PERGAMON

Pattern Recognition 35 (2002) 825–834

PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/patcog

Feature analysis through information granulation and fuzzy sets

Witold Pedrycz[a,b,*], George Vukovich[c]

[a] *Department of Electrical and Computer Engineering, University of Alberta, Edmonton, Alta., Canada T6G 2G7*
[b] *Systems Research Institute, Polish Academy of Sciences, 01-447 Warsaw, Poland*
[c] *Canadian Space Agency, Spacecraft Engineering 6767 Route de l'Aeroport Saint-Hubert, Que., Canada J3Y 8Y9*

Received 24 August 2000; accepted 20 April 2001

Abstract

Feature analysis and feature selection are fundamental pursuits in pattern recognition. We revisit and generalize an issue of feature selection by introducing a mechanism of soft (fuzzy) feature selection. The underlying idea is to consider features to be granular rather than numeric. By varying the level of granularity, we modify the level of contribution of the specific feature to the overall feature space. We admit an interval model of the features meaning that their values assume a form of numeric intervals. The intervalization of the features exhibits a clear-cut interpretation. Moreover a contribution of the features to the formation of the feature space can be easily controlled: the broader the interval, the less essential contribution of the feature to the entire feature space. In limit, when the intervals get broad enough, one may view the feature to be completely eliminated (dropped) from the feature space. The quantification of the features in terms of their importance is realized in the setting of the clustering FCM model (namely, a process of the binary or fuzzy feature selection is carried out and numerically quantified in the space of membership values generated by fuzzy clusters). As the focal point of this study concerns an interval-like form of information granules, we reveal how such feature intervalization helps approximate fuzzy sets described by any type of membership function. Detailed computations give rise to a detailed quantification of such granular features. Numerical experiments provide a comprehensive numerical illustration of the problem.

Keywords: Feature space; Clustering; Information granularity and information granules; Set approximation of fuzzy sets; Pattern recognition

1. Introduction: fuzzy sets in feature formation and feature selection

Since the very inception of fuzzy sets, their role in pattern recognition has been advocated quite vigorously, cf. Refs. [1–5]. This role has been manifested at the conceptual level as well as materialized in a vast number of specific algorithms of unsupervised and supervised learning [3,15]. There are numerous and comprehensive development environments of fuzzy classifiers. Features and their ensembles forming a feature space become a core of any endeavor of pattern recognition. Fuzzy sets, by their nature, seem to exhibit a primordial impact on the design of pattern classifiers, not only those constructs being fully embedded in the framework of the fuzzy set technology. As commonly envisioned, fuzzy sets are

* Corresponding author. Department of Electrical and Computer Engineering, University of Alberta, Edmonton, Alta., Canada T6G 2G7. Tel.: +1-780-492-4661; fax: +1-780-492-1811.

E-mail address: pedrycz@ee.ualberta.ca (W. Pedrycz).

0031-3203/02/$22.00
PII: S0031-3203(01)00102-9

perceived as linguistic granules that spread over the domains of the variables (features) encountered in the problem at hand. The respective membership functions are formed in order to focus attention on some regions of the universe of discourse that are perceived of importance to the classification task. In other words, they may eventually facilitate the learning and result in a better performance of the constructed classifier. Computationally, fuzzy sets introduce nonlinearity to the classification problem by *nonlinearly* transforming (normalizing) the original space. In other words, instead of the original space $\boldsymbol{X}$ (that may eventually call for further normalization), we develop nonlinear terms (fuzzy sets), say A, B, C defined in $\boldsymbol{X}$. The ensuing classifier "perceives" the environment through these fuzzy set constraints (namely, each pattern manifests through the respective membership grades of $A(x), B(x)$, and $C(x)$, respectively). This form of the nonlinear transformation has a lot in common with receptive fields used in radial basis function neural networks. In this role, fuzzy sets form a preprocessing module that is followed by the computationally intensive architecture (viz. a multilayer neural network, inference scheme of approximate reasoning and others). Even in this simple example it becomes apparent that by introducing fuzzy sets we tend to increase the dimensionality of the classification problem. More specifically, one variable (feature) has been expanded to three features (that is A, B, and C). The dimensionality expansion may not be acceptable, in particular if we admit a larger number of the linguistic terms for each original variable. The increase is still linear, that is the number of new features is n' where $n' = p^* n$, with "p" being the number of the linguistic terms—fuzzy sets (assuming that we use the same number of the linguistic terms for each variable). Our expectations are that the increase of dimensionality is compensated by some tangible advantages at the learning side and the overall performance of the classifier.

Fuzzy clustering, especially FCM, has occupied a dominant role as an efficient vehicle of information granulation (that is building fuzzy sets) [1,3,6]. FCM helps combat the curse of dimensionality in the classification problem by developing fuzzy *relations* rather than fuzzy *sets*. This is obvious that as all variables are involved in the clustering process at once, the number of new features is equal to the number of the clusters being generated (say, c). Obviously, as "c" is independent from the dimensionality of the original space ("n"), thus we may encounter also an effect of dimensionality reduction. This type of feature formation exhibits two important aspects that are worth underlining:

- first, the linguistic granules are developed (at least to significant extent) based upon the available patterns (data) so the statistical characteristics of the experimental data can be properly captured.
- second, all original variables are taken into account at the same time—the resulting constructs are fuzzy relations rather than fuzzy sets. This helps us take into consideration interrelationships occurring within the patterns.

Feature selection has been a cornerstone of pattern recognition, see Refs. [7–9]. We can envision a lot of fundamental results arising in the realm of statistical classification techniques. The reduction of the feature space, viz. an elimination of features that tend to be less "informative" (that is less discriminative) and the ensuing determination of the best subset of features is a computationally intensive problem. The enumeration, brute-force approach will not work in the case of classification problems of higher dimensionality. The proper performance index guiding this selection is another problem to deal with. Various techniques were studied including those confined to neurocomputing , cf. Refs. [10,11] and exploiting discrete optimization techniques, see e.g., Refs. [12,13]. Interestingly enough, not so much can be found as far as feature selection is concerned in the setting of fuzzy pattern recognition; one can refer to Refs. [3,6,14,15] that tackle this matter to some extent.

The objectives of this study are threefold:

- First, to formulate a problem of feature selection both in its binary (two-valued) and soft (fuzzy) version. Both versions are embedded in a cluster-driven environment.
- Second, to show how feature intervalization gives rise to the fuzzy version of the problem of feature selection. Our conjecture is that feature granulation is closely tied with the issue of partial elimination of variables (features).
- Third, to quantify an effect of feature granulation (intervalization) in terms of feature selection. In particular, we are interested in studying a correlation between binary and fuzzy feature selection and its ensuing quantification governed by the granularity of the features.

Following the identified thrust of the study, the paper is organized in the following manner. In Section 2, we formulate the problem in the space of membership values (that is the space being generated through fuzzy clustering). Section 3 concentrates on soft (fuzzy) feature selection and elaborates on the pertinent computational details. In particular, set-based approximation of fuzzy sets is discussed. Numerical studies are included in Section 4.

From now on, we confine ourselves to the following notation. The set of patterns to be clustered is located in the n-dimensional space of reals, $\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\}$

where $x_k \in R^n$. The number of clusters is equal to "c". The distance function $||.||$ used here is a weighted Euclidean distance where each coordinate (feature) of the pattern is normalized by its standard deviation, namely

$$||a - b|| = \sum_{i=1}^{n} \frac{(a_i - b_i)^2}{\sigma_i^2}.$$

Both a and b are the vectors (patterns) in R^n. Here σ_i is a standard deviation of the ith feature. This type of the weighted distance implies that all features exhibit a similar impact when expressing similarity between the patterns. The key outcomes of the clustering arise in the form of the prototypes (centroids) of the clusters $v_1, v_2, \ldots, v_c$ (essentially given the prototypes, we can easily "reconstruct" all membership values contained in the partition matrix).

Furthermore, a standard FCM algorithm will be used throughout the study; the algorithmic details are not reported here as those are widely available in the existing literature, see e.g., Ref. [3], the classic reference in this area.

2. Feature selection in the space of membership values: a selection criterion

Membership functions of the fuzzy clusters (fuzzy relations) reflect the very nature of the original feature space. In other words, our conjecture is that when we traverse the original feature space X (that is, as mentioned, an n-dimensional Euclidean space of reals, R^n), this movement becomes fully reflected in the corresponding move occurring in the unit hypercube of the membership grades, that is $[0,1]^c$. The classification regions of any classifier are also formed based on the membership values. We may expect that an elimination of some coordinates of R^n (thus a reduction of the original feature space), will become fully reflected in the departure from the original membership values. Denote by I the set of indices (numbers of features) contained in $N = \{1, 2, \ldots, n\}$ that is

$$I = \{i_1, i_2, \ldots, i_m\}$$

We use the notation $I \subset N$ to summarize a certain subset of the overall feature set

$$j \in I \overset{\text{def}}{=} \text{feature "}j\text{" is included in subset } I.$$

Subsequently denote by $u(x_k)$ a c-dimensional vector of the membership grades produced by the clustering algorithm. Similarly, by $u(x_k, I)$ we express a c-dimensional vector of membership grades computed for the features contributing to I. More specifically, we have

$$u_i(x_k, I) = \frac{1}{\sum_{j=1}^{c} (||x_k - v_i||_I / ||x_k - v_j||_I)^{2/p-1}}$$

with $p > 1$. Moreover $||.||_I$ denotes the distance function restricted to the features in I and computed as follows:

$$||a - b||_I = \sum_{l \in I} \frac{(a_l - b_l)^2}{\sigma_l^2}.$$

Let us recall that the original membership grades are calculated based on the well-known expression [3]

$$u_i(x_k) = \frac{1}{\sum_{j=1}^{c} (||x_k - v_i|| / ||x_k - v_j||)^{2/p-1}}$$

(to maintain uniformity, we could have alternatively used the distance notation of the form $||.||_N$ for all the features exploited in the respective membership computations).

Now any combination of the features I can be evaluated by expressing how far $u(x_k, I)$ differs from the original $u(x_k)$. Again, a simple Euclidean distance could be used here. A sum of these distances over all patterns gives rise to the expression

$$Q(I) = \sum_{k=1}^{N} \sum_{i=1}^{c} (u_i(x_k, I) - u_i(x_k))^2.$$

The higher the value of this performance index, the more essential is the combination of the features removed from N that is the features contained in $N \setminus I$. Obviously, $Q(N)$ is a boundary condition stating that no features have been eliminated; hence $Q(N)$ yields a zero value. In general, there is no monotonicity condition satisfied namely the statement

$$\text{if } I \subset I' \text{ then } Q(I) > Q(I')$$

may not hold in general. This observation does not seem to be very surprising: we may anticipate that an optimal subset of features may not be the largest one that is left.

3. Uncertainty in feature description and soft feature selection

The feature selection procedure we have discussed so far is a standard one: a certain feature is either in (becomes an element of I) or out (that is included in $N \setminus I$). That is we are concerned with a Boolean (two-valued) feature selection. An interesting generalization of this version of the selection problem can be referred to as a soft (fuzzy) feature selection. The idea is not to drop a given feature but to granulate it, viz. admit its nonnumeric values, especially intervals. This *intervalization* of the feature will give rise to a soft character of feature selection. Intuitively, when the interval describing the

value of the feature gets broader, the feature contributes to a lesser extent to the modified feature space yet the abrupt "in–out" character of the Boolean reduction is not present.

3.1. Detailed computations of the membership functions in clusters

We start with a one-dimensional case ($\boldsymbol{X}=\boldsymbol{R}$) as this scenario becomes the most tangible and easy to visualize and interpret. Given the set of prototypes $\{v_i\}$, $i=1,2,\ldots,c$, the membership values (viz. the resulting fuzzy partition) are computed in the following way:

$$u_i(x)=\frac{1}{\sum_{j=1}^{c}(x-v_i/x-v_j)^2} \tag{1}$$

(here the fuzzification factor "p" was set to 2). By sweeping the value of "x" throughout the entire space $\boldsymbol{X}$, the plots of the membership functions are instantaneously constructed. Now let us consider that the nonnumeric (granular) input X regarded as an interval, namely $X=[a,b]$ is taken into consideration. The calculations may follow (1) yet the determination of the distance function has to be revisited. Let us introduce the following definition, see also Fig. 1.

$$||X-v|| \overset{\text{def}}{=} \begin{cases} 0, \text{ if } v\in[a,b], \\ \min(|a-v|,|b-v|). \end{cases} \tag{2}$$

In limit, when $X=\boldsymbol{R}$ the distance evidently, $|X-v|$ is equal identically to zero.

This distance (2) is used in the computations of the membership functions. The corresponding calculations are carried out in the following way:

$$u_i(X)=\frac{1}{\sum_{j=1}^{c}(||X-v_i||/||X-v_j||)^2}. \tag{3}$$

The above calculations require some modifications when the distance between X and v_i becomes equal to zero (this is obviously a non-issue in the case of numeric inputs $X=\{x\}$). To overcome this deficiency, we redefine (3) by accepting $u_i(X)$ to be equal 1 under such circumstances,

$$u_i(X)=\begin{cases} 1, \text{ if } |X-v_i|=0, \\ (3), \text{ otherwise.} \end{cases} \tag{4}$$

Note that as $u_i(x)$ is equal to one, the remaining membership values (for v_j, $j=1,2,\ldots,\mathrm{c}$, $i\neq j$) are forced to be set up to zero.

The interval-valued feature X captures the effect of uncertainty and it can be expressed in the form $[x-\delta,x+\delta]$ that is an interval centered around "x" being of a length of 2δ. The length expressed in this explicit manner helps quantify the factor of uncertainty. Using this model, we calculate the membership values (as indicated by (4)) for selected values of δ. The plots of these membership functions are shown in Fig. 2. Noticeably, once the values of δ go up, the membership functions assume wider flat regions that become distributed around the prototypes. They get sharper and more "localized" for lower values of δ.

NB. One can eventually drop this unity constraint; by doing so we are in line with the possibilistic clustering. It is noticeable that the granular input implies a certain departure from the commonly accepted constraint and the relationship between the size of the granules and the violation of the constraint can be easily quantified. We will not be pursuing this issue as being somewhat marginal to the main vein of the topic.

The above finding is extended to the multivariable case by studying the n-dimensional feature vector. First, we introduce a concise notation to capture an effect of uncertainty (granulation) of the features. Introduce a vector of uncertainties Δ

$$\Delta^{\mathrm{T}}=[\delta_1 \quad \delta_2 \quad \ldots \quad \delta_n],$$

where the ith feature of $\boldsymbol{x}_k$ associated with the uncertainty factor that is quantified as an interval $[x_{ki}-\delta_i,\ x_{ki}+\delta_i]$. We use the notation $u(\boldsymbol{x},\Delta)$ to express the effect of uncertainty across all features. The distance function is a generalization of that given by Eq. (2), namely the calculations are completed coordinatewise and the results are summed up.

If δ_i increases, then this effectively gives rise to the reduction of the feature as the expression $(X_i-v_i)^2$ attains zero. The performance index describing the fuzzy reduction of the feature space is formulated in the form

$$Q(\Delta)=\sum_{k=1}^{N}\sum_{i=1}^{c}(u_i(\boldsymbol{x}_k,\Delta)-u_i(\boldsymbol{x}_k))^2.$$

3.2. Set approximation of fuzzy sets

The above discussion was confined to the granular data represented as intervals (or hypercubes). The reason was evident: all calculations were simple. The use of fuzzy sets (X) instead would add a lot of computational burden that may not be fully legitimized. If a fuzzy set were encountered, it would be advisable to convert (approximate) it by a set and use such an approximation afterwards. Fortunately, this approximation is obvious and intuitively appealing. The main finding can be formulated as follows

Proposition. *Consider a unimodal normal fuzzy set A defined in* $\boldsymbol{R}$ *with a continuous membership function.*

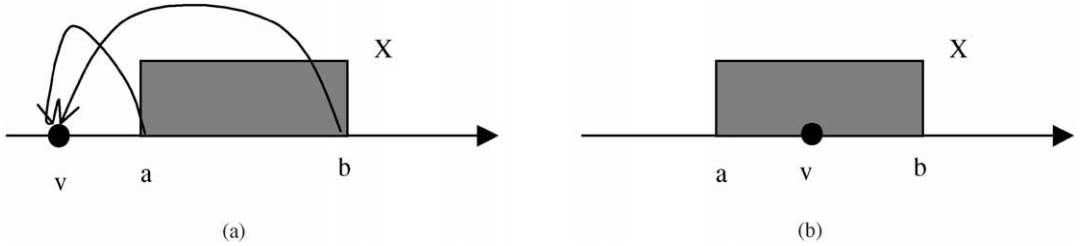

Fig. 1. Computing distance between numeric entity (v) and a granular (interval-valued) quantity X: v outside X (a) X covers v (b).

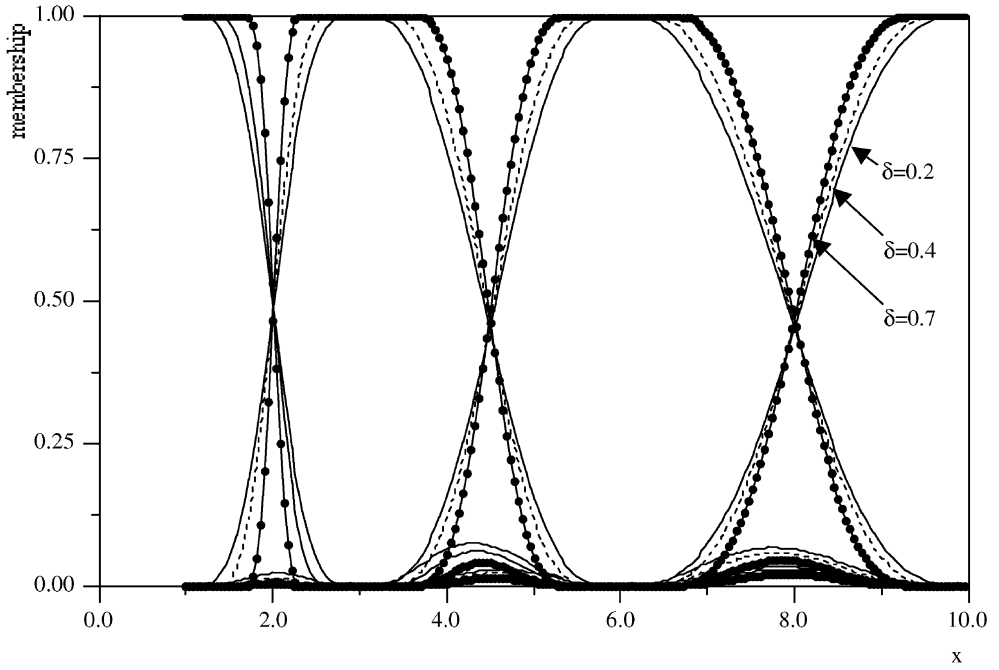

Fig. 2. Membership functions for selected values of δ.

Its best approximation (*in the sense of the Minkowski distance*) *is a set* A^* *with the characteristic function* (*that is an* 1/2*-cut of* A)

$$A^*(x) = A_{1/2}(x).$$

Proof. Let us consider the performance index Q expressing a distance between A and A^*

$$Q(\alpha) = \int_a^b |A(z) - A^*(z)|^p \, \mathrm{d}z. \tag{5}$$

The power (p) with $p > 1$ standing in the performance index gives rise to the Minkowski distance between the fuzzy set and its approximation. Bearing in mind the unimodality of the fuzzy set (which is quite general, anyway), we rewrite Eq. (5) in the form of the series of integrals

$$Q(\alpha) = \int_a^{x_0} A^p(z) \, \mathrm{d}z + \int_{x_0}^{m} (1 - A(z))^p \, \mathrm{d}z + \int_m^{y_0} (1 - A(z))^p \, \mathrm{d}z + \int_{y_0}^{b} A^p(z) \, \mathrm{d}z.$$

Note that the optimal threshold level (α) identifies two elements in the universe of discourse $\boldsymbol{X}$, say x_0 and y_0 (as seen in Fig. 3) and being already used in the above formula. That is, we have $A(x_0) = \alpha$, $A(y_0) = \alpha$.

The optimization of Q carried out with respect to α is equivalent to the optimization of Q with respect to x_0 and y_0 meaning that

$$\text{Min } Q(\alpha) = \text{Min } Q(x_0, y_0).$$

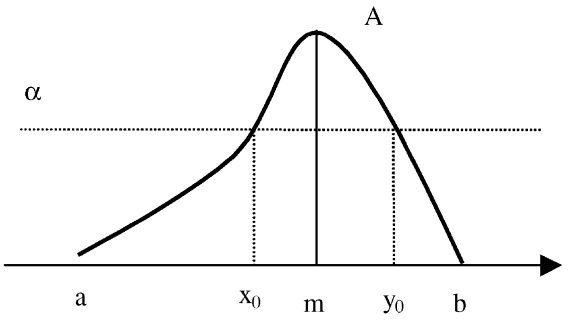

Fig. 3. Approximating fuzzy set (A) by A_α through α-cut optimization.

The necessary conditions leading to the minimum of Q read as

$$\frac{\partial Q}{\partial x_0} = A^p(x_0) - (1 - A(x_0))^p = 0 \tag{6}$$

and

$$\frac{\partial Q}{\partial y_0} = (1 - A(y_0))^p - A(y_0)^p = 0. \tag{7}$$

Let us derive Eq. (6) step by step (the derivations of Eq. (7) are carried out in an analogous manner). First, we have two evident relationships

$$\frac{\mathrm{d}}{\mathrm{d}x_0}\int_a^{x_0} F(z)\,\mathrm{d}z = F(x_0)$$

and

$$\frac{\mathrm{d}}{\mathrm{d}x_0}\int_{x_0}^{c} F(z)\,\mathrm{d}z = -F(x_0)$$

that hold for any function F for which the above integrals make sense. The use of these expressions in computing the derivative of Q leads us to Eq. (6), namely

$$\frac{\mathrm{d}Q}{\mathrm{d}x_0} = \frac{\mathrm{d}}{\mathrm{d}x_0}\int_a^{x_0} A^p(z)\,\mathrm{d}z + \frac{\mathrm{d}}{\mathrm{d}x_0}\int_{x_0}^{m} (1 - A(z))^p\,\mathrm{d}z$$
$$= A^p(x_0) - (1 - A(x_0))^p.$$

Solving Eq. (6) with respect to x_0 we get $A(x_0) - 1 + A(x_0) = 0$. This leads to $A(x_0) = 1/2$ viz. x_0 is a point where the membership function attains $\alpha = 1/2$. Similarly, we handle the second equation (7) which leads to the same result as before, namely $A(y_0) = 1/2$.

Interestingly, the threshold (that is equal to 1/2—an intuitively appealing finding) does not depend on the form of the membership function itself. The generality of the finding (that is, anyway, quite intuitively appealing) clearly points out how fuzzy sets can be converted into sets. An important observation is that not all fuzzy sets are equally easy to approximate: the higher the performance index $Q(1/2)$, the more *difficult* is to approximate the fuzzy set under discussion and more questionable this approximation is. In general, it is advisable to consider the value of the performance index along with the resulting set approximation of the fuzzy set under discussion. As an example, let us discuss how a triangular fuzzy set is approximated, see Fig. 4.

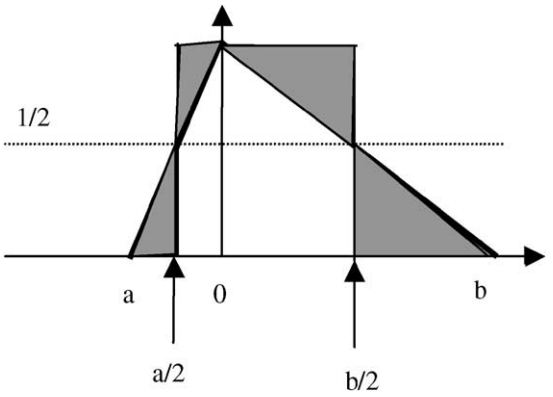

Fig. 4. Triangular fuzzy set and its set-based approximation. The shadowed regions quantify the approximation error.

Put also $p = 1$ (so we are concerned with the Hamming distance). The approximation error is visualized in the form of four triangular regions which, in total, amounts to

$$Q = 1/4(a + b).$$

This error is a linear function of the lower and upper bound of the fuzzy set. We can rewrite Q in the form underlining its relation with the support of A

$$Q = 1/4\ \mathrm{supp}(A) + 1/2a.$$

Obviously, the support of A, $\mathrm{supp}(A)$, is equal to $b - a$. □

4. Experimental studies

The experiment uses one of the datasets available at the UCI at Irvine [16]. It concerns a part of the Boston housing data and consists of 250 8-dimensional patterns (we reduced the size of the data to visualize all results). First, the clustering was completed with the use of the standard FCM with the fuzzification factor equal to 2 and $c = 7$ clusters. The prototypes are summarized in Table 1. The distance function $||.||$ guiding the clustering mechanism is the weighted (normalized) Euclidean distance (that is each feature is normalized by dividing its value by the corresponding standard deviation).

Next, all combinations ($2^8 = 256$) of the features are investigated. These combinations are coded in binary. For instance, the vector 0 0 0 0 0 0 0 1 identifies a set of

Table 1
Prototypes of the clusters distributed in the 8-dimensional feature space

Cluster no.	Prototype							
1	0.8116	3.9122	14.2760	0.0897	0.5916	6.1054	84.7475	2.9705
2	0.4682	10.1878	9.4311	0.0838	0.5210	6.3177	67.2972	4.0732
3	0.2922	17.2803	7.2518	0.0539	0.4837	6.4615	53.7833	4.8058
4	0.5323	8.6016	10.1900	0.0900	0.5332	6.2723	71.2547	3.8639
5	0.3122	16.0072	7.5226	0.0580	0.4885	6.4423	55.6263	4.6952
6	0.5889	7.3481	10.8828	0.0930	0.5441	6.2329	74.5271	3.6809
6	0.3617	13.5992	8.1571	0.0676	0.4996	6.3987	59.7660	4.4633

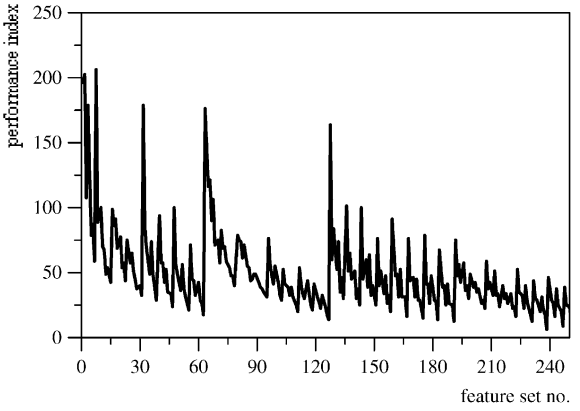

Fig. 5. The values of the performance index Q for successive feature sets.

features having only a single element (the last feature). Its decimal equivalent is one. The values of the performance index Q for all the combinations (the x-axis includes a decimal equivalent of the collection of the features) are summarized in Fig. 5.

This figure shows clearly that there are a lot of differences in performance of the different collections of the features. Evidently, each set of features has to be decoded by transforming the integer number into a binary string. The best results (collections of the features) are identified by the binary string 1 1 1 0 1 1 1 1 leading to $Q = 6.0253$ and 1 1 1 1 0 1 1 1 with the performance index equal to 8.2682. This means that all but one feature (either 3 or 4) give rise to the best results. The spikes in Fig. 5 correspond to a single-element feature set which, as could have been anticipated, contributes to a poor performance. An excerpt from the entire list of the combinations of the features is given in Table 2.

There is another way of visualizing the results. Instead of analyzing each combination of the features, it is of interest to investigate how the number of features affects the values of the performance index. In general, one may anticipate that the larger number of the feature set may lead to the lower values of the performance index.

Table 2
Selected combinations of features along with their performance index

Feature set	Performance index	Feature set	Performance index
1 1 1 1 0 0 0 0	45.5831	1 1 1 1 0 1 1 1	8.2682
1 1 1 0 1 1 1 1	6.0253	1 1 1 1 0 1 1 0	18.8382
1 1 1 0 1 1 1 0	17.4416	1 1 1 1 0 1 0 1	24.0651
1 1 1 0 1 1 0 1	20.7247	1 1 1 1 0 1 0 0	36.8041
1 1 1 0 1 1 0 0	31.2429	1 1 1 1 0 0 1 1	15.4062
1 1 1 0 1 0 1 1	19.5656	1 1 1 1 0 0 1 0	25.1947
1 1 1 0 1 0 0 1	29.3116	1 1 1 0 0 1 1 0	22.0126
1 1 1 0 1 0 0 0	42.6012	1 1 1 0 0 1 0 1	26.8939
1 1 1 0 0 1 1 1	12.1960	1 1 1 0 0 1 0 0	39.5637
1 1 1 0 0 0 1 1	21.8574	1 1 1 0 0 0 1 0	31.3970
1 1 1 0 0 0 0 1	34.0258	1 1 1 0 0 0 0 0	51.4628
1 1 1 0 1 0 1 0	28.5243	1 1 1 1 0 0 0 1	28.1017

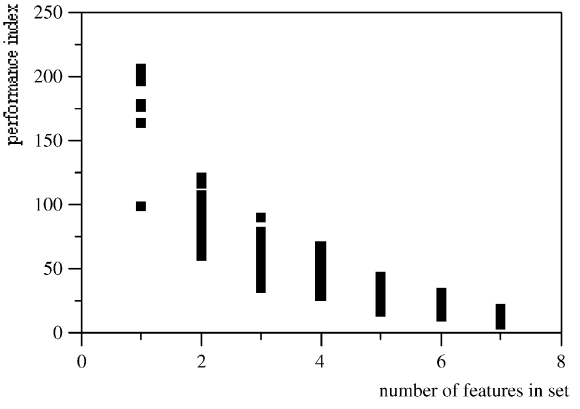

Fig. 6. The values of the performance index versus the number of features in the feature set.

Obviously, this is not a monotonic relationship. This is clearly summarized in Fig. 6. It is noticeable that some combinations that involve a smaller number of the features perform better over some more numerous collections of the features.

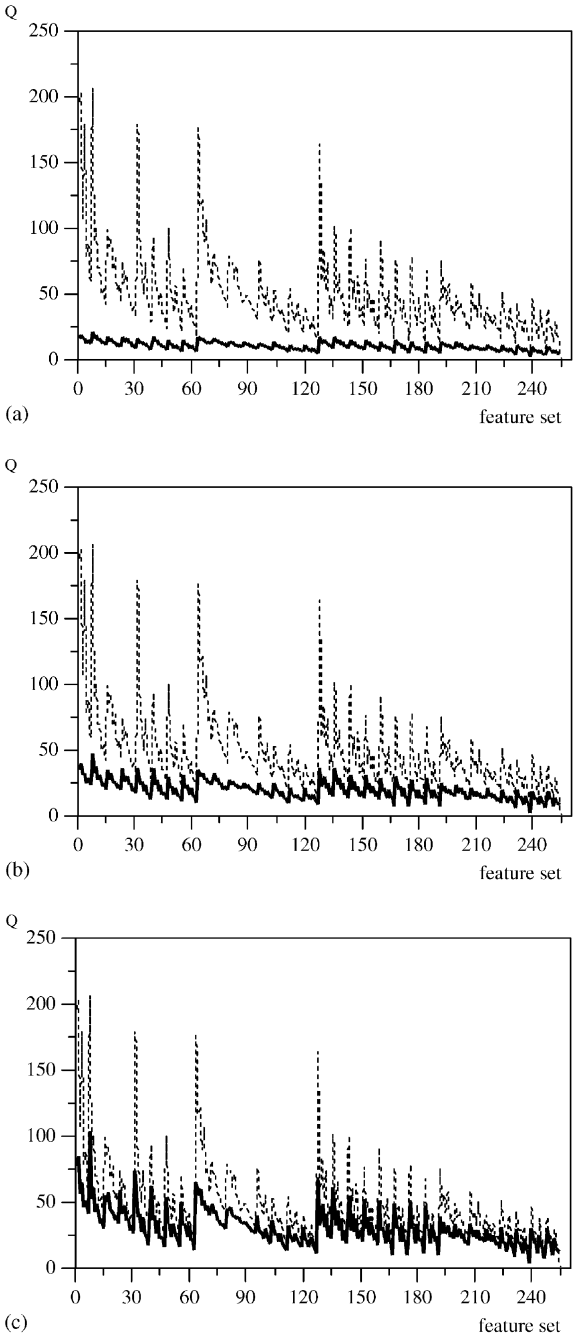

Fig. 7. Performance index Q for the Boolean feature reduction (elimination) indicated by a dotted line and their granulation (shown by a solid line) for selected values of δ: (a) $\delta = 0.2$, (b) $\delta = 0.4$, (c) $\delta = 0.7$.

The series of figures, Fig. 7, shows an effect of soft feature elimination that is accepting granular features instead of using numeric features. The experiment was completed for three selected sizes of the information granules. To maintain a consistency of the experiment that will help us work out a comparative analysis, the granular input is formed by an interval $[a, b]$ centered around an original numeric input "x". The size of the information granule itself is taken proportionally to the standard deviation of this particular feature. Thus

$$X = [x - \delta\sigma, x + \delta\sigma],$$

where $\delta \in [0, 1]$ controls the size of the interval and σ denotes the standard deviation of the respective feature. The experiments were conducted for $\delta = 0.2, 0.4$, and 0.7, see Fig. 7. Instead of eliminating the feature, we replace it by the granular version. The results show up a clear tendency: the performance of the granular features (and the associated combinations) follows the results reported for the subsets of features (viz. the features being eliminated). All figures include also the results of the Boolean feature reduction. When δ increases, the results become very similar meaning that there is a certain point where it does not matter if the feature has been eliminated or granulated (intervalized).

There is another way of looking into the same results by plotting the values of the performance index for the Boolean and fuzzy feature elimination. This gives us a certain insight into strength of correlation between the values of such performance indexes. Refer to Fig. 8.

For higher values of δ, say $\delta = 3.0$, one can observe almost a strong correlation (the points become distributed along a straight line) with the correlation coefficient being equal to 1.

5. Conclusions

In this study, we have investigated an issue of granular information in feature selection. It is shown that granular features (as opposed to their numeric counterparts) give rise to a generalization of the two-valued mechanism of feature selection providing with its continuous version. The soft (fuzzy) form of feature selection sheds light on some essential links between the granularity of the variables (features) and its "size" (expressed with the aid of the width of the numeric interval) as well as an impact of it on the relevance of such information granules.

The algorithmic fabric of the investigations is based on the well-known FCM algorithm. We have discussed a way of quantifying the effect of granular data on the membership functions. The proposed set-based approximation of fuzzy sets helps reduce computational burden of the feature selection procedure. The fundamental result obtained in this setting is very general (the existence of the optimal value of the α-cut) and this becomes instrumental in better understanding of the essence and quality of approximation delivered by the set theory.

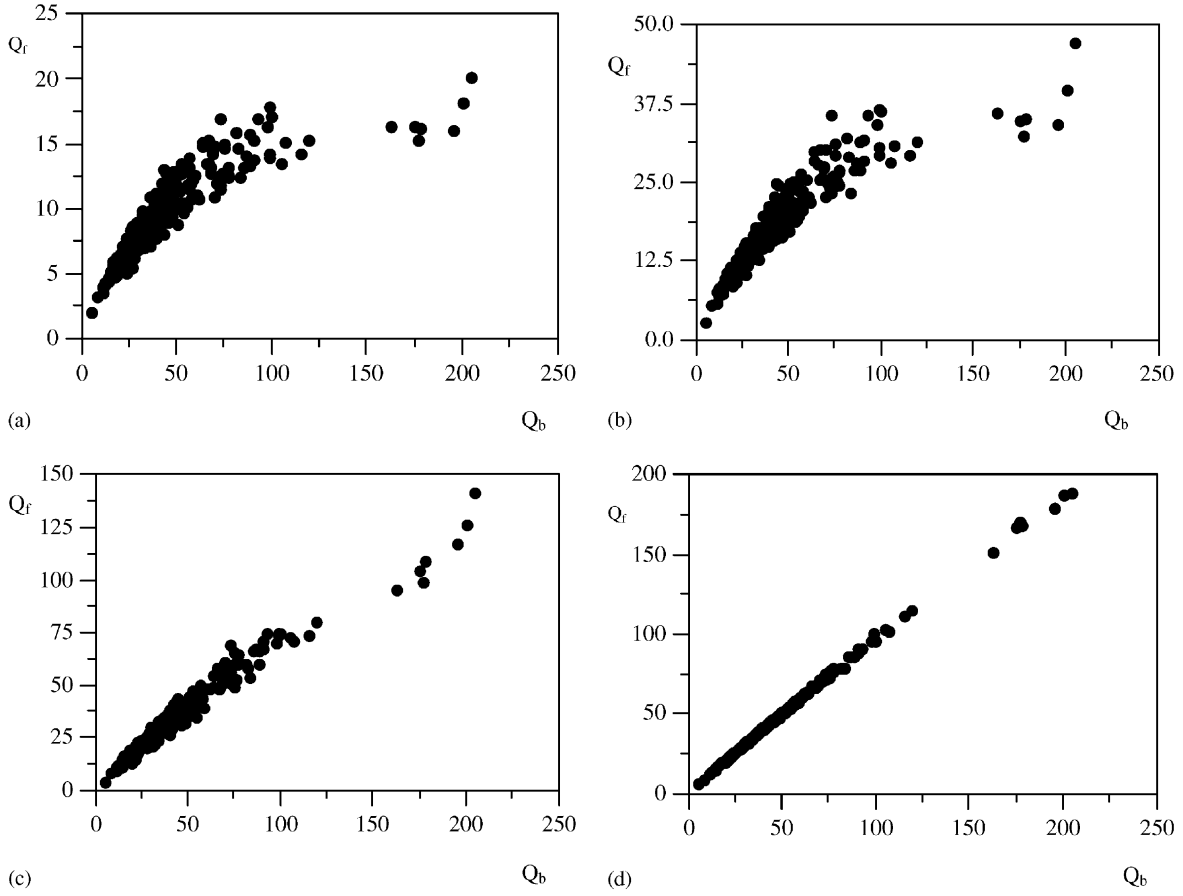

Fig. 8. Performance index of the fuzzy (Q_f) and Boolean (Q_b) feature selection for selected valued of δ: (a) $\delta = 0.2$, (b) $\delta = 0.4$ (c) $\delta = 1.0$, (e) $\delta = 3.0$.

Interestingly, granular information can be effectively used to incorporate our knowledge about a lack of precision concerning a problem at hand. For instance, in Ref. [17] it has been shown that dynamic systems with unknown (yet nonzero) delay can be modeled and controlled by exploiting granular (interval-valued) information about the system rather than relying on the pure numeric reading of the sensors. The granularity of information introduced on purpose was intended to capture our ignorance about the delay value of the system under control.

There is another interesting issue linked with the subject of this study. It concerns clustering heterogeneous data where some features of data (patterns) are granular and represented in the form of intervals. While close to the idea discussed in Refs. [18,19], the intervalization of the data can be captured in the format of granular prototypes themselves. This issue of granular clustering itself goes beyond the scope of this study and will be discussed in the foregoing study.

References

[1] R. Bellman, R. Kalaba, L.A. Zadeh, Abstraction and pattern classification, J. Math. Anal. Appl. 13 (1966) 1–7.
[2] J.C. Bezdek, Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum Press, New York, 1981.
[3] A. Kandel, Fuzzy Techniques in Pattern Recognition, Wiley, New York, 1982.
[4] E.T. Lee, Fuzzy tree automata and syntactic pattern recognition, IEEE Trans. Pattern Anal. Mach. Intell. PAMI-4 (4) (1982) 445–449.
[5] W. Pedrycz, Fuzzy sets in pattern recognition, Pattern Recognition 2/3 (1990) 121–146.
[6] S.K. Pal, B. Chakraborty, Fuzzy set theoretic measure for automatic feature evaluation, IEEE Trans. Systems, Man, Cybernet. SMC-16 (5) (1986) 754–760.
[7] K. Fukunaga, Introduction to Statistical Pattern Recognition and Machine Learning, Academic Press, New York, 1972.
[8] Jain, D. Zongker, Feature selection: evaluation, application, and small sample performance, IEEE Trans. Pattern Anal. Mach. Intell. 19 (2) (1997) 152–158.

[9] S. Theodoridis, K. Koutroumbas, Pattern Recognition, Academic Press, San Diego, 1999.
[10] J. Mao, A.K. Jain, Artificial neural networks for feature extraction and multivariate data projection, IEEE Trans. Neural Networks 6 (2) (1997) 296–317.
[11] R. Setino, H. Liu, Neural network feature selector, IEEE Trans. Neural Networks 8 (3) (1997) 654–662.
[12] P. Pudil, J. Novovicova, J. Kittler, Floating search methods in feature selection, Pattern Recognition Lett. 15 (1994) 1119–1125.
[13] B. Yu, B. Yuan, A more efficient branch and bound algorithm for feature selection, Pattern Recognition 26 (6) (1993) 883–889.
[14] J.C. Bezdek, P.F. Castelaz, Prototype classification and feature selection with fuzzy sets, IEEE Trans. Systems Man Cybernet. SMC-7 (2) (1977) 87–92.
[15] V. Di Gesu, M.C. Maccarone, Feature selection and possibility theory, Pattern Recognition 19 (1986) 63–72.
[16] C. Blake, E. Keogh, C.J. Merz, UCI repository of machine learning databases [http://www.ics.uci.edu/~mlearn/MLRepository.html], University of California, Department of Information and Computer Science, Irvine, CA.
[17] W. Pedrycz, G. Vukovich, Data-based design of fuzzy sets, J. Fuzzy Logic Intelligent Systems 9 (3) (1999) 255–263.
[18] R. Hathaway, J.C. Bezdek, W. Pedrycz, A parametric model for fusing heterogeneous fuzzy data, IEEE Trans. Fuzzy Systems 4 (1996) 270–281.
[19] W. Pedrycz, J.C. Bezdek, R.J. Hathaway, G.W. Rogers, A non-parametric model for fusing heterogeneous data, IEEE Trans. Fuzzy Systems 6 (1998) 411–425.

About the Author—WITOLD PEDRYCZ is a Professor and Director of Computer Engineering and Software Engineering in the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, Canada. He is actively pursuing research in Computational Intelligence, fuzzy modeling, knowledge discovery and data mining, fuzzy control including fuzzy controllers, pattern recognition, knowledge-based neural networks, relational computation, and Software Engineering. He has published numerous papers in this area. He is also author of 7 research monographs covering various aspects of Computational Intelligence and Software Engineering. Dr. Pedrycz is an IEEE Fellow. He currently serves as an Associate Editor of IEEE Transactions on Systems Man and Cybernetics and IEEE Transactions on Fuzzy Systems. He also served on the Editorial Board of IEEE Transactions on Neural Networks.

About the Author—GEORGE VUKOVICH received his Ph.D. from the University of Toronto in Control Systems in 1982. After working at Northrop Corp. and Honeywell Inc., he joined the Canadian Space Agency as a research scientist in 1989. Currently he is the Director of Spacecraft Engineering. He maintains an active research program in Spacecraft Control Systems, Structural Control and Fuzzy Logic.