IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727
PP 25-31
www.iosrjournals.org

Locating Unevenly Formed Clusters Based on Entropy

[1]Mrs. M. Rajalakshmi MCA.,M.Phil., [2]Mrs.A.Kalaiselvi,
[1]Assistant Professor, Department Of MCA, Sankara College Of Science And Commerce Saravanampatti
[2]Research Scholar, Sankara College Of Science And Commerce Coimbatore

Abstract: *In data clustering the more traditional algorithms are based on similarity criteria which depend on a metric distance. This fact imposes important constraints on the shape of the clusters found. These shapes generally are hyper spherical in the metric's space due to the fact that each element in a cluster lies within a radial distance relative to a given center. This paper propose a clustering algorithm that does not depend on simple distance metrics and. Therefore, it allows us to find clusters with arbitrary shapes in n-dimensional space. The proposal is based on some concepts stemming from Shannon's information theory and evolutionary computation. Here each cluster consists of a subset of the data where entropy is minimized. This is a highly non-linear and usually non- convex optimization problem which disallows the use of traditional optimization techniques. To solve it we apply a rugged genetic algorithm. In order to test the efficiency of our proposal we artificially created several sets of data with known properties in a tridimensional space. The result of applying our algorithm has shown that it is able to find highly irregular clusters that traditional algorithms cannot. Some previous work is based on algorithms relying on similar approaches (such as ENCLUs' and CLIQUE's). The differences between such approaches and techniques are discussed.*

Keywords: *clustering, data mining, information theory, genetic algorithms.*

I. Introduction

Clustering is an unsupervised process that allows the partition of a data set *X* in *k* groups or *clusters* in accordance with a similarity criterion. This process This process is unsupervised because it does not require a priori knowledge about the clusters. Generally the similarity criterion is a distance metrics based in *Minkowsky Family of metrics* [1] which is given by:

$$d_{mk}(P,Q) = \sqrt[p]{\sum_{i=1}^{n} \left| P_i - Q_i \right|^p} \qquad (1)$$

where P and Q are two vectors in an n-dimensional space. From the geometric point of view, these metrics represent the spatial distance between two points. However, this distance is sometimes not an appropriate measure for our purpose. For this reason sometimes the clustering methods use statistical metrics such as *Mahalanobis'* [2], *Bhattacharyya's* [3] or *Hellinger's* [4], [5]. These metrics statistically determine the similarity of the probability distribution between random variables P and Q. In addition to a similarity criterion, the clustering process typically requires the specification of the number of clusters. This number frequently depends on the application domain. Hence, it is usually calculated empirically even though there are methodologies which may be applied to this effect [6].

1.1 A Hierarchy of Clustering Algorithms

A large number of clustering algorithms has been proposed which are usually classified as follows:

Partitional. Which discover clusters iteratively relocating iteratively elements of the data set between subsets. These methods tend to build clusters of proper convex shapes. The most common methods of this type are k-means [7], k-medoids or PAM (Partitioning Around Medoids) and CLARA (Clustering Large Applications) [8].

Hierarchical. In which large clusters are merged successively into smaller clusters. The result is a tree (called a dendrogram) whose nodes are clusters. At the highest level of the *dendrogram* all objects belong to the same cluster. At the lowest level each element of the data set is in its own unique cluster. Thus, we must select the adequate cut level such that the clustering process is satisfactory. Representative methods in this category are BIRCH [9], CURE and ROCK [10].

Density Based. In this category a cluster is a dense (in some pre-specified sense) region of elements of the data set that is separated by regions of low density. Thus, the clusters are identified as areas highly populated with elements of the data set. Here each cluster is flexible in terms of their shape. Representative algorithms of this category are DBSCAN [11] and DENCLUE [12].

Grid Based. Which use space segmentation through a finite number of cells and from these performs all operations. In this category are STING (Statistical Information Grid-based method) described by Wang et al. [13] and Wave Cluster [14].

Additionally, there are algorithms that use tools such as fuzzy logic or neural networks giving rise to methods such as Fuzzy C-Means [15] and Kohonen Maps [16], respectively. The performance of each method depends on the application domain. However, Halkidi [17] present several approaches that allow to measure the quality of the clustering methods via the so-called "quality indices".

1.2 Desired Properties of Clustering Algorithms

ln general a good clustering method must:
Be able to handle multidimensional data sets.
Be independent of the application domain.

In units based on CLIQUE (Clustering in Quest) [18] algorithm where a unit is dense if the fraction of the elements contained in the unit is greater than a certain threshold. A cluster is the maximum set of connected dense units. Another work is the so-called COOLCAT algorithm [19] which also approaches the clustering problem on entropic considerations but is mainly focused on categorical sets of data. The difference of this proposal is that the space is quantized through a hypercube that encapsulates all elements of the data set. The hypercube is composed of units or quantization that called “hypervoxels” or, simply, “voxels”. The number of voxels determines the resolution of the hypercube. Contrary to ENCLUS, this algorithm does not iterate to find the optimal space quantization. Here the hypercube is unique and its resolution is given a priori as a parameter. The units of quantization become the symbols of the sources alphabet which allow an analysis through information theory. This working hypothesis is that areas with high density have minimum entropy with respect to areas with low density.

II. Generalities

In what follows to make a very brief mention of most of the theoretical aspects having to do with the proper understanding of this algorithm. The interested reader may see the references.

2.1.InformationTheory

Information theory addresses the problem of collecting and handling data from a mathematical point of view. There are two main approaches: the statistical theory of communication (proposed by Claude Shannon [20]) and the so-called algorithmic complexity(proposed by Andrei Kolmogorov [21]). In this paper rely on the statistical approach in which information is a series or symbols that comprise a *message.* which is produced by an *information source* and is received by a *receiver* through a *channel.*

Where:

Message. It is a finite succession or sequence of symbols.

Information Source. It is a mathematical model denoted **by** S which represents an entity which produces a sequence of symbols (message) randomly. The space or all possible symbols **is** called source alphabet and is denoted as ∑ (see [22]).

Receiver. It is the end **of** the communication's channel which receives **the** message.

Channel. It is the medium used to convey a message from an *information source* to a *receiver.* In this document we apply two key concepts which are very important for theproposal.

Self Information. It is the information contained in a symbol s_i. which is defined as[1]:

$$I(s) = —\log2\ p(s4) \qquad (2)$$

where $p(s_i,)$ is the probability that the symbol s_i, is generated by the source S. we can see that the information of a symbol is greater when its probability is smaller. Thus,the self information of a sequence or statistically independent symbols is:

$$I(S_1S_2 \ldots S_n) = I(S_1) + I(S_2) + \ldots + I(S_n) \qquad (3)$$

Entropy. The entropy is the expected value of the information of the symbols generated by the source S. This value may be expressedas

$$H(S) = \sum^{n}_{i=1} p(s_i) I\ p(s_i) = -\sum^{n}_{i=1} p(s_i) \log2\ p(s_i) \qquad (4)$$

where *n* is the size of the alphabet ∑. Therefore, we see that entropy is greater the more uniform the probability distriburion of symbols is.

III. Geneticalgorithms

Genetic Algorithms (GA) (a very interesting introduction to genetic algorithms and other evolutionary algorithms may be found in [23]) are optimization algorithms which are frequently cited as "partially simulating the process of natural evolution". Although this is -a suggestive analogy behind which, indeed, lies the original motivation for their inception. it is better to understand them as a kind of algorithms which take advantage of the implicit (indeed, unavoidable) granularity or the search space which is induced by the use of the finite binary representation in a digital computer. In such finite space numbers originally thought of as existing in $\mathfrak{R}$ "actually map into ***B"*** space. Thereafter it is simple to establish that a genetic algorithmic process is a finite Markov chain (MC) whose states are the populations arising from the so- called genetic *operators:* (typically) selection, crossover and mutation. As such they display all of the properties of a MC. from this fact one may infer the following mathematical properties or a GA: 1) The results of the evolutionary process are independent of the initial population and 2) A GA preserving the best individual arising during the process will converge to the global optimum (albeit the convergence process is not bounded in time).
For a proof of these facts the interested reader may see [24]. Their most outstanding feature is that, as opposed to other more traditional optimization techniques, the GA iterates simultaneously over *several* possible solutions. Thereafter, other plausible solutions are obtained by combining *(crossing over)* the *codes* or these solutions to obtain hopefully better ones. The solution space (SS) is, therefore, traversed stochastically searching for increasingly better plausible solutions. In order to guarantee that the SS will be globally explored some bits or the encoded solution are randomly selected and changed (a process called *mutation).* The main concern or GA-practitioners (given the fact that well designed GAs, in general, will find the best solution) is to make the convergence as efficient as possible. The work of Forrest et al. has determined the characteristics of the so-called *Idealized GA* (IGA) which is improvious to GA-hard problems[25].

3.1vasconcelos'GeneticAlgorithms

The implementation or the **IGA** is unattainable in practice. However, a practical approximation called the Vasconcelos' GA (VGA) has been repeatedly tested and proven to be highly efficient [26]. The VGA, therefore, turns out to be an optimization algorithm of broad scope of application and demonstrably high efficiency. A statistical analysis was performed by minimizing a large number or functions and comparing the relative performance of six optimization methods[2] of which five are GAs. The ratio of every GA's absolute minimum (with probability *P=O.95* relative to the best GA's absolute minimum may be found in Table 1 under the column "Relative Performance". The number of functions which were minimized to guarantee (he mentioned confidence level is shown under Number of Optimized Functions".

Algorithm	Relative Performance	NumberOf Optimized Functions
VGA	1.000	2.736
EGA	1.039	2.484
TGA	1.233	2.628
SGA	1.236	2.772
CGA	1.267	3.132
RHC	3.830	3.600

Table 1. Relative Performance of Different Breeds of Genetic Algorithms

It may be seen that the so-called Vasconcelos' GA (VGA) in this study was the best of all the analyzed variations. Interestingly the CGA (the classical or canonical" genetic algorithm) comes at the bottom of the list with the exception of the random mutation hill climber (RHC) which is not an evolutionary algorithm. According to these results, the minima found with the VGA are, on the average, more than 25% better than those found with the CGA. Due to its tested efficiency,we now describe in more detail theVGA.
Outline of vasconcelos' Genetic Algorithms (VGA)
1. Generate random population of **n** individuals (suitable solutions for the problem).
2. Evaluate the fitness f(x) of each individual x in the population.
3. Order the ***n*** individuals from best (top) **to** worst (bottom) for *i=l, 2.. . . .* ***n*** according to their fitness.
4. Repeat steps A-D (see below) for *i = l,2....,[n/2].*

A. Deterministically select the i - th and the **(n** — i+ l) - th, individuals (the parents) from the population.
B.With probability Pc cross over the selected parents to form two new individuals (the offspring). If no crossover is performed. Offspring are an exact copy of the parents.
C**.**with probability *Pm* mutate new offspring at each locus (position in individual).
D. Add the offspring to a new population

5. Evaluate the fitness f(x) of each individual x in the new population
6. Merge the newly generated and the previous populations
7. If the end condition is satisfied, stop. and return the best solution.
8. Order the *n* individuals from best to worst ***(i=1,*** 2, ... *n)* according to their fitness
9. Retain the top *n* individuals; discard the bottom *n* individuals
10. Go to step 4

As opposed to the CGA. the VGA selects the candidate individuals deterministically picking the two extreme (ordered according to their respective fitness) performers or the generation for crossover. This would seem to fragrantly violate the survival-of-the-fittest strategy behind evolutionary processes since the genes or the more apt individuals are mixed with those of the least apt ones. However. the VGA also retains the best *n* individuals out of the *2n* previous ones. The net effect of this dual strategy is to give variety to the genetic pool (the lack of which is a cause for slow convergence) while still retaining a high degree of elitism. This sort or elitism, or course. guarantees that the best solutions are not lost. On the other hand. the admixture or apparently counterpointed plausible solutions is aimed at avoiding the proliferation of similar genes in the pool. In nature as well as in GAs variety is needed in order to ensure the efficient exploration of the space of solutions[3]. As stated before. all GAs will eventually converge to a global optimum. The vGA does so in less generations. Alternatively we may say that the **VGA** will outperform other GAs given the same number of generations. Besides, it is easier to program because we need not to simulate a probabilistic process. Finally. the VGA is impervious to negative fitness's values .We, thus, have a tool which allows us to identify the best values tor a set or predefined metrics possibly reflecting complementary goals. For these reasons we use in our work the **VGA** as the optimization method. In what follows we explain our proposal based in the concepts mentioned above.

IV. Evolutinary Entropic Clustering

Let X be a data set of elements x_i, such that $x_i=\{x1,x2. ..... ,x_{in}\}$. let *D* he an *n* dimensional space such that xi C *D* and let c_j be a subset or D called cluster. Then we must find a function that associates each element of X to the *j-th* cluster c_j, as:

$$f(x_i)=c_j; \square x_i \varepsilon X \wedge 2 \leq j \leq k \qquad (5)$$

where K is the number of clusters and ***f(xi)*** is called the membership function. Now we describe a method which attempts to identify those elements within the data set which share common properties. These properties are a consequence of (possibly) high order relationships which we hope to infer via the entropy of a quantized vector space. This space. in what follows, will be denoted as the *Hpercubic Wrapper.*

4.1 Hpercubic Wrapper

A Hpercubic Wrapper denoted as HW is an n-dimensional subspace of D such that:

x_i € HW □ x_i€ X (6)

HW is set of elements is called voxels, which are units in ,n-dimensional that can contain zero or more elements of the set X. The cardinality or HW depends on the maximum number of voxels that we specify in each dimension of the space **D** such that:

$$|HW| = \prod^{n}_{i=1} L_i \qquad (7)$$

where $\mathbf{L_i}$ is the number or voxels in the i-th dimension and **n** is the dimension number of D. From equation (7) it follows that □ V_m € HW:

$$0<m\leq\prod^{n}_{i=1} L_i \qquad (8)$$

where α is the number the elements that belong to the data set X and β is the number or symbol within cluster i. This constraint ensures that entropy is minimal within any given cluster. The algorithm yields a best individual which represents a set of clusters or symbols that are map into sets of voxels in the subspace *HW,* as shown inFig.3.

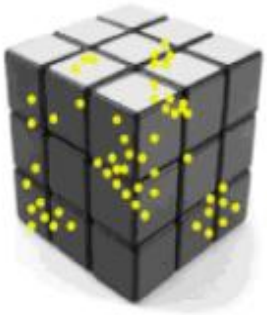

Fig.1. Hypercubic Wrapper in a tri-dimensional space where the points represent elements of the data set and the subdivisions are voxels.

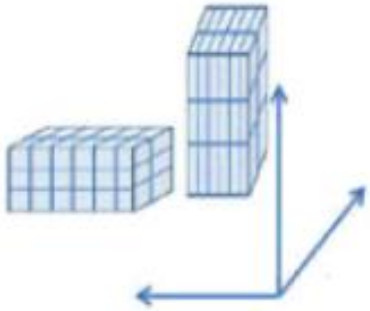

Fig.2. Hypercubes with different lengths per dimension

In what follows we show some experiments that allow us to test the effectiveness of the algorithm presented previously

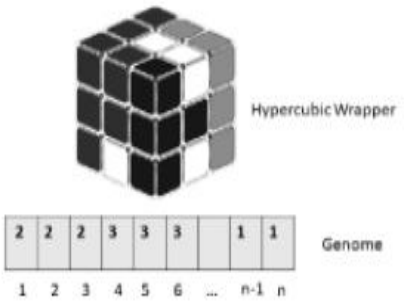

Fig.3. Possible clustering delivered by the VGA. Different intensities in the cube represent a different cluster .(White voxels are empty).

V. ExperimentaI Results

Our algorithm was tested with a synthetic data set which consists of a set of points contained by three disjoint spheres. The features and parameters of the first test are given in Table 2. The values of the parameters were determined experimentally.

The VGA was run 20 times (with different seeds or the pseudo random number generator yielding an average effectiveness of 98%. Notice that no information other than the number of clusters is fed to FGEEA. The same data set was tested with other algorithms such as *Kohonen Maps* and *Fuszi C-Means,* The results obtained are showninTable3, This us allow see that the result of our proposal is similar to result given by me alternative algorithms. The high effectiveness in all cases is probably due to the spatial distribution of data set. Next, we test with other data set whose spatial distribution yields presents overlapping clusters as is shown in

Fig. 4. For clarity we show a bi-dimensional example. The actual runs consisted of threedimensionaldata.

Feature	Value	Parameter	Value
Sample Size	192	N(Number of individuals)	500
Elements per cluster	64	G(Generations)	1000
Dimensions	3	Pm(Mutation Probability)	0.001
Cardinality	Disjoint sphere	Pc(Crossover Probability)	0.99

Table 2. Features and Parameters first test

Algorithm	Average Effectiveness
Kohonen Maps	0.99
Fuzzy C.Means	0.98

Table 3. Results obtained with Kohonene Maps and Fuzzy C-Means

In this case the size sample is 192. Its elements are, a priori, distributed in three clusters. The cardinality is 64 in all cases. The results obtained are shown in Table 4.

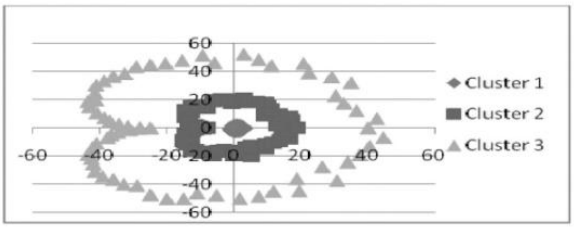

Fig .4.Overlapping clusters

Algorithm	Average Effectivenes
Kohonen Maps	0.62
Fuzzy C-Means	0.10
FGEEA	0.73

Table 4.Results of FGEEA with overlapping data set

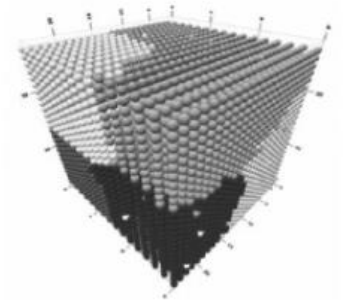

Fig.5 .Irregular clusters. The number of voxels is 15625(25 voxels per dimension).The White voxels are empty.

Here the effectiveness decreases significantly in general. But FGEEA showed the better results. Finally we tested our algorithm with a data set in tridimensional space with an unknown spatial distribution. For $k = 3$ (number or clusters) the algorithm found a solution that is shown in Fig. 5. Here the clusters are irregularlyshaped. These last results were not compared with other clustering algorithms. However. we can see that in principle this approach is feasible.

VI. Conclusions And Future Work

These results allow us to test the feasibility of our algorithm. This is not enough. However, to assume its effectiveness in general. To achieve this proof we require testing with several data sets and applying more solid clustering validation techniques. Computationally, the analysis of the geometric and spatial membership relation between elements or a multidimensional data set is hard. Our approach showed that in principle, membership relations in a data set can be round through or its entropy without an excessive demand on computational resources. Even though the results obtained are limited (since they correspond to particular cases

and in tri-dimensional data) they are promissory. Therefore, future work requires to generalize our method for a data set in n-dimensional space (with *n>3).* to analyze its computational complexity and to test its detailed mathematical formulation. we will report on these issues shortly.

References

[1]. Cha. Sil.: Taxonomy of Nominal Type histogram Distance Measures. Massachusetts (2008)
[2]. Mahalanobis. P.C.: On the generalized distance instatistics(1936)
[3]. Bhattucharyya. A.: On a measure of divergence between two statistical populations defined by probability distributions.Calcutta(1943)
[4]. Pollard. D.E.: A user's guide to measure theoretic probability. Cambridge University Press,Cambridge(2002)
[5]. Yang. G.L.. Le Cam. L.M.: Asymptotics in Statistics: Some Basie Concepts. Springer.Berlin(2000)
[6]. Li. X.. wal, M. Kwong Li. C.: Determining the Optimal Number or Clusters by an Extended RPCL Algorithm. hlong Kong Polytechnic University, thong Kong(l999)
[7]. MacQueen. J.B.: Same Methods for Classification and Analysis or Multivariate Observations. In: Proceedings of 5th Berktey Sysmposium on Mathematical Statises and Probabilit y.? Berklcy. pp. 281—297 (1967)
[8]. Ny. R.. han. J.: Eflecient and Effective Clustering Methods for Spatial Data Mining. Saw tiago de Chile (1994)
[9]. Zhang. T., Raniakrishnman. R.. Linvy. M.: BIRChl: An efficient Method for Very Large Databases. Montreal, Canada (19%)