Classification Rules in Methods of Clustering

Sadaaki Miyamoto, *Member, IEEE,*

***Abstract*—While classification rules are essential in supervised classification methods, they are not noticed well in methods of clustering. Nevertheless, some clustering techniques have clear rules of classification, while they are not obvious in other methods. This paper discusses classification rules or classification functions in the former class including K-means, fuzzy c-means, and the mixture of distributions, and shows theoretical properties that exhibit the nature of a method in this class. In contrast, linkage methods of agglomerative hierarchical clustering do not appear to have classification rules. We show, however, the single linkage method has the rule of nearest neighbor classification, while other linkage methods not. An advanced method using positive-definite kernels is also discussed.**

***Index Terms*—Agglomerative hierarchical clustering, K-means, fuzzy K-means, mixture of distributions, classification rules, inductive property.**

I. Introduction

DATA clustering, or simply clustering, is becoming one of major tools for analyzing large scale data in this world of the 'big data'. Many years ago, clustering techniques have supplementary roles to supervised classification. Due to the increase of necessities to survey and examine huge and unorganized data collections, we are confronting with more unsupervised cases, and thus unsupervised classification is being noted to be important.

Although there are various methods of unsupervised classification, we discuss solely clustering which has a long history in this class of methods. At least its age is more than 60 years, and on the other hand new methods are developed and various applications are actively studied.

Most papers on methods of clustering have a simple structure:

1) Propose a new algorithm.
2) Apply it to a number of examples and compare results with those by typical existing methods.
3) Show that the proposed method is superior to the compared old methods.

Many studies have been like this, but a fundamental question is: is this way of discussion really useful?

Such discussions may expand methods of clustering, but do not serve deeper understanding of methods of clustering, for which theoretical studies are needed.

Theoretical considerations are minor in foregoing literature but important studies have been done: a typical example is K-means++ [2] where the efficiency of the K-means [9] is improved and theoretical properties on the efficiency of the algorithm is discussed.

S. Miyamoto is with the Department of Risk Engineering, University of Tsukuba, Ibaraki 305-8573 Japan email: (see http://www.risk.tsukuba.ac.jp/miyamoto/index.html).

In this paper we do not consider the efficiency of algorithms, but we study theoretical properties of well-known classes of methods.

What we focus upon is classification rules in some methods of clustering. A classification rule obviously exists in a method of supervised classification, whereas it is ambiguous or unclear in clustering, since clustering implies generation of classes on a set of given objects and nothing more, and thus to have a classification rule does not seem to be a matter of interest. Classification rules are, however, essential to understand theoretical properties of methods of clustering, which we will show in this paper.

We consider two well-known classes of methods for this purpose: first class is the K-means and related methods; second class is the agglomerative hierarchical clustering including different linkage methods.

Some methods in these classes have clearly defined classification rules, while others not. Note also that classification rules may include fuzzy rules or probabilistic rules.

Most discussions in this paper is methodological and examples are simple and for the purpose of illustration.

The rest of this paper is organized as follows. Chapter 2 discusses the K-means and related methods. Not only fuzzy K-means [3], [5] but also the model of mixture of distributions [10] are considered to be related methods to the K-means. Chapter 3 studies agglomerative hierarchical clustering where the single linkage and other linkage methods [6] are contrasted. Chapter 4 finally concludes the paper.

To save space, we omit the proofs of the propositions; they are not difficult and readers may refer to the literature, e.g., [13].

II. K-Means and Related Methods

We begin with giving notations. $X = \{x_1, x_2, \ldots, x_N\}$ is the set of objects for clustering, in which x_k $(k = 1, 2, \ldots, N)$ is a point in $\boldsymbol{R}^p$, $x_k = (x_k,^1, \ldots, x_k^p)^\top \in \boldsymbol{R}^p$. $\boldsymbol{R}^p$ is the p-dimensional Euclidean space with the Euclidean norm $\|x\| = \sqrt{x^\top x}$.

Clusters of X denoted by $G_1, \ldots, G_K$ are subsets of X that form a partition of X:

$$\bigcup_{i=1}^{K} G_i = X, \quad G_i \cap G_j = \emptyset \ \ (i \neq j). \tag{1}$$

However, this property holds only for hard clusters. When we consider fuzzy clusters and probabilistic clusters, the above property should be weakened.

A. The Basic K-Means

The name of K-means comes from the well-known paper of MacQueen [9], but the basic algorithm of the K-means

mentioned in the literature is simpler than the one described in [9]. Actually the name of K-means indicates a class of related algorithms instead of a single algorithm.

We first describe a prototypical procedure for K-means:

A Prototype Procedure for K-Means

1) Generate initial clusters randomly.
2) Determine a prototype vector for each cluster.
3) Allocate each object to the nearest prototype
4) If clusters are convergent, stop. Else go to Step 2).

The above procedure is not an algorithm in a strict sense, since how prototypes are determined is not described.

The reason why we describe this prototype is that different algorithms are expressed as variations of this prototypical procedure; hence they are regarded as members of a family related to K-means prototype. Note also that the number of clusters K should be decided beforehand.

1) The hard K-means: The K-means, which is also called hard K-means, uses centroids, in other words, centers of gravity as the prototypes:

$$v_i = \frac{1}{|G_i|} \sum_{x_k \in G_i} x_k \tag{2}$$

where $|G_i|$ is the number of elements in G_i.

Hence the algorithm **KM** of the K-means becomes as follows:

KM1: Generate initial clusters $G_1, \ldots, G_K$ randomly.

KM2: Calculate cluster prototypes v_i $(1 = 1, \ldots, K)$ by (2).

KM3: Allocate every object $x_k \in X$ to the cluster of the nearest prototype:

$$x_k \to G_i \iff i = \arg \min_{1 \le j \le K} \|x_k - v_i\|^2 \tag{3}$$

KM4: If clusters are convergent, stop. Else go to Step **KM2**.

Another way to determine cluster prototype is in Kohonen's SOM [8]: the VQ (vector quantization) algorithm can be used for clustering, where a learning scheme

$$v_i^{(t+1)} = v_i^{(t)} + \alpha(t)(x_k - v_i^{(t)}) \tag{4}$$

is used for cluster prototypes. Note that t is the number of iterations and $\alpha(t)$ is the learning parameter; x_k in this equation is the last element allocated to cluster G_i.

2) Fuzzy K-means: Fuzzy K-means [5], [3] is a variation of the K-means, where cluster prototypes are fuzzy centroids v_i. Instead of the nearest allocation, fuzzy nearest allocation using membership u_{ki} is used:

$$u_{ki} = \left[\sum_{j=1}^{K} \left(\frac{\|x_k - v_i\|^2}{\|x_k - v_j\|^2} \right)^{\frac{1}{m-1}} \right]^{-1} \tag{5}$$

$$v_i = \frac{\sum_k (u_{ki})^m x_k}{\sum_k (u_{ki})^m} \tag{6}$$

where $m > 1$ is a fuzzifying parameter. It has been shown that as $m \to 1$, the solutions approach to those of the K-means [13]. The algorithm of fuzzy K-means repeats (5) and (6) until convergence, and hence fuzzy K-means can be regarded as a variation of the K-means.

3) Mixture of Distributions: Usually the model of the mixture of distributions is different from the K-means and fuzzy K-means, but this model can be related to those in a sense. Let us take the most typical case of the Gaussian mixture.

Let us also suppose that covariances for clusters are known and given by

$$\sigma^2 I = \frac{1}{2\lambda} I, \tag{7}$$

where λ is a given positive parameter and I is the identity matrix. This assumption appears a bit strange but is convenient for our purpose.

Then the parameter estimation is only for the averages of the Gaussian distribution. Let v_i is the mean vector and let $u_{ki} = P(G_i|x_k)$. Using the EM algorithm [10], we have

$$u_{ki} = \frac{\exp(-\lambda \|x_k - v_i\|^2)}{\sum_{j=1}^{K} \exp(-\lambda \|x_k - v_j\|^2)}, \tag{8}$$

$$v_i = \frac{\sum_k u_{ki} x_k}{\sum_k u_{ki}}. \tag{9}$$

The algorithm repeats (8) and (9) until convergence. Note that these equations are similar to those for fuzzy K-means. Thus the Gaussian mixture in this restricted form is regarded as a variation of K-means.

4) Fuzzy K-means and Gaussian mixture: We can observe relations between the Gaussian mixture and fuzzy K-means in more detail. For this purpose we review the formulation of fuzzy K-means, which is an alternate optimization of the following objective function:

$$J(U, V) = \sum_{i=1}^{K} \sum_{k=1}^{N} (u_{ki})^m \|x_k - v_i\|^2, \quad (m > 1), \tag{10}$$

with simplified notations of membership matrix $U = (u_{ki})$ and matrix V collecting the prototypes: $V = (v_1, \ldots, v_K)$. A constraint is imposed upon U:

$$M = \{\, U = (u_{ki}) : \sum_{j=1}^{c} u_{kj} = 1, \forall j;\ u_{kj} \ge 0, \forall k, j \,\}. \tag{11}$$

The alternate optimization means that, with a give random initial value of U and/or V, we optimize $J(U, V)$ with respect to U with the previously determined V, and then we optimize $J(U, V)$ with respect to V with the previously determined U, until convergence. As a result the solutions (5) and (6) are obtained and thus we repeat (5) and (6).

We introduce here another objective function:

$$J_E(U, V) = \sum_{i=1}^{K} \sum_{k=1}^{N} \left\{ u_{ki} \|x_k - v_i\|^2 + \lambda^{-1} u_{ki} \log u_{ki} \right\}, \tag{12}$$

where $\lambda > 0$. This function has been considered as a variation of fuzzy K-means by a number of researchers [12], [13]. By

the alternate optimization described above using $J_E(U,V)$ instead of $J(U,V)$, and with the same constraint (11), we have the solutions (8) and (9). Thus the restricted form of the Gaussian mixture is equivalent to fuzzy K-means using $J_E(U,V)$, which is sometimes called entropy-based fuzzy K-means.

General Gaussian mixture model has the covariance matrix in addition to the mean vector. For this general case we still have similar relations in which generalized entropy-based fuzzy K-means express a generalization of the solutions derived from the EM algorithm [7], of which we omit the details for simplicity. See, e.g., [7] or [13].

B. Classification Rules in K-Means and Related Methods

We consider the method of hard K-means again and study classification rule associated with it.

1) Voronoi regions as classification rule: Let us introduce characteristic function $u_i(x)$ for cluster G_i: For each $x_k \in X$,

$$u_i(x_k) = 1, \quad x_k \in G_i, \tag{13}$$
$$u_i(x_k) = 0, \quad x_k \notin G_i. \tag{14}$$

Thus $u_i \colon X \to \{0,1\}$. For such a function defined on a discrete set X, it is difficult to observe a mathematical property. However, extending u_i to the whole space $\boldsymbol{R}^p$ is straightforward, as we will see below.

A key for this extension is the Voronoi region (see Fig. 1) which is actually referred to in vector quantization [8]. Thus the K-means is understood as the algorithm to generate Voronoi regions with centers of the cluster prototypes.

Let us denote the Voronoi regions be $W_i(V)$ $(i = 1, \ldots, K)$ with centers $V = (v_1 \ldots, v_K)$:

$$W_i(V) = \{x \in \boldsymbol{R}^p \colon \|x - v_i\| \le \|x - v_j\|, \ \forall j, \ j \ne i \ \}. \tag{15}$$

Assume that the K-means algorithm is repeated and the converged cluster prototypes are $\bar{V}$. Also suppose that the obtained clusters are $G_1, \ldots, G_K$. We then have

$$G_i = W_i(\bar{V}) \cap X, \quad i = 1, \ldots, K. \tag{16}$$

Thus the extended function $u_i \colon \boldsymbol{R}^p \to \{0,1\}$ is:

$$u_i(x) = 1, \quad x \in G_i, \tag{17}$$
$$u_i(x) = 0, \quad x \notin G_i. \tag{18}$$

They are respectively derived from (13) and (14) by replacing object symbol x_k with variable symbol x.

2) Mixture of distributions and fuzzy K-means: Let us move to the discussion of the mixture of distributions. The basic model of the mixture of distributions is as follows:

$$P(G_i|x) = \frac{p(x|G_i)P(G_i)}{\sum_{j=1}^{K} p(x|G_j)P(G_j)} \tag{19}$$

where $p(x|G_i)$ is the probability density with the condition of class G_i; $P(G_i)$ is the prior probability of class G_i. As the result $P(G_i|x)$, the posterior probability that x belongs to class G_i, is calculated. This equation is common between supervised and unsupervised classifications. Thus $P(G_i|x)$ is actually the probabilistic allocation rule, whereby the probabilistic membership of x_k to G_i is obtained by substituting $x = x_k$.

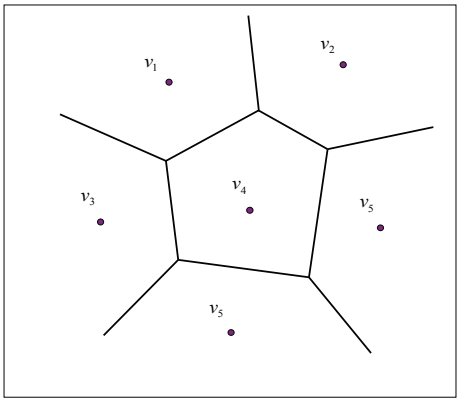

Fig. 1. A simple example of Voronoi regions with six centers on a plane.

Let us turn to fuzzy K-means and consider what we have in relation to the mixture of distributions. As in the case of the K-means, let us replace object symbol x_k by variable x in (5). We then have $u_{ki} \to U_i(x)$:

$$U_i(x) = \left[\sum_{j=1}^{K} \left(\frac{\|x - v_i\|^2}{\|x - v_j\|^2} \right)^{\frac{1}{m-1}} \right]^{-1}. \tag{20}$$

The entropy-based method has another fuzzy rule

$$U_i^E(x) = \frac{\exp(-\lambda \|x - v_i\|^2)}{\sum_{j=1}^{K} \exp(-\lambda \|x - v_j\|^2)}, \tag{21}$$

by replacing x_k in (8) by x. This equation can also be derived as $U_i^E(x) = P(G_i|x)$ using the Gaussian mixture with fixed variances (7) as above.

3) Theoretical properties: The Voronoi regions are relatively clear but the properties of the probabilistic and/or fuzzy allocation rules are not trivial, which we study in this section.

We first note that the probabilistic or fuzzy rules are closely related to Voronoi regions.

When we use probabilistic or fuzzy clustering, we often want to have hard reallocations of the objects. In such a case reallocation rule using the maximum of fuzzy memberships is natural:

$$u_i(x) = 1 \iff i = \arg \max_{1 \le j \le K} U_i(x), \tag{22}$$

where $u_i(x)$ is the final hard allocation rule and $U_i(x)$ is a fuzzy or probabilistic allocation rule.

We then have the following propositions.

Proposition 1: The characteristic function $u_i(x)$ derived from (22) defines the Voronoi region: $W_i(V)$, where V is the collection of prototypes derived from (6) or (9). The same property holds for $U_i^E(x)$.

Proposition 2: $U_i(x)$ given by (20) satisfies

$$\max_{x \in \boldsymbol{R}^p} U_i(x) = U_i(v_i) = 1, \tag{23}$$

$$\lim_{\|x\| \to \infty} U_i(x) = \frac{1}{K}. \tag{24}$$

On the other hand, $U_i^E(x)$ which has been derived from the entropy-based fuzzy K-means does not have the above properties in Proposition 2. Generally,

$$U_i^E(v_i) < 1 \tag{25}$$

and moreover

$$\sup_{x \in \boldsymbol{R}^p} U_i^E(x) > U_i^E(v_i). \tag{26}$$

The behavior of $U_i^E(x)$ as $\|x\| \to \infty$ is more complicated than that of $U_i(x)$. We need a new definition for this purpose.

Definition 1: A set of points $v_1, \ldots, v_K$ is called to have a general position when no three points of them are on a same line.

When cluster centers $v_1, \ldots, v_K$ are in a general position, no two boundaries of the Voronoi regions are parallel.

In addition, note that some Voronoi regions are bounded in the sense that a sufficiently large sphere can include the region, while others are unbounded.

We now have the next proposition.

Proposition 3: Assume that cluster prototypes $v_1, \ldots, v_K$ are in a general position. Let $V = (v_1, \ldots, v_K)$. If the Voronoi region $W_i(V)$ is bounded, then the corresponding fuzzy rule satisfies

$$\lim_{\|x\| \to \infty} U_i^E(x) = 0, \tag{27}$$

whereas if the Voronoi region $W_j(V)$ is unbounded, then the corresponding fuzzy rule satisfies

$$\lim_{\|x\| \to \infty} U_j^E(x) = 1, \tag{28}$$

provided that x moves inside the region $W_j(V)$.

4) Implications of fuzzy rules: We thus observe theoretical properties of probabilistic or fuzzy rules. We see they give Voronoi regions when clusters are made hard. This means that fundamental property of allocating objects are same for hard and fuzzy K-means. In other words, cluster boundaries are piecewise linear. If we want to have clusters with nonlinear boundaries which we call here nonlinear clusters for simplicity, we should use other methods.

There are two methods to have nonlinear cluster boundaries: one is to use additional variables [13] which we omit here to save space. Another is to use positive-definite kernel functions which we describe below.

C. Kernel-Based Clustering

The development of support vector machines [15], [14] stimulated the use of positive definite kernels [14]. Application of kernels to K-means clustering is described as follows.

1) High-dimensional mapping: Remember that X is a subset of $\boldsymbol{R}^p$. Assume H be another Euclidean space which may be finite or infinite dimensional. The norm of H is denoted by $\|\cdot\|_H$ and its inner product is $\langle \cdot, \cdot \rangle_H$. Let $\Phi \colon \boldsymbol{R}^p \to H$ a function which is called a high-dimensional mapping. We assume that $\Phi(x)$ itself is unknown but its inner product $\langle \Phi(x), \Phi(y) \rangle_H$ is represented by an explicit function:

$$K(x, y) = \langle \Phi(x), \Phi(y) \rangle_H. \tag{29}$$

A typical example is the Gaussian kernel:

$$K(x, y) = \exp(-\lambda \|x - y\|^2). \tag{30}$$

Let us consider a variation of the objective function of fuzzy K-means:

$$J(U, W) = \sum_{j=1}^{K} \sum_{k=1}^{N} (u_{kj})^m \|\Phi(x_k) - w_i\|_H^2 \tag{31}$$

where $m \geq 1$: when $m > 1$, the above function is for kernel-based fuzzy K-means; when $m = 1$, it implies hard K-means. Note that $W = (w_1, \ldots, w_K)$ is the collection of prototypes in H.

The alternate optimization with respect to U and W cannot be carried out, since

$$w_i = \frac{\sum_k u_{ki} \Phi(x_k)}{\sum_k u_{ki}} \tag{32}$$

cannot be calculated due to unknown $\Phi(x_k)$.

The alternate optimization is hence replaced by the iterative calculation of U and $D(x_k, w_i) = \|\Phi(x_k) - w_i\|_H^2$:

$$\begin{aligned} D(x_k, w_i) &= \|\Phi(x_k) - w_i\|_H^2 \\ &= K(x_k, x_k) - \frac{2}{\sum_{k=1}^{K} (u_{ki})^m} \sum_{l=1}^{N} (u_{li})^m K(x_l, x_k) \\ &\quad + \frac{1}{(\sum_{k=1}^{K} (u_{ki})^m)^2} \sum_{j=1}^{N} \sum_{l=1}^{N} (u_{ji} u_{li})^m K(x_l, x_j), \end{aligned} \tag{33}$$

while the membership is given by the same equations as before:

$$u_{ki} = \left[\sum_{j=1}^{K} \left(\frac{D(x_k, w_i)}{D(x_k, w_j)} \right)^{\frac{1}{m-1}} \right]^{-1} \quad (m > 1) \tag{34}$$

$$u_{ki} = \begin{cases} 1 & \iff i = \arg \min_{1 \leq j \leq K} D(x_k, w_i) \\ 0 & \text{otherwise} \end{cases} \quad (m = 1) \tag{35}$$

Note that not only the function $\Phi(x)$ but also the space H need not be explicitly given in this derivation.

2) Fuzzy classification rule: The fuzzy classification rule of kernel-based clustering is derived from replacing x_k by x

in (33) and (34):

$$D(x,w_i) = \|\Phi(x) - w_i\|_H^2 = K(x,x) - \frac{2}{\sum_{k=1}^K (u_{ki})^m} \sum_{l_1}^N (u_{li})^m K(x,x_l) + \frac{1}{(\sum_{k=1}^K (u_{ki})^m)^2} \sum_{j=1}^N \sum_{l_1}^N (u_{ji}u_{li})^m K(x_l,x_j), \tag{36}$$

$$U_i(x) = \left[\sum_{j=1}^K \left(\frac{D(x,w_i)}{D(x,w_j)} \right)^{\frac{1}{m-1}} \right]^{-1} \qquad (m>1) \tag{37}$$

The hard classification rules are omitted, since they are easy to derive from (35).

3) Another algorithm of kernel-based clustering: In addition to the above method, there is another algorithm. Let $\mathcal{K} = (K(x_i,x_j))$ be $N \times N$ matrix derived from a given kernel function. Note that $\mathcal{K}^{\frac{1}{2}}$ is well-defined using the square root of the positive eigenvalues, as all eigenvalues are non-negative. Assume that $\boldsymbol{e}_i$ $(1 \leq i \leq N)$ be ith elementary vector: $\boldsymbol{e}_1 = (1,0,0,\ldots)$, $\boldsymbol{e}_2 = (0,1,0,\ldots)$, and so on.

Let $y_k = \mathcal{K}^{\frac{1}{2}}\boldsymbol{e}_k$, and substitute y_k into x_k $(x_k \leftarrow y_k)$. Repeat the ordinary formula (5) and (6) until convergence. In short, we use the ordinary algorithm to $Y = (y_1,\ldots,y_N)$. Then what we have is the same as the ones by (34):

Proposition 4: Let $\hat{u}_{ki}$ be the solution by putting $x_k = y_k = \mathcal{K}^{\frac{1}{2}}\boldsymbol{e}_k$ in (5) and (6). Then we have $\hat{u}_{ki} = u_{ki}$, where u_{ki} is the solution of (34).

The proof is omitted, but readers can easily check that the solutions are the same.

4) Inductive and non-inductive clustering: Where is the difference between the methods to derive $\hat{u}_{ki}$ and u_{ki} in the last proposition?

Note that $\Phi\colon \boldsymbol{R}^p \to H$, whereas the map $x_k \mapsto \mathcal{K}^{\frac{1}{2}}\boldsymbol{e}_k$ is defined on X with the range $\boldsymbol{R}^N$. In this way, although the solutions are the same, the domains of definition and the ranges are different. This means that the former method has the fuzzy rule of classification $U_i(x)$ defined on $\boldsymbol{R}^p$, while the latter does not have a classification rule outside of X.

We have seen that a family of methods related to the K-means has classification rules defined over the whole object space. Thus when a new object occurs after clustering, each method can classify it to a certain cluster.

The last algorithm, in contrast, does not have such a classification rule. It simply generates clusters of X but a new object cannot be classified.

The former method is called here *inductive clustering*, while the latter is called *non-inductive*. This name is after Vapnik's concept of *inductive inference* and *transductive inference* in semi-supervised learning [4]. Thus methods related to the K-means are inductive, while the last algorithm is that of non-inductive clustering.

We see another class of methods and consider whether they are inductive or not.

III. Agglomerative Hierarchical Clustering

Another class of methods popular in various application fields is agglomerative hierarchical clustering which outputs dendrograms [1], [6]. This class of methods is very old but the number of users in applications are maybe as large as those of the K-means.

We assume that a distance between points $D(x,y)$ in this section is defined in some way, and it need not be an Euclidean distance or squared Euclidean distance.

The general procedure of agglomerative hierarchical clustering is in the following, where $D(G,G')$ is a distance between clusters, which will be defined after the procedure. Note also that the procedure has a given real parameter α.

1) Let initial clusters be individual objects $G_i = \{x_i\}$, $i = 1,\ldots,N$, and let the number of clusters be $K = N$.
2) Find the pair of clusters of minimum distance:
$$(G_p.G_q) = \arg \min_{1 \leq i,j \leq K} D(G_i,G_j) \tag{38}$$
3) If $D(G_p,G_q) > \alpha$, then stop the merging and output clusters $\mathcal{G}(\alpha) = \{G_h,\ldots,G_l\}$ and the clustering process as a dendrogram and stop.
4) Merge: $G_r = G_p \cup G_q$. Delete $G_p.G_q$ and add G_r to the collection $\mathcal{G}$ of clusters. Reduce the number of clusters $K = K - 1$.
5) If $K = 1$, then output the trivial cluster $\mathcal{G} = \{X\}$ and the clustering process as a dendrogram and stop.
6) Update distances $D(G_r,G_j)$ for all other clusters G_j in $\mathcal{G}$. Go to step 2).

Note that this procedure has two different kinds of outputs: an output is $\mathcal{G} = \{G_h,\ldots,G_l\}$ and another is a dendrogram (a dendrogram is undefined here, but readers can refer to standard texts of clustering like Everitt et al. [6]).

There are different methods of updating $D(G_r,G_j)$ which are called linkage methods. We consider three linkage methods below.

Single linkage: The distance is defined to be the minimum of distances between two points in the two clusters:

$$D(G,G') = \min_{x \in G, y \in G'} D(x,y) \tag{39}$$

Frequently, an updating formula which calculates $D(G_r,G_j)$ from $D(G_p,G_j)$ and $D(G_q,G_j)$ is used:

$$D(G_r,G_j) = \min\{D(G_p,G_j), D(G_q,G_j)\} \tag{40}$$

Complete linkage: The distance is defined to be the maximum of distances between two points in the two clusters which can be contrasted with the single linkage:

$$D(G,G') = \max_{x \in G, y \in G'} D(x,y) \tag{41}$$

The corresponding updating formula is as follows:

$$D(G_r,G_j) = \max\{D(G_p,G_j), D(G_q,G_j)\} \tag{42}$$

Average linkage: The distance is defined to be the average of distances between every combination of two points in the two clusters:

$$D(G,G') = \frac{1}{|G||G'|} \sum_{x \in G, y \in G'} D(x,y) \tag{43}$$

where $|G|$ is the number of elements in G. The corresponding updating formula is as follows:

$$D(G_r, G_j) = \frac{|G_p|}{|G_r|} D(G_p, G_j) + \frac{|G_q|}{|G_r|} D(G_q, G_j). \quad (44)$$

$$|G_r| = |G_p| + |G_q| \quad (45)$$

There are other two linkage methods of the centroid method and the Ward method, but we omit the detail of them.

It appears that these linkage methods do not have the inductive property, in other words, they do not have a particular classification rule for classifying another object. As we see in the next section, however, the single linkage method has a sound classification rule.

A. Inductive property of the single linkage

Unlike other linkage methods, the single linkage method is known to have a number of good theoretical properties: It is essentially equivalent to the minimum spanning tree of a weighted graph [1] and the max-min transitive closure of a fuzzy relation [11].

The single linkage method is closely related to the nearest neighbor classification rule, as shown by the definition of the distance (39).

Let us redefine the collection of clusters $\mathcal{G}$ of the output; as it has parameter α and it is applied to set X, we write the output as:

$$\mathcal{G}(\alpha; X) = \{G_h, \ldots, G_l\}. \quad (46)$$

Suppose that we have a new object y to some cluster. We find the nearest neighbor $z \in G_i$ of y and allocate y to G_i. This rule is written here as

$$\mathcal{G}(\alpha; X) \leftarrow y. \quad (47)$$

Note that

$$\mathcal{G}(\alpha; X) \leftarrow y = \{G_h, \ldots, G_i \cup \{y\}, \ldots G_l\}. \quad (48)$$

We have the following.

Proposition 5: Let

$$\alpha > \min_{x \in X} D(x, y). \quad (49)$$

We then have

$$\mathcal{G}(\alpha; X \cup \{y\}) = \mathcal{G}(\alpha; X) \leftarrow y. \quad (50)$$

In other words, clusters obtained from the single linkage with adding y to X before the algorithm starts and the allocation of y after clusters of X are obtained leads to the same result, provided that (49) holds. Note that if α is too small and (49) does not hold, then $\{y\}$ forms an isolated cluster in the left hand of (50).

This means that the single linkage clustering includes the nearest neighbor allocation rule as its essence. Hence we can say that the single linkage method has an inductive property.

Figure 2 is a complicated figure in which 20 points with numbers $1 - 20$ on the plane are objects for clustering. The segments connecting these points forms minimum spanning trees for the three clusters. They have been derived from the minimum spanning tree connecting all points by deleting three longest segments from it. Thus we have three clusters. It is known that these three clusters are obtained using the single linkage. Curved arcs are with the radius of a certain value of α, and the regions inside the arcs mean that when y is given within a region, y is allocated to the respective cluster, and (49) and (50) are satisfied. The dotted segments show unions of the Voronoi regions for the three clusters. If y is given in a region of the union of the Voronoi region, y will be allocated to the respective cluster, but (50) is not satisfied in general.

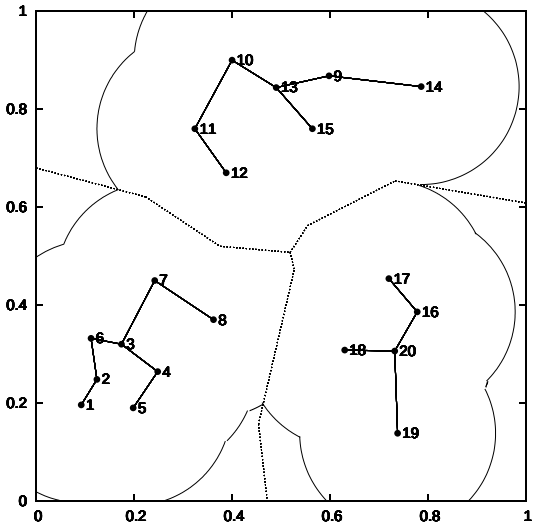

Fig. 2. Three clusters generated from 20 points by the single linkage. The regions surrounded by curves satisfy (50), while those regions outside of the curves and within the dotted lines produce isolated points that finally belong to the respective clusters.

What about the other linkage methods? We can define a furthest neighbor allocation rule related to the complete linkage method and an average allocation rule related to the average linkage method. However, a result like the one in Proposition 5 is not derived. Hence we cannot say the other two methods have the inductive property.

IV. Conclusion

Methods related to the K-means, kernel-based K-means, and agglomerative hierarchical clustering have been overviewed. The discussion is focused upon classification rules which include fuzzy rules and probabilistic rules. Methods with such rules are called inductive, while those without classification rules are called non-inductive.

We omitted complicated discussions of exceptional cases, e.g., when an object is on a prototype, or it is on the boundary of more than one Voronoi regions for simplicity, as such detailed discussion for exceptional cases will not alter the essential part of the present results.

The motivation for such clustering rules is mainly methodological: investigation of theoretical properties of rules will help deeper understanding of the method under consideration. However, such a methodological consideration will help us

when we want to choose a suitable method of clustering in a variety of applications.

Other subjects related to classification rules related to clustering discussed in this paper were omitted due to page limitation. Readers will find other methods and applications in [13].

ACKNOWLEDGMENT

The authors would like to thank the editors for inviting the author to this work. This study has partly been supported by the Grant-in-Aid for Scientific Research, JSPS, Japan, no.26330270.

REFERENCES

[1] M. R. Anderberg, *Cluster Analysis for Applications*, Academic Press, 1973.
[2] D. Arthur, S. Vassilvitskii, k-means++: the advantages of careful seeding, *Proc. of SODA 2007*, pp.1027-1035.
[3] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum, 1981.
[4] O. Chapelle, B. Schölkopf, A. Zien, eds., *Semi-Supervised Learning*, The MIT Press, 2006.
[5] J. C. Dunn, A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters, *J. of Cybernetics*, Vol.3, pp.32-57, 1974.
[6] B. S. Everitt, S. Landau, M. Leese, D. Stahl, *Cluster Analysis, 5th ed.*, Wiley, 2011.
[7] H. Ichihashi, K. Miyagishi, K. Honda, Fuzzy *c*-means clustering with regularization by K-L information, *Proc. of 10th IEEE Intern. Conf. on Fuzzy Systems*, Vol.2, pp.924-927, 2001.
[8] T. Kohonen, *Self-Organizing Maps 2nd ed.*, Springer, 1997.
[9] J. B. MacQueen, Some methods of classification and analysis of multivariate observations, In: Proc. of 5th Berkeley Symposium on Math. Stat. and Prob., pp.281-297, 1967.
[10] G. McLachlan, D. Peel, *Finite Mixture Models*, Wiley, 2000.
[11] S. Miyamoto, *Fuzzy Sets in Information Retrieval and Cluster Analysis*, Springer, 1990.
[12] S. Miyamoto, M. Mukaidono, Fuzzy *c*-means as a regularization and maximum entropy approach, *Proc. of the 7th IFSA World Congress*, Vol.2, pp.86-92, 1997.
[13] S. Miyamoto, H. Ichihashi, K. Honda, *Algorithms for Fuzzy Clustering*, Springer, 2008.
[14] B. Schölkopf, A. J. Smola, *Learning with Kernels*, The MIT Press, 2002.
[15] V. N. Vapnik, *Statistical Learning Theory*, Wiley, New York, 1998.