A Fuzzy Information Density Based Clustering Algorithm in Medical Data Analysis

V. Ponomarenko

***Abstract* —** The paper is devoted to the transformation of numerical values into linguistic values in medical data analysis. A software tool, based on clustering, was developed for this purpose. The implementation results are presented in this paper. A modification of the fuzzy clustering algorithm FEBFC is also introduced, which supposes using a fuzzy information density instead of fuzzy entropy. The proposed algorithm is compared with several well-known clustering algorithms on medical data sets.

Keywords — cluster validity indices, fuzzification, clustering algorithms, fuzzy clustering, medical data analysis.

I. Introduction

During the last few decades, the face of the modern world was qualitatively changed by information technologies. Various technical means and information delivery channels, based on progressive information and communication systems, have revolutionized human life, that nowadays is inseparably associated with huge flows of data. These data are involved in all spheres of human activity – social, economic, political, spiritual. However, the information contained in such data can be whether very valuable or completely useless. Extracting various kinds of useful information from the data sets is one of the main tasks of the modern science called data mining.

Data mining proposes methods, that today are widely used in healthcare, which is in one of the fundamental fields of the social sphere of modern life. Such popularity of the data mining methods was formed mainly due to the rapid development of medical devices and therapy technologies, that allow to produce and to store large amount of data. Producing of new data is mostly achieved in a process of providing medical services. For example, imagine a patient who came to a polyclinic to examine his digestive system. A medical worker, using a special probe or ultrasound device, measures necessary medical indicators and records them in a medical report. Obtained in a such way medical data are usually persisted in some database for possible using in future. Therefore, storing medical data is achieved through the use of various database systems.

The information contained in medical data is extremely important for solving diagnostic, therapeutic, statistical, administrative and other tasks in the field of medicine, e.g. determination of a correct treatment, definition of a patient's group of risk and prevention of diseases. Solving of these tasks has a huge impact on a quality of medical services, life expectancy, mortality and time of illnesses of population.

In the field of medicine, both numeric (continuous) and nominal (linguistic) types of data are used. The numeric data type is used for representing value of a continuous medical indicator, e.g. age of a patient, body mass index, resting blood pressure, albumin and globulin ratio. Variables of the nominal data type usually keep a name of some state of a categorical medical indicator, e.g. a patient's appetite can be good or poor, a tumor can be malignant or benign. In medical data analysis obtaining a continuous value is often not enough informative to make a

V. Ponomarenko, University of Žilina, Žilina, Slovakia (e-mail: volod.ponomar@gmail.com).

conclusion about patient's state for determining necessary treatment, so the medical worker has to associate this value with an appropriate nominal value. This is a typical situation when a transformation from numeric into linguistic values comes into play. The numeric values of an attribute obtained as a result of the transformation can be relatively easy converted into fuzzy data specified by a membership function. The whole process of obtaining fuzzy values from numeric values is called fuzzification.

Fuzzy medical data, obtained as a result of fuzzification, are very valuable. They can be used, in particular, for increasing the healthcare system reliability through the reducing potential medical failures, that is discussed in papers [1]–[2].

The process of transformation from numeric into linguistic values, that is based on cluster analysis, is the subject of study in this paper. Various of clustering algorithms, that perform cluster analysis, can be found in literature. This variety includes fuzzy clustering algorithms, that implies belonging of an object to several clusters simultaneously. A promising clustering algorithm Fuzzy Entropy Based Fuzzy Classifier (FEBFC), proposed in paper [3], assumes using a fuzzy entropy, based on Shannon's entropy, as a criterion of optimality. In this paper a modification of the FEBFC algorithm is introduced, based on different approach to optimality criterion: a fuzzy information density is used instead of the fuzzy entropy measure. This study also includes development of a software for transformation values of any numeric attribute of the medical data set into fuzzy values, based on clustering algorithms. Several fuzzy clustering algorithms are implemented in this software solution as well as the mentioned FEBFC algorithm and its modification. Implementation details and accuracy comparison of clustering algorithms are discussed below in this paper.

II. Fuzzy Information Density Based Fuzzy Classifier

The mentioned above FEBFC clustering algorithm was proposed by Hahn-Ming Lee, Chih-Ming Chen et al. [3]. According to the proposed approach, the fuzzy entropy $FE(\tilde{A})$ is defined on the universal set $X = \{r_1, r_2, \dots, r_n\}$, where $i = 1, 2, \dots, n$, for the elements within an interval (cluster) in a non-probabilistic way:

$$FE(\tilde{A}) = \sum_{j=1}^{m} FE_{C_j}(\tilde{A}) = \sum_{j=1}^{m} -D_j \log_2 D_j \tag{1}$$

where $\tilde{A}$ is a fuzzy set defined on an interval of pattern space which contains k elements ($k < n$); $C_1, C_2, \dots, C_m$ represent m classes into which the n elements are divided; $FE_{C_j}(\tilde{A})$ is the fuzzy entropy of the elements of class j in an interval, defined as $FE_{C_j}(\tilde{A}) = -D_j \log_2 D_j$; D_j is the match degree with fuzzy set $\tilde{A}$ for the elements of class j in an interval, where $j = 1, 2, \dots, m$, defined as $D_j = \frac{\sum_{r \in S_{C_j}(r_n)} \mu_{\tilde{A}}(r)}{\sum_{r \in X} \mu_{\tilde{A}}(r)}$; $\mu_{\tilde{A}}(r_i)$ is the mapped membership degree of the element r_i with the fuzzy set $\tilde{A}$; $S_{C_j}(r_n)$ is a set of elements of class j on the universal set X (subset of the universal set X).

The FEBFC algorithm assumes using a fuzzy entropy cluster validity index (I_{FE}), also known as a total fuzzy entropy, for determining an optimal number of clusters. The index is defined as a sum of fuzzy entropies of all clusters:

$$I_{FE} = \sum_{i=1}^{k} FE_i^* \tag{2}$$

where k – the number of clusters; FE_i^* – the fuzzy entropy of the i-th cluster, calculated as a sum of fuzzy entropies of all fuzzy sets on the i-th interval.

The proposed cluster validity index would be perfect if all intervals had equal length and quantity of patterns on them. But usually there are several clusters of different size among one data set. Distances between the patterns are also different. Thus, a simple addition of fuzzy entropy values of intervals may lead to inaccurate results. Alternatively, the fuzzy information density measure can be used instead of the fuzzy entropy measure as an optimality criterion, that will potentially lead to more accurate clustering results, because the fuzzy information density takes cluster sizes into account.

The information density measure was defined in paper [4]. Based on it, of the fuzzy information density FD_q of the q-th cluster ($q = 1,2,\dots,k$) is defined as:

$$FD_q = \begin{cases} FE_q / \log_2(n_q + 1), & \text{if } n_q > 0 \\ 0, & \text{if } n_q = 0 \end{cases} \tag{3}$$

where FE_q – the fuzzy entropy on the q-th interval; n_q – the number of patterns on the q-th interval.

The fuzzy entropy cluster validity index (I_{FE}) should be then replaced by the fuzzy information density cluster validity index (I_{FD}), that is also called the total information density. It is defined as:

$$I_{FD} = \sum_{q=1}^{k} \omega_q \times FD_q \tag{4}$$

where FD_q – the fuzzy information density on the q-th interval; $\omega_q = \frac{n_q}{n}$ is a weight coefficient; n – the number of patterns in the data set; n_q – the number of patterns on the q-th interval. Then the optimal number of clusters k^* is calculated as

$$I_{FD}(k^*) = \min_{2 \le k \le n-1} I_{FD}(k) \tag{5}$$

As a result of applying the proposed changes to the original FEBFC clustering algorithm, a new algorithm was obtained. It was called a Fuzzy Information Density Based Fuzzy Classifier (FIDBFC).

The FIDBFC clustering algorithm consists of the following steps:

Step 1. Set the initial number of clusters (intervals) $k := 2$.

Step 2. Locate the centers of intervals using following subsequence of steps:

2A. Find the initial centers of intervals $c_1, c_2, \dots, c_k$ using formula:

$$c_q = x_{min} + (x_{max} - x_{min}) \times \frac{q-1}{k-1}, \qquad q = 1,2,\dots,k.$$

2B. Assign each element of the distribution to a corresponding interval with the smallest Euclidian distance to the interval center:

$$|x_i - c_q^*| = \min_{1 \le q \le k} |x_i - c_q|$$

where c_q^* is the closest center to the element x_i.

2C. Recompute the cluster centers.

$$c_q = \frac{\sum_{i=1}^{n_q} x_i^q}{n_q}$$

where n_q is the total number of patterns x_i^q, that belong to q-th cluster.

2D. Compare recomputed cluster centers with previous. If any center was changed then go to *Step 2B*. Otherwise, go to *Step 3*.

Step 3. Assign the membership function for each interval according to:

$$\mu_1 = \begin{cases} 1, & \text{for } x \le c_1 \\ \frac{c_2 - x}{c_2 - c_1}, & \text{for } c_1 < x \le c_2 \\ 0, & \text{otherwise} \end{cases} \qquad \mu_q = \begin{cases} 0, & \text{for } x \le c_{q-1} \\ \frac{x - c_{q-1}}{c_q - c_{q-1}}, & \text{for } c_{q-1} < x \le c_q \\ \frac{c_{q+1} - x}{c_{q+1} - c_q}, & \text{for } c_q < x \le c_{q+1} \\ 0, & \text{otherwise} \end{cases}$$

$$\mu_k = \begin{cases} 0, & \text{for } x < c_{k-1} \\ \frac{x - c_{k-1}}{c_k - c_{k-1}}, & \text{for } c_{k-1} \le x < c_k \\ 1, & \text{otherwise} \end{cases}$$

where $q = 2,3,\ldots,k-1$.

Step 4. Compute the $I_{FD}(k)$ for k clusters and $I_{FD}(k-1)$ for $k-1$ clusters according to formula (5).

Step 5. If $I_{FD}(k) < I_{FD}(k-1)$, then partition again $(k := k+1)$ and go to *Step 2*; otherwise, $k-1$ is the optimal number of clusters.

For illustrating the FIDBFC algorithm, we use the following example. Let *X* be a distribution of three classes of objects represented by values of some attribute of these objects. The distribution divided into three and four intervals is shown in Figure 1 (a) and (b) respectively as a set of objects △, □ and ○, placed on x axis. Position on the axis corresponds with a value of the attribute.

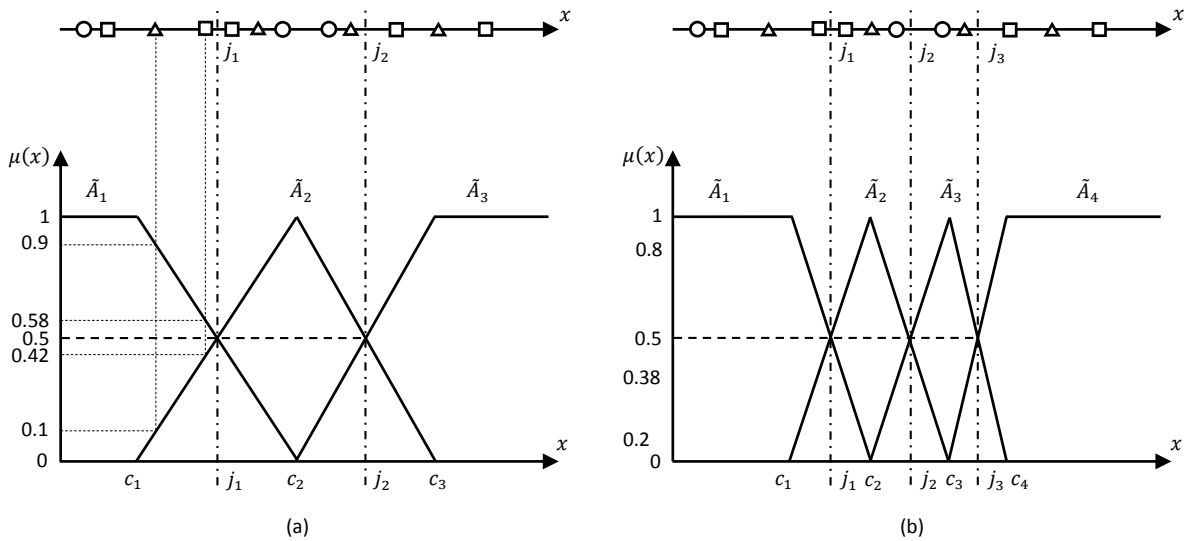

Fig. 1. Example of a distribution of 3 classes of objects (△, □ and ○ denote class 1, class 2 and class 3 respectively) with corresponding membership functions

In the first considered case (Figure 1 (a)), the distribution is divided into three intervals, that

are $(-\infty; j_1)$, $[j_1; j_2)$ and $[j_2; \infty)$. On these intervals fuzzy sets $\tilde{A}_1$, $\tilde{A}_2$ and $\tilde{A}_3$ are obtained using a membership function. The centers of fuzzy sets are denoted as c_1, c_2, c_3. Let us introduce a calculation process of the total fuzzy entropy of the distribution with mentioned dividing.

In the below paragraphs the sequence of steps needed to calculate the fuzzy entropy measure of the interval $(-\infty; j_1)$ is detailly described. The measure calculation process for other intervals is skipped, because the same method is used.

At first, we find a total membership degree for each class on values of the fuzzy set $\tilde{A}_1$ from the interval:

- total membership degree of "$\triangle$" is 0.9;
- total membership degree of "$\square$" is $1 + 0.58 = 1.58$;
- total membership degree of "$\circ$" is 1.

Then we calculate match degrees:

- $D_{\triangle} = 0.9/(0.9 + 1.58 + 1) = 0.9/3.48 = 0.25862$;
- $D_{\square} = 1.58/3.48 = 0.45402$;
- $D_{\circ} = 1/3.48 = 0.28736$.

In the next step we calculate fuzzy entropies of $\tilde{A}_1$ on $(-\infty; j_1)$:

- $FE_{\triangle}(\tilde{A}_1) = -0.25862 \times \log_2 0.25862 = 0.50459$;
- $FE_{\square}(\tilde{A}_1) = -0.45402 \times \log_2 0.45402 = 0.51721$;
- $FE_{\circ}(\tilde{A}_1) = -0.28736 \times \log_2 0.28736 = 0.51698$;
- $FE_1(\tilde{A}_1) = FE_{\triangle}(\tilde{A}_1) + FE_{\square}(\tilde{A}_1) + FE_{\circ}(\tilde{A}_1) = 1.53878$.

Similarly, the fuzzy entropies of $\tilde{A}_2$ and $\tilde{A}_3$ on $(-\infty; j_1)$ are calculated:

- $FE_1(\tilde{A}_2) = FE_{\triangle}(\tilde{A}_2) + FE_{\square}(\tilde{A}_2) + FE_{\circ}(\tilde{A}_2) = 0.70627$;
- $FE_1(\tilde{A}_3) = FE_{\triangle}(\tilde{A}_3) + FE_{\square}(\tilde{A}_3) + FE_{\circ}(\tilde{A}_3) = 0$.

The fuzzy entropy of the interval $(-\infty; j_1)$ equals:

- $FE_1^* = FE_1(\tilde{A}_1) + FE_1(\tilde{A}_2) + FE_1(\tilde{A}_3) = 2.24505$.

Similarly, the fuzzy entropies FE_2^* and FE_3^* of the corresponding intervals $[j_1; j_2)$ and $[j_2; \infty)$ can be obtained:

- $FE_2^* = FE_2(\tilde{A}_1) + FE_2(\tilde{A}_2) + FE_2(\tilde{A}_3) = 3.67795$;
- $FE_3^* = FE_3(\tilde{A}_1) + FE_3(\tilde{A}_2) + FE_3(\tilde{A}_3) = 0.95096$.

After obtaining the fuzzy entropy values, the information density measures can be calculated for each interval:

- $FD_1 = FE_1^*/\log_2(k+1) = 2.24505/\log_2 5 = 0.96689$.
- $FD_2 = FE_2^*/\log_2(k+1) = 3.67795/\log_2 6 = 1.42283$;
- $FD_3 = FE_3^*/\log_2(k+1) = 0.95096/\log_2 4 = 0.47548$.

Finally, according to (4) the total fuzzy information density measure of the distribution (a) is calculated in the following way:

- $I_{FD}^{(a)} = (4/12) \times 0.96689 + (5/12) \times 1.42283 + (3/12) \times 0.47548 =$
  $= 1.03401$.

In the second considered case (Figure 1 (b)), the distribution is divided into four intervals, that are $(-\infty; j_1)$, $[j_1; j_2)$, $[j_2; j_3)$ and $[j_3; \infty)$. Using the same method, as described before, the total fuzzy information density measure for this case of dividing of the distribution is calculated as:

- $I_{FD}^{(b)} = (3/12) \times 0.78318 + (4/12) \times 0.99318 + (2/12) \times 0.90356 +$
  $+ (3/12) \times 0.47122 = 0.79526$.

According to obtained results ($I_{FD}^{(b)} < I_{FD}^{(a)}$), dividing the distribution into four intervals is preferable. Comparison of clustering results of the introduced algorithm is described below.

III. A Software Tool for Cluster Analysis and Fuzzification

A significant part of the study was devoted to development of a software for transformation of values of any numeric attribute of a medical data set into fuzzy values. This software is based on fuzzy clustering algorithms and satisfies following functional requirements:

- reading data set from a file;
- basic analysis of the initial data set;
- graphical visualization of basic analysis results;
- fuzzy clustering algorithms implemented;
- importing clustering results from external software;
- graphical visualization of clustering results;
- fuzzification of the initial data set depending on clustering results;
- writing fuzzification result to a file.

According to the above list, the most of functionality of the developed software is related to fuzzy clustering. Therefore, it was called the Fuzzy Clustering Tool. The software was implemented in C++ programming language using Qt 5.10 framework. The main window of it is shown in Figure 2.

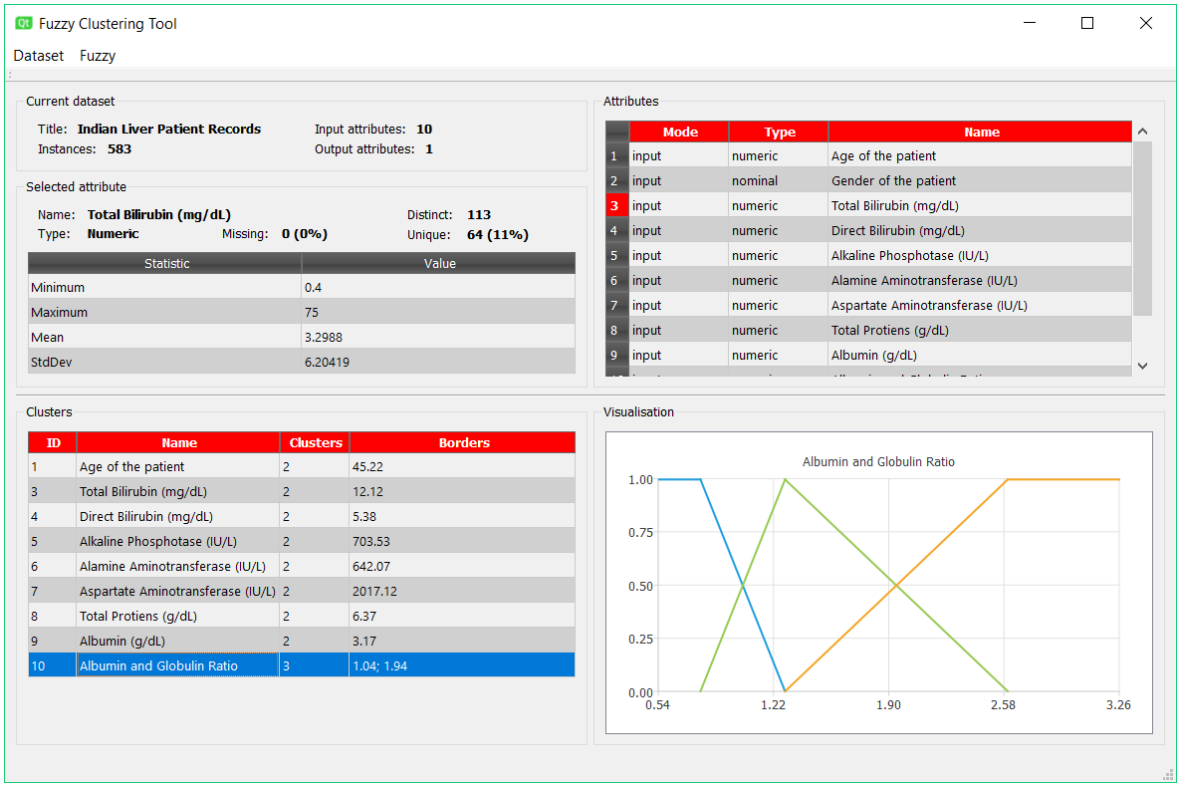

Fig. 2. The main window of the Fuzzy Clustering Tool.

IV. Comparison of the FIDBFC Algorithm with Other Fuzzy Clustering Algorithms on Medical Data

Using the implemented Fuzzy Clustering Tool, the introduced in this paper FIDBFC algorithm was compared with following fuzzy clustering algorithms: Fuzzy c-Means (FCM) [5]–[6], Gustafson-Kessel (GK) algorithm [7], Gath-Geva (GG) algorithm [8], Multi-Interval Discretization (MID) [4] and Fuzzy Entropy Based Fuzzy Classifier (FEBFC) [3]. The first three algorithms (Fuzzy c-Means, Gustafson-Kessel algorithm and Gath-Geva algorithm) need

a cluster validity index to define an optimal number of clusters, therefore the Pairing Frequency index, proposed in paper [9], was additionally implemented into the software tool.

Evaluation of fuzzy clustering results was performed through the Clustering Accuracy Indices, that can be divided into two groups: Internal indices (uses only input attributes data) and External indices (uses information about belonging of a pattern to some class of data) [10]–[11]. The Internal indices can be used in both supervised and unsupervised learning, but the External indices can be used in case of supervised learning only.

In a literature of clustering a lot of various internal indices can be found, but the most well-known of them are the following [12]–[18]:

- Partition Coefficient index;
- Partition Entropy index;
- Fukuyama-Sugeno index;
- Xie-Beni index.

Among the external indices the following can be highlighted [19]–[20]:

- Purity index;
- Normalized Mutual Information index.

The listed above clustering algorithms were evaluated and compared using these internal and external indices on the following medical data sets, obtained from the Kaggle [21] and UCI [22] Machine Learning Repositories:

- Pima Indians Diabetes;
- Heart Disease;
- Breast Cancer Wisconsin;
- Indian Liver Patient Records;
- Chronic Kidney Disease.

Table 1. Clustering Accuracy Indices calculated for the fuzzification performed on the Pima Indians Diabetes Dataset

Algorithm	Partition Coefficient index	Partition Entropy index	Fukuyama-Sugeno index	Xie-Beni index	Purity index	Normalized Mutual Information index
FCM	0.74530	0.55755	15.49595	0.13241	0.67337	0.04168
GK	0.81605	0.39656	9.10605	62.95703	0.66701	0.04180
GG	0.80680	0.42239	7.03228	129.22207	0.66321	0.03090
MID	0.96787	0.06993	9.38145	187.26315	0.65865	0.02023
FEBFC	0.84070	0.34661	5.35528	0.09274	0.65654	0.02903
FIDBFC	0.83577	0.35898	5.69637	0.09239	0.65654	0.02913

According to the obtained values of the Partition Coefficient and the Partition Entropy indices, the most accurate is the MID algorithm (see Table 1). The FEBFC algorithm is the most accurate according to the Fukuyama-Sugeno index. The FIDBFC algorithm is the most accurate according to the Xie-Beni index. The Purity indices identify the FCM as the most accurate algorithm and Normalized Mutual Information indicates the results of the GK algorithm as the best. The FIDBFC, which is the modification of the FEBFC, gives better results than the FEBFC according to the Xie-Beni index, that is one of the most relevant internal indices, and according to the Normalized Mutual Information, that is one of the most relevant external indices. Thus, modification of the FEBFC algorithm, proposed in this paper, leads to

better clustering results of the Pima Indians Diabetes Dataset according to two significant indices.

Table 2. Clustering Accuracy Indices calculated for the fuzzification performed on the Cleveland Heart Disease data set

Algorithm	Partition Coefficient index	Partition Entropy index	Fukuyama-Sugeno index	Xie-Beni index	Purity index	Normalized Mutual Information index
FCM	0.74495	0.56002	5.92440	0.11528	0.54916	0.04926
GK	0.80537	0.42032	3.17181	9.16700	0.54603	0.05179
GG	0.73040	0.58724	5.84767	62.60825	0.55168	0.04751
MID	0.98750	0.02764	4.50228	313.08871	0.54940	0.06126
FEBFC	0.81525	0.40006	2.25668	0.10192	0.54125	0.03940
FIDBFC	0.82118	0.38605	2.00213	0.10587	0.54125	0.03905

As we can see in Table 2, the MID algorithm gives the most accurate clustering results according to the Partition Coefficient, the Partition Entropy and the Normalized Mutual Information indices. Clustering results of the GG algorithm are the most accurate only according to the Purity index as well as clustering results of the FEBFC algorithm are the most accurate according to the Xie-Beni index.

The FIDBFC algorithm is the most accurate according to the Fukuyama-Sugeno index. It also better than the original FEBFC algorithm according to the Partition Coefficient and the Partition Entropy indices, has the same accuracy according to the Purity index and a bit worse according to other two indices. Thus, modification of the FEBFC algorithm, proposed in this paper, leads to better clustering results of the Heart Disease data set according to at least half of considered indices.

Table 3. Clustering Accuracy Indices calculated for the fuzzification performed on the Breast Cancer Wisconsin data set

Algorithm	Partition Coefficient index	Partition Entropy index	Fukuyama-Sugeno index	Xie-Beni index	Purity index	Normalized Mutual Information index
FCM	0,70180	0,65043	9,97471	0,17636	0,78141	0,20863
GK	0,80869	0,41571	3,11619	136,80074	0,77748	0,19883
GG	0,71908	0,61341	8,52217	30,85465	0,78417	0,20820
MID	0,91949	0,17782	8,03281	243,40205	0,76021	0,19926
FEBFC	0,83261	0,36418	3,60073	0,09625	0,75014	0,19384
FIDBFC	0,78031	0,47621	6,69028	0,10761	0,76131	0,20961

As we can see in Table 3, the MID algorithm gives the most accurate clustering results according to the Partition Coefficient and the Partition Entropy indices. According to the Fukuyama-Sugeno, Xie-Beni and Purity indices, clustering results of the GK, FEBFC and GG algorithms are the most accurate correspondingly.

The FIDBFC algorithm is the most accurate according to the Normalized Mutual Information index. It also more accurate than the original FEBFC algorithm according to the Purity index, but less accurate according to the other indices. Thus, for the Breast Cancer Wisconsin data set the FEBFC algorithm in general gives better results than its modification (the FIDBFC algorithm).

Table 4. Clustering Accuracy Indices calculated for the fuzzification performed on the Indian Liver Patient Records data set

Algorithm	Partition Coefficient index	Partition Entropy index	Fukuyama-Sugeno index	Xie-Beni index	Purity index	Normalized Mutual Information index
FCM	0,76521	0,51781	7,31219	0,23369	0,71355	0,03833
GK	0,84262	0,33978	4,81232	235,85757	0,71355	0,03798
GG	0,80658	0,41956	6,89524	5,91104	0,71355	0,03553
MID	0,94466	0,12144	7,53296	150,68521	0,71520	0,04287
FEBFC	0,88244	0,25931	3,02895	0,06235	0,71355	0,02400
FIDBFC	0,87023	0,28597	3,33581	0,06514	0,71355	0,02874

As we can see in Table 4, the MID algorithm gives the most accurate clustering results according to the Partition Coefficient, the Partition Entropy, the Purity and the Normalized Mutual Information indices. According to the Fukuyama-Sugeno as well as the Xie-Beni indices, clustering results of the FEBFC algorithm are the most accurate.

Table 5. Clustering Accuracy Indices calculated for the fuzzification performed on the Chronic Kidney Disease data set

Algorithm	Partition Coefficient index	Partition Entropy index	Fukuyama-Sugeno index	Xie-Beni index	Purity index	Normalized Mutual Information index
FCM	0,77893	0,49223	5,33962	0,22215	0,74692	0,17449
GK	0,86890	0,28460	2,22786	169,93521	0,75999	0,19142
GG	0,79741	0,44822	4,80674	22,14820	0,68667	0,13921
MID	0,93745	0,13790	5,55817	301,79223	0,63449	0,07141
FEBFC	0,87330	0,28549	1,50738	0,06734	0,65568	0,09795
FIDBFC	0,82725	0,38268	3,01684	0,08254	0,67831	0,12089

The FIDBFC algorithm on the Indian Liver Patient Records data set is more accurate than the original FEBFC algorithm according to the Normalized Mutual Information index, but less accurate or has the same accuracy according to the other indices. Thus, for the Indian Liver Patient Records data set the FEBFC algorithm in general gives better results than its modification (the FIDBFC algorithm).

As we can see in Table 5, the MID algorithm gives the most accurate clustering results according to the Partition Coefficient and the Partition Entropy indices; the FEBFC algorithm gives the most accurate clustering results according to the Fukuyama-Sugeno and the Xie-Beni indices; the GK algorithm gives the most accurate clustering results according to the Purity and the Normalized Mutual Information indices.

The FIDBFC algorithm is more accurate than the original FEBFC algorithm according to the Purity and the Normalized Mutual Information indices, but less accurate according to the other indices. Thus, for the Chronic Kidney Disease data set the FEBFC algorithm in general gives better results than its modification (the FIDBFC algorithm).

V. Conclusion

In this paper a modification of the FEBFC clustering algorithm was introduced. It is based on using a fuzzy information density instead of the fuzzy entropy measure. The modification, called a Fuzzy Information Density Based Fuzzy Classifier, with several other clustering algorithms were implemented in special software tool, that allowed to make comparison of clustering results. It was experimentally approved, that on some medical data set the introduced FIDBFC algorithm gives better results than the original FEBFC algorithm. Thus, for such data set using the FIDBFC algorithm for transformation from numeric into linguistic and fuzzy values is preferable.

References

[1] Zaitseva E., Levashenko V., Kvassay M., Barach P. *Healthcare system reliability analysis addressing uncertain and ambiguous data.* 2017 International Conference on Information and Digital Technologies (IDT), Žilina, 2017, pp. 442-451.

[2] Barach P., Levashenko V., Zaitseva E. *New Methods for Healthcare System Evaluation Using Human Reliability Analysis.* Proceedings of the Human Factors and Ergonomics Society, vol. 61(1), 2017, pp. 583-587.

[3] Lee H. M., Chen Ch. M., Chen J. M., Jou Y. L. *An efficient fuzzy classifier with feature selection based on fuzzy entropy.* In IEEE Transactions on systems, Man, and Cybernetics – Part B: Cybernetics, vol. 31, no. 3, 2001, pp. 426-432.

[4] Popel D. V. *From continuous to multiple-valued data.* In Proceedings IEEE International Symposium on Multiple-Valued Logic, 2003, pp. 367-372.

[5] Bezdek J. C. *Pattern recognition with fuzzy objective function algorithms.* New York, NY: Plenum Press, 1981, 272 pages.

[6] Dunn J. *A fuzzy relative of the ISODATA process and its use in detecting compact well separated clusters.* Journal of Cybernetics, 3 (3), 1974, pp. 32-57.

[7] Gustafson D. E., Kessel W. C. *Fuzzy clustering with a fuzzy covariance matrix.* Proc. IEEE CDC, 1979, pp. 761-766.

[8] Gath I., Geva A. B. *Unsupervised optimal fuzzy clustering.* IEEE Trans. Patt. Anal. Machine Intell., vol. 11, no 7, 1989, pp. 773-780.

[9] Cui H., Zhang K., Fang Y., Sobolevsky S., Ratti C., Horn B. K. P. *A Clustering Validity Index Based on Pairing Frequency.* In IEEE Access, vol. 5, 2017, pp. 24884-24894.

[10] Rendón E., Abundez I., Arizmendi A., Quiroz E. M. *Internal versus external cluster validation indexes.* International Journal of Computers and Communications, vol. 5(1), 2011, pp. 27-34.

[11] Abonyi J., Feil B. *Cluster Analysis for Data Mining and System Identification.* Birkhäuser Verlag AG, 2007, 319 pages.

[12] Rezaee M. R., Lelieveldt B. P. F., Reiber J. H. C. *A new cluster validity index for the fuzzy c-mean.* Pattern Recognition Letters, vol. 19, 1998, pp. 237–246.

[13] Bezdek J. C. *Cluster validity with fuzzy sets.* Journal of Cybernetics, vol. 3(3), 1973, pp. 58-72.

[14] Bezdek J. C. *Mathematical models for systematics and taxonomy*. In: Proceedings 8th International Conference in Numerical Taxonomy, Freeman, San Francisco, 1975, pp. 143-166.
[15] Fukuyama Y., Sugeno M. *A new method of choosing the number of clusters for the fuzzy C-means method.* In Proc. 5th Fuzzy Syst. Symp., 1989, pp. 247-250.
[16] Xie X. L., Beni G. *A validity measure for fuzzy clustering.* IEEE Trans. Pattern Anal. Mach. Intell., vol. 13, no. 8, Aug. 1991, pp. 841–847.
[17] Zhou K., Ding S., Fu C., Yang S. *Comparison and weighted summation type of fuzzy cluster validity indices.* Int J Comput Commun Control, vol. 9, 2014, pp. 370-378.
[18] Pakhira M. K., Bandyopadhyay S., Maulik U. *Validity Index for Crisp and Fuzzy Clusters.* In. Pattern Recognition, vol. 37(3), 2004, pp. 487-501.
[19] Sripada S. Ch., Rao M. S. *Comparison of purity and entropy of K-means Clustering and Fuzzy C means Clustering.* Indian Journal of Computer Science and Engineering (IJCSE), Vol. 2, No. 3, 2011, pp. 343-346.
[20] Kvålseth T. O. *On Normalized Mutual Information: Measure Derivations and Properties.* Entropy, vol. 19(11), 2017, 631 pages.
[21] *The Home of Data Science & Machine Learning*, https://www.kaggle.com.
[22] *UCI Machine Learning Repository: Center for Machine Learning and Intelligent Systems*, https://archive.ics.uci.edu/ml/index.php.