

Numeric Data to Information Granules and Computing with Words

Witold Pedrycz

Dept. of Electrical and Computer Engineering
 University of Alberta
 Edmonton, AB T6G 2G6
 pedrycz@ee.ualberta.ca

Stuart Rubin

Dept. of Information Surveillance and Reconnaissance
 SSC-PAC
 San Diego, CA, USA
 stuart.rubin@navy.mil

Abstract— The underlying intent of this study is to show how numeric data, fuzzy sets (and information granules, in general) as well as information granules of higher type build a knowledge-based conceptual hierarchy. The bottom-up organization of the paper starts with a concept and selected techniques of data compactification. Compactification is the process, which involves information granulation and in successive phases may give rise to higher type constructs (say, type-2 fuzzy sets, interval-valued fuzzy sets and alike). The detailed algorithmic investigations are provided where we show how membership grades of higher type constructs are formed. In the sequel, we focus on Computing with Words (CW) which in this context is regarded as a general paradigm of processing information granules. We stress the relationships between numeric computing and processing information granules of well-defined semantics (which constitutes the essence of Computing with Words).

Keywords—compactification, computing with words, fuzzy sets, information granules, numeric computing

I. INTRODUCTION AND PROBLEM FORMULATION

Assessing quality of available data, especially in situations when they are significantly scattered and of high dimensionality, becomes crucial for their further usage in a variety of reasoning schemes. The nature of data and their distribution implies different levels of quality of results of inference [8].

The data usually come with some redundancy which is detrimental to most of processing they are involved in. It could be also inconvenient to interpret them considering the size of the data set itself. Taking these aspects into consideration, it could be of interest to represent the whole data set D by a selected subset of elements F where $F \subset D$. While there is a wealth of approaches that exist nowadays, most of them are concerned with some form of averaging meaning that at the end we come up with the elements which have never existed in the original data meaning that they usually may not have any straightforward interpretation. Fuzzy clustering is an example of the detailed framework within which fuzzy clusters are formed [2]. In contrast, if F is a subset of D , the interpretability does not cause difficulties. It is also evident that the choice of the elements of F as well as their number implies the quality of representation of original data D . This set being treated as a “condensation” of D can be a result of a

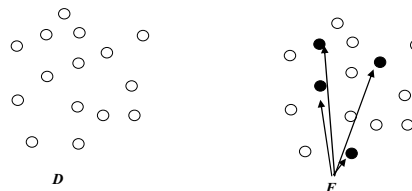
certain optimization. The cardinality of F , which is far lower than the cardinality of D helps alleviate the two problems we have identified at the very beginning.

Let us start with a formal presentation of the problem where we also introduce all required notation. We are provided with a collection of data $D = (x_k, y_k), k=1, 2, \dots, N$ forming an experimental evidence and coming from a certain process or phenomenon. We assume that x_k and y_k are vectors in \mathbf{R}^n and \mathbf{R}^m , respectively. The semantics of x_k and y_k depends on the setting of the problem (and will be exemplified through several examples); in general we can regard y_k to be a certain indicator (output) associated with the given x_k .

Graphically, we can portray the essence of the problem as in Figure 1. The crux of the optimization criterion guiding the construction of F is to represent D by the elements of F to the highest extent; later on we will elaborate on details of the objective function articulating this requirement. Each element of D is expressed via a certain relationship whose “c” arguments ($x_{i1}, x_{i2}, \dots, x_{ic}$) are elements of F , refer also to Figure 1. More specifically, we can describe it concisely as follows

$$\hat{y}_k = \Phi(x_k; x_{i1}, x_{i2}, \dots, x_{ic}) \quad (1)$$

where $k \in N-I$ and we strive for the satisfaction of the relationship $\hat{y}_k = y_k$ which can be achieved through some optimization of the mapping itself as well as a way in which the set F has been constructed [15]. As the form of the mapping stipulates, we are concerned with a certain way of compactification of data.



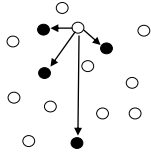


Figure 1. Compactification of data: from original data D to its compact representation F Indicated is a way in which elements of $D-F$ are represented by the content of F .

In what follows, we use some additional notation: let N stands for the set of indexes, $N=\{1, 2, \dots, N\}$ while I be a subset of “c” indexes of N , $I \subset N$, $I=\{i_1, i_2, \dots, i_c\}$ used to denote the elements of F . The structure of the data as presented above could be suitable in a variety of contexts:

- *decision-making processes.* For instance, in assessing terrorist threats we are provided (on a basis of some previous cases or decision scenarios) a collection of characterizations of a threat situation (\mathbf{x}_k) and the associated actions along with their preference (relevance) \mathbf{y}_k , say $\mathbf{y}_k = [0.8 \ 0.4 \ 0.05]$ with actions such as e.g., “enhance surveillance”, “deploy patrol”, “issue warning” [16].
- *prediction* Here \mathbf{x}_k is concerned with a vector of variables describing a certain process at a given time moment (k) while \mathbf{y}_k is a vector of the same variables with the values assumed in the consecutive time moment. The concept can be used in various schemes of learning including those exploited in neural networks [6].
- *classification* In this case \mathbf{x}_k is viewed as a vector of features positioned in the n -dimensional space while \mathbf{y}_k is a Boolean vector of class allocation; in particular for a two-class problem \mathbf{y}_k assumes a single Boolean value.

It is worth noting that a well-known scheme of case-based reasoning (CBR) [4] emerges as one of the general alternatives which take advantage of the format of the data used here. In general, CBR embraces four major processes: (a) retrieving cases from memory that are relevant to a target problem; (b) mapping the solution from the closest (the most similar) retrieved case to the target problem; (c) possible modification of the solution (its adaptation to the target problem); and (d) retaining the solution as a new case in memory. This study shows that the successive phases of processing can be realized and the reasoning results quantified in terms of information granules [17].

One of the problems addressed by this paper is not only that of *quantitative* granularization and its attendant mechanics and algorithmic details, but also touches upon qualitative effects of granulation which relates directly to Computing with Words.

The paper is structured in a bottom-up manner. We start with the formulation of the optimization problem (Section 2); here we clearly identify the main phases of the process of

optimization by distinguishing between parametric and structural enhancements. The structural aspect of optimization is handled by running one of the techniques of evolutionary optimization, namely Particle Swarm Optimization (PSO) [9]. The pertinent discussion is covered in Section 3. Section 4 is concerned with the development of higher-order information granules which are inherently associated with the essence of the compactification process. We show that on a conceptual level the resulting constructs become interval-valued fuzzy sets or type-2 fuzzy sets, in a general setting. Illustrative experiments are reported in Section 6. While those sections are of more detailed nature, in the sequel we build upon these findings and focus on Computing with Words (CW) [19] [20] as a general paradigm of processing information granules.

II. THE OPTIMIZATION PROCESS

Proceeding with the formulation of the problem, there are two essential design tasks involved, that is (a) determination of F , and (b) formation of the prediction (estimation) mechanism of the output part associated with $\mathbf{x}_k \in F$. We start in a bottom-up fashion by considering (b) and assuming that at this phase the set F has been already determined and becomes available to us.

A. Reconstruction and its Underlying Optimization

In the reconstruction procedure, our intent is to express (predict) the conclusion part associated with $\mathbf{x}_k \in F$ in such a way that this prediction $\hat{\mathbf{y}}_k$ is made as close as possible to \mathbf{y}_k [15]. Intuitively \mathbf{y}_k can be expressed on a basis of what is available to us that is a collection of elements $\mathbf{y}_i \in F$. A general view can be expressed in the form of the following aggregation

$$\hat{\mathbf{y}}_k = \sum_{i \in I} u_i(\mathbf{x}_k) \mathbf{y}_i \quad (2)$$

where $u_i(\mathbf{x}_k)$ is sought as a level of activation, closeness, proximity, or relevance of $\mathbf{x}_k \in D-F$ and the i -th element of F . The closer the two elements are, the higher the value of $u_i(\mathbf{x}_k)$ is. In some sense, $u_i(\mathbf{x}_k)$ can be treated as a receptive field constructed around \mathbf{x}_i and in this manner capturing the influence \mathbf{x}_i has on its neighborhood. The closeness is quantified through some distance. Here we may benefit from a variety of ways in which the distance could be expressed. In addition to the commonly encountered distance functions, one can also consider those based on tensor representation of the space, cf. [5]. The receptive fields are adjusted by minimizing the following performance index

$$V = \sum_{i \in I} u_i^p(\mathbf{x}_k) \|\mathbf{x}_k - \mathbf{x}_i\|^2$$

Min V with respect to $\mathbf{x}_i \in I$. (3)

where we assume that $u_i(\mathbf{x}_k) \in [0,1]$ and as usual require that these values sum up to 1 (with the sum taken over all elements of \mathbf{I}).

The extra parameter (p), $p > 1$ helps form a shape of the receptive field by quantifying how much influence \mathbf{x}_i exerts on its neighborhood. If $p \approx 1$ the influence is the most visible whereas with the higher values of “ p ” the impact tempers off quite quickly. Say, for the values of p around 3-4, the receptive field becomes very localized by exhibiting “spikes” positioned at \mathbf{x}_i only with a very fast decline of the values of the field.

Proceeding with the optimization of (3) we convert the optimization with the constraint $\sum_{i \in \mathbf{I}} u_i(\mathbf{x}_k) = 1$ into unconstrained optimization making use of Lagrange multiplier λ . Now the minimized performance reads as

$$V_1 = \sum_{i \in \mathbf{I}} u_i^p(\mathbf{x}_k) \|\mathbf{x}_k - \mathbf{x}_i\|^2 + \lambda \left(\sum_{i \in \mathbf{I}} u_i(\mathbf{x}_k) - 1 \right) \quad (4)$$

By solving the equations $\frac{dV_1}{du_i(\mathbf{x}_k)} = 0$, $i \in \mathbf{I}$ and $\frac{dV_1}{d\lambda} = 0$, we obtain

$$u_i(\mathbf{x}_k) = \frac{1}{\sum_{j \in \mathbf{I}} \left(\frac{\|\mathbf{x}_k - \mathbf{x}_i\|}{\|\mathbf{x}_k - \mathbf{x}_j\|} \right)^{\frac{2}{p-1}}} \quad (5)$$

$i \in \mathbf{I}$, $k \in N - \mathbf{I}$. Turning on to the first problem, the minimization of (3), with the $u_i(\mathbf{x}_k)$ computed as given by (5), depends on the choice of the representative subset of \mathbf{D} , that is

$$Q = \sum_{k \in N - \mathbf{I}} (y_k - \hat{y}_k)^2$$

$$\text{Min}_r Q \quad (6)$$

This is an optimization problem of combinatorial character and as such it requires a very different optimization scheme. Techniques of evolutionary optimization [9] are of interest in this setting. In particular, we consider an algorithmic vehicle of Particle Swarm Optimization (PSO).

B. Particle Swarm Optimization as a Vehicle of Population-Oriented Optimization

A particle swarm is a population of particles – possible solutions in the multidimensional search space [7] [10] [18]. A significant number of applications of the PSO have been reported in the literature, cf. [10]. Each particle explores the search space and during this search it adheres to some quite intuitively appealing guidelines navigating the search process: (a) it tries to follow its previous direction, and (b) it looks back

at the best performance both at the level of the individual particle and the entire population. In this sense, the algorithm exhibits some societal facets as there is some collective search of the problem space along with some component of memory incorporated as an integral part of the search mechanism [14].

The performance of each particle during its movement is assessed by means of some performance index. A position of a swarm in the search space $\mathbf{S} (\subset \mathbf{R}^n)$, is described by some vector $\mathbf{z}(\text{iter}) \in \mathbf{S}$ where “iter” denotes consecutive discrete time moment (iteration). The next position of the particle is governed by the following update expressions concerning the particle, $\mathbf{z}(\text{iter}+1)$ and its speed, $\mathbf{v}(\text{iter}+1)$.

$$\mathbf{z}(\text{iter}+1) = \mathbf{z}(\text{iter}) + \mathbf{v}(\text{iter}+1)$$

//update of position of the particle

$$\mathbf{v}(\text{iter}+1) = \xi \mathbf{v}(\text{iter}) + \phi_1(\mathbf{p} - \mathbf{x}(\text{iter})) + \phi_2(\mathbf{p}_{\text{total}} - \mathbf{x}(\text{iter})) //$$

update of speed of the particle

(7)

where \mathbf{p} denotes the best position (the lowest performance index) reported so far for this particle while $\mathbf{p}_{\text{total}}$ is the best position overall developed so far across the whole population. ϕ_1 and ϕ_2 are random numbers drawn from the uniform distribution $U[0,2]$ defined over $[0,2]$ that help build a proper mix of the components of the speed; here different random numbers affect the individual coordinates of the speed [3] [11] [13]. The second expression governing the change in the velocity of the particle is particularly interesting as it nicely captures the relationships between the particle and its history as well as the history of the overall population in terms of their performance reported so far.

The three components contributing to the modified speed of the particle, namely the current speed $\mathbf{v}(t)$ scaled by the inertial weight (ξ) smaller than 1 (whose role is to quantify resistance to the changes of the current speed), memory of the particle (alluding to the best position achieved so far), and some societal aspect of the optimization scheme expressed by its reliance on the best performance reported across the whole population.

While the PSO scheme is of general nature and independent from the specific optimization problem, the critical to its effective usage is the representation of the search space \mathbf{S} in terms of its components. Let us note that the combinatorial nature of the minimization problem comes with the set of indexes \mathbf{I} which imply a subset of “c” data used to represent the remaining elements. The vector \mathbf{z} comprises “N” entries in $[0,1]$. Its entries are ranked and the first “c” locations return a set of indexes of \mathbf{I} . An illustrative example is shown in Figure 2.

III. INFORMATION GRANULES OF HIGHER ORDER AS CONSTRUCTS OF COMPACTIFICATION

The compactification procedure returns a collection of representatives \mathbf{I} using which we represent all elements in $\mathbf{K} - \mathbf{I}$. Because of the far lower cardinality of \mathbf{F} we could anticipate that its elements regarded as descriptors of all elements in the

far larger set $D-F$, we could capture this representation aspect by reflecting the representation aspect of the elements of F through constructing information granules on a basis of original numeric representation. In more detail, in case of numeric vectors \mathbf{x}_k we form information granules – intervals or more generally fuzzy sets. In case of the corresponding \mathbf{y}_k 's (which could be fuzzy sets to start with), we end up with higher order fuzzy sets and type-2 fuzzy sets, in particular. The overall view of the underlying idea is visualized in Figure 3.

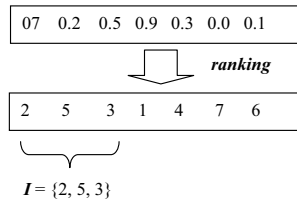


Figure 2. Example of \mathbf{z} , ranking of its entries producing the set of indexes I ; here $N = \{1, 2, \dots, 7, c = 3\}$.

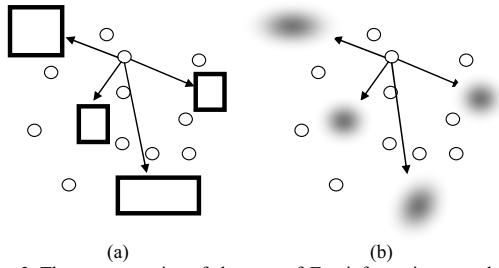


Figure 3. The representation of elements of F as information granules: (a) intervals, and (b) fuzzy sets

The essential task is about a construction of the information granules that is a way in which we can arrive at the characteristic functions or membership functions of the elements of F .

Before proceeding with the detailed algorithmic developments which realize a concept of *justifiable granularity* (where we consider that the granularity of information has to be legitimized by the diversity of the detailed information upon which it is constructed), we recall some organizational aspects. Each element of $D-F$ produces a degree of membership to the i -th element in F which is computed with the use of (4). Consider a fixed element of F , say $i_0 \in I$. The membership degrees associated with it constitute a set $\{u_{i_0}(\mathbf{x}_k)\}$ $k \in N-I$. Next we concentrate on the individual coordinates of \mathbf{x}_{i_0} as well as \mathbf{x}_k , $k \in N-I$. Fix the coordinate: this results in pairs of the form $\{z_k, u_{i_0}(\mathbf{x}_k)\}$ $k \in N-I$. In addition, include the corresponding coordinate of \mathbf{x}_{i_0} , denote it by z_0 , for which the membership degree is equal to 1. Altogether we obtain the set of pairs

$$\{z_k, u_{i_0}(\mathbf{x}_k)\}, k \in N-I, (z_0, 1) \} \quad (8)$$

The essence of the principle of is to quantify the variability existing in a set of available membership degrees in the form of some information granule such as an interval or another fuzzy set.

Given this set of pairs (8), see also Figure 4, we are interested in representing these membership values by spanning an interval $[z_-, z_+]$ around z_0 so that it realizes an intuitively appealing procedure: increase high membership values to 1 and reduce to 0 low membership values. In this sense we form an interval as a suitable information granule capturing the diversity residing within all the pairs (8); refer again to Figure 4. Being more formal, we develop an interval around z_0 whose bounds are expressed as follows.

if $z_i \in [z_-, z_+]$ then elevate to membership grades to 1

if $z_i \notin [z_-, z_+]$ then reduce membership grades to 0

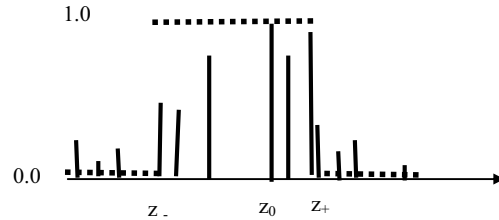


Figure 4. Computing the interval representation of numeric values through the principle of justifiable granularity by optimizing the effect of elevation and suppression of the corresponding membership degrees

The bounds of the interval $[z_-, z_+]$ are subject to optimization with the criterion such that the total changes to membership degrees (being equal either to $1-u_i(\mathbf{x}_k)$ or $u_i(\mathbf{x}_k)$) are made as small as possible. The changes in the values of z_- and z_+ are made in such a way that we minimize the following performance index

$$\text{Min}_{z_1, z_2 \in \mathbb{R}, z_1 \leq z_2} \left\{ \sum_{a_i \in [z_1, z_2]} (1 - u_i(\mathbf{x}_k)) + \sum_{a_i \notin [z_1, z_2]} u_i(\mathbf{x}_k) \right\} \quad (9)$$

The information granule can be expressed as some fuzzy set [14]. In particular, triangular fuzzy sets can be easily formed. As shown in Figure 5 the modal value of the membership function is z_0

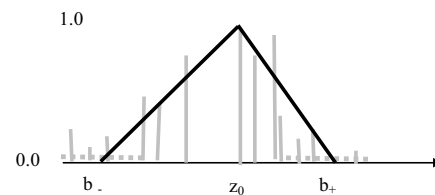


Figure 5. Triangular fuzzy set representing individual membership grades; the cutoff points b_- and b_+ are optimized by running standard linear regression

The optimized fuzzy set is spanned over $[0, 1]$ with the slopes of the membership functions optimized individually for the data positioned below and above z_0 . The standard linear regression applied here returns the parameters b and b_+ in the membership function.

Alluding to Figure 4, we note that depending upon the way in which the principle of justifiable granularity is being realized, the information granules are expressed in the language of set theory (interval analysis to be more precise) or fuzzy numbers.

This construct is fully reflective of the essence of the compactification process: we start with numeric data, reduce their number (in other words, compactify them) but by doing this the resulting subset of D is no longer composed of numeric entities but consists of information granules. The character of these information granules is fully reflective of the level of compactification and the distribution of elements of F around which we span information granules.

Similarly the result formed by the use of the principle of justifiable granularity applied to membership grades comes in the form of some interval or type-2 fuzzy set. In the first case, we form interval-valued fuzzy sets with membership intervals given by $[z, z_+]$. In the second case, we typically end up having triangular fuzzy sets defined in the unit interval.

We can note two essential differences between the compactification procedure established here and vector quantization and data compression as those are encountered in the literature. First, we choose elements from the D rather than constructing representatives (say, prototypes, centroids) as this has been commonly practiced in vector quantization. Second, we develop a granular representation of D in terms of information granules built around the elements of F .

The set of information granules built with the use of elements of F can be quantified with regard to their properties characterizing the granular properties. From this perspective, two fundamental characterizations are worth highlighting:

(a) in case of interval-like nature of information granules built for F , the structural property of the compactification set is quantified in two ways: (i) size of information granules expressed as a sum of “volumes” (vol) of the hyperboxes F_1, F_2, \dots, F_c resulting from spanning them over the elements of F , say $vol(F_1, F_2, \dots, F_c)$. This sum quantifies an overall level of granularity being the effect of the compactification process, (ii) distinguishability of the hyperboxes (and this property could be of direct relevance when dealing with Computing with Words) which states whether any of the two hyperboxes intersect, that is $F_i \cap F_j \neq \emptyset$ for any pair (i, j) .

(b) in case of information granules being represented as fuzzy sets, the distinguishability can be expressed by computing the levels of possibility between F_i and F_j , that is $Poss(F_i, F_j)$. The volume of the fuzzy set is measured by its energy measure of fuzziness. Furthermore as we encounter intermediate membership degrees (which do not exist in set theory), one can consider entropy measure of fuzziness which reflects the cumulative uncertainty in terms of the membership degrees.

IV. COMPUTING WITH WORDS: PROCESSING INFORMATION GRANULES

The paradigm of Computing with Words emphasizes an important and unique feature of processing carried out in this setting: the entities being processed here are information granules [1] of well-defined semantics so that the meaning of such granules is easily comprehended and processed. Hence the starting point of this type of computing is a collection of information granules. This phase is completed by the procedure presented in the previous sections of this study. The general view of the processing and a way in which Computing with Words relates to numeric computing is portrayed in Figure 6.

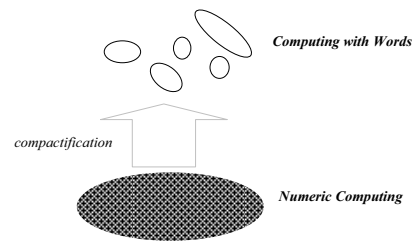


Figure 6. Relationships between Computing with Words and numeric computing: emergence of CW through compactification

There is an important aspect of computational efficiency: while the numeric data are quite voluminous, the collection of information granules (formed around the elements of F) is fairly limited. The compactification plays an important role to support all pursuits of Computing with Words.

	x_k					y_k
0.7	0.9	0.1	0.4	0.2	1.0	
0.1	0.4	0.9	0.5	0.1	0.7	
1.0	0.1	0.7	0.8	0.5	0.2	
0.6	0.8	0.7	0.2	0.6	1.0	
0.6	0.0	0.1	0.2	0.8	0.4	
0.7	0.9	1.0	1.0	0.7	0.3	
1.0	0.2	1.0	0.1	0.8	0.6	
1.0	0.3	0.0	0.4	1.0	0.9	
0.2	0.9	0.1	0.4	0.2	1.0	
0.0	0.5	0.8	0.9	0.9	0.5	

V. EXPERIMENTAL STUDIES

The suite of experiments reported here, which consists of synthetic as well as Machine Learning data (<http://www.ics.uci.edu/~mllearn/MLSummary.html>) is intended to illustrate the performance of the method.

Synthetic data We consider a small data set where $N = 10$ and $n = 3$ (see above). The population size is 20 and the PSO was run for 80 generations. The method was run for several values of “c” ranging in-between 2 and 6. The values of the performance index along with the selected subsets F are shown below.

TABLE I. THE PERFORMANCE INDEX

c	Q	# of data
2	0.298	1,3
3	0.302	1,3,8
4	0.157	1,3,4,6
5	0.157	1,3,4,6,7
6	0.033	1,3,4,5,6,8

It is interesting to note that several data points are the same across all sets F , say data #1, 3, 6 and 6 which points at their stability and sustainable relevance when it comes to the representation of the whole data set.

The proposed approach comes with an important design parameter (p) whose adjustable values could impact the performance of the results. As a matter of fact, this effect is clearly present in the experiments. For $c=2$ and 5 we experimented with the values of “ p ” which produced the results shown in Figure 3.

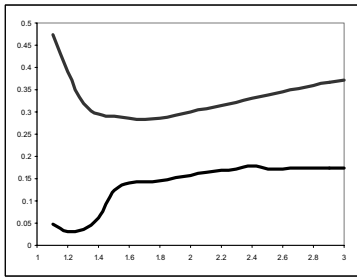


Figure 7. Values of the performance index Q versus p for $c=2$ and 5 (upper and lower curve)

The results presented there are quite convincing. The increase in the number of points used for representation leads to better performance [12]. There are optimal values of “ c ” and they depend upon the cardinality of F . When $c=2$ the optimal value of “ p ” is equal to 1.7. For the larger number of elements in F ($c=5$), the optimal value of “ p ” is lower ($p=1.2$), which points at the very different nature of the receptive fields which in this case are quite extended.

Boston housing Here, we consider the data set coming from the Machine Learning Repository. It consists of 390 13-dimensional data points. A single output is the price of real estate. The parameters of the PSO were set as: size of the population is equal to 100 and the number of generations was set to 150. The compactification was run for F with 5, 10, 15, 20, and 25 elements, respectively. The value of the parameter “ p ” was set to 2.0. The obtained results are shown in Figure 4.

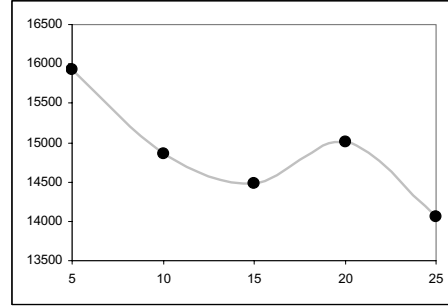


Figure 8. Values of Q versus “ c ”

VI. CONCLUSIONS

In this study, we have showed that the compactification of data sets leads to their representation through some granular representatives. The resulting granularity is inherently implied as an effect of representation of data by a few of their representatives. The nature of information granules itself is reflective of the data and the way the compactification has been completed. This effect is quantified by several associated measures such as volume, non-overlap of the granules, and their entropy. The compactification procedure can be repeated in a hierarchical manner which then leads to the emergence of information granules of higher type (say, type- p fuzzy sets).

We can envision that the intertwined processes of granularization, CW, and Type 2 fuzzy sets all serve to decrease the entropy of a system by way of extracting information granules along some defined metric (including the cases exploiting tensor analysis [5]) space. Indeed, to do so was one of Arbib’s goals in the creation of denotational programming semantics and possibly Hoare’s axiomatic semantics. The reason for the practical failure of these approaches is now clear. They simply do not respect all available contexts in granular formation [11] [14]. Such context may be dynamic, weighted, and context sensitive. Representational formalism is the key. While there is no single general representation [3], this paper has treated grammatical and granular representations of knowledge. These topics definitely warrant future study.

ACKNOWLEDGMENT

The authors gratefully acknowledge the support of SSC-PAC for providing funding for this research under DTRA grant no. BA08MSB008. This paper is approved for public release with an unlimited distribution.

REFERENCES

- [1] A. Bargiela and W. Pedrycz, *Granular Computing: An Introduction*, Kluwer Academic Publishers, Dordrecht, 2003.
- [2] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Functions*, Plenum, N.Y. 1981
- [3] G.J. Chaitin, Randomness and mathematical proof, *Scientific American*, Vol. 232, 1975, 47-52.
- [4] R.O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, John Wiley: NY, 2001.
- [5] C.T.J. Dodson and T. Poston, *Tensor Geometry: The Geometric Viewpoint and its Uses* 2nd ed., New York, NY, Springer-Verlag, 1997.
- [6] R.L. Harvey, *Neural Network Principles*, NJ: Prentice-Hall Inc., 1994.
- [7] J. Kennedy and R.C. Eberhart, 1995. Particle swarm optimization. *Proc. IEEE Int.Conf. on Neural Networks*, IEEE Press, NJ, 1995, vol. 4, 1942–1948.
- [8] J. Murthy and S.H. Rubin, Multi-Sensor fusion for photonic realization, *Proc. 1st Ann. Infor. Integration and Reuse Conf.*, Atlanta, GA, 1999, 79-82.
- [9] D. Gastelum, T. Jones, A. Agarwal, J. Kothari, S. Bhat, H.K. Lee, E. Grant, A. Nelson, S.H. Rubin, and G.K. Lee, The development of a testbed for evolutionary learning algorithms for mobile robotic colonies, *Proc. the 2004 Intern. Symposium on Collaborative Technologies and Systems (CTS'04)*, San Diego, CA, 18-23 Jan., 2004, pp. 212-217.
- [10] S. Paterlini and T. Krink, Differential evolution and particle swarm optimisation in partitional clustering, *Computational Statistics & Data Analysis*, 50, 5, 2006, 1220-1247.
- [11] S.H. Rubin, A Typogenetic application of random seeded crystal learning, *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, Chicago, IL, 1992, 283-289.
- [12] S.H. Rubin (ed.) *Special Issue of ISA Trans. on Artificial Intelligence for Engineering, Design, and Manufacturing*, 1992.
- [13] S.H. Rubin, A Heuristic logic for randomization in fuzzy mining, *J. Control and Intelligent Systems*, 27, 1, 1999, 26-39.
- [14] S.H. Rubin, *3rd World Multiconference on Systemics, Cybernetics, and Informatics (SCI'99)* and *5th International Conference on Information Systems Analysis and Synthesis (ISAS'99)*, Orlando, FL, "The role of computational intelligence in the new millenium," pp. 3-13 (plenary), 31 Jul. 1999.
- [15] S.H. Rubin, S.N.J. Murthy, M.H. Smith, and L. Trajkovic, KASER: knowledge Amplification by Structured Expert Randomization," *IEEE Transactions on Systems, Man, and Cybernetics – Part B: Cybernetics*, 34, 6, 2004, 2317-2329.
- [16] S.H. Rubin, System and *Method for Real-Time Natural Language Type 2 KASER GUI*, Navy Office of Patent Council, received 25 Jan. 2008, disclosed 29 Jan. 2008.
- [17] S.H. Rubin, Knowledge amplification by structured expert randomization – KASERs in SoS design, in *System of Systems – Principles and Applications*, (M. Jamshidi, ed.) Boca Raton: Taylor & Francis Group, 2008.
- [18] F. van den Bergh and A.P. Engelbrecht, A study of particle swarm optimization particle trajectories, *Information Sciences*, 176, 8, 2006, 937-971.
- [19] L.A. Zadeh, From computing with numbers to Computing with Words - From manipulation of measurements to manipulation of perceptions, *IEEE Trans. Circuits and Systems*, 45, 1999, 105-119.
- [20] L. A. Zadeh, Toward a generalized theory of uncertainty (GTU) – an outline, *Information Sciences*, 172, 1-2, 2005, 1-40.