Approximate Clustering Via the Mountain Method

Ronald R. Yager & Dimitar P Filev

Machine Intelligence Institute

Iona College

New Rochelle, NY 10801

Technical Report #MII-1305R

ABSTRACT

We develop a simple and effective approach for approximate estimation of the cluster centers on the basis of the concept of a mountain function. We call the procedure the mountain method. It can be useful for obtaining the initial values of the clusters that are required by more sophisticated cluster algorithms. It also can be used as a stand alone simple approximate clustering technique. The method is base is base upon a griding of the space, the construction of a mountain function based upon the data and then a destruction of the mountains to obtain the cluster centers.

1. Introduction

The method of Fuzzy C-Means (Fuzzy ISODATA) has been the dominant approach in both the theory and practical applications of fuzzy techniques to unsupervized classification for almost two decades.[1]

According to [2-5] if our data, $\{x_1, ..., x_n\} \subset R^s$, is n points in the s-dimensional space, the Fuzzy C-Means (FCM) defines a soft clustering into m < n clusters, characterized by cluster centers:

$$x_i^* = \frac{\sum_{k=1}^{n} v_{ik}^* x_k}{\sum_{k=1}^{n} v_{ik}^*}, \; i = (1, m)$$

and membership grades:

$$v_{ik}^* = \frac{\frac{1}{|x_k - x_i^*|^2}}{\sum_{j=1}^{m} \frac{1}{|x_k - x_j^*|^2}}, \; i = (1, m), \; k=(1,n)$$

where by $|\,.\,|$ we denote any inner product norm induced on R^s (in particular the Euclidean norm). If for some k and i $|x_k - x_i^*| = 0$, a singularity occurs , then v_{ik}^* for all i are any nonnegative numbers such that:

$$\sum_{i=1}^{m} v_{ik}^{*} = 1 \quad \text{and} \quad v_{ik}^{*} = 0 \quad \text{if} \quad |x_k - x_i^{*}| \neq 0$$

Such a partitioning provides a necessary condition for the local minimization of the FCM objective function J_C:

$$J_C = \sum_{k=1}^{n} \sum_{i=1}^{m} v_{ik}^{2} |x_k - x_i|^2$$

The FCM approach is essentially an iterative technique which starting with cluster centers generates membership grades which are used to induce new cluster centers.. The process continues it a stabilizes.

One difficulty with this approach is the estimation of the initial values of the cluster centers. This fact along with the local minimizing property of the FCM algorithm can sometimes complicate the process of clustering.

In this paper we suggest a heuristic algorithm for the initial estimation of the cluster centers. In this mode the aim of this approach is to provide a tool for supporting the FCM method. However, in situations where only approximate, not too exact, values of the cluster centers are needed, this approach can act as a stand alone clustering algorithm. The method introduced here is called the **Mountain Method (MM)** and its spirit is based upon what a human does in visually forming clusters. As we shall subsequently see this approach would be particularly useful in an interactive mode. The idea of this approach is rather straight forward. The first step is to form a discretetization of the object space R^S by forming a grid on R^S(see figure #1). The intersection of the grid lines, which occurs at what we shall call the vertex or node points , provides our desired discretetization. We shall denote the finite subset of R^S consisting of the vertices as V. The set of points in V constitute our condidates for cluster centers. Thus we see the degree approximation of our final centers is very sensitive to the fineness of our griding. The finer the griding the less approximate but, more calculations are needed. The second step is the introduction of the data and construction of what we call the mountain function. The mountain function which we denote as M is defined on V and is constructed as follows. For each piece of data, x_i, we add an amount to the M value at each point v in V. The amount added depends on the distance of v from x_i, the closer the

two the more added. In this way after all the data points have been considered we have a function on R^S, actually V, which looks like a mountain range reflecting the distribution of the data. The next step is the selection of the cluster centers. This is accomplished by the destruction of the mountains. We find the point in V, c_1, which has the greatest value for M, the peak of the mountain range, this becomes our first cluster center. At all points v in V we subtract from their M value a quantity dependent upon its distance from c_1 and the value $M(c_1)$. The effect of this subtraction is to reduce the mountains. We next look for the new peak. This becomes our next cluster center. We now use this to reduce our mountain function. We continue in this manner until the mountain function is virtually destroyed.

2. The Concept of a Mountain Function

Let us assume a collection of n data points $\{x_1, ..., x_n\}$ in the s dimensional space R^s. We shall denote by x_{kj} the j-th coordinate of the k-th point, where k = 1, 2, . ., n and j= 1, 2, . ., s. We shall without loss of generality, restrict the s dimensional space R^s to an s dimensional hypercube $I_1 \times .. \times I_s$ where the intervals I_j, j = 1, 2, . ., s are defined by the ranges of the coordinates x_{kj}, i.e.:

$$I_j = [\min_k(x_{kj}), \max_k(x_{kj})]$$

Evidently the hypercube contains all the points of the data set $\{x_1, ..., x_n\}$. Further we shall discretize each of the intervals I_j into r_j equidistant points. Such a discretization forms an s dimensional grid in the hypercube with nodes $N_{(i_1, ..., i_s)}$, where indices $i_1, ..., i_s$ take values from the sets $[1, ..., r_1], ..., [1, ..., r_s]$. We shall denote the equidistant coordinates of the grid nodes by $X_1^{(j)}, ..., X_{r_j}^{(j)}$, where points quantize the interval I_j.and j = (1, s). The grid discretizes the space restricted by the hypercube. We emphasize the significance of this discretization, the grid nodes are are potential cluster centers. On one hand the coarser the discretetization, less nodes, the less calculations required but also the coarser the final cluster center values. In the following we shall use the simpler notatiion of N_i to indicate a node with the implicit fact that i equals some tuple of the form $(i_1, ..., i_s)$.

We shall look at any of the grid nodes as a possible candidate to become a cluster center. Let $d(x_k, N_i)$ be the distance from the data point x_k to the grid nodes N_i. Consider the simple case of two dimensional data, S = 2. We assume a discretization of I_1 into 3 and I_2 into 4 points. In this case we can express $x_k = (x_{k1}, x_{k2})$. Furthermore our discretetization gives us

$$I_1 = \{X_1^{(1)}, X_2^{(1)}, X_3^{(1)}\} \text{ and } I_2 = \{X_1^{(2)}, X_2^{(2)}, X_3^{(2)}, X_4^{(2)}\}.$$

In the case where i = (1, 3) the distance is defined as follows:

$$d(x_k, N_i) = (|x_{k1} - X_1^{(1)}|^p + |x_{k2} - X_3^{(2)}|^p)^{\frac{1}{p}}$$

In particular we can consider p = 1 or p = 2 (Euclidean distance).

As we discussed in the introduction, we construct a mountain function defined over the set grid nodes by adding, for each data, an amount to each node proportional to its distance from the data point.

In the following we shall use the distance measure $d(x_k, N)$ to score the membership of each grid node N_i to the data points $\{x_1, ..., x_n\}$. We shall assign higher scores to the nodes that are closer to a data point. The following function provides our mountain function at the vertex point N_i

$$M(N_i) = \sum_{k=1}^{n} e^{-\alpha\, d(x_k, N_i)} \qquad \text{(I)}$$

In the above α is a positive constant.

It is evident from the form of function (I) that the closer a grid node N_i is to the data point x_k the higher is the score supplied by the exponential term of (I); it is maximal for a data point that coincides with the grid node; it is decreasing exponentially for the farther grid nodes. We can look at the values of function $M(N_i)$ as heights of a mountain range, having as a base the grid. We call the function $M(N_i)$ the **mountain function**. The value of the mountain function can be seen to be closely related to the *density* of data points in the neighborhood of the node. As such it can be seen to represent the potential ability of any of the grid nodes to be a cluster center. The higher the mountain function value the larger is this potential ability. Therefore the mountain function value can be used as indicators of the clustering of the the hypercube from the data.

3. Estimation of the Cluster Centers from the Mountain Function

The basic idea of identification of cluster centers estimates is simple and straightforward. By using of the mountain function we are looking systematically for global maxima and associate them with cluster centers. We start by looking for the grid node with maximal mountain value, the mountain range peak. If there are more than one maxima, we select randomly one of them. Let us denote the maximal value of the mountain function as M_1^* thus

$$M_1^* = \text{Max}_i[M(N_i)].$$

We note it is the global maximum of the mountain function

Let the grid node N_1^* indicate the point in the hypercube where this maximal score of the mountain function is attained, it is a peak of the mountain range formed by the data set. It should be noted that this peak is usually surrounded by a number of grid points that also have high scores this is due to process of constructing the mountain function and the inherent continuity of the hypercube. We shall assign this peak node, N_1^*, as the first cluster center.

In order to find the next cluster center we must first eliminate the effects of the cluster center just identified. In order to accomplish this we subtract from our current mountain function a value at each node, this gives us a revised mountain function. The amount subtracted at each node point is proportional to the distance of the point from the maximal, the newly assigned cluster center, as well as being proportional to the current maximal value, M_1^*. In this manner we now form a new revised mountain function $\widehat{M}^2$ defined on the set of all nodes:

$$\widehat{M}^2(N_i) = \widehat{M}^1(N_i) - M_1^* \sum_{k=1}^{n} e^{-\beta\, d(N_1^*, N_i)} \qquad \text{(II)}$$

In the above $\widehat{M}^1(N_i)$ is actually the original mountain function, the value M_1^* is the current maximal and β is a constant.

To find the next cluster center we proceed in a similar manner. We find a node point, N_2^*, that maximizes the new reduced mountain function, $\widehat{M}^2$, this becomes our second cluster center we then remove the effects of this center in a manner similar to II giving us a new modified mountain

function $\widehat{M}^3$ which we use in the next stage. More generally expression II can be written as

$$\widehat{M}^k(N_i) = \widehat{M}^{k-1}(N_i) - M^*_{k-1} \sum_{k=1}^{n} e^{-\beta\, d(N^*_{k-1},\, N_i)} \qquad \text{(III)}$$

In the above $\widehat{M}^k$ is the new mountain function, $\widehat{M}^{k-1}$ the old mountain function, M^*_{k-1} the peak value of the location of this peak value, the newly identified cluster center, N^*_{k-1}.

We can look at this process as a process of destroying the mountain function. This process will guarantee that those nodes closer to the new identified cluster center will have their mountain more strongly reduced then the those further away. In some sense the idea of this approach is in the same spirit as Kohonen's [6] approach to self organization.

In the formulation we have suggested we have used an exponential function to introduce the effects of adding data and removing cluster centers one could use other functions, in particular a linear function may we used.

In the examples that follow we have found it more effective to bound the new mountain functions below by zero, thus

$$\widehat{M}^k(N_i) = \text{Max}\, [\widehat{M}^{k-1}(N_i) - M^*_{k-1} \sum_{k=1}^{n} e^{-\beta\, d(N^*_{k-1},\, N_i)},\ 0].$$

We shall demonstrate the workings of the mountain method in the following example.

Example 1. We consider a set of 10 data points in R^2 space shown in figure #1 and listed in table 1. The intervals I_1, I_2 are [0, 1] and [0, 1]. The grid for discretization, with $r_1 = r_2 = 6$, is shown in the same figure. The numerical values of the mountain function calculated according to (I) are listed in table 2. In figure #2 we depict the mountain function in the 3D space. It is seen that the maximal value of the mountain function appears at node (1, 0.6) of the grid. Therefore we associate this node with the first cluster center estimate. To find the next (less important) cluster center we remove the effect of the first cluster by destroying the mountain function. The modified mountain function, after removing the first custer center, is depicted in Fig.3. Its numerical values are listed in table 3. It can be seen by comparison of the entries of table 2 and table 3 the extensive destroying of the mountain function around the cluster center estimate (1.0, 0.6). By inspecting the modified mountain function we find the global maximum at grid point (0.6, 1.0). It becomes the estimated

second cluster center.

k	X	Y
1	0.36	0.85
2	0.65	0.89
3	0.62	0.55
4	0.50	0.75
5	0.35	1.00
6	0.90	0.35
7	1.00	0.24
8	0.99	0.55
9	0.83	0.36
10	0.88	0.43

Table 1. Given points in the R^2 space.

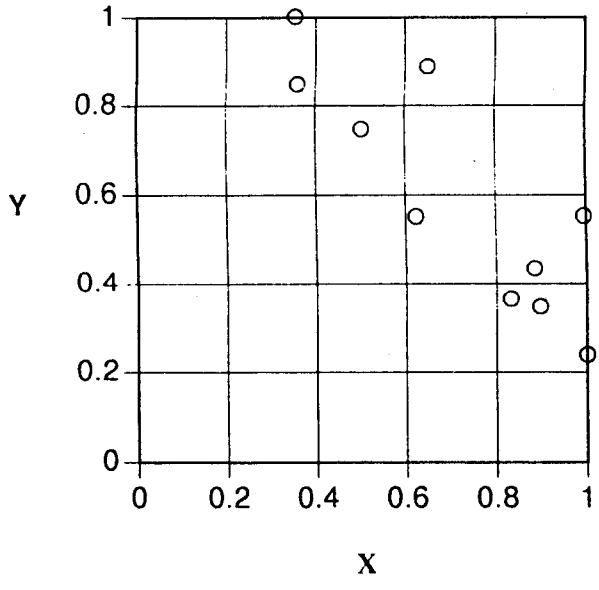

Fig.1. Example 1: Given data set.

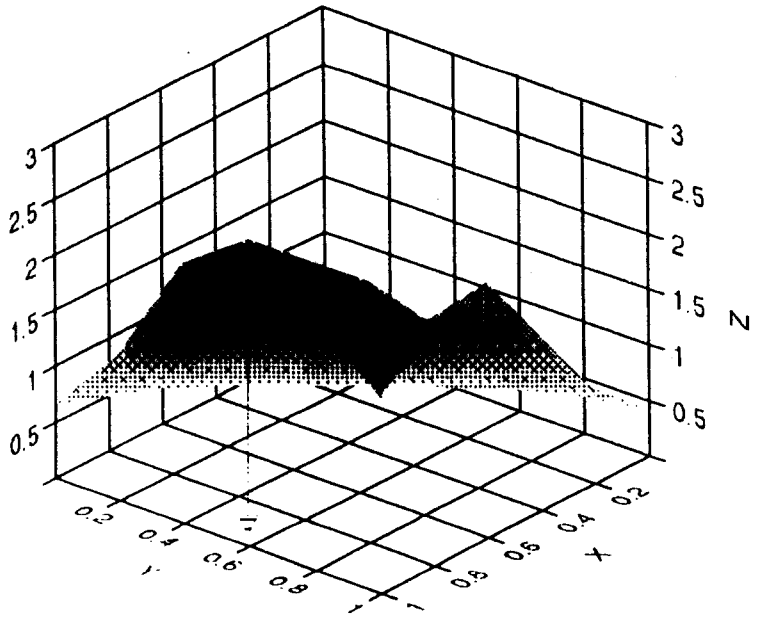

Fig.2. Example 1: The mountain function.

		X					
		0.00	**0.20**	**0.40**	**0.60**	**0.80**	**1.00**
	0.00	0.06	0.11	0.19	0.31	0.48	0.65
	0.20	0.10	0.18	0.32	0.54	0.88	1.31
Y	**0.40**	0.16	0.28	0.50	0.85	1.42	2.33
	0.60	0.23	0.43	0.75	1.25	1.95	2.75
	0.80	0.32	0.62	1.08	1.70	2.33	2.24
	1.00	0.41	0.84	1.54	2.28	2.14	1.74

Table 2. Numerical values of the mountain function.

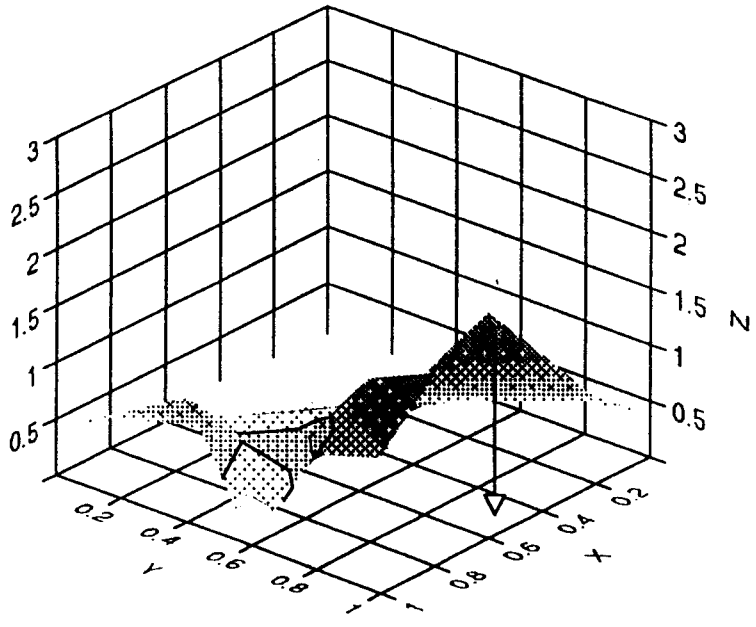

Fig.3. Example 1: The mountain function after removing the first center.

		X					
		0.00	**0.20**	**0.40**	**0.60**	**0.80**	**1.00**
	0.00	0.04	0.06	0.10	0.16	0.26	0.40
	0.20	0.06	0.10	0.16	0.25	0.42	0.75
	0.40	0.11	0.18	0.28	0.39	0.53	1.09
Y	**0.60**	0.18	0.31	0.50	0.70	0.72	0.00
	0.80	0.28	0.52	0.87	1.24	1.44	1.01
	1.00	0.38	0.77	1.39	2.00	1.68	1.19

Table 3. Numerical values of the mountain function after removing the first center.

We continue the production of new cluster centers and reduction of the mountain function until the level of the current maximum M^*_{k-1}, compared with the original maximum M^*_1 becomes too low. This means that there are only very few points around this cluster center and it can be omitted. We shall stop the process of destroying the mountain function when the ratio:

$$\frac{M^*_1}{M^*_{k-1}} < \delta \qquad \text{(V)}$$

where δ is given parameter; we shall denote by p* the step that satisfies the stop criterion (V). Obviously it defines (p*-1) cluster centers.

We shall summarize the above discussion in the following algorithm:

Algorithm

1. Calculate the intervals I_j, j=(1,s).
2. Quantize the intervals and form the grid.
3. Calculate the mountain values according to (I).
4. Find the cluster centers estimates and modify the mountain function according to (III) until stopping rule (IV) is satisfied.

Example 2. We consider the two dimensional data set, that is presented on figure #4. Its mountain function is depicted in figure #5.

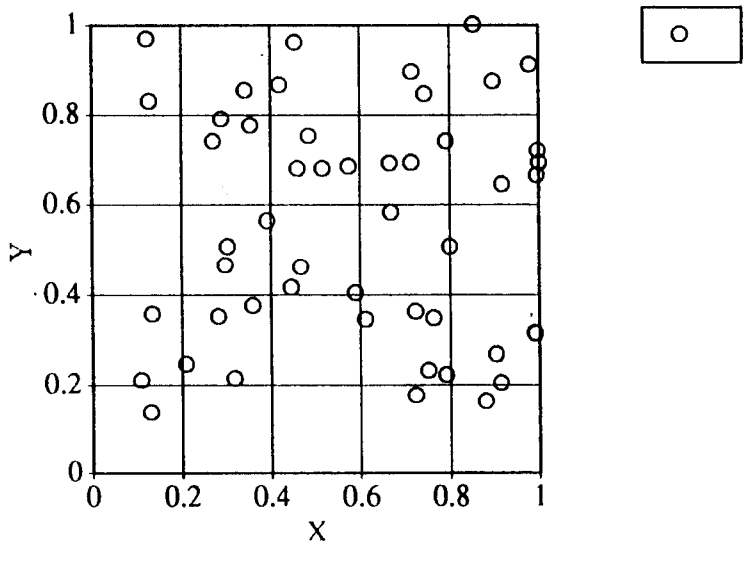

Fig.4. Example 2: Original data set.

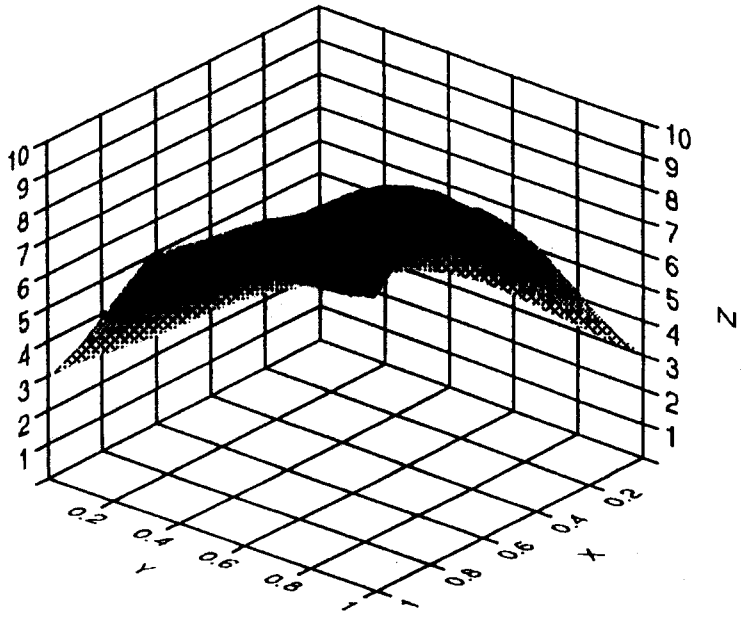

Fig.5. Example 2: The mountain function

The first estimated cluster center estimate is found at the node (0.7, 1.0). The absolute maximum of the mountain function at this node is 9.99. The process of estimating the cluster centers and destroying the mountain function is illustrated in figures 6 - 9. The second center estimate is at node (0.9, 0.5); it is associate with an absolute maximum 7.49 of the modified mountain function that is presented on Fig.6.

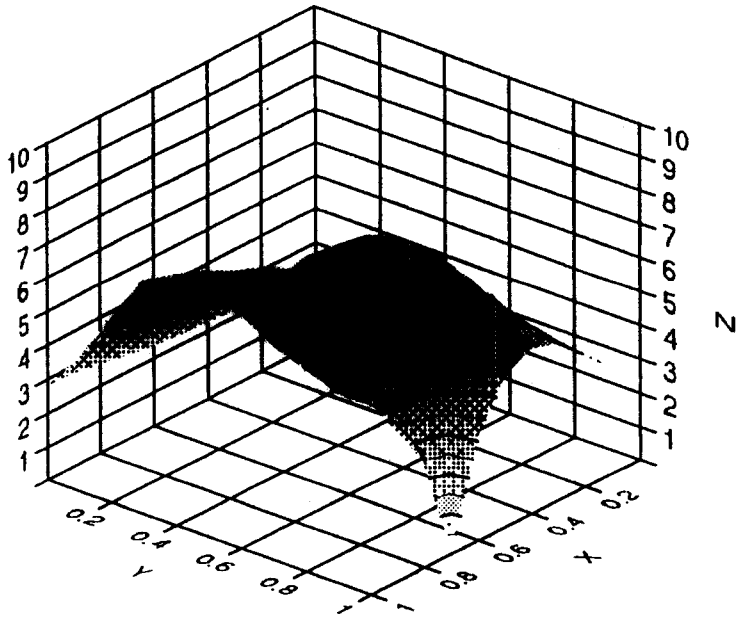

Fig.6. Example 2: The mountain function after removing the first center.

Similarly we obtain the next cluster center estimate at node (0.4, 0.6), with an with peak value of the modified mountain function 5.78, see figure #7.

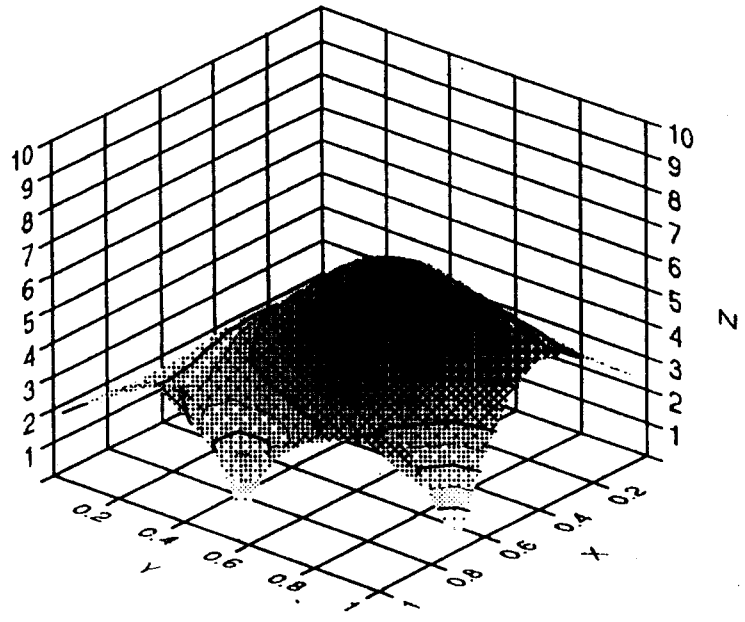

Fig.7. Example 2: The mountain function after removing the second center

In the same manner we proceed finding the next cluster center estimates by destroying the mountain function, see figure #8 and figure #9.

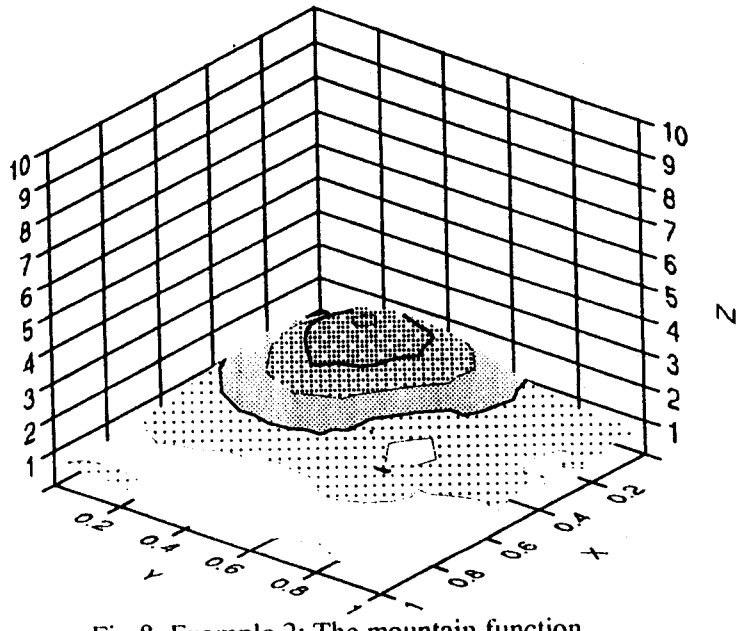

Fig.8. Example 2: The mountain function after removing the sixth center.

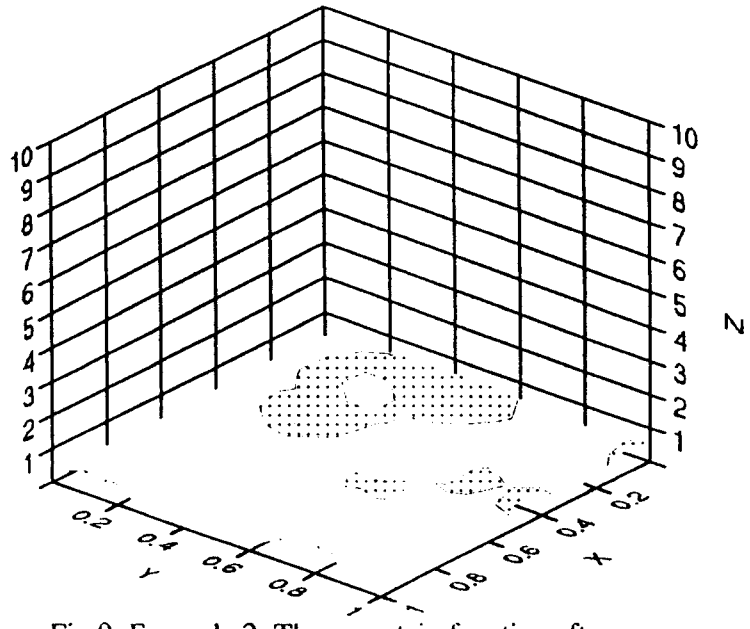

Fig.9. Example 2: The mountain function after removing the seventh center.

The first five identified cluster centers are shown in figure #10.

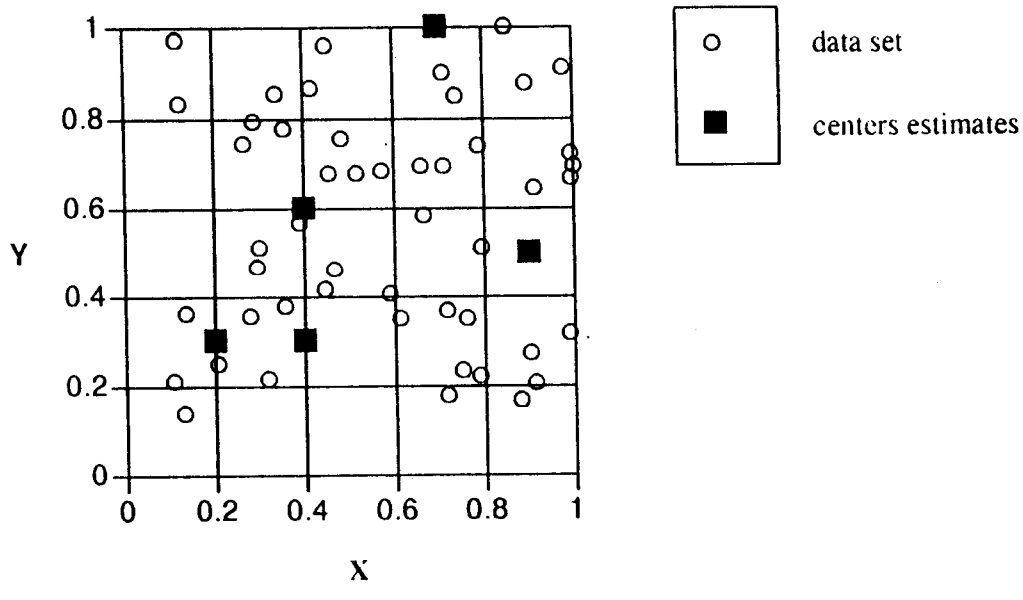

Fig.10. Example 2: Original data set and centers estimates

4. Conclusion

A simple, easy to implement algorithm, for approximate clustering was presented in this paper. It is based upon the idea of griding the space and the concept of constructing and destroying the a mountain function. This method which allows us to identify estimates of the cluster centers can be used to identify initial starting cluster centers values for use in more sophisticated cluster seeking algorithms such as fuzzy c-means. This technique can also be used as an end in itself to provide a quick and approximate technique for locating cluster centers. Its eventual combination with computer graphics will allow us to develop a natural and effective approach to clustering.

5. References

[1]. Bezdek, J. C. and Pal, S. K., Fuzzy Models for Pattern Recognition, IEEE Publications: New York, 1992.

[2]. Bezdek, J. C., "Numerical taxonomy with fuzzy sets," Journal of Mathematical Biology 1, 57-71, 1974.

[3] Bezdek, J., Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum: New York, 1981.

[4]. Kandel, A., Fuzzy Techniques in Pattern Recognition, Wiley-Interscience: New York, 1982.

[5] Bezdek, J., Hathaway, R., Sabin, M. and Tucker, W., "Covergence theory for Fuzzy C-Means: counterexamples and repairs," IEEE Trans. on Systems, Man & Cybernetics, vol. SMC-17, 873-877, 1987.

[6] Kohonen, T., Self-Organization and Associative Memory, Springer-Verlag: Berlin, 1984.