International Journal of Hybrid Information Technology
Vol.8, No.12 (2015), pp. 225-230
http://dx.doi.org/10.14257/ijhit.2015.8.12.16

A Framework of Granular Computing Clustering Algorithms

Hongbing Liu, Chunhua Liu and Chang-an Wu

School of Computer and Information Technology, Xinyang Normal University, Xinyang 464000, China
liuhbing@126.com

Abstract

A framework of granular computing clustering algorithms is proposed in the paper. Firstly, granules are represented as the normal forms, such the diamond granule in 2-domensional space and hyperdiamond granule in N-dimensional space, sphere granule in 2-dimensional space and hypersphere granule in N-dimensional space. Secondly, operations between two granules are designed to realize the transformation between two spaces with different granularities. Thirdly, the threshold of granularity is used to control the join process between two granules. The performance of granular computing algorithms is evaluated by the experimental results on the data sets selected from machine learning repository.

Keywords: *Granular Computing, hyperdiamond, join Operation*

1. Introduction

Granular computing (GrC) concerns the processing of complex information entities called information granules, which arise in the process of data abstraction and derivation of knowledge from information or data [1, 2]. In the philosophical sense, granular computing can describe a way of thinking that relies on the human ability to recognize the real world under various levels of granularity in order to abstract and consider only those things that serve a specific interest and to switch among different granularities. By focusing on different levels of granularity, one can obtain different levels of knowledge, as well as a greater understanding of the inherent knowledge structure. Granular computing is thus essential in human problem solving and hence has a very significant impact on the design and implementation of intelligent systems, such as classification problems [3-7].

In this paper, we present a framework of granular computing clustering algorithms (GrCC). Firstly, the granule is represented as the normal form. Secondly, the operations $\vee$ and $\wedge$ are introduced to realize the transformation between two granule spaces with different granularities. Thirdly, the threshold of granularity is used to control the operation between two granules. Finally, a framework of granular computing clustering is formed by operation between two granules and the user-defined threshold ρ.

2. A Framework of GrCC

For the data set $S=\{\boldsymbol{x}_i|i=1,2,...,n\}$ in N-dimensional space, GrC algorithm is formed in terms of the following steps. Firstly, the representation method of granule is proposed. Secondly, operations between two granules are designed. Thirdly, the fuzzy inclusion relation between two granules is measured by fuzzy inclusion measure. Finally, the GrC algorithms are designed by operations between two granules.

ISSN: 1738-9968 IJHIT

2.1. Representation of Granules

A granule is represented as a subset of S which is composed by the data with the similar features, and the size of granule is measured by the granularity induced by the maximal distance between data belonging to the same granule. In order to facilitate the study of granular computing, such as the operations between two granules, the granules are represented as the standard form, for example, the granule with the shape of diamond in 2-dimensional space and the shape of hyperdiamond in N-dimensional space.

A granule is represented as the hyperdiamond $\boldsymbol{G}=(\boldsymbol{C},R)$, where $\boldsymbol{C}$ is the center of granule, R is radii of granule, and refers to the granularity of granule $\boldsymbol{G}$ which is measured by the maximal distance between center and the data included in granule. Particularly, a point $\boldsymbol{x}$ is represented by a atomic granule with the center $\boldsymbol{x}$ and granularity 0 in N-dimensional space. The distance between center $\boldsymbol{C}=(c_1,c_2,...,c_N)$ and datum $\boldsymbol{x}=(x_1,x_2,...,x_N)$ can be defined as follows

$$d_1(x,C)=|x_1-c_1|+|x_2-c_2|+...+|x_N-c_N|$$

2.2. Operations between Two Granules

The operations between two granules reflect the transformation between macroscopic and microcosmic of human cognitions. When a person want to observe the object more carefully, the object is partitioned into some suitable sub-objects, namely the universe is transformed into some parts in order to study the object in detail in the view of microscopic. Conversely, there is the same attributes of some objects, we regard the objects as a universe to simple the process in the view of macroscopic. The operations between two granules are designed to realize the transformation between macroscopic and microscopic. Set-based models of granular structures are special cases of lattice-based models, where the lattice join operation $\vee$ coincides with set union operation $\cup$ and lattice meet operation $\wedge$ coincides with set intersection operation $\cap$.

Join operation $\vee$ and meet operation $\wedge$ are used to realize the transformation between macroscopic and microcosmic. Operation $\vee$ unites the granules with small granularities to the granules with the large granularities. Inversely, Operation $\wedge$ divides the granules with large granularities into the granules with small granularities. Join operation $\vee$ and meet operation $\wedge$ are designed as follows.

Any points are regarded as atomic granules which are indivisible, the join process is the key to obtain the larger granules compared with atomic granules. Likewise, the whole space is a granule with the maximal granularity, the meet process produces the smaller granules compared with original granules.

For two hyperdiamond granules $\boldsymbol{G}_1=(\boldsymbol{C}_1, R_1)$ and $\boldsymbol{G}_2=(\boldsymbol{C}_2, R_2)$ in N-dimensional space, the join hyperdiamond granule is

$$\boldsymbol{G}=\boldsymbol{G}_1\vee\boldsymbol{G}_2=(\boldsymbol{C}, R)$$

The center $\boldsymbol{C}$ and the granularity R of $\boldsymbol{G}$ are computed as follows.

Firstly, the vector from $\boldsymbol{C}_1$ to $\boldsymbol{C}_2$ and vector from $\boldsymbol{C}_2$ to $\boldsymbol{C}_1$ are computed. If $\boldsymbol{C}_1=\boldsymbol{C}_2$, then $\boldsymbol{C}_{12}=0$ and $\boldsymbol{C}_{21}=0$. If $\boldsymbol{C}_1\neq\boldsymbol{C}_2$, then $\boldsymbol{C}_{12}=(\boldsymbol{C}_2-\boldsymbol{C}_1)/d(\boldsymbol{C}_1,\boldsymbol{C}_2)$ and $\boldsymbol{C}_{21}=(\boldsymbol{C}_1-\boldsymbol{C}_2)/d(\boldsymbol{C}_2,\boldsymbol{C}_1)$.

Secondly, the crosspoints of $\boldsymbol{G}$ and $\boldsymbol{G}_1$ are $\boldsymbol{P}_1=\boldsymbol{C}_1-\boldsymbol{C}_{12}R_1$ and $\boldsymbol{P}_2 = \boldsymbol{C}_1+\boldsymbol{C}_{12}R_1$. The crosspoints of $\boldsymbol{G}$ and $\boldsymbol{G}_2$ are $\boldsymbol{Q}_1 = \boldsymbol{C}_2-R_2\boldsymbol{C}_{21}$ and $\boldsymbol{Q}_2 = \boldsymbol{C}_2+R_2\boldsymbol{C}_{21}$.

Thirdly, the center C and granularity R of the join hypersphere granule $\boldsymbol{G}$ is computed by algorithm1.

Algorithm1. computing C and R of join hypersphere granule G between G_1 and G_2

Input: $G_1=(C_1,R_1)$ and $G_2=(C_2,R_2)$
Output: $G=(C,R)$
if $R_1>=R_2$
  if d$(C_1,C_2)<=R_1-R_2$ $C=C_1$ $R=R_1$
  else $C=(P_1+Q_1)/2$ $R=d(P_1,Q_1)/2$
  end
else
  if $d(C_1,C_2)<=R_2-R_1$ $C=C_2$ $R=R_2$
  else $C=(P_1+Q_1)/2$ $R=d(P_1,Q_1)/2$
  end
end

Figure 1 shows the join process of the hyperdiamond granule $\boldsymbol{G}_1$ = [0.2 0.15 0.1] and the hyperdiamond granule $\boldsymbol{G}_2$ = [0.1 0.2 0.1]. The crosspoints of hyperdiamond granule $\boldsymbol{G}_1$ and the line crossing vector $\boldsymbol{C}_{12}$=[-0.6667,0.3333] are $\boldsymbol{P}_1$=[0.2667, 0.1167] and $\boldsymbol{P}_2$=[0.1333,0.1833]. The crosspoints of hyperdiamond granule $\boldsymbol{G}_2$ and the line crossing vector $\boldsymbol{C}_{21}$=[0.3333 -0.6667] are $\boldsymbol{Q}_1$=[0.0333,0.23333] and $\boldsymbol{Q}_2$=[0.1667,0.1667]. According to algorithm1, the central vector and granularity of the join hyperdiamond granule $\boldsymbol{G}$ are $\boldsymbol{C}$=[0.15,0.175] and $\boldsymbol{R}$=0.175, namely $\boldsymbol{G}$=[0.15 0.175 0.175].

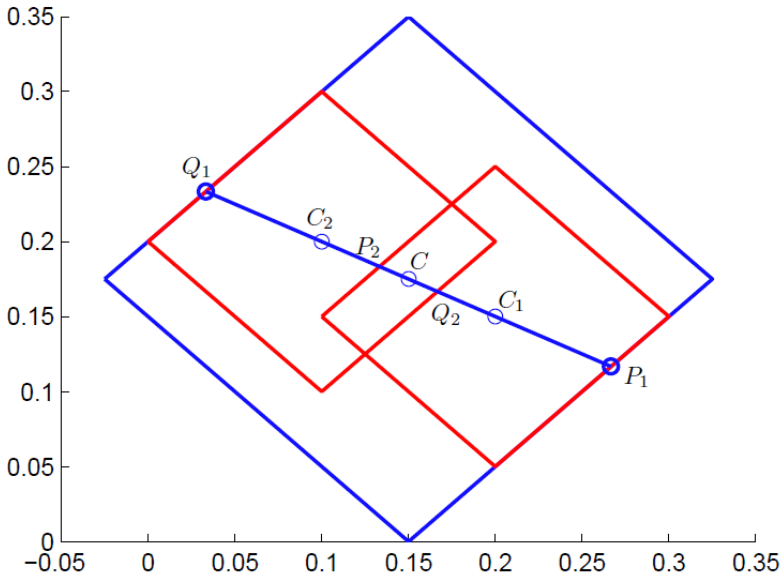

Figure 1. The Join Hyperdiamond Granule of Two Hyperdiamond Granules

2.3. The Framework of GrCC

For data set S, the granular computing clustering algorithms are proposed by the following steps. Firstly, the samples are used to form the atomic granule. Secondly, the threshold of granularity is introduced to conditionally union the atomic granules by the aforementioned join operation, and the granule set is composed of all the join granules. Thirdly, if all atomic granules are included in the granules of GS, the join process is terminated, otherwise, the second process is continued. The GrC clustering algorithms are described as follows.

Suppose the atomic hyperdiamond granules induced by S are g_1, g_2, g_3, g_4, g_5. The training process can be described as the following tree structure shown in Figure6, leafs denote the atomic hyperdiamond granules, root denotes GS including its child nodes G_1, G_2, and g_3. G_1 is induced by join operation of child nodes g_1 and g_2, G_2 is the join hyperdiamond granule of g_4 and g_5, g_3 is the atomic hyperdiamond granule. The whole process of obtaining GS is the bottle up process.

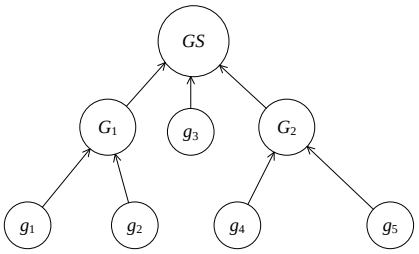

Figure 2. The Framework of GrCC

The GrCC framework is described as algorithm2.

Algorithm2. GrC clustering process
Input: Data set S, threshold ρ of granularity
Output: Granule set GS

S1. initialize the granule set $GS=\varnothing$
S2. $i=1$
S3. for the ith sample $\boldsymbol{x}_i$ in S, form the corresponding atomic granule $\boldsymbol{G}_i$
S4. $j=1$
S5. form the join granule $\boldsymbol{G}_i \vee \boldsymbol{G}_j$ of $\boldsymbol{G}_i$ and $\boldsymbol{G}_j \in GS$, if the granularity of $\boldsymbol{G}_i \vee \boldsymbol{G}_j$ is less than or equal to ρ, then $\boldsymbol{G}_j = \boldsymbol{G}_i \vee \boldsymbol{G}_j$, else
S6. $j=j+1$
S7. if all the granularities of $\boldsymbol{G}_i \vee \boldsymbol{G}_j$ are greater than ρ, then $GS=GS \cup \{G_i\}$
S8. remove $\boldsymbol{x}_i$ until S is empty.

3. Experiments

We verified the feasibility of GrCC by data set selected from UCI benchmark data set. The data set in 2-dimensional space is used to show the data and the achieved granules, and the data sets in N-dimensional space are used to verify the extension of GrCC. We compare GrCC with the traditional clustering algorithms by objective function that is the sum of distances between data and their granule's centers, such as fuzzy c means (FCM) clustering and k-means clustering. All the experiments are performed with an 3.2GHz Intel(R) Core(TM) i5 CPU and 8GB RAM, running Microsoft Windows7 and Matlab2008.

For the selection of parameter ρ of GrCC, we used the stepwise refinement strategy, which made the same cluster number with the cluster number provided by references.

Firstly, the data set named D31 including 31 clusters in [8] is used to verify the clustering feasibility of GrCC. The achieved granules, granule centers, and the cluster centers [8] are shown in Figure 3. We compared GrCC with FCM, Kmeans by objective value (Obj), the sum of squared errors (SSE) and standard deviation (STD) between the achieved cluster center and the clustering provided by [8], the comparisons of GrCC, FCM, and Kmeans are listed in Table 1. From the table we can see, GrCC is better than FCM and Kmeans on the aspects of SSE and STD, we can draw the same conclusion by Figure 4.

Secondly, data sets, such as iris, image, pendigit, optdigits, are selected from machine learning repository to verify the extension of GrCC shown in Table 2. For the data set

optdigits, GrCC is better than FCM because Obj, SSE, and STD of GrCC are less than FCM, GrCC is better than Kmeans in terms of Obj and SSE, and GrCC is worse than Kmeans in terms of STD. So GrCC, FCM, and Kmeans have their own advantages and disadvantages, and are comparative. GrCC, FCM, and Kmeans also are comparative for the data sets pendigits, iris, and image.

Table 1. Comparisons of GrCC, FCM, and Kmeans on Data Set D31

Algorithms	Obj	SSE	STD
GrCC	3169	13.4708	0.2755
FCM	2293.5	13.9843	0.8338
Kmeans	3444.5	14.0707	0.7155

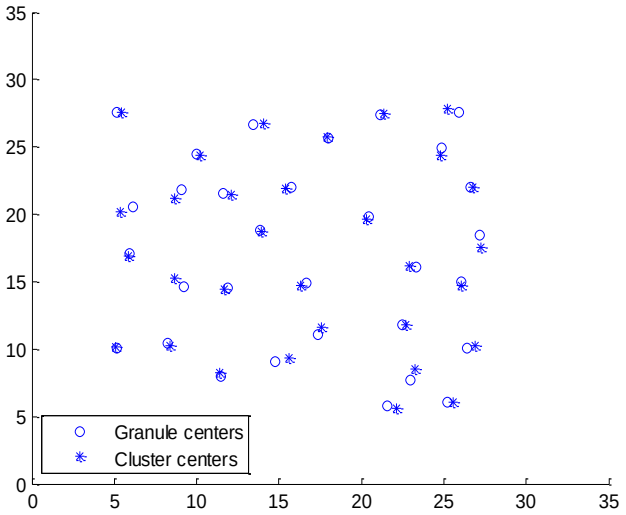

Figure 3. The Granule Centers and Cluster Centers by GrCC for Data Set D31

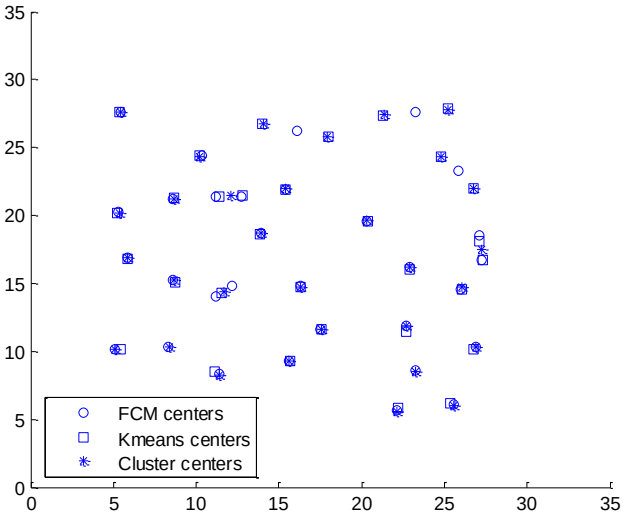

Figure 4. The Cluster Centers of FCM, Kmeans, and GrCC

Table 2. Comparisons of GrCC, FCM, and Kmeans on Data Sets

Data sets	Algorithms	Obj.	SSE	STD
Optdigits	GrCC 240	12066	4805.4	**2.4815**
	FCM	13718	5200.3	4.3909
	Kmeans	**9549.5**	**834.2**	5.7785
Pendigits	GrCC 485	746660	53331	**10.4912**
	FCM	637570	26075	23.5398
	Kmeans	**494280**	**16833**	26.1413
Iris	GrCC 2.7	101.57	0.1922	0.2033
	FCM	**96.93**	**0.0697**	**0.0993**
	Kmeans	97.32	0.1479	0.1709
Image	GrCC 307	17082	19436	**24.3531**
	FCM	13292	**17216**	32.1416
	Kmeans	**12523**	17218	25.7416

4. Conclusions

A framework of granular computing clustering algorithms is proposed in the paper. The granule was represented as the normal form, such as hyperdiamond, hypersphere, and hypercube in N-dimensional space, the join operation was used to generate the granule with larger granularity compared with the original granules, and the threshold of granularity was used to control the join process, the machine learning data sets were used to verify the performance and comparison of GrCC, FCM, and Kmeans. For the future work, the proposed framework of GrCC can be used for image segmentation, image reconstruction, *etc*.

Acknowledgment

This work was supported in part by the Natural Science Foundation of China (Grant no. 61170202, 61202287).

References

[1] L. A. Zadeh, "Fuzzy Sets and Information Granulation", North Holland Publishing, **(1979)**.
[2] L. A. Zadeh, "Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic", Fuzzy Sets and Systems, vol. 90, **(1997)**, pp. 111-127.
[3] V. G. Kaburlasos, I. N. Athanasiadis and P. A. Mitkas, "Fuzzy Lattice Reasoning (FLR) classifier and its application for ambient ozone estimation", International Journal of Approximate Reasoning, vol. 45, no. 1, **(2007)**, pp. 152-188.
[4] V. G. Kaburlasos and S. E. Papadakis, "A granular extension of the Fuzzy-ARTMAP (FAM) neural classifier based on Fuzzy Lattice Reasoning (FLR)", Neurocomputing, vol. 72, no. 10-12, **(2009)**, pp. 2067-2078.
[5] Y. Jamshidi and V. G. Kaburlasos, "gsaINknn: A GSA optimized, lattice computing knn classifier," Engineering Applications of Artificial Intelligence, vol. 35, **(2014)**, pp. 277-285.
[6] V. G. Kaburlasos and T. Pachidis, "A Lattice-Computing ensemble for reasoning based on formal fusion of disparate data types, and an industrial dispensing application", Information Fusion, vol. 16, **(2014)**, pp. 68-83.
[7] G. A. Papakostas, A. G. Hatzimichailidis and V. G. Kaburlasos, "Distance and similarity measures between intuitionistic fuzzy sets: A comparative analysis from a pattern recognition point of view", Pattern Recognition Letters, vol. 34, no. 14, **(2013)**, pp. 1609-1622.
[8] C. J. Veenman, M. J. T. Reinders and E. Backer, "A maximum variance cluster algorithm," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 9, **(2002)**, pp. 1273-1280.

