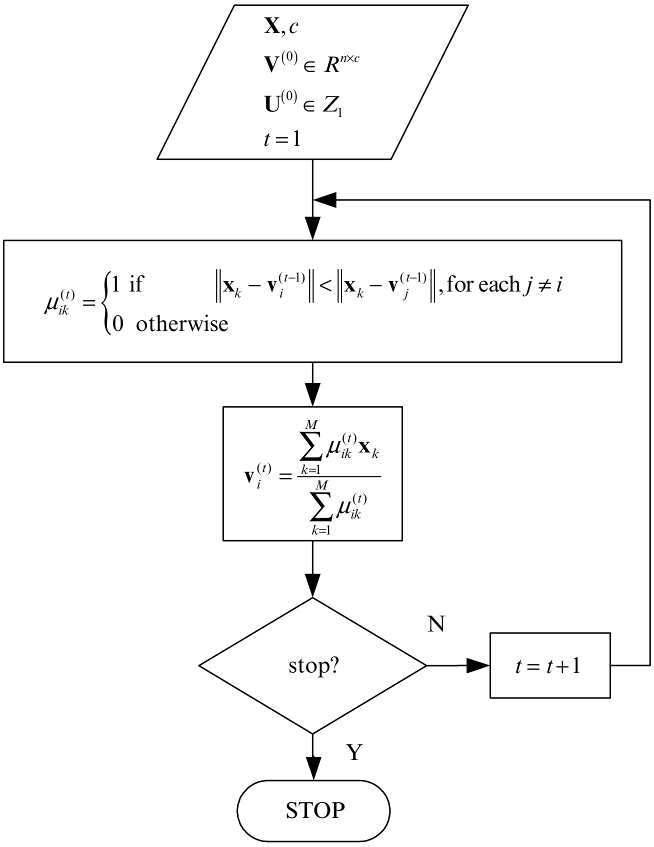
$\mathbf{X}, c$
$\mathbf{V}^{(0)} \in R^{n \times c}$
$\mathbf{U}^{(0)} \in Z_1$
$t = 1$
$\mu_{ik}^{(t)} = \begin{cases} 1 \text{ if } & \left\|\mathbf{x}_k - \mathbf{v}_i^{(t-1)}\right\| < \left\|\mathbf{x}_k - \mathbf{v}_j^{(t-1)}\right\|, \text{for each } j \neq i \\ 0 \text{ otherwise} \end{cases}$
$\mathbf{v}_i^{(t)} = \frac{\sum_{k=1}^{M} \mu_{ik}^{(t)} \mathbf{x}_k}{\sum_{k=1}^{M} \mu_{ik}^{(t)}}$
stop?
N
$t = t + 1$
Y
STOP

FIGURE 8.7. Flowchart of the HCM algorithm

of cluster centers location, i.e. $\left\|\mathbf{V}^{(t+1)}-\mathbf{V}^{(t)}\right\|<\varepsilon$. The HCM algorithm may give various results, depending on the initial location of the cluster centers.

8.5 FCM algorithm

Now let us present the FCM algorithm (*Fuzzy C-Means*) which allows assigning the same objects to various clusters with appropriate membership degrees. The FCM algorithm is the most frequently used algorithm of fuzzy clustering. It detects clusters with prototypes which are points in the data space. All clusters have the same shape dependent on the norm chosen in advance since the algorithm has no possibility to adjust the matrix $\mathbf{A}$ to existing data. This algorithm is derived by minimization of the criterion

$$J\left(\mathbf{X};\mathbf{U},\mathbf{V}\right)=\sum_{i=1}^{c}\sum_{k=1}^{M}\left(\mu_{ik}\right)^{m}\left\|\mathbf{x}_k-\mathbf{v}_i\right\|_{\mathbf{A}}^{2}, \tag{8.19}$$

where

$$\mathbf{U}=\left[\mu_{ik}\right]\in Z_2 \tag{8.20}$$

is the matrix of the set $\mathbf{X}$ partition, whereas

$$\mathbf{V}=\left[\mathbf{v}_1,\mathbf{v}_2,\ldots,\mathbf{v}_c\right] \tag{8.21}$$

is the vector of centers which are to be defined as a result of the algorithm operation, $\mathbf{v}_i\in R^n$, $i=1,...,c$. The following term appearing in formula (8.19)

$$D_{ik\mathbf{A}}^{2}=\left\|\mathbf{x}_k-\mathbf{v}_i\right\|_{\mathbf{A}}^{2}=\left(\mathbf{x}_k-\mathbf{v}_i\right)^{T}\mathbf{A}\left(\mathbf{x}_k-\mathbf{v}_i\right) \tag{8.22}$$

permits to compute the distance between vector $\mathbf{x}_k$ and cluster center $\mathbf{v}_i$, and $m\in(1,\infty)$ is a coefficient indicating the fuzziness degree of formed clusters. When $m\to 1$, the partition becomes less and less fuzzy. When $m\to\infty$, the partition becomes more and more fuzzy (then $\mu_{ik}=1/c$). In practice the value $m=2$ is chosen. In order to execute the algorithm, having a given data set $\mathbf{X}$, we must choose the number of clusters c, fuzziness degree m, parameter ε in the algorithm stopping criterion and initiate at random matrix $\mathbf{U}^{(0)}\in Z_2$ and vector of clusters prototypes $\mathbf{V}^{(0)}$. The algorithm stopping criterion is the same as in case of the HCM algorithm. The FCM algorithm, like HCM, may give various results depending on the initialization. The shape of clusters depends on the adopted distance measure. The flowchart of the FCM algorithm operation is illustrated in Fig. 8.8.

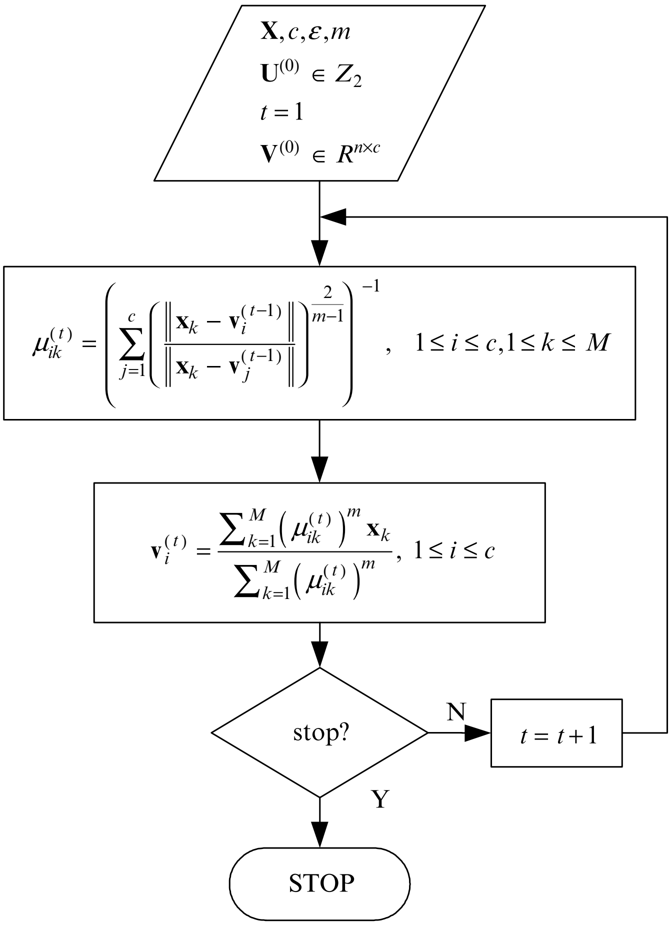
$\mathbf{X}, c, \varepsilon, m$
$\mathbf{U}^{(0)} \in Z_2$
$t = 1$
$\mathbf{V}^{(0)} \in R^{n \times c}$
$\mu_{ik}^{(t)} = \left(\sum_{j=1}^{c} \left(\frac{\left\| \mathbf{x}_k - \mathbf{v}_i^{(t-1)} \right\|}{\left\| \mathbf{x}_k - \mathbf{v}_j^{(t-1)} \right\|} \right)^{\frac{2}{m-1}} \right)^{-1}, \quad 1 \le i \le c, 1 \le k \le M$
$\mathbf{v}_i^{(t)} = \frac{\sum_{k=1}^{M} \left(\mu_{ik}^{(t)} \right)^m \mathbf{x}_k}{\sum_{k=1}^{M} \left(\mu_{ik}^{(t)} \right)^m}, \quad 1 \le i \le c$
stop?
N
$t = t + 1$
Y
STOP

FIGURE 8.8. Flowchart of the FCM algorithm

8.6 PCM algorithm

When deriving the FCM algorithm it is assumed that the sum of the membership degrees of a given object to each of the clusters always equals 1. This restriction may cause undesirable shift of cluster centers in a situation when single incidental objects (noise) occur which sometimes lie far away from the proper clusters. Giving up this constraint, we will get the PCM algorithm (*Possibilistic C-Means*) which may be obtained as a result of minimization of the following objective function:

$$J(\mathbf{X}, \eta; \mathbf{U}, \mathbf{V}) = \sum_{i=1}^{c} \sum_{k=1}^{M} (\mu_{ik})^m \left\| \mathbf{x}_k - \mathbf{v}_i \right\|_{\mathbf{A}}^2 + \sum_{i=1}^{c} \eta_i \sum_{k=1}^{M} (1 - \mu_{ik})^m, \quad (8.23)$$

where η_i is a certain positive constant. The first term of criterion (8.23) is the same as in criterion (8.19) concerning the FCM algorithm. The second term, however, makes it necessary for membership degrees to be as big as possible, without which the solution would be achieved for matrix $\mathbf{U}$ with elements equal 0. Such a solution would result from giving up the assumption saying that the sum of membership degrees of a given object to each of the clusters always equals 1. It is easy to notice that the global objective function (8.23) can be decomposed into c objective functions for particular clusters. As a result of minimization we get

$$\mu_{ik} = \left(1 + \left(\frac{D_{ik\mathbf{A}}}{\eta_i}\right)^{\frac{2}{m-1}}\right)^{-1}, \tag{8.24}$$

where distance $D_{ik\mathbf{A}}$ is given by (8.22). The coefficient η_i defines the so-called *width of resulting possibilistic distribution.* We can choose the same value of the coefficient η_i for all clusters or compute it separately for each of the clusters, proportionally to the average distance of the objects from the center of a given cluster, i.e.

$$\eta_i = \frac{\sum_{k=1}^{M} (\mu_{ik})^m D_{ik\mathbf{A}}^2}{\sum_{k=1}^{M} (\mu_{ik})^m}. \tag{8.25}$$

The algorithm stopping criterion is chosen in the same way as in case of the HCM algorithm. We must note the fact that improper initialization of the PCM algorithm may lead to partitioning in which all membership degrees are equal. Therefore the initial partitioning in the PCM algorithm usually takes place with use of the FCM algorithm. The flowchart of the PCM algorithm is illustrated in Fig. 8.9.

8.7 Gustafson-Kessel algorithm

In the algorithms presented so far the type of norm must be defined in advance. Therefore we must know what cluster shapes occur in the data. The main disadvantage of the algorithms with a constant norm is searching for clusters with the shape which may not occur in the data set. The Gustafson-Kessel algorithm (GK) is a modification of the FCM algorithm. In this algorithm each cluster is associated with a separate matrix $\mathbf{A}_i$, and the distance between object $\mathbf{x}_k$ and the cluster center $\mathbf{v}_i$ equals

$$D_{ik}^2 = (\mathbf{x}_k - \mathbf{v}_i)^T \mathbf{A}_i (\mathbf{x}_k - \mathbf{v}_i). \tag{8.26}$$

During the algorithm operation also matrices $\mathbf{A}_i$, inducing the distance measure, $i = 1, ..., c$, are modified. The objective function in the GK algorithm is defined in the same way as in the FCM algorithm (8.19),

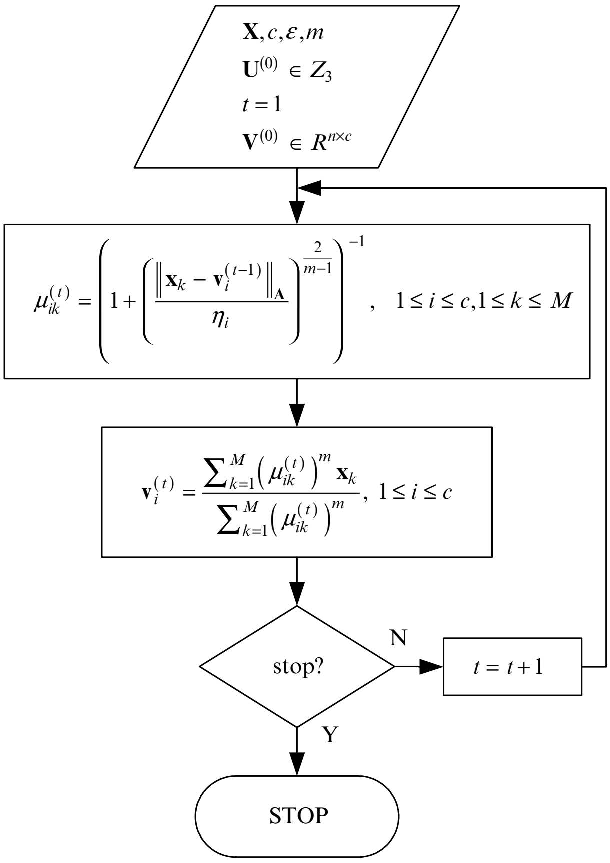
$\mathbf{X}, c, \varepsilon, m$
$\mathbf{U}^{(0)} \in Z_3$
$t = 1$
$\mathbf{V}^{(0)} \in R^{n \times c}$
$\mu_{ik}^{(t)} = \left(1 + \left(\frac{\left\| \mathbf{x}_k - \mathbf{v}_i^{(t-1)} \right\|_{\mathbf{A}}}{\eta_i}\right)^{\frac{2}{m-1}}\right)^{-1}, \quad 1 \le i \le c, 1 \le k \le M$
$\mathbf{v}_i^{(t)} = \frac{\sum_{k=1}^{M} \left(\mu_{ik}^{(t)}\right)^m \mathbf{x}_k}{\sum_{k=1}^{M} \left(\mu_{ik}^{(t)}\right)^m}, \quad 1 \le i \le c$
stop?
N
$t = t + 1$
Y
STOP

FIGURE 8.9. Flowchart of the PCM algorithm

but now distance measure (8.26) is used. Thus the objective function takes the form of

$$J(\mathbf{X}; \mathbf{U}, \mathbf{V}, \mathbf{A}) = \sum_{i=1}^{c} \sum_{k=1}^{M} (\mu_{ik})^m D_{ik}^2, \tag{8.27}$$

where $\mathbf{A} = (\mathbf{A}_1, \mathbf{A}_2, \ldots, \mathbf{A}_c)$. Let us notice that the direct minimization of criterion (8.27) does not lead to an effective solution as the value of this criterion may have any small value, e.g. for matrix $\mathbf{A}_i$ with almost exclusively zero-value elements. In order to obtain a correct result, matrices $\mathbf{A}_i$ must be constrained, e.g. by setting the values of their determinants, i.e.

$$\det(\mathbf{A}_i) = \rho_i, \quad \rho_i > 0, \quad \forall i, \quad i = 1, \ldots, c, \tag{8.28}$$

where ρ_i is a chosen constant reflecting the information of data subject to clustering. In case of lack of such information it is assumed that $\rho_i = 1$ for

$i = 1, ..., c$. The constraint (8.28) causes that the volumes of clusters are constant, and we only permit a change of the shape of clusters. As a result of the minimization of criterion (8.27) with respect to matrix $\mathbf{A}_i$ we get

$$\mathbf{A}_i = \left[\rho_i \det\left(\mathbf{F}_i\right)\right]^{\frac{1}{n}} \mathbf{F}_i^{-1}, \tag{8.29}$$

where $\mathbf{F}_i$ is the so-called *fuzzy covariance matrix* of the i-th cluster

$$\mathbf{F}_i = \frac{\sum_{k=1}^{M} \left(\mu_{ik}\right)^m \left(\mathbf{x}_k - \mathbf{v}_i\right) \left(\mathbf{x}_k - \mathbf{v}_i\right)^T}{\sum_{k=1}^{M} \left(\mu_{ik}\right)^m}. \tag{8.30}$$

The algorithm initialization requires determining the same parameters as in the FCM algorithm, and additionally coefficients ρ_i defining the volumes of particular clusters (if we do not have the knowledge on the problem, we may assume that $\rho_i = 1$). The GK algorithm finds clusters of any shapes but requires more computations than the FCM algorithm due to the necessity to compute the determinant and inverse of the matrix $\mathbf{F}_i$. The flowchart of the GK algorithm is illustrated in Fig. 8.10.

8.8 FMLE algorithm

In the FMLE clustering algorithm (*Fuzzy Maximum Likelihood Estimates*) the distance measure refers to the form of maximum likelihood estimates. This measure is given by the following formula:

$$D_{ik\mathbf{G}_i} = \frac{\left[\det\left(\mathbf{G}_i\right)\right]^{\frac{1}{2}}}{P_i} \exp\left[\frac{1}{2} \left(\mathbf{x}_k - \mathbf{v}_i\right)^T \mathbf{G}_i^{-1} \left(\mathbf{x}_k - \mathbf{v}_i\right)\right], \tag{8.31}$$

where $\mathbf{G}_i$ is the covariance matrix of the i-th cluster

$$\mathbf{G}_i = \frac{\sum_{k=1}^{M} \mu_{ik} \left(\mathbf{x}_k - \mathbf{v}_i\right) \left(\mathbf{x}_k - \mathbf{v}_i\right)^T}{\sum_{k=1}^{M} \mu_{ik}}, \tag{8.32}$$

and P_i is the *a priori* probability of choosing the i-th cluster

$$P_i = \frac{1}{M} \sum_{k=1}^{M} \mu_{ik}. \tag{8.33}$$

Membership degree μ_{ik} *may be interpreted as the probability* of assigning object $\mathbf{x}_k$ to the i-th cluster. Convergence of the FMLE algorithm strongly depends on the initialization as it often gets stuck in the local minimum. Contrary to the GK algorithm, realization of the FMLE algorithm does not require the knowledge or arbitrary assumption of the value of parameter ρ_i for $i = 1, ..., c$. The algorithm flowchart is illustrated in Fig. 8.11.

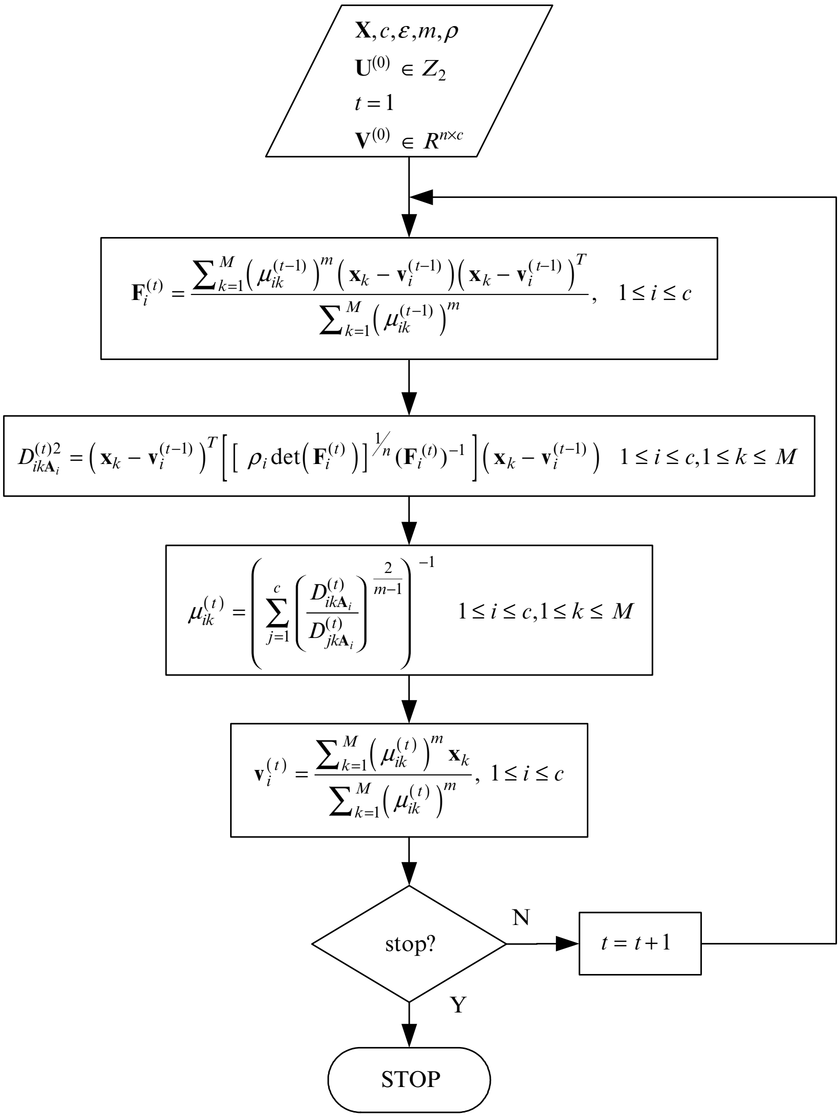
$\mathbf{X}, c, \varepsilon, m, \rho$
$\mathbf{U}^{(0)} \in Z_2$
$t = 1$
$\mathbf{V}^{(0)} \in R^{n \times c}$
$\mathbf{F}_i^{(t)} = \frac{\sum_{k=1}^{M} \left(\mu_{ik}^{(t-1)}\right)^m \left(\mathbf{x}_k - \mathbf{v}_i^{(t-1)}\right)\left(\mathbf{x}_k - \mathbf{v}_i^{(t-1)}\right)^T}{\sum_{k=1}^{M} \left(\mu_{ik}^{(t-1)}\right)^m}, \quad 1 \le i \le c$
$D_{ik\mathbf{A}_i}^{(t)2} = \left(\mathbf{x}_k - \mathbf{v}_i^{(t-1)}\right)^T \left[\left[\rho_i \det\left(\mathbf{F}_i^{(t)}\right)\right]^{1/n} (\mathbf{F}_i^{(t)})^{-1} \right] \left(\mathbf{x}_k - \mathbf{v}_i^{(t-1)}\right) \quad 1 \le i \le c, 1 \le k \le M$
$\mu_{ik}^{(t)} = \left(\sum_{j=1}^{c} \left(\frac{D_{ik\mathbf{A}_i}^{(t)}}{D_{jk\mathbf{A}_i}^{(t)}} \right)^{\frac{2}{m-1}} \right)^{-1} \quad 1 \le i \le c, 1 \le k \le M$
$\mathbf{v}_i^{(t)} = \frac{\sum_{k=1}^{M} \left(\mu_{ik}^{(t)}\right)^m \mathbf{x}_k}{\sum_{k=1}^{M} \left(\mu_{ik}^{(t)}\right)^m}, \quad 1 \le i \le c$
stop?
N
$t = t + 1$
Y
STOP

FIGURE 8.10. Flowchart of the Gustafson-Kessel algorithm

8.9 Clustering validity measures

The number of clusters is an important factor influencing clustering validity. It should reflect the actual number of clusters of objects similar to each other in the set **X**. The proper number of clusters may be found by clustering the data set for a different number of clusters and different values

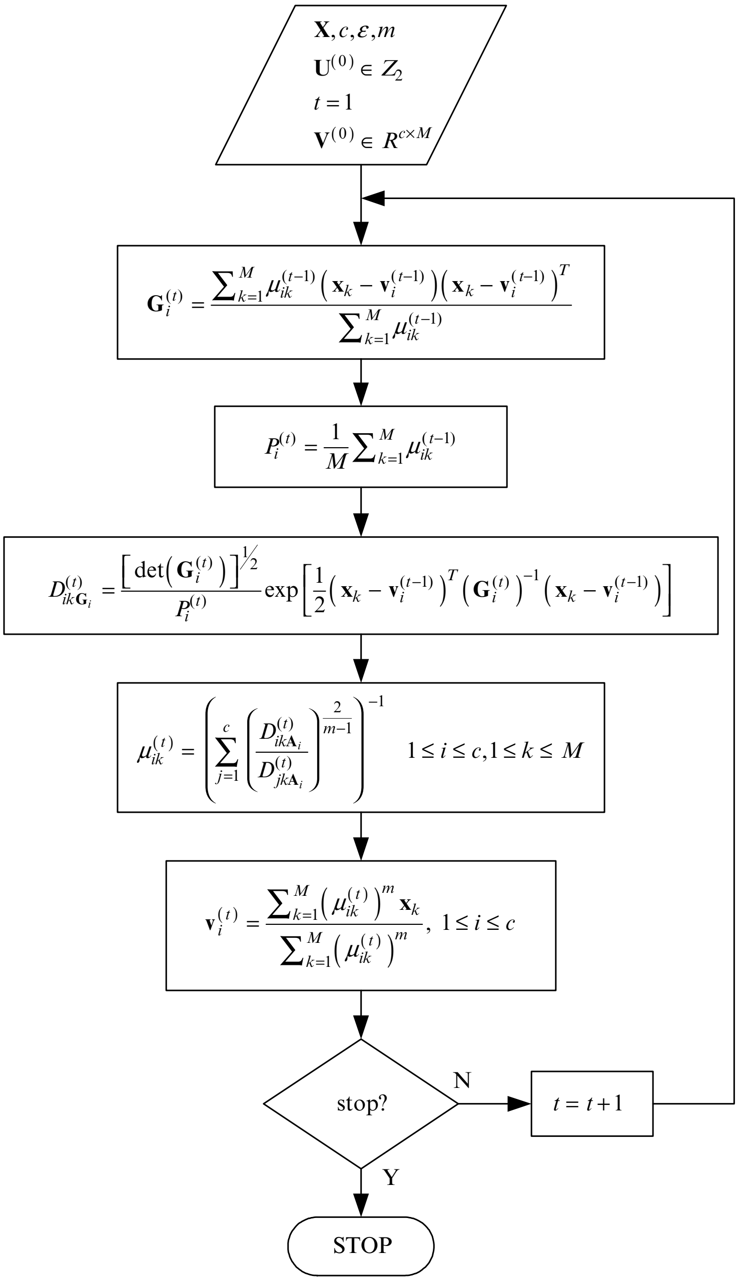
$\mathbf{X}, c, \varepsilon, m$
$\mathbf{U}^{(0)} \in Z_2$
$t = 1$
$\mathbf{V}^{(0)} \in R^{c \times M}$
$\mathbf{G}_i^{(t)} = \frac{\sum_{k=1}^{M} \mu_{ik}^{(t-1)} \left(\mathbf{x}_k - \mathbf{v}_i^{(t-1)}\right)\left(\mathbf{x}_k - \mathbf{v}_i^{(t-1)}\right)^T}{\sum_{k=1}^{M} \mu_{ik}^{(t-1)}}$
$P_i^{(t)} = \frac{1}{M} \sum_{k=1}^{M} \mu_{ik}^{(t-1)}$
$D_{ik\mathbf{G}_i}^{(t)} = \frac{\left[\det\left(\mathbf{G}_i^{(t)}\right)\right]^{1/2}}{P_i^{(t)}} \exp\left[\frac{1}{2}\left(\mathbf{x}_k - \mathbf{v}_i^{(t-1)}\right)^T \left(\mathbf{G}_i^{(t)}\right)^{-1} \left(\mathbf{x}_k - \mathbf{v}_i^{(t-1)}\right)\right]$
$\mu_{ik}^{(t)} = \left(\sum_{j=1}^{c} \left(\frac{D_{ik\mathbf{A}_i}^{(t)}}{D_{jk\mathbf{A}_i}^{(t)}}\right)^{\frac{2}{m-1}}\right)^{-1} \quad 1 \le i \le c, 1 \le k \le M$
$\mathbf{v}_i^{(t)} = \frac{\sum_{k=1}^{M} \left(\mu_{ik}^{(t)}\right)^m \mathbf{x}_k}{\sum_{k=1}^{M} \left(\mu_{ik}^{(t)}\right)^m}, \ 1 \le i \le c$
stop?
N
$t = t + 1$
Y
STOP

FIGURE 8.11. Flowchart of the FMLE algorithm

of parameters (e.g. parameter m in the FCM algorithm). Each time we must also evaluate the partition obtained. Such evaluation is performed by means of special indexes called *clustering validity indexes*. Below is the list of several best-known clustering validity indexes.

a) Fuzziness in partition matrix $\mathbf{U}$

It is the simplest index measuring the fuzziness degree of a partition matrix

$$V_1(\mathbf{U}) = \frac{1}{M}\sum_{i=1}^{c}\sum_{k=1}^{M}(\mu_{ik})^2 . \tag{8.34}$$

The best partition is a partition where index $V_1(\mathbf{U})$ reaches the maximum value, that is

$$\max_c\left\{\max_{Z_2} V_1(\mathbf{U})\right\}, \quad c = 2, \ldots, M-1. \tag{8.35}$$

Coefficient $V_1(\mathbf{U})$ evaluates the distance of all objects to the cluster centers. If each data is strongly connected with one cluster only, i.e. if for each k membership degree μ_{ik} is big for only one cluster i, the uncertainty of data is low, and consequently $V_1(\mathbf{U})$ takes a high value. It is easy to notice that the value of index (8.34) depends on the distance of particular objects $\mathbf{x}_k$ from the centers of created clusters. Index (8.34) is connected with the index defining the entropy of data partition

$$V_2(\mathbf{U}) = -\frac{1}{M}\sum_{i=1}^{c}\sum_{k=1}^{M}\mu_{ik}\ln(\mu_{ik}) . \tag{8.36}$$

The best partition is a partition which minimizes index (8.36) that is

$$\min_c\left\{\min_{Z_2} V_2(\mathbf{U})\right\} \quad c = 2, \ldots, M-1. \tag{8.37}$$

When all degrees have values close to $1/c$, which means a high degree of clusters fuzziness, then measure $V_2(\mathbf{U})$ takes high values, which means that the result of clustering is unsatisfactory. By analogy, if all membership degrees μ_{ik} take values close to 0 or 1, then measure $V_2(\mathbf{U})$ takes low values, which indicates a good result of clustering.

b) Fukuyama-Sugeno index

The inconvenience of the above indexes is dependence of their values on the number of clusters c and lack of connection between those values and geometric shape of clusters.

The Fukuyama-Sugeno index enables connection of partition with geometric properties of clustered data. It is given by the following formula:

$$V_3(\mathbf{U}, \mathbf{V}; \mathbf{X}) = \sum_{i=1}^{c}\sum_{k=1}^{M}(\mu_{ik})^m\left(\|\mathbf{x}_k - \mathbf{v}_i\|_{\mathbf{A}}^2 - \|\mathbf{x}_k - \overline{\mathbf{v}}\|_{\mathbf{A}}^2\right), \tag{8.38}$$

where $\overline{\mathbf{v}}$ is an average of all points in the data set, i.e.

$$\overline{\mathbf{v}} = \frac{1}{M}\sum_{k=1}^{M} x_k. \tag{8.39}$$

The optimal data partition minimizes index V_3.

c) Xie-Beni index

Xie-Beni index is given by the formula

$$V_4\left(\mathbf{U},\mathbf{V};\mathbf{X}\right) = \frac{\sum_{i=1}^{c}\sum_{k=1}^{M}\left(\mu_{ik}\right)^m \left\|\mathbf{x}_k - \mathbf{v}_i\right\|^2}{M\left(\min_{i,j}\left\{\left\|v_i - v_j\right\|\right\}^2\right)}, \tag{8.40}$$

and the optimal selection of the number of classes is given by the formula

$$\min_{c}\left\{\min_{M_2} V_4\left(\mathbf{U}\right)\right\} \quad c = 2,\ldots,M-1. \tag{8.41}$$

The best partition minimizes index (8.40) which is a quotient of the average of all distances between clusters and objects and the smallest distance between clusters. The proper clustering procedure should result in a situation in which all data will be as close to the centers of the respective clusters as possible and all centers will be as far from each other as possible.

8.10 Illustration of operation of data clustering algorithms

The most frequently used algorithm of data clustering is the FCM algorithm. Therefore in this chapter we will perform a simulation of this algorithm and compare it with the HCM and PCM algorithms.

Example 8.5
Figure 8.12 presents an exemplary data set composed of 9 two-dimensional objects, i.e. $M = 9$ and $n = 2$. Matrix $\mathbf{X}$ corresponding to this set is in the form of

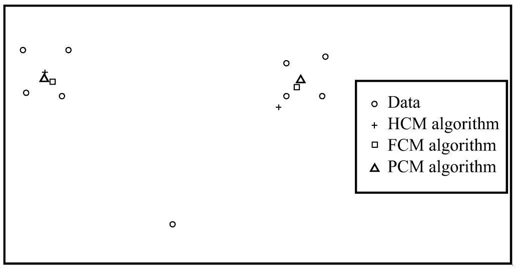

FIGURE 8.12. Comparison of three clustering algorithms

$$\mathbf{X} = \begin{bmatrix} 98 & 97 & 111 & 109 & 178 & 178 & 190 & 189 & 143 \\ 86 & 99 & 99 & 85 & 85 & 95 & 97 & 85 & 46 \end{bmatrix}. \tag{8.42}$$

It is easy to notice two separate groups of objects and one object number 9 which "does not fit" in those groups. Symbol "+" has been used to mark the cluster centers obtained by means of the HCM algorithm. As one can see, object 9 has been qualified to cluster 2 and influenced the position of this cluster's center, by "drawing" this center towards itself. As a result of the HCM algorithm operation, the following partition matrix has been created:

$$\mathbf{U}_{\mathrm{HCM}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{8.43}$$

The FCM algorithm assigned object number 9 to both groups with the same membership degree equal to 0.5. In such a case the centers of both clusters are moved towards object number 9. As a result of clustering using the FCM algorithm, the following partition matrix has been created:

$$\mathbf{U}_{\mathrm{FCM}} = \begin{bmatrix} 0.99 & 0.98 & 0.98 & 0.99 & 0.00 & 0.01 & 0.02 & 0.01 & 0.50 \\ 0.01 & 0.02 & 0.02 & 0.01 & 1.00 & 0.99 & 0.98 & 0.99 & 0.50 \end{bmatrix}. \tag{8.44}$$

The problem of noise has been best dealt with by the PCM algorithm which assigned membership degrees equal to 0.04 and 0.03 to the object number 9, which can be seen when analyzing the partition matrix created

$$\mathbf{U}_{\mathrm{PCM}} = \begin{bmatrix} 0.76 & 0.56 & 0.53 & 0.73 & 0.03 & 0.03 & 0.02 & 0.02 & 0.04 \\ 0.02 & 0.02 & 0.02 & 0.02 & 0.74 & 0.73 & 0.54 & 0.65 & 0.03 \end{bmatrix}. \tag{8.45}$$

The centers of both clusters have only slightly been moved comparing to the results of the HCM and PCM algorithms. Table 8.1 shows the coordinates of cluster centers which were created as a result of clustering set (8.42).

TABLE 8.1. Coordinates of cluster centers which were created as a result of clustering the set (8.42)

HCM algorithm	x_1	x_2
Cluster 1	175.60	81.60
Cluster 2	103.75	92.25
FCM algorithm	x_1	x_2
Cluster 1	106.13	89.38
Cluster 2	181.20	87.73
PCM algorithm	x_1	x_2
Cluster 1	103.58	90.19
Cluster 2	182.41	89.86

8.11 Notes

This chapter presents only basic methods of data clustering. In order to illustrate their operation we compared the HCM, FCM and PCM algorithms. The most frequently used FCM method is sensitive to occurrence of noisy data. This method may serve as an initialization of the PCM algorithm, which is resistant to noise and outliers. It is also applied for preliminary setup of membership functions during design of neural and fuzzy systems (Chapters 9 and 10). Data clustering methods constitute an extremely important research tool in computational intelligence and have numerous applications. Both, basic algorithms presented in this chapter as well as more advanced methods, e.g. oriented at detection of clusters of specific shapes, have been discussed in detail in the literature [3, 9, 23, 34, 83]. It is worth noting that pioneers in the field of data clustering methods are James C. Bezdek and Enrique H. Ruspini whose original works have been reprinted in the book [10].