Fuzzy Cluster Algorithm for the Automatic Identification of Joint Sets

R. E. HAMMAH†
J. H. CURRAN†

The task of identifying and isolating joint sets or subgroups of discontinuities existing in data collected from joint surveys is not a trivial issue and is fundamental to rock engineering design. Traditional methods for carrying out the task have been mostly based on the analysis of plots of the discontinuity orientations or their clustering. However, they suffer from their inability to incorporate the extra data columns collected and also lack in objectivity. This paper proposes a fuzzy K-means algorithm, which has the capability of using the extra information on discontinuities, as well as their orientations in exploratory data analysis. Apart from taking into account the hybrid nature of the information gathered on joints (orientation and non-orientation information), the new algorithm also makes no a priori assumptions as to the number of joint sets available. It provides validity indices (performance measures) for assessing the optimal delineation of the data set into fracture subgroups. The proposed algorithm was tested on two simulated data sets in the paper. In the first example, the data set demanded the analysis of discontinuity orientation only, and the algorithm identified both the number of joint sets present and their proper partitioning. In the second example, additional information on joint roughness was necessary to recover the true structure of the data set. The algorithm was able to converge on the correct solution when the extra information was included in the analysis. © 1998 Elsevier Science Ltd. All rights reserved.

INTRODUCTION

When underground or surface excavations are made in rock masses, the behavior of the surrounding rock material can be greatly influenced by the presence of discontinuities [1, 2]. Various modes of rock slope and wedge failure can be attributed to the existence of fractures in a rock mass [3]. Discontinuities also play a crucial role in the classification of rock masses [4]. It is therefore of paramount importance, in both civil engineering and mining applications, to carefully collect and analyze data on the fractures present in a rock medium. The results of the analysis of joint survey data are subsequently used in the design of excavations in rock [1–3].

Methods for taking the measurements of the characteristics of discontinuities can be divided into two broad classes—borehole sampling methods and the mapping of exposed rock surfaces. Borehole sampling methods involve the extraction of rock cores from subsurface regions by means of diamond drilling. The cores recovered can be among other things analyzed for discontinuity orientation, surface geometry, etc. In addition, the walls of boreholes can be examined with remote cameras or television equipment [1, 3]. Also boreholes may be used to provide access for geophysical equipment in order to gather data on the structure of rock masses [1, 3].

When discontinuities intersect with rock surfaces, linear traces of these discontinuities arise. Consequently, the mapping of exposed rock surfaces allows for direct measurements of discontinuity properties to be made [3]. Exposed rock surfaces can be mapped in systematic fashion using either scanline or window sampling. In scanline sampling, a scanline (usually a tape of length between 2 to 30 m) [3] is stretched along an exposed planar or near planar surface. Discontinuity traces that intersect the scanline are recorded and measurements of their properties are made. Window or scanarea sampling is very similar to scanline sampling with the exception that properties of a discontinuity are recorded, if a section of its trace falls within a defined area of the exposed rock surface being mapped. More comprehensive discussions on the various sampling methods can be found in Refs [1–3].
On the basis of geological genetic criteria discontinuities encountered in rock masses can be classified into faults, bedding planes, joints, fractures and cleavages. The existence of the different types of discontinuities can be attributed to a range of geological processes that might have occurred in the rock mass being mapped. For example, joints are formed as a result of cooling in igneous rock, shrinkage due to drying in sedimentary rocks, and for all rock types can be attributed to tectonic stresses. Detailed accounts on discontinuity types can be found in Ref. [3]. Discontinuities generally occur in sets (sub-parallel planar groups) and tend to have a pattern. As geologists or geotechnical engineers map rock surfaces, they record observed geological relations between groups of discontinuities, and information on their geological genetic types. Therefore, the information provided by geologists or geotechnical engineers about the discontinuities encountered during mapping is of great importance to the analysis of joint survey data.

A typical joint survey data set consists of information on the orientation of the discontinuities encountered, the types of discontinuities, roughness of their surfaces, persistence, spacing and other attributes deemed important by the recording geologist or geotechnical engineer. Some of the characteristics recorded during the mapping of discontinuities, such as orientation and spacing, are quantitative in nature, while others like color and geological genetic type are qualitative variables. Based on the similarity across variables, the discontinuities are then separated into subgroups or sets.

Tools for analyzing survey data and delineating them into subgroups or clusters are mainly based on contouring on stereographic plots projections of the discontinuity poles (normals to the planes of discontinuities). Computer software packages exist today that make the plotting and contouring of pole information on stereograms a near trivial issue. (One such package is DIPS, a program developed by the Rock Engineering Group of the University of Toronto [5]. It was used for displaying and contouring all the pole data in this paper.) The data analyst thereafter proceeds to identify the different fracture sets revealed on the plots.

The above-described process has two principal weaknesses. The first deficiency is that all the other information, some or all of which may also assist in delineating the discontinuities into sets, have been excluded from the analysis. Also there exists the possibility that some joint sets may be very similar in orientation, but greatly differ in some other attributes, say, joint roughness, spacing or geologic genetic type. For example, the orientation of a fault may be close to that of a set of joints, but it cannot be grouped with the joints, because it is a major discontinuity with properties very different from that of joints. A separation, therefore, based on only orientations may lead to significant error. Such errors can be sources of severe problems during construction of structures in rock masses, or during the course of their use. This disadvantage can, in principle, be overcome by sorting the survey information on a spreadsheet in addition to the contouring [6]. However, this approach can be very tedious, if the data set is sufficiently large and/or the number of variables numerous, and it can fail to reveal the true structure apparent in the data.

One other disadvantage of the separation of fracture sets using the contouring of pole orientations on stereographic plots is that the process is very subjective. Different data analysts can arrive at very different answers based on their background, experience, and personal biases and inconsistencies [3]. The differences in results are even more pronounced in cases where the boundaries between clusters are unclear [3].

Past attempts at tackling the problem of subjectivity have tried to use cluster analysis. Shanley and Mahtab [7] were the first to propose an algorithm, which provided some objectivity to the process. That algorithm was further enhanced by Shanley and Yegulalp [8]. However, this algorithm still lacks the capability of including the recorded additional data in the analysis. Dershowitz et al. [9] describe a stochastic algorithm for clustering discontinuities, which can handle the extra information provided. The method is based on defining probability distributions for each of the fracture characteristics, and involves the integration of these probability distribution functions. Numerical integration usually is very computationally intensive. Also the method requires knowledge of the number of discontinuity sets in the data under analysis.

Statistical tools such as discriminant analysis, regression, decision analysis and cluster analysis exist for the exploratory analysis of data. More recently, artificial intelligence techniques such as neural networks have been developed also for the purposes of analyzing and interpreting data. However, in an environment where data has to be classified into homogeneous groups of objects in the absence of a priori information on the groups, cluster analysis is the tool most suitable for application [10, 11]. Techniques for clustering data abound in literature, and they can be separated into two broad categories—hierarchical cluster algorithms and partitional cluster methods. Hierarchical clustering techniques generate nested clusters of data, i.e. some clusters may be embedded in others and as a result different numbers of clusters can be obtained based on the level at which they are observed. Partitional algorithms generate a unitary partitioning of the data into homogeneous groups with all of them being at the same level. The authors are of the belief that of the major types of clustering algorithms, a K-means partitional algorithm would best suit the purposes of the rock mechanics expert. This can be attributed to the nature of the classification problem in the exploratory analysis of discontinuity data for rock mechanics pur-
poses, and to the fact that $K$-means algorithms are less susceptible to outliers than hierarchical schemes [12].

On the basis of the nature of the boundaries between the clusters identified, partitional algorithms can be divided into “hard” or “crisp” algorithms and “soft” classification schemes. In “hard” clustering an observation is assigned to one or the other cluster. However, this “all or nothing” classification does not adequately represent reality, because it is often the case that a data object in one cluster may bear similarity in some characteristics to other objects in a different cluster [13]. Soft cluster algorithms, on the other hand, assign to every observation degrees of membership to the clusters in a data set. These membership degrees often assume real values between zero and one. When observations belong to a cluster, they tend to have high values of degrees of membership to that cluster.

The cluster algorithm being proposed in this paper belongs to this class of cluster techniques. It is founded on one of the most widely used and successful classes of cluster analysis, the fuzzy $K$-means algorithm [13–15]. Fuzzy cluster algorithms have generally been developed and used in solving problems of computer vision and medical imaging, and are gaining popularity in other areas [11, 13, 14]. In general, it is believed that fuzzy clustering techniques yield results superior to those of other cluster methods [13]. This can be partially attributed to the fact that no observation is fully assigned to any cluster in any iteration of the cluster process. Also the soft boundaries of fuzzy algorithms allow them to escape traps in data that can fool hard classification schemes [15].

The fuzzy $K$-means approach is especially appealing in solving the problem of separating discontinuities into sets, because it explicitly accounts for the uncertainty present in both the collection and the analysis of the survey data. It also provides a computationally attractive alternative to the algorithm proposed by Dershowitz et al. in Ref. [9].

Fuzzy set theory provides the framework within which uncertainty in data can be accounted for in a natural and realistic manner [13]. It was originated by Zadeh [16] when he defined a new concept of sets that allowed an object to belong to a set with a degree of membership lying between zero and one. Under the fuzzy theory, the greater the certainty that an object belongs to a set, the closer its membership value is to one. Based on this idea of fuzzy sets, Ruspiní [17] defined an algorithm for separating data into sets through the minimization of an objective function. Practically all the subsequent work on fuzzy clustering that has followed can derive its foundation from Ruspiní’s work [13].

The earliest work, known to the authors, on the application of fuzzy $K$-means clustering to the analysis of discontinuity data was by Harrison [18]. The present work has made a number of modifications to the fuzzy $K$-means algorithm, which are specific to the clustering of orientations. These modifications include a different distance norm, a novel approach to determining the centroids (means) of clusters of directional (spherical data), and modifications of existing cluster performance measures peculiar to the clustering of discontinuity orientations.

THE FUZZY $K$-MEANS ALGORITHM

The fuzzy $K$-means algorithm uses Picard iterations to solve for the minimum of an objective function [13]. Given a data set comprising of $N$ observations, each described by a vector of $P$ attributes, $X_p=(X_{p1}, X_{p2}, \ldots, X_{pP})$, the algorithm seeks to partition the data set into $K$ subgroups or clusters on the basis of the measured similarities among the vectors (observations) of the data. The algorithm basically seeks for regions of high density in data. The prototype or centroid of each cluster is the vector most representative of the cluster, i.e. it is the geometric mean of all the vectors belonging to that cluster. Cluster prototypes are represented in the $K$-means algorithm as $V_i$. The solution to the problem of separating data objects into $K$ sub-groups can be achieved by minimizing the fuzzy objective function:

$$J_m(U, V) = \sum_{j=1}^{N} \sum_{i=1}^{K} (u_{ij})^m d^2(X_j, V_i); \quad K \leq N \quad (1)$$

The quantity $d^2(X_j, V_i)$ is the distance between observation $X_j$ and the cluster centroid $V_i$. The distance is a measure of the dissimilarity between the two points. When the two points exactly coincide, the distance between them is zero. It increases, as the two objects become more and more dissimilar.

The choice of the distance measure is determined by the space in which variables lie. In $\mathbb{R}^P$ space, the Euclidean norm

$$d^2(X_j, V_i) = \sum_{p=1}^{P} (X_{jp} - V_{ip})^2, \quad (2)$$

can be used. In order to measure the distance between orientations they are first converted to unit normals and thus the space becomes points lying on the surface of a unit sphere (non-Euclidean space). In this space the sine of the angle between two normals is the measure that represents how far apart they are (the cosine is a similarity and not a dissimilarity measure). The distance norm therefore becomes:

$$d^2(X_j, V_i) = 1 - (X_j \cdot V_i)^2, \quad (3)$$

where $X_j \cdot V_i$ is the dot product of vectors $X_j$ and $V_i$ (In Ref. [19] reasons are supplied as to why Equation (3) is a valid distance metric).

It must be noted that the distance measure used in a cluster algorithm implicitly imposes a topology on the data under analysis [13]. A distance metric causes a
fuzzy cluster algorithm to search for clusters that exhibit shapes defined by the metric [13,18,19]. The metrics described by Equations (2) and (3) tend to favor "spherical" clusters or clusters that are geometrically homogeneous, i.e., clusters shapes of which do not exhibit preferred directions around their means. For example, in two-dimensional space, elliptical clusters show preferred directions (along their principal axes) whereas circular clusters show no such preferences. There are ways of dealing with this problem, but their discussion is beyond the scope of the current paper (see Ref. [19] for more comprehensive coverage on this topic).

\( u_{ij} \) is the degree of membership (measure of belonging) of observation \( X_j \) in cluster \( i \) and is dependent on the distance. It is computed from the formula:

\[
(1 - \psi(X_j, V_i)) \left( \frac{1}{\psi(X_j, V_i)} \right)^{1/(m-1)} \left( \sum_{k=1}^{K} \left( \frac{1}{\psi(X_j, V_k)} \right)^{1/(m-1)} \right)^{-1}.
\]

(4)

Equation (5) simply computes the coordinates of cluster prototypes as the weighted average of the observations in the data set, the weights being the memberships, \( u_{ij} \), raised to the \( m \)th power. (The derivation of Equation (5) can be found in Ref. [13].)

The extent to which observations influence the determination of cluster prototypes is controlled by \( m \)—the fuzziness exponent. When Equation (5) is used for computing the locations of cluster centroids, weights of observations that are far from cluster centroids approach zero more rapidly than those of pattern vectors in the vicinity of cluster prototypes. This allows for the much-reduced influence of outlying points on the determination of centroids [22]. This establishes one of the reasons why the fuzzy K-means algorithm is less susceptible to outliers than some other classification methods.

In order to compute new cluster centroids for points on the surface of a unit sphere the formula Equation (5) even with normalization (to ensure that the centroids are also unit vectors) cannot be used, since it can lead to completely erroneous prototypes being determined. An example of a case where an erroneous mean can be calculated is considered in Fig. 1. The vectors shown on the diagram are normals to two sub-vertical planes perpendicular to the \( y-z \) plane. If Equation (5) (with the weighting factors, \( u_{ij} \), all set to be equal to 1) is used in determining the mean, the answer obtained is \((0.0, -0.0209, 0.9998)\) which is the normal to a near horizontal plane! To arrive at the correct answer, the sign of vector \( V_1 \) would have to be reversed before the application of formula Equation (5). When several vectors are involved, keeping track of signs becomes very cumbersome.

An approach for finding means of vectors that avoids the need for reversing signs is the eigenanalysis.

\[
\hat{V}_j = \frac{\sum_{j=1}^{N} (u_{ij})^m X_j}{\sum_{j=1}^{N} (u_{ij})^m}.
\]

(5)

Fig. 1. Calculation of mean of normals to two sub-vertical planes.
method outlined in Refs [23–25]. Given \( N \) unit vectors in the form of direction cosines \((x_j, y_j, z_j)\), where \( j = 1, \ldots, N \), their mean can be determined in the following manner:

(i) Compute the orientation matrix \( S \) using the formula

\[
S = \begin{bmatrix}
\sum_{j=1}^{N} x_j x_j & \sum_{j=1}^{N} x_j y_j & \sum_{j=1}^{N} x_j z_j \\
\sum_{j=1}^{N} y_j x_j & \sum_{j=1}^{N} y_j y_j & \sum_{j=1}^{N} y_j z_j \\
\sum_{j=1}^{N} z_j x_j & \sum_{j=1}^{N} z_j y_j & \sum_{j=1}^{N} z_j z_j
\end{bmatrix}.
\] (6)

(ii) Find the eigenvalues \((\tau_1, \tau_2, \tau_3)\) of \( S \) and their respective normalized eigenvectors \( (\xi_1, \xi_2, \xi_3) \), where \( \tau_1 < \tau_2 < \tau_3 \). Vector \( \xi_3 \), which is the eigenvector associated with the maximum eigenvalue, will be the mean vector of the group of \( N \) vectors.

In the form described above the eigenanalysis procedure cannot be used to determine centroids, since it would give the overall mean for the \( N \) vectors being analyzed. The authors have successfully adapted this method for computing the prototypes of clusters of unit vectors in the fuzzy \( K \)-means routine by including the weighting factors, \((u_{ij})^m\), in the orientation matrix. A modified orientation matrix \( S^* \) is defined to be

\[
S^* = \begin{bmatrix}
\sum_{j=1}^{N} (u_{ij})^m x_j x_j & \sum_{j=1}^{N} (u_{ij})^m x_j y_j & \sum_{j=1}^{N} (u_{ij})^m x_j z_j \\
\sum_{j=1}^{N} (u_{ij})^m y_j x_j & \sum_{j=1}^{N} (u_{ij})^m y_j y_j & \sum_{j=1}^{N} (u_{ij})^m y_j z_j \\
\sum_{j=1}^{N} (u_{ij})^m z_j x_j & \sum_{j=1}^{N} (u_{ij})^m z_j y_j & \sum_{j=1}^{N} (u_{ij})^m z_j z_j
\end{bmatrix}.
\] (7)

The normalized eigenvector corresponding to the largest eigenvalue of \( S^* \) is the new cluster centroid, i.e.

\[
\hat{V}_i = \xi_3.
\] (8)

Proof of the validity of using this approach of computing cluster centroids for spherical data is given in Appendix A at the end of this paper. There are several compelling reasons for using this approach as opposed to the use of a routine involving Equation (5) with suitable housekeeping of the signs of vectors. One of the reasons for the usage of the eigenanalysis approach is that the resulting eigenvalues and vectors contain important information on cluster shapes [19]. This information can be utilized with an appropriate distance metric for finding elliptical clusters on a sphere that are ill-placed relative to each other [19]. (For example, two elliptical clusters are ill-placed relative to each other, when their centroids are close and their principal axes are perpendicular to each other.)

Secondly, the two smallest eigenvalues obtained from the spectral decomposition of the matrix \( S^* \) have a physical meaning that allows them to be used in forming critical measures for determining the validity of various cluster partitions. This will be discussed in greater detail in Section 5. Furthermore, these advantages are gained without paying steep penalties in computational effort or speed. Memory requirements are not much more than those needed for a sign tracking sub-routine based on Equation (5), and there are very fast subroutines, in Numerical Analysis literature, for finding the eigenvalues and eigenvectors of matrices that can be easily implemented.

**SEQUENCE OF STEPS FOR EXECUTING THE ALGORITHM**

The following scheme gives the sequence of computations needed to solve for the minimum of the fuzzy objective function.

(i) The algorithm starts off with the selection of initial guesses for the \( K \) cluster centroids. The initial prototypes are chosen such that they are realistic vectors that fail in the general region of the data being analyzed. Their selection can be realized in different ways. One method would be to select the first \( K \) input vectors as the initial guesses of the centroids [11]. Another way of picking initial guesses is to randomly select \( K \) vectors from the data set as seed points for the algorithm [11]. To guarantee the selection of \( K \) well separated initial prototypes, the mean of the data set can be chosen as the first initial centroid. Thereafter, each subsequent initial guess is picked such that its distance from each of those already chosen is not less than a specified minimum [11, 21].

The authors experimented by initializing the algorithm with \( K \) randomly chosen cluster prototypes. One way of choosing a random vector in \( S^p \) space is by selecting each of the \( P \) attributes of the prototype as a random real number in the interval \((\bar{X}_p - 3\sigma_p, \bar{X}_p + 3\sigma_p)\), where \( \bar{X}_p \) is the mean of the \( p \)th attribute of the \( N \) input vectors \( X_n \) and \( \sigma_p \) is its corresponding standard deviation. To select a random point on the surface of a unit sphere, all it takes is to select three random real numbers between \((-1,1)\) and normalize them. Vectors selected in the manner described are realistic in that they lie in space of the input vectors and in the general region of the data.

Different choices of initial guesses of cluster prototypes can lead to different partitions of the same data. This is because the algorithm may or may not converge on the global minimum of the objective function Equation (1); it is only guaranteed to settle at a local minimum [11, 21]. This comes into play especially for data sets with poorly separated clusters. However, confidence in cluster results can be built by running the algorithm a few times and carefully observing resulting partitions. The authors have observed in their practice that this problem is much less pronounced when the
number of clusters $K$ is correct, i.e., corresponds to the actual number of clusters in a data set.

(ii) Compute the distances $d^2(X_i, V_j)$ of all $N$ observations from the $K$ cluster centroids using formula Equation (3) and/or Equation (2).

(iii) Calculate the degrees of membership, $u_{ij}$, of all $N$ observations in the $K$ clusters with formula Equation (4).

(iv) Evaluate new cluster prototypes using the eigen-analysis of the fuzzy orientation matrix, $S^*$, for directional (spherical) data. If non-orientation data (extra-recorded information) is also involved in the analysis, then Equation (5) can be used to determine those components of the coordinates of cluster prototypes. The validity of this treatment of mixed-type data (data with both spherical and non-spherical components) can be proved, if it is recognized that the objective function for such data consists of two distinct sums. The first sum involves the spherical component of observations (distances for this part of the objective function are measured with Equation (3)) while the second sum represents the contribution of the Euclidean components of points (distances involved in this component are computed using Equation (2)). Since the variables of the two sums that form the composite objective function are completely different from each other, minimization of the objective function can be accomplished through the minimization of the individual sums. Thus the proof for mixed-type data is a combination of that found in Appendix A of the current paper and that provided by Bezdek [13].

(v) Compute new distances using formula(s) Equation (3) and/or Equation (2), and new degrees of membership, $\hat{u}_{ij}$ (Equation (4)), for all $N$ observations.

(vi) If

$$\max_{i,j} |u_{ij} - \hat{u}_{ij}| < \epsilon$$

stop [12], otherwise go to step (iv) of the procedure. $\epsilon$ is a tolerance limit that acts as the criterion for terminating the iterations. It lies between 0 and 1. For all examples given in this paper, $\epsilon = 0.001$.

The fuzzy objective function for any specified number of clusters tends to have multiple stationary points (local minima). There is no guarantee that the Picard iterations of the algorithm will converge on the global minimum of $J_m$. Also, even if the algorithm converged on the global minimum, it would still not be guaranteed that this minimum of the objective function would necessarily provide the best possible partitioning of the data under examination [13]. However, in the practice of the authors, the proposed algorithm has strong tendency to converge on the correct solution when the number of clusters is correct for data sets, and especially when the clusters are well separated. This may be attributed to the general advantages of fuzzy algorithms given in the introduction to this paper. The examples provided in the paper show that there is at least strong evidence that the technique provides rational answers in clustering discontinuity data.

Since the main thrust of the current work is to present the theoretical aspects of the algorithm highlighted with two examples, an in-depth analysis of its performance over a broader range of geological conditions shall be provided in Ref. [26].

An alternate way of implementing the algorithm would be to start off by choosing random values for all the degrees of membership, $u_{ij}$, for the $N$ observations and proceeding to compute in sequence the cluster prototypes, distances, and new membership values, $\hat{u}_{ij}$ [13]. The approach of selecting initial cluster centroids as opposed to initializing memberships is preferred in this work, because it allows the data analyst to easily utilize any a priori knowledge that may exist on the means of discontinuity sets (cluster prototypes) to start off the algorithm. It usually is the case that when information is available on the structure of a data set, it is in the form of the number of clusters and their centroids.

**ASSIGNMENT OF OBSERVATIONS TO CLUSTERS**

Observations are assigned to one cluster or the other based on the values of the degrees of membership. Vector $X_i$ is considered to belong to cluster $I$, if its membership, $u_{ij}$, in cluster $J$ is higher than its membership in all the other clusters. By definition, outliers do not fit into the general patterns of any of the clusters very well. In fuzzy $K$-means clustering, they can be detected because they tend not to have high membership values in any clusters. It is therefore possible to exclude such ambiguous observations from being assigned to any cluster by establishing a threshold value for the maximum memberships of data observations. When the largest cluster membership value of a vector falls below the threshold, it may be excluded from getting assigned. Great care must be exercised, if points are not going to be assigned since not every vector with low cluster memberships is an outlier.

Jain and Dubes [11] advocate that outliers be identified and removed from cluster analysis because they tend to distort cluster shapes. Other experts caution on the exclusion of outliers from data sets. It is their belief that unless there are very compelling reasons for suspecting that an observation that is far from the rest of the data is a result of erroneous measurement it must be left alone. It might just be a rare occurrence of a feature that is present in the data and its removal can have serious repercussions. If an outlier is known to be a measurement error and it can be corrected, then this should be done. All that can be stated here by the authors is that if so desired, outliers can be removed from data sets after clustering with the fuzzy $K$-means algorithm.

**CLUSTER VALIDITY**

So far in the discussion of the method no mention has been made of how to determine the number of clusters in a data set, or establish that the partitioning
obtained is correct. After all the fuzzy K-means algorithm will provide K clusters whether that partitioning is true or not (in certain cases null clusters can exist). This question is one of the most important in cluster analysis and is not an easy one to answer. The answer to it is fundamentally tied in to the definition of what a cluster is.

One important issue to bear in mind is that any cluster algorithm has the potential to impose a structure on a data set that might not reflect the actual cluster structure present in the data. For example, as a result of the distance measure (Equation (3)) used in the above-described algorithm, it primarily seeks for rotationally symmetric clusters whether or not they exist in data under analysis. If the clusters possess rotational asymmetry and are unfavorably oriented relative to each other, the algorithm may be incapable of ever producing the right answers and the resulting validity measures would not be of much use. (If the placement of non-circular clusters relative to each other is not critical, which is the most commonly encountered case, the algorithm as it is can still perform very well in isolating these clusters [19].) However, the authors have proposed a distance measure that is capable of detecting rotationally asymmetric clusters of spherical data. Discussion of this measure lies outside the scope of this paper, but a comprehensive exposé on it is given in Ref. [19]. It suffices to mention, however, that the cluster validity measures that will be proposed in this section of the current paper for spherical data, and in themselves do not suffer from the above-described limitations can be used in conjunction with the methods discussed in Ref. [19] for finding ill-placed non-circular clusters.

Often times the data analyst uses a clustering algorithm to acquire information and knowledge on the structure existing in the data set [11]. It is therefore imperative that the algorithm provides the analyst with performance measures on the “validity” of the answers that are obtained from running the algorithm on data. Cluster validity is the section of cluster analysis that attempts to deal with the issues of validating clustering results.

A number of validity measures have been proposed in the literature in attempt to establish indices of cluster validity, but in this paper we shall restrict ourselves to indices for fuzzy cluster algorithms. This is because the algorithm under consideration belongs to this group of classification schemes, and that these performance measures have been widely accepted among researchers of fuzzy clustering.

Gath and Geva [21] have proposed indices that are based on the “crispiness” of resulting clusters, their hypervolume, and their densities. Their idea is that the clusters determined should be as less fuzzy as possible, should possess low hypervolumes (the term hypervolume is used, because the variable space not necessarily 3-dimensional), and should be of highest densities. Although their original performance measures were designed for a fuzzy implementation of maximum likelihood estimation, they can be generalized to apply to other algorithms in the family of fuzzy K-means methods.

In $\mathbb{R}^p$ space the fuzzy covariance matrix [27] is defined as

$$F_i = \frac{\sum_{j=1}^{N}(u_{ij})^n(x_j - v_i)(x_j - v_i)^T}{\sum_{j=1}^{N}(u_{ij})^n}.$$  (9)

The fuzzy hypervolume can then be defined as

$$F_{HV} = \left(\sum_{i=1}^{K} |\det(F_i)|\right)^{1/2}.$$  (10)

The average partition density of the clustering is computed using the formula

$$D_{PA} = \frac{1}{K} \sum_{i=1}^{K} \frac{S_i}{|\det(F_i)|^{1/2}},$$  (11)

where $S_i$ is known as the “sum of central members”, and is calculated as

$$S_i = \sum_{j=1}^{N} u_{ij}, \quad \forall X_j \in \{X_j(x_j - v_i)^T F_i^{-1}(x_j - v_i) < 1\}.$$  (12)

The partition density which is representative of the general density of the clustering is defined by Gath and Geva [21] as

$$P_D = \frac{S}{F_{HV}},$$  (13)

where

$$S = \sum_{i=1}^{K} \sum_{j=1}^{N} u_{ij}, \quad \forall X_j \in \{X_j(x_j - v_i)^T F_i^{-1}(x_j - v_i) < 1\}.$$  (14)

The optimal partitioning based on these criteria is that for which the hypervolume is minimal and the density measures are maximal.

For orientation data, the hypervolume and “sum of central members” used in computing the other performance measures of Ref. [21] cannot be used in the form in which they are originally given. The analogs of these cluster validity measures specific to the clustering of spherical data (or for determining the contribution of the spherical components of mixed type vectors) have to be proposed. The eigenvalues obtained from the spectral decomposition of the orientation matrix, $S^*$, are of great significance in the derivation of the hypervolume and “central members” for spherical data. Although a full discussion of the derivations of the spherical analogs of the fuzzy hypervolume and “sum of central members” lies outside the
scope of the current work, an attempt shall be made to explain the rationale behind the derived equations.

First, we shall recapitulate ourselves with the fact that the spectral decomposition of $S^*$ for a cluster identifies three orthogonal axes, $\xi_1$, $\xi_2$ and $\xi_3$, and their corresponding eigenvalues, $\tau_1$, $\tau_2$ and $\tau_3$. As is already known from the discussion of the calculation of prototypes, $\xi_3$ is the mean of a cluster of points. The other two vectors coincide with the minor and major principal axes of the elliptical distribution of points generated, when the observations in the cluster are projected onto a plane perpendicular to the mean. The eigenvalues can be expressed as (see Appendix A):

$$\tau_1 = \xi_i^T S^* \xi_1 = \sum_{j=1}^{N} (u_{ij})^m (X_j \cdot \xi_j)^2, \quad t = 1, 2 \text{ or } 3.$$  (15)

From the expression Equation (15), it can be seen that the eigenvalues are measures of the spread of the projections of vectors in the principal directions. The eigenvalues $\tau_1$ and $\tau_2$ provide information on cluster shapes. If the distribution of vectors in a cluster has rotational symmetry, i.e. if its shape is circular or near circular, then $\tau_1 = \tau_2$ or $\tau_2 = \tau_1 = \tau_1$. The greater the ratio $\tau_2/\tau_1$ is, the greater the departure of the cluster shape is from a circle (the eccentricity of the resulting elliptical shape increases). Also, the eigenvalues represent a measure of the deviation of vectors from the centroid of a cluster in the directions of the principal axes.

When a vector exactly coincides with the prototype of a cluster, its projections onto principal directions 1 and 2 are zero. Large deviations from the cluster mean result in larger projections on either or both of these two directions. This, precisely, is the idea captured by the variance of a set of observations in Euclidean $(\theta^m)$ space. Thus it can be deduced from Equation (15) that the eigenvalues divided by the quantity, $\Sigma_{j=1}^{N} (u_{ij})^m$, i.e.

$$s_1^2 = \frac{\tau_1}{\sum_{j=1}^{N} (u_{ij})^m},$$  (16)

are measures of the variance of the projections of spherical data observations in three orthogonal directions for the different clusters. Therefore, the hypervolume of a cluster of points on the surface of a sphere is the surface area determined from the integral:

$$F_i = \int_{0}^{\alpha_i} \int_{0}^{\beta_i} \frac{1}{\sqrt{1-x^2-y^2}} \, dx \, dy.$$  (17)

Because a closed-form solution of this integral cannot be easily established, it is handier to evaluate it using numerical quadrature methods. (It is possible to approximate this surface by simply calculating the area of the projection of the above-defined surface onto a plane, i.e. by using the product $s_1 \times s_2$. This approximation, however, works well only for small values of $s_1$ and $s_2$.)

Using the understanding of what the eigenvalues are, the "sums of central members" for spherical data can be defined as:

$$S_i = \sum_{j=1}^{N} u_{ij}, \quad \forall X_j \in \left\{ \frac{\sum_{j=1}^{N} (X_j \cdot \xi_{ij})^2}{s_i^2} < 1 \right\}. \quad (18)$$

The spherical analogue of the numerator of Equation (13) for computing the partition density becomes:

$$S = \sum_{j=1}^{K} \sum_{i=1}^{N} u_{ij}, \quad \forall X_j \in \left\{ \frac{\sum_{j=1}^{N} (X_j \cdot \xi_{ij})^2}{s_i^2} < 1 \right\}. \quad (19)$$

When mixed-type data is being analyzed (when extra data columns of information on discontinuities are included in an analysis), the cluster performance measures become composites of those measured for spherical data and for Euclidean data. To facilitate the definition of the composite measures, the vector $X_j$ representing the $j$th data point will conceptually be divided into two components such that it can be represented as: $X_j = (X_j^{(s)}, X_j^{(e)})$. The superscripts (s) and (e) refer to the spherical component (direction cosines) and Euclidean component (extra data), respectively. The hypervolume for clusters then becomes the product of the hypervolumes of the spherical and Euclidean constituents, i.e.:

$$F_i = F_i^{(s)} F_i^{(e)}, \quad (20)$$

where $F_i^{(s)}$ is the hypervolume determined from Equation (17), and $F_i^{(e)}$ obtained from Equation (9). The hybrid condition for establishing "central members" assumes the form:

$$X_j \in \left\{ \frac{\sum_{j=1}^{N} (X_j^{(s)} \cdot \xi_{ij})^2}{s_i^2} + (X_j^{(e)} - V_i^{(e)}) \cdot F_i^{-1} (X_j^{(e)} - V_i^{(e)}) < 1 \right\}. \quad (21)$$

Two other fuzzy cluster validity measures were used in determining the optimal partitioning of our sample data set. These were the Xie–Beni validity index [28] and the Fukuyama–Sugeno validity functional [20].

The Xie–Beni index is defined as

$$\nu_{XB}(U, V; X) = \frac{\sum_{i=1}^{K} \sum_{j=1}^{N} (u_{ij})^m \|X_j - V_i\|^2}{N(\min_{i \neq k} \{\|V_i - V_k\|^2\})}. \quad (22)$$

$\|\|$ is the norm for measuring distances in the classification space and therefore can be replaced with either Equation (3) (for spherical data only), or the sum of Equations (2) and (3) for mixed-type data. This index measures the compactness and separation of the fuzzy $K$-partition obtained from the clustering algorithm. The compactness is the ratio of the weighted total vari-
ation of the data set to the number of observations. The minimum distance between the prototypes of the clusters is known as the separation [28]. When a good partitioning of the data set is obtained the numerator of the index is small, because the membership degree, \( u_{ij} \), is high whenever \( \|X_i - V_j\|^2 \) assumes a small value and is small whenever \( \|X_i - V_j\|^2 \) becomes large. In the case where the clusters obtained are well separated, the minimum distance between the cluster centres is relatively high. It is therefore taken that small values of \( v_{SB} \) indicate a better clustering of the data set than larger values.

The Fukuyama–Sugeno validity functional is calculated from the formula

\[
v_{FS}(U,V;X) = \sum_{i=1}^{K} \sum_{j=1}^{N} (u_{ij})^m \left( \|X_i - V_j\|^2 - \|V_j - \bar{V}\|^2 \right)
\]

\[
= J_m - K_m,
\]

(23)

where \( \bar{V} \) is the geometric mean of the cluster centroids.

\( J_m \) in the above equation is the fuzzy objective function defined in Equation (1) and

\[
K_m = \sum_{i=1}^{K} \sum_{j=1}^{N} (u_{ij}^m) \|V_j - \bar{V}\|^2.
\]

The physical interpretation of the Fukuyama–Sugeno validity index is generally not very clear [20]. However, since the objective function, \( J_m \), decreases monotonically for increasing \( K \) (the number of clusters), the function \( K_m \) may be interpreted as a cost term intended to penalize the use of increasing values of \( K \) to minimize the objective function. Fukuyama and Sugeno propose that small values (minima) of \( v_{FS} \) attest to good partitionings of the data set into clusters. \( v_{FS} \) has the tendency to decrease monotonically with increasing \( K \).

In order to use the Xie–Beni and Fukuyama–Sugeno validity indices for establishing the optimal partitioning of directional data, the sine-squared norm \( 1 - (X_i \cdot V_j)^2 \) has to be used in place of the Euclidean metric \( \|X_i - V_j\|^2 \).

**SAMPLE APPLICATIONS OF FUZZY K-MEANS ALGORITHM**

The algorithm was run on two sample data sets. The first example is a demonstration of the use of the fuzzy K-means clustering algorithm for delineating a sample joint survey data set into distinct clusters based on orientation data.

The second example illustrates how the fuzzy clustering algorithm facilitates the automatic classification of joints into sets based not only on orientation information, but also when an extra data column is present. The extra data column could be any numeric property of discontinuities such as trace length, spacing, discontinuity frequency, or even joint roughness coefficient (JRC). In Section 6.2, the possible misclassification of discontinuities when delineation is solely founded on discontinuity orientation, is contrasted with the case when the inclusion of additional data on discontinuities significantly improves the quality of cluster separation.

For both examples, it is assumed that all samples in the data set come from one structural domain (a zone of a rock mass homogeneous or uniform in properties. If the discontinuities being analyzed are not from a structural domain, then they must be divided into homogeneous sub-groups and each of them analyzed separately. Failure to do so introduces error into the analysis of discontinuity data [2]. Hoek, Kaiser and Bawden in Ref. [6] recommend that the number of discontinuities measured in any structural domain be not less than a 100.

**Example 1**

The first data set consists of 195 rock joints (this data set is saved under the name Exampmin.dip in the EXAMPLES subdirectory of the DIPS program). For purposes of illustration, only orientation information
was used in the analysis. Figure 2 contains illustrations of a plot of the poles of the discontinuities in data set 1 and also a contour plot of these normals. The contours on the stereogram in Fig. 2(b) and on subsequent stereographic plots in this paper are contours of equal probability (in percent) of pole occurrence [25]. On the contour plot it can be seen that there are 4 possible fracture sets.

The data set was analyzed for values of $K$ between 2 and 7. For each number of clusters, $K$, the algorithm was run multiple times and performance values averaged. The values of the validity indices are displayed in Table 1 for the various numbers of clusters, $K$. In the table, values of the criteria that correspond to the optimal number of clusters, as determined by each the criteria, have been highlighted. Also the results in Table 3 are shown graphically in Fig. 3. Other than the Xie–Beni index, all the other validity indices indicate the optimal number of fracture sets to be 4. The fact that one index did not identify $K = 4$ as the optimal partition underlies the necessity of using more than one performance measure. No single cluster validity index

![Graphs of cluster validity indices for data set in Example 1.](image)

Fig. 3. Graphs of cluster validity indices for data set in Example 1. (a) Plot of fuzzy hypervolume, $F_{HV}$, against number of clusters, $K$. Minimum is seen at $K = 4$. (b) Plot of partition density, $P_{D}$, against number of clusters, $K$. Maximum is seen at $K = 4$. (c) Plot of average partition density, $D_{PA}$, against number of clusters, $K$. Maximum is seen at $K = 4$. (d) Plot of Xie–Beni index, $X_{BB}$, against number of clusters, $K$. Minimum is seen at $K = 3$. Note that value of index for $K = 4$ is second best. (e) Plot of Fukuyama–Sugeno index, $r_{FS}$, against number of clusters, $K$. Minimum is seen at $K = 4$. 
can correctly detect the optimal structure of data sets all the time. This is what necessitates the use of multiple validity criteria. Notwithstanding, the $K = 4$ partition is still ranked second by the Xie–Beni index.

Figure 4 has contour plots of the clusters, ensuing from the optimal partitioning of the data set. The algorithm has not only determined the correct number of clusters, but has also apparently identified the structure in the data. The results of clustering with the $K$-means algorithm, we believe, are in agreement with the partitioning most human analysts would come up with.

**Example 2**

In the second example of the application of the fuzzy $K$-means clustering algorithm, a simulated data set consisting of four joint sets, each of which had 50 observations, was generated (Fig. 5). In addition to data on discontinuity orientation, values of a quantitative characteristic were generated from uniform distributions for each of the joint sets. The parameters necessary for generating the observations in this artificial data set are given in Table 2. Before clustering the data, the values of the extra data column were standardized using the formula:

![Table 1. Performance measures for number of clusters, $K = 2$ to 7, for Example 1](image)

![Figure 4. Contour plots of poles of resulting clusters after fuzzy cluster analysis of first data set. (a) Contour plot of poles of discontinuities assigned to Set 1. (b) Contour plot of poles of discontinuities assigned to Set 2. (c) Contour plot of poles of discontinuities assigned to Set 3. (d) Contour plot of poles of discontinuities assigned to Set 4.](image)
where $\bar{X}$ and $S$ are, respectively, the mean and standard deviation of the values of the additional data column. The standardization was performed in order to scale down these values, in order not to allow them to overwhelm the orientation component in the cluster analysis of the mixed data. (The above-mentioned issue of equalizing, in some way, the influence of variables on clustering results introduces the topic of standardization and weighting of variables in cluster analysis. Its detailed treatment in relation to the clustering of discontinuities can be found in Ref. [29].)

Joint sets 2 and 3 overlap in orientation, but differ greatly in the additional quantitative characteristic. From viewing the plot of the all poles in the data set, it would be correct to say that 3 clusters were present, but with a possibility of the existence of a fourth cluster. The possible existence of a fourth joint set is suspected because the contour plot of the poles exhibits twin peaks in the region of orientation $225/65$. This may be cause for further investigation, since the presence of the twin peaks may be either caused by the presence of two close lying joint sets that differ in the extra property, or may be simply due to a sampling aberration.

Table 3 contains the values of the validity indices for $K$ partitions, $K$ ranging from 2 to 7, based on orientation only. The first three indices (those based on hypervolumes and densities) point to the existence of 3 clusters of discontinuities (although the hypervolume index assigns close values for the cases of $K = 3$ and $K = 4$). These observations are characterized in Fig. 6(a)–(c) by the stationary points of the plots of the performance indices against the number of clusters. The other two cluster performance measures —the Xie–Beni and Fukuyama–Sugeno indices— indicate 4 clusters. The results in Table 3 and the plots in Fig. 6(d), (e) evidence this choice. This example highlights the ability of the algorithm to provide very logical answers. The selection of 3 or 4 clusters by the validity indices mirrors the debate that would occur in the minds of data analysts as to how many clusters exist in the data. It is also noticeable that different validity indices indicate the various partitions with differing strength or conviction. For example, in Table 3 the Xie–Beni index strongly points to the 3 cluster partitioning as being the best, whereas the hypervolume criterion barely distinguishes between the cases of $K = 3$ and $K = 4$.

At this stage of the analysis of data set 2, it is impossible to correctly separate joint sets 2 and 3, because the additional discontinuity property has not been included in the clustering process. The inability to

**Table 2. Simulation parameters for generating discontinuities in Example 2**

<table>
<thead>
<tr>
<th>Joint set #</th>
<th>Orientation</th>
<th>Extra data column</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Distribution</td>
<td>$\kappa^a$</td>
</tr>
<tr>
<td></td>
<td>Dip</td>
<td>Dip direction</td>
</tr>
<tr>
<td>1</td>
<td>Fisher</td>
<td>21.0</td>
</tr>
<tr>
<td>2</td>
<td>Fisher</td>
<td>41.5</td>
</tr>
<tr>
<td>3</td>
<td>Fisher</td>
<td>32.5</td>
</tr>
<tr>
<td>4</td>
<td>Fisher</td>
<td>21.5</td>
</tr>
</tbody>
</table>

$^a\kappa$ is the concentration parameter of the Fisher distribution.

Note: For each of the discontinuity sets in Table 1, 50 points were generated.

**Table 3. Performance measures for Example 2 for number of clusters, $K = 2$ to 7, for orientation data only**

<table>
<thead>
<tr>
<th>$K$</th>
<th>Fuzzy hypervolume</th>
<th>Partition density</th>
<th>Average partition density</th>
<th>Xie–Beni</th>
<th>Fukuyama–Sugeno</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0.172562</td>
<td>473.4203</td>
<td>512.3597</td>
<td>0.147898</td>
<td>-43.7507</td>
</tr>
<tr>
<td>3</td>
<td>0.120048</td>
<td>566.159</td>
<td>608.8765</td>
<td>0.075526</td>
<td>-56.0455</td>
</tr>
<tr>
<td>4</td>
<td>0.119768</td>
<td>669.3643</td>
<td>724.9808</td>
<td>0.36519</td>
<td>-48.8923</td>
</tr>
<tr>
<td>5</td>
<td>0.142071</td>
<td>561.2356</td>
<td>651.3806</td>
<td>0.363421</td>
<td>-45.5348</td>
</tr>
<tr>
<td>6</td>
<td>0.132959</td>
<td>589.6222</td>
<td>647.515</td>
<td>0.386207</td>
<td>-45.3712</td>
</tr>
<tr>
<td>7</td>
<td>0.140767</td>
<td>539.7078</td>
<td>618.295</td>
<td>0.32375</td>
<td>-43.3776</td>
</tr>
</tbody>
</table>
properly delineate these two clusters leads to biased results for the distributions of the extra data column for these joint sets. (Also, some skewness in the distributions of the orientations for these two discontinuity sets arises.) The average value of the non-orientation characteristic determined for the two superimposed joint sets (sets 2 and 3), if the \( K = 3 \) partitioning is chosen, is 9.86 when clustering of the data is based on orientation information only.

Performing the analysis of the data set anew, but this time with the inclusion of the extra data column leads to the correct recovery of the true structure of the data set (the assignment of observations to clusters is near identical to the original structure of the simulated data set). As depicted by the graphs in Fig. 7 and the results in Table 4, all the validity measures select the partitioning of the data into 4 clusters as the best clustering of the observations. The contour plots in Fig. 8 show the resulting 4 clusters. A comparison of Fig. 8(b),(c) reveals the overlap of joint sets 2 and 3 in orientation. However, the algorithm correctly assigned the observations to their appropriate clusters.
as a result of the information contributed by the additional data column.

It must be noted that the contours of the plots in Fig. 8 do not exactly match the contours of the initial data set in Fig. 5. The differences in contouring can be attributed to the fixed number of grid points used for contouring poles in the program DIPS. An analogous situation arises in the drawing of histograms for Euclidean data. When observations from a number of statistical distributions are grouped together into one data set, histograms of the observations after they have been separated into clusters would not exactly match the histogram of the grouped observations, if the number of bins for all histogram plots is kept constant.
Table 4. Performance measures for Example 2 for number of clusters, $K = 2$ to 7, when extra data column is included in analysis

<table>
<thead>
<tr>
<th>$K$</th>
<th>Fuzzy hypervolume</th>
<th>Partition density</th>
<th>Average partition density</th>
<th>Xie–Beni</th>
<th>Fukuyama–Sugeno</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0.149958</td>
<td>517.4812</td>
<td>520.15858</td>
<td>0.13512</td>
<td>-41.0821</td>
</tr>
<tr>
<td>3</td>
<td>0.075335</td>
<td>1142.174</td>
<td>1316.936</td>
<td>0.127685</td>
<td>-140.8</td>
</tr>
<tr>
<td>4</td>
<td><strong>0.027801</strong></td>
<td><strong>2640.082</strong></td>
<td><strong>2974.768</strong></td>
<td><strong>0.0509983</strong></td>
<td><strong>-209.137</strong></td>
</tr>
<tr>
<td>5</td>
<td>0.034258</td>
<td>2054.298</td>
<td>2496.128</td>
<td>0.525081</td>
<td>-200.292</td>
</tr>
<tr>
<td>6</td>
<td>0.036405</td>
<td>1946.632</td>
<td>2224.118</td>
<td>0.917034</td>
<td>-170.757</td>
</tr>
<tr>
<td>7</td>
<td>0.036514</td>
<td>1871.21</td>
<td>2162.864</td>
<td>0.861568</td>
<td>-153.365</td>
</tr>
</tbody>
</table>

In the case where the extra data column is included in the analysis, the average value for this discontinuity characteristic for joint set 2 is 18.1, and 1.61 for joint set 3! These values differ greatly from the averaged value (9.86) obtained when the cluster analysis of the data set was performed without the extra data column. Use of the averaged value of the non-orientation discontinuity attribute in engineering calculations could have serious repercussions. This example is only a simple illustration of the extents to which answers can be biased, when information pertinent to the identification of clusters is omitted from analysis. At this stage it must be noted that key information for correctly separating clusters might reside in some geological data such as genetic type. In such a case, the appropriate conversion of the geological data into quantitative form so that it could be incorporated into

Fig. 8. Contour plots of poles of clusters obtained from analysis of second data set, with extra data column included. (a) Contour plot of poles of discontinuities assigned to Set 1. (b) Contour plot of poles of discontinuities assigned to Set 2. (c) Contour plot of poles of discontinuities assigned to Set 3. (d) Contour plot of poles of discontinuities assigned to Set 4.
a subsequent cluster analysis, would be greatly beneficial.

**DISCUSSION AND CONCLUSIONS**

The application of the popular and widely used fuzzy K-means algorithm [13–15] offers an elegant, simple and yet very powerful tool for overcoming some of the most fundamental problems of delineating rock fracture data into sets. Through the use of the algorithm the data analysis process acquires objectivity and can be performed at several times the speeds human analysts are capable of. Boredom and tedium can be either eliminated or greatly reduced by using an automatic classifier, especially in the case where the number of variables and categories existing in the data set is large. The approach also does not require any a priori knowledge of the structure present in the data set. In cases where a priori information exists on the structure of a data set, it can be utilized in a cluster analysis by supplying the algorithm with the known number of clusters. In addition, the algorithm can be initialized with the known mean features of the clusters as the seed points for prototypes.

The information on other features useful in separating discontinuities into distinct clusters can now be included in the automated analysis of joint survey data. It places in the hands of the user a powerful tool for trying several different options of clustering, which may otherwise escape detection by even the most skilled and experienced human analysts.

Performance measures of the partitioning results of fuzzy K-means algorithms allow the data analyst to obtain an idea of which particular clustering (or clusterings) is best or sensible. The plethora of validity indices available implies that the possibility of being misled by any one index is significantly reduced. Although certain data configurations or structures can make the algorithm susceptible to the “seed” points used in initializing it, when it is combined with human perception in practice, it greatly facilitates the recovery of the structure of discontinuity data.

The fuzzy K-means algorithm provides the appropriate framework for creating an excellent data analysis tool that provides the rock mechanics expert the opportunity to obtain objective and accurate answers in the task of delineating clusters. In addition, it is also capable of revealing structures in data that might otherwise escape human attention.

The use of the fuzzy K-means classification tool for analyzing discontinuity data must be encouraged, since it can be performed in a fraction of the time it would take a human analysts to examine the data for patterns. As a result, the analyst can instead spend his/her time in reviewing cluster results, which allows him/her to gain deeper appreciation and insight of the structure of discontinuity data. Also greater use of the method will lead to improved understanding of its behavior and most certainly to enhancements, by the research community, of its various components.

**REFERENCES**


APPENDIX A

The proof of the use of the fuzzy orientation matrix for determining new cluster prototypes for directional data can be established through the constrained optimization of the fuzzy objective function,

\[ J_0(U, V) = \sum_{i=1}^{N} \sum_{j=1}^{N} (u_{ij})^p \delta(X_i, V_j). \]  

(A.1)

However, before proceeding with the necessary derivations, we shall briefly look at the definition of eigenvalues and eigenvectors. If A is an \( n \times n \) matrix, and

\[ AX = \omega X. \]  

(A.2)

then there exist choices for the scalar quantity \( \omega \), called eigenvalues of A, that produce non-zero solutions X, called eigenvectors of A. Also, the eigenvalues \( \omega \) can be expressed in terms of A and \( X \) as:

\[ \omega = X^T AX. \]  

(A.3)

From the theory of the eigenanalysis of square matrices, it is established that there exist \( n \) eigenvalues, \( \omega_p \) (\( p = 1, \ldots, n \)), of the matrix A, with \( n \) corresponding eigenvectors, \( \xi_p \). The eigenvalues are arranged in order of magnitude so that \( \omega_1 \leq \omega_2 \leq \ldots \leq \omega_n \). If A is symmetric, then all its eigenvalues are real numbers.

For the minimization of \( J_0(U, V) \) with respect to cluster centroids, the degrees of membership of observations, the \( u_{ij} \), of the objective function are fixed while the cluster prototypes, the \( V_j \), are variables. For greater clarity, let each directional data observation and cluster centroids expressed in direction cosines be represented as:

\[ X_j = \begin{bmatrix} x_j \\ y_j \\ z_j \end{bmatrix}, \]

and

\[ V_j = \begin{bmatrix} \xi_j \\ \mu_j \\ \nu_j \end{bmatrix}, \]

respectively, with the constraint that \( \xi_j^2 + \mu_j^2 + \nu_j^2 = 1 \).

In that case, the distance measured between observation \( X_j \) and prototype \( V_j \) for the fuzzy algorithm becomes:

\[ d^2(X_j, V_j) = 1 - (X_j \cdot V_j)^2 = 1 - (x_j \xi_j + y_j \mu_j + z_j \nu_j)^2. \]  

(A.4)

Rewriting the objective function using the new notations and expressions from above, we obtain:

\[ J_0(U, V) = \sum_{i=1}^{N} \sum_{j=1}^{N} (u_{ij})^p (1 - (x_i \xi_j + y_i \mu_j + z_i \nu_j)^2). \]  

(A.5)

or

\[ J_0(U, V) = \sum_{i=1}^{N} \sum_{j=1}^{N} (u_{ij})^p - \sum_{i=1}^{N} \sum_{j=1}^{N} (u_{ij})^p (x_i \xi_j + y_i \mu_j + z_i \nu_j)^2. \]  

(A.6)

Upon close examination of Equation (A.6) it can be seen that the extrema of \( J_0 \) occur only when the second term on its right-hand-side: \( \sum_{i=1}^{N} \sum_{j=1}^{N} (u_{ij})^p (x_i \xi_j + y_i \mu_j + z_i \nu_j)^2 \), assumes an extremal value. The problem then of optimizing the fuzzy objective function reduces to that of solving the constrained optimization problem for the second term.

The solution of the constrained optimization problem can be obtained using the method of Lagrange multipliers. For each ith term of the expression to be optimized, let the Lagrangian be:

\[ F_i(\omega, V) = \sum_{j=1}^{N} (u_{ij})^p (x_i \xi_j + y_i \mu_j + z_i \nu_j)^2 - \omega (\xi_j^2 + \mu_j^2 + \nu_j^2 - 1), \]

where \( \omega \) is a Lagrange multiplier.

Partial differentiation of Equation (A.7) with respect to \( \xi_j \) yields the result:

\[ \frac{\partial F_i}{\partial \xi_j} = 2 \sum_{j=1}^{N} (u_{ij})^p x_i \xi_j + x_i y_i \mu_j + x_i z_i \nu_j - 2 \omega \xi_j. \]  

(A.8)

In similar fashion, partial differentiation of \( F_i(\omega, V) \) with respect to \( \mu_j \) and \( \nu_j \) leads to the following equations:

\[ \frac{\partial F_i}{\partial \mu_j} = 2 \sum_{j=1}^{N} (u_{ij})^p y_i \mu_j + y_i z_i \nu_j - 2 \omega \mu_j, \]  

(A.9)

and

\[ \frac{\partial F_i}{\partial \nu_j} = 2 \sum_{j=1}^{N} (u_{ij})^p z_i \nu_j - 2 \omega \nu_j. \]  

(A.10)

The stationary points of \( F_i(\omega, V) \) are attained only when its partial derivatives are equal to zero. Equating each of Equations (A.8), (A.9) and (A.10) to zero, results in the system of equations:

\[ \begin{bmatrix} \sum_{j=1}^{N} (u_{ij})^p y_i \xi_j + x_i y_i \mu_j + x_i z_i \nu_j - \omega \xi_i \\ \sum_{j=1}^{N} (u_{ij})^p y_i \mu_j + y_i \nu_j - \omega \mu_j \\ \sum_{j=1}^{N} (u_{ij})^p z_i \nu_j - \omega \nu_j \end{bmatrix} = \begin{bmatrix} \xi_i \\ \mu_i \\ \nu_i \end{bmatrix}. \]  

(A.11)

which can be rewritten in matrix form as:

\[ \begin{bmatrix} \sum_{j=1}^{N} (u_{ij})^p y_j x_j \\ \sum_{j=1}^{N} (u_{ij})^p x_j y_j \\ \sum_{j=1}^{N} (u_{ij})^p y_j z_j \end{bmatrix} \begin{bmatrix} \xi_i \\ \mu_i \\ \nu_i \end{bmatrix} = \begin{bmatrix} \omega \xi_i \\ \omega \mu_i \\ \omega \nu_i \end{bmatrix}. \]  

(A.12)

Comparing Equation (A.12) with Equation (2.2) it can be seen that \( \omega \) and

\[ \begin{bmatrix} \xi_i \\ \mu_i \\ \nu_i \end{bmatrix} \]

are respectively the eigenvalues and eigenvectors of the symmetric matrix:

\[ S^* = \begin{bmatrix} \sum_{j=1}^{N} (u_{ij})^p x_j \xi_j \\ \sum_{j=1}^{N} (u_{ij})^p \xi_j y_j \\ \sum_{j=1}^{N} (u_{ij})^p \xi_j z_j \end{bmatrix} \begin{bmatrix} \sum_{j=1}^{N} (u_{ij})^p y_j x_j \\ \sum_{j=1}^{N} (u_{ij})^p y_j y_j \\ \sum_{j=1}^{N} (u_{ij})^p y_j z_j \end{bmatrix}. \]  

(A.13)

Using the relationship of eigenvalues to eigenvectors given by Equation (A.3), it is possible to establish that:

\[ \omega = \sum_{j=1}^{N} (u_{ij})^p (x_j \xi_j + y_j \mu_j + z_j \nu_j)^2. \]  

(A.14)

Therefore, from Equation (A.6), the fuzzy objective function, \( J_0(U, V) \), is minimized when the eigenvalue is largest, i.e. when \( \omega = \omega_j \). As a result, the eigenvector, \( \xi_j \), that corresponds to the largest eigenvalue is the mean vector of a fuzzy cluster of directional (spatial) data points, since it is the vector that minimizes the fuzzy objective function.