MFWK-Means: Minkowski Metric Fuzzy Weighted *K*-Means for high dimensional data clustering

L. Svetlova, *Moscow Institute of Physics and Technology, RF*[1]
University of Texas at Brownsville, USA
80 Fort Brown, Brownsville, TX, USA, 78520
mila.s.svetlova@gmail.com

B. Mirkin, *Birkbeck University of London, UK*
Malet Street, London, UK WC1E 7HX
mirkin@dcs.bk.ac.uk

H. Lei, *University of Texas at Brownsville, USA*
80 Fort Brown, Brownsville, TX, USA, 78520
hansheng.lei@utb.edu

Abstract

This paper presents a clustering algorithm, namely MFWK-Means, which is a novel extension of K-Means clustering to the case of fuzzy clusters and weighted features. First, the Weighted K-Means criterion utilizing Minkowski metric is adopted to solve the problem of feature selection for high dimensional data. Then, a further extension to the case of fuzzy clustering is presented to group datasets with natural fuzziness of cluster boundaries. Also, we adopt an intelligent version of K-Means, using Mirkin's method of Anomalous Pattern for initialization. Our new Minkowski metric Fuzzy Weighted K-Means (MFWK-Means) is experimentally validated on both benchmark datasets and synthetic datasets. MFWK-Means is shown to be competitive and more stable against noise in comparison with a variety of versions of K-Means based methods. Moreover, in most situations it reaches the highest clustering accuracy at wider intervals of Minkowski exponent.

1. Introduction

Clustering is a popular computational activity to organize the objects into groups of similar objects (clusters). The overwhelming amount of published papers discussing various clustering techniques in a variety of application fields tells of the importance as well as difficulties in designing a general purpose clustering algorithm. The problem of defining a 'good' clustering is not straightforward and far from being clear. Yet there is a single clustering algorithm which is arguably the most popular among practitioners: that is *K*-Means [5]. Two main advantages of the *K*-Means algorithm are its intuitive simplicity and fast convergence [1]. It works well with a variety of probability distributions [26]. But there are disadvantages too. One serious drawback is that the method's results depend on the initial setting to a large degree, thus leading to the initialization problem [27]. Another drawback is that its results are highly dependent on the feature weights or scales accepted, which motivates us to consider a feature weighting problem [27]. We address these two most vital clustering issues in the context of fuzzy clustering. Fuzzy clusters may better reflect real world data structures, in which boundaries between groups are essentially fuzzy, and a more nuanced description of the object's belongingness is required.

We propose a novel method extending the conventional *K*-Means to the case of fuzzy clusters with weighted features and Minkowski distance metric [9-11, 12]. The method is used in an intelligent manner to compute the initial location of centroids and their number for a given dataset [1]. The remainder of the paper is organized as follows. Section 2 reviews background and related work. Section 3 describes previous versions of *K*-Means and the proposed novel extension. Section 4 specifies our experimental setting and describes the results of our experiments. Section 5 summarizes the paper.

2. Background

This paper concerns the feature weighting and initialization for *K*-Means in the context of fuzzy clustering approach.

2.1. Feature selection and the choice of metric

The conventional *K*-Means algorithm uses pre-assigned weights, or scale factors, when scoring the distance between objects. However, a meaningful cluster

IEEE IRI 2013, August 14-16, 2013, San Francisco, California, USA
978-1-4799-1050-2/13/$31.00 ©2013 IEEE

may fall in a subspace defined by a subset of features. Currently, there are two popular approaches to consider feature selection issue: subspace clustering [8] and feature weighting [12]. In this paper we focus on the feature weighting approach as related to the *K*-Means clustering criterion. To identify the importance of different features, variable feature weights are introduced: the more important the feature, the greater the weight it should have.

Our paper continues a long line of research started by De Soete for ultrametric or additive tree structures [6]. Makarenkov and Legendre extended De Soete's feature weighting approach to the *K*-means algorithm by alternatingly minimizing the *K*-Means summary distance criterion over three groups of variables: entity memberships, cluster centroids, and feature weights [7]. Huang et al. extended the *K*-Means criterion by admitting an arbitrary exponent to the weights [10, 11]. They also extended this method to the cluster-specific weighted features [9]. Amorim and Mirkin further extended the approach by using Minkowsi metric so that the weight exponent is used as the metric exponent, whereas the weights reflect just the feature scaling factors as usual [12].

2.2. Initialization

To initialize the conventional *K*-means method one should specify the number of clusters and their initial centroids. Pelleg and Moore proposed the popular *X*-Means method to solve initialization problem, which increases the number of initial centroids, according to posterior probabilities, until the upper bound is reached [13]. Another popular algorithm is *G*-Means algorithm, which is based on a statistical test for the hypothesis that a subset of data follows a Gaussian distribution [14]. Lei et al. considered this issue by noticing that the assumption of Gaussian distribution is not necessary for different application domains [15]. Instead of using models which are represented by the mixture of Gaussians they presented a similarity-driven clustering approach (*S*-Means). An overview of the solutions to the problem of initialization was provided by Chiang and Mirkin, who compared some popular approaches at data generated from Gaussian clusters with the controlled parameters of between- and within-cluster spread to model cluster intermix [16]. They demonstrated that the number of clusters is best reproduced by Hartigan's [3] method, though it fails on the issues of cluster and centroid recovery. In contrast, *iK*-Means method by Mirkin [1] leads to most accurate results in terms of cluster and centroid recovery; however, it may formidably overestimate the number of clusters.

2.3. Fuzzy clustering

A modification of the conventional *K*-Means to the case of fuzzy clustering, Fuzzy *C*-Means clustering FCM, was proposed by Dunn [5] and improved by Bezdek [4]. A fuzzy cluster is represented by its membership function, in which each component is interpreted as the degree of membership of an entity to a cluster.

Modifications of this method include well-known fuzzy possibility partition clustering algorithms, such as PCM, which involves the possibility of an object to belong to a cluster, and FPCM, which includes both memberships and possibilities [17, 18]. Further extensions of fuzzy clustering techniques are described in [19, 20, 21, 22, 23].

3. Methodology

This paper proposes a novel extension of the *K*-Means clustering method, addressing the issues discussed above. To address the initialization issue, the intelligent version of *K*-Means (*iK*-Means [1]) is utilized throughout.

3.1. The conventional *K*-Means clustering

The set of objects for the conventional *K*-Means is represented by a quantitative entity-to-feature matrix $Y = (y_{iv})$ where y_{iv} is the value of feature $v \in V$ at entity $i \in I$: I - a set of N entities, V - a set of M features. The method produces a partition $S = \{S_1, \dots, S_K\}$ of I in K non-empty, non-overlapping subsets $S_k \subset I$, clusters, each represented by a centroid c_K, an M-dimensional vector in the feature space $k = \overline{1,K}$. Starting from initial centroids, *K*-Means updates clusters according to the Minimum distance rule and centroids, as the cluster gravity centers.

It is well known that the *K*-Means is an alternating minimization algorithm for the square-error criterion:

$$W(S,c) = \sum_{k=1}^{K} \sum_{i \in S_k} \sum_{v=1}^{M} (y_{iv} - c_{kv})^2 \qquad (1)$$

3.2. Intelligent *K*-Means clustering

To initialize the *K*-means method, we apply the method of Anomalous Pattern [1] to find just one cluster S and its centroid c by alternatingly minimizing:

$$W(S,c) = \sum_{i \in S} d(i,c) + \sum_{i \in S} d(i,0) \qquad (2)$$

After an S is found, it is removed from the data, and process applies to the remaining entities, etc. In the end, all singletons are removed and remaining "anomalous" centroids initialize *K*-Means.

3.3. Feature weighting

A modification of *K*-Means criterion, named *MWK*-Means was proposed by Amorim and Mirkin [12]. They

modified the criterion (1) to include unknown weights w_v of the features v, $v = \overline{1, M}$:

$$W_\beta(S, c, w) = \sum_{k=1}^{K} \sum_{i \in I} \sum_{v=1}^{M} s_{ik} |w_v y_{iv} - w_v c_{kv}|^\beta = \sum_{k=1}^{K} \sum_{i \in S_k} d^\beta(y'_i, c'_k) \quad (3)$$

where weights are assumed to be non-negative and sum up to unity: $w_v \geq 0$, $\sum_{v \in M} w_v = 1$. The exponent β is defined by the user or by a semi-supervised procedure [12].

The right-hand expression refers to β power Minkowski metric between the rescaled entity points $y'_i = (w_v y_{iv})$ and centroids $c'_k = (w_v c_{kv})$ in the same feature space. Therefore, criterion (3) leads us to the criterion (1) by identifying the weights with feature scaling coefficients. Thus, minimization of Minkowski metric Weighted *K*-Means (*MWK*-Means) (3) is an alternating minimization algorithm in iterations over three groups of variables, i.e. clusters S_k, centroids c_k, and weights w_v.

3.4. Proposed algorithm: Minkowski metric Fuzzy Weighted *K*-Means

3.4.1. Weighted features for fuzzy clusters. A fuzzy cluster is represented by its membership function $s_k = (s_{ik})$, $i \in I$, in which s_{ik} $(0 \leq s_{ik} \leq 1)$ is the degree of membership of entity i to cluster k. The condition $\sum_k s_{ik} = 1$ is assumed. An additional modification is to calculate feature weights for each cluster separately. This follows earlier work by Huang et al. [9] and Amorim and Mirkin [12] to bring a greater flexibility to the procedure. Thus, w_{vk} instead of w_v is introduced, with w_{vk} being weights of feature v $(v = 1..M)$ for cluster k $(k = 1..K)$. Therefore, the criterion is finally modified to:

$$W_{F\beta}(S, c, w) = \sum_{k=1}^{K} \sum_{i \in I} \sum_{v=1}^{M} s_{ik}{}^\alpha w_{vk}^\beta |y_{iv} - c_{kv}|^\beta \quad (4)$$

Thus, minimization of *MFWK*-Means criterion (5) is also an alternating minimization algorithm in iterations over three groups of variables.

3.4.2. Minkowski metric Fuzzy Weighted K-Means method. Presented below is our formulation of the algorithm to alternatingly minimize the proposed *MFWK*-Means criterion (4) at given $\boldsymbol{\alpha}, \boldsymbol{\beta}$:

1. For given centroids c_k and weights $w_v = (w_{vk})$, update clusters according to the formula followed from the first order optimality condition for the problem of minimization of (4) constrained by $\sum_k s_{ik} = 1$:

$$s_{ik} = \frac{1}{\sum_{t=1}^{K} \left[\frac{d^\beta(y'_i, c'_k)}{d^\beta(y'_i, c'_t)} \right]^{\frac{1}{\alpha - 1}}} \quad (5)$$

The distance between y'_i and c'_k is calculated in β power Minkowski metric between the rescaled entity points $y'_i = (w_{vk} y_{iv})$ and centroids $c'_k = (w_{vk} c_{kv})$:

$$d^\beta(y'_i, c'_k) = \sum_{v=1}^{M} w_{vk}^\beta |y_{iv} - c_{kv}|^\beta \quad (6)$$

Some positive value is also added to each item in the sum to avoid the division by zero, and it is assumed $\alpha > 1$.

2. For given fuzzy clusters $s_k = (s_{ik})$ and weights $w_v = (w_{vk})$, update centroid $c_k = (c_{kv})$ of each cluster s_k as its Minkowski center so that, at each v, c_{kv} is a value minimizing β power Minkowski's distance to the cluster's objects (so the membership of all objects to a cluster *k* should be taken into account), such as:

$$d(c) = \sum_{i \in I} s_{ik}^\alpha w_{vk}^\beta |y_{iv} - c|^\beta \rightarrow min \quad (7)$$

To calculate the Minkowski metric center, the Steepest descent computation method is applied [1, 18].

3. For given clusters $s_k = (s_{ik})$ and centroids $c_k = (c_{kv})$, update weights according to the formula followed from the first order optimality condition for the problem of minimization of (4) constrained by $\sum_{v=1}^{M} w_{vk} = 1$:

$$w_{vk} = \frac{1}{\sum_{u=1}^{M} \left[\frac{\sum_{i \in I} s_{ik}^\alpha |y_{iv} - c_{kv}|^\beta}{\sum_{i \in I} s_{ik}^\alpha |y_{iu} - c_{ku}|^\beta} \right]^{\frac{1}{\beta - 1}}} \quad (8)$$

Some positive value is added to each item in the sum, and it is assumed that $\beta \neq 1$ in order to avoid division by 0.

4. Experiment Design, Results and Discussion

The goal of the current section is to explore the performance of *MFWK*-Means in comparison with existing analogues. The results of experiments for both real and synthetic datasets are described.

4.1. Experiment design

4.1.1. The input and the output of the algorithm. The input of the algorithm is a dataset $Y = (y_{iv})$. A user does not have to specify the number of clusters and the position of initial centroids, as the intelligent version of the method is implemented. The output of the algorithm is a final fuzzy clustering result: matrix $S = (s_{ik})$ of belongingness of objects i to clusters k. The algorithm also outputs the set of final feature weights and multi-dimensional centroids.

To start the algorithm, the user needs to specify exponent values α and β. The exponent α affects the fuzziness of the solution. The value β is a power of distance in Minkowski metric. To examine optimal values

of parameters, two different approaches were considered. Basically, the proposed method was tested at different values of α and β, varying them from 1 to 5, with the increment of 0.1. The most accurate results have been always achieved at $\alpha = \beta$. Additionally, a semi-supervised approach was implemented, learning α and β from exposing the class labels on a 20% data sample, as proposed in [12].

4.1.2. Deffuzification and evaluation of clustering accuracy on real datasets. There is no benchmark dataset with known fuzzy partition. In order to extract crisp clusters from fuzzy clustering partition to be able to evaluate the accuracy in recovery of pre-assigned cluster labels in real datasets, the following defuzzification procedure was used: object i belongs to a cluster k with a maximum value of s_{ik} for all $k = \overline{1,K}$. However, each object may belong, at most, to one crisp cluster.

In order to evaluate crisp clustering accuracy result for benchmark datasets, the idea from the paper of Huang et al. is applied [9-11]. They represented the accuracy of clustering as a proportion of points correctly clustered by the algorithm. Specifically, each of the K defuzzified clusters from the output of the algorithm is mapped to that of the pre-labeled K clusters with the largest overlap. Finally, the number of common objects in the largest overlaps among all clusters is summarized and divided by the size of the whole dataset.

4.1.3. Algorithms under comparison. All the experiments are performed on both initial datasets and on the data with added noise. Feature weighting procedure is analyzed in order to select the subspace of the most important features for each cluster and to detect noise features in data.

Our *MFWK*-Means method is compared with its analogues: the conventional *K*-Means (*K*-Means), Weighted *K*-Means (*WK*-Means) [9-11], and Minkowski metric Weighted *K*-Means (*MWK*-Means) [12].

Two versions of each of these methods are considered:

(i) *Random initialization approach* (*K*-Means, *WK*-Means, *MWK*-Means, *MFWK*-Means): 100 runs of each algorithm are performed, starting from random initializations.

(ii) *Intelligent initialization approach* (*iK*-Means, *iWK*-Means, *iMWK*-Means, *iMFWK*-Means), where intelligent version of the algorithm is considered, applying the Anomalous pattern method for initialization.

Real datasets for experiments - the Iris dataset, the Pima Indians Diabetes dataset, the Hepatitis dataset - are taken from the UCI Machine Learning Repository [24].

4.1.4. Representation of Results. To present all the results within a uniform structure, each method is considered as a special case of *WK*-Means as in [12]. For each method the authors considered "exponent β at distance" and "exponent β at weight". Exponents that correspond to the highest clustering accuracy results for different versions of *K*-Means method are reported. For the random initialization approach, the average value of accuracy, its standard deviation, and the maximum value among 100 executions of the algorithm are presented. This model of reporting is carried through all subsequent results.

4.2. Experiments on benchmark datasets

4.2.1 Iris dataset. The Iris dataset contains 3 classes of iris plants with 50 objects in each class, represented by 4 features: sepal length, sepal width, petal length, and petal width. Table 1 presents clustering accuracy results for the Iris dataset, which are achieved by applying methods under consideration.

Table 1. Accuracy level for different versions of *K*-Means at the Iris dataset

Method	Exponent β at		Accuracy (%)		
	Distance	Weight	Mean	Std dev	Max
K	2	0	84	12.3	89.3
WK	2	1.8	87.1	13.8	96.0
MWK	1.2	1.2	93.3	8.3	96.7
MFWK	2	2	96.7	0	96.7
iK	2	0	88.7		
iWK	2	1.1	96.7		
iMWK	1.2	1.2	96.7		
	2	2	94.7		
	3	3	90.0		
iMFWK	1.7	1.7	96.7		
	1.8	1.8	96.7		
	1.9	1.9	96.7		
	2	2	96.7		
	2.1	2.1	96.7		

The best result shows 96.7% accuracy. Within the random initialization approach, this result has been achieved only for *MWK*-Means method and only once in 100 runs [12]. Our *MFWK*-Means gives the maximum value for each of the runs, so the method does not depend on the initial centroids. Within the intelligent initialization approach, the accuracy of 96.7% has been achieved by all the methods, except the conventional one. However, *MFWK*-Means method reaches this result for a wide range of Minkowski exponent values. This shows that the proposed method is superior in terms of its stability and low sensitivity to the α, β parameters and initial setting.

4.2.2. Pima Indian Diabetes dataset. This dataset of 768×8 size contains information about patients and it involves 2 classes: 500 patients tested positive for diabetes, and 268 patients tested negative. Pima Indian

Diabetes data has a rather complex class structure, so clustering accuracy results for algorithms under consideration are not close to 100%. However, the results in Table 2 show that our *MFWK*-Means method outperforms all other version of K-Means in the case of the random initialization approach with 71.48% of accuracy. It also provides the highest average clustering result among 100 runs of the random initialization approach. On the other hand, when applying the intelligent initialization approach, *iMFWK*-Means outputs accuracies of 67.6% and 68.1%, which are higher than the accuracy of *iK*-Means and *iWK*-Means, but slightly less than the result achieved by *iMWK*-Means method.

Table 2. Accuracy level for different versions of *K*-Means at the Pima Indians Diabetes dataset

Method	Exponent β at		Accuracy (%)		
	Distance	Weight	Mean	Std dev	Max
K	2	0	66.67	0.55	66.8
WK	2	4.5	64.5	2.98	66.28
MWK	3.9	3.9	68.18	2.85	71.35
MFWK	4.6	4.6	68.54	2.76	71.48
iK	2	0	66.8		
iWK	2	1.8	64.7		
iMWK	4.9	4.9	69.4		
iMFWK	4.6	4.6	68.1		
	4.1	4.1	67.6		

4.2.3. Hepatitis dataset. This dataset of 155×19 size contains information about patients with hepatitis and involves 2 classes: 32 patients who died from the disease and 123 patients who survived. This dataset includes 12 categorical features. It is worth mentioning that there is a large amount of missing values in the data that were not included in the input of the algorithm. Accuracy levels that are achieved applying algorithms under consideration are presented in Table 3.

Table 3. Accuracy level for different versions of *K*-Means clustering method at the Hepatitis dataset

Method	Exponent β at		Accuracy (%)		
	Distance	Weight	Mean	Std dev	Max
K	2	0	71.51	1.36	72.26
WK	1	1	78.72	0.13	80
MWK	1	1	79.02	0.84	80
MFWK	2.6	2.6	82.75	5.75	86.25
iK	2	0	72.26		
iWK	1	1	78.71		
iMWK	2.3	2.3	84.52		
iMFWK	2.6	2.6	86.25		
	2.7	2.7	86.25		
	2.8	2.8	86.25		
	2.9	2.9	86.25		
	3.3	3.3	86.25		

Our method in both random and intelligent versions (*MFWK*-Means and *iMFWK*-Means, respectively) outperforms all other versions of *K*-Means, with the maximum accuracy of 86.25% reached at a wide range of exponent values. Replacing all missing values in data with the grand means of the corresponding features, the algorithm outputs a slightly less accurate result – 81.2%, which is higher than all other results of the methods considered, except *MWK*-Means.

Summarizing, the proposed method is shown to be competitive by outperforming all of its analogues in the experiments on the Pima Indians Diabetes dataset at random initialization, as well as on the Hepatitis dataset at both random and intelligent versions. Furthermore, for the Iris and Hepatitis datasets the proposed method outputs a wide set of exponent values at which the maximum accuracy is achieved, which tells to the method's stability and low sensitivity to initial settings. The idea of applying feature weighting procedure is substantiated as it assigns considerably higher weights to informative features, causing a successful clustering result.

4.3. Experiments on benchmark datasets with added noise

This section compares results achieved in the experiments on the Iris dataset and Pima Indians Diabetes dataset with added noise by the algorithms under consideration. The cardinality of each dataset is extended by adding a pre-specified number of noise features, which are uniformly distributed within the range of the original data distribution. For each dataset, noise features are generated 10 times and average accuracy results are presented.

4.3.1. Iris dataset with added noise features. For the test of robustness against added noise features, the Iris dataset described by 4 real features is supplemented with the same amount of noise features. As it was reported for *MWK*-Means, noise versions of the Iris dataset led to the same optimal values of exponents β, though the accuracy results slightly decreased after adding noise for both *WK*-Means and *MWK*-Means methods. Table 4 shows that the proposed *MFWK*-Means method is more reliable with respect to added noise. The accuracy result provided by our method stays unchanged after adding noise to data.

Table 4 shows that the maximum accuracy result of 96.7% on the Iris dataset with added noise is reached only by our *MFWK*-Means method. Moreover, it provides this accuracy at each run of the algorithm. On the contrary, *K*-Means, *WK*-Means, and *MWK*-Means do not reach the maximum accuracy at any of the 100 runs of the method. *iMFWK*-Means also presents 96.7% of accuracy. This result has not been achieved by other intelligent versions of *K*-Means under consideration. It is worth mentioning that due to the feature weighting procedure, our algorithm

is capable of identifying noise features, assigning to them close to zero weights.

Table 4. Accuracy levels achieved at the different versions of weighted *K*-Means clustering method at the noisy Iris dataset 4 noise features

Method	Exponent β at		Accuracy (%)		
	Dist	Weight	Mean	Std dev	Max
K	2	0	66.7	7.0	80.0
WK	2	1.2	88.7	12.6	96.0
MWK	1.2	1.2	90.0	12.8	96.0
MFWK	2	2	96.7	0	96.7
iK	2	0	69.3		
iWK	2	1.1	96.0		
iMWK	1.1	1.1	96.0		
	2	2	91.3		
	3	3	87.3		
iMFWK	1.9	1.9	96.7		
	2	2	96.7		
	2.1	2.1	96.7		

4.3.2. Pima Indians Diabetes dataset with added noise features. The hypothesis that the proposed *MFWK*-Means method is not sensitive to the added noise is substantiated by experiments on the Pima Indians Diabetes dataset as well. According to Table 5, the accuracy of clustering the data with added noise decreases by less than 3% for *iMFWK*-Means and by less than 5 % on average for *MFWK*-Means in the case of adding 8 noise features to 8 real ones. Curiously, the accuracy of clustering is higher for the data supplemented with 8 noise features than for the data supplemented with 4 noise features.

Table 5. Accuracy levels achieved by *MFWK*-Means clustering method at the noisy Pima Indians Diabetes dataset with 4 noise features and 8 noise features (top and bottom, respectively)

Method	Exponent β at		Accuracy (%)		
Features	Dist	Weight	Mean	Std dev	Max
MFWK					
8 real	4.6	4.6	68.54	2.76	71.48
+ 4	4.0	4.0	63.58	3.15	67.45
+ 8	4.0	4.0	64.18	2.98	67.71
iMFWK					
8 real	4.6	4.6	68.1		
+ 4	4.0	4.0	65.6		
+ 8	4.0	4.0	66.4		

Summarizing, in all cases the method is shown to be more reliable than the various versions of *K*-Means, with respect to the noise. Due to the feature weighting procedure the proposed algorithm is shown to be capable of eliminating unimportant features from the distance calculation, assigning them low weight value.

4.4. Experiments on synthetic data

4.4.1. Generating synthetic datasets and fuzzy clustering accuracy evaluation. In order to evaluate a fuzzy accuracy clustering result without applying the defuzzification procedure, synthetic datasets are considered, since so far there is no known real dataset with a pre-specified fuzzy partition. In order to construct datasets with pre-labeled fuzzy clusters we applied a Synthetic Data generator, which generates clusters as points of a multivariate Gaussian distribution [25]. In this case, the probability density function value is considered as the value of fuzzy belongingness of an object i to a cluster k. So, we can construct the matrix $T = (t_{ik})$ of the actual belongingness of objects to clusters. In order to evaluate fuzzy clustering accuracy we need to compare this matrix to the final fuzzy matrix $S = (s_{ik})$, produced by the proposed algorithm. Both S and T are normalized so that the within row sums are equal to unity. The accuracy of the fuzzy clustering result is:

$$accuracy = (1 - \frac{\sum_{i=1}^{N}\sum_{k=1}^{K}|t_{ik} - s_{ik}|}{2N}) * 100\%$$

It is worth mentioning that it is much harder to reach 100% recovery in a fuzzy case, because the values of the belongingness of objects to clusters – s_{ik} and t_{ik} – are between 0 and 1.

4.4.2. Synthetic datasets: low intermix case. In order to evaluate the clustering accuracy, 20 synthetic datasets of a low cluster intermix with 3 clusters, each consisting of 100 objects and described by 4 features, have been generated. Both the fuzzy clustering average accuracy result and the clustering result of its defuzzified version are presented in Table 6.

Table 6. The fuzzy clustering accuracy for the synthetic datasets of a low cluster intermix

Best setting: $\alpha = \beta = 1.9$	Fuzzy clustering accuracy	Clustering accuracy of defuzzified version	Time (sec)
4 features	96.27%	100%	8.27
+ 4 noise	96.03%	100%	10.23
+ 8 noise	95.79%	100%	12.11
+ 12 noise	72.56%	94.23%	15.53

Furthermore, after adding 8 noisy features to 4 real ones, the accuracy decreases on average by less than 1%. The clustering accuracy result of the defuzzified version has reached 100% of accuracy on all the 20 synthetic datasets generated. In this case, 8 noise features, which is twice as much as the number of features describing data, do not affect the clustering accuracy result at all. These results support the validity of the method and a high resistance to the noise added to data.

4.4.3. Synthetic datasets: high intermix case. In order to consider a more challenging case – high cluster intermix -

20 synthetic datasets of the same size have been constructed. In this case, *iMFWK*-Means, in general, identifies more initial centroids than have been constructed by the Synthetic Data generator. However, the inconsistency may be considered as reasonable result. Specifically, clusters have been constructed as points of multivariate normal distribution with a high cluster intermix, so objects in the intersection of two or more clusters may automatically form a new cluster. In this case, it is an advantage of the proposed algorithm that it can detect similarities of objects in the intersection of artificially constructed clusters and combine them to a new cluster. However, in order to compare the accuracy of a clustering result, the number of clusters was pre-specified. Both the fuzzy clustering average accuracy and the accuracy of its defuzzified version are presented in Table 7.

Table 7. The fuzzy clustering accuracy and the for the synthetic datasets of a high cluster intermix

Best setting: $\alpha = \beta = 2.9$	Fuzzy clustering accuracy	Clustering accuracy of defuzzified version	Time (sec)
4 features	81.09%	98.3%	7.17
+ 4 noise	78.05%	94.7%	13.01

As one can see from Table 10, the fuzzy clustering accuracy for the datasets of a high cluster intermix is not as close to 100% as it is for the datasets of a low cluster intermix. However, it is quite stable against added noise features. The reason for the decrease in accuracy may refer to the fact that the proposed algorithm, in general, finds more initial centroids that have been constructed by the Synthetic Data generator.

On the other hand, the clustering accuracy of the defuzzified version, even in the case of a high cluster intermix, is characterized by high levels of accuracy. On average, only 5 objects out of 300 have been misclassified, which gives an accuracy of 98.3%. After adding 4 noise features to 4 real ones, the accuracy decreases by 4% on average, which supports the view of the robustness of the method against added noise.

4.4.4. Large scale synthetic datasets. In order to evaluate the performance of the method on large scale datasets, only the case of a low cluster intermix has been considered. Table 8 represents both the fuzzy clustering average accuracy and the average clustering accuracy of its defuzzified version achieved on datasets of a specified size. The results on large scale datasets illustrate the high reliability of the proposed method. Specifically, the fuzzy clustering accuracy result keeps stable when the size of the dataset is increased, fluctuating around 99%. Moreover, the clustering accuracy of the defuzzified version on datasets of a low cluster intermix is always 100%. Sometimes the algorithm defines more clusters than has been designed by the Synthetic Data generator. However, the structure of clusters stays unchanged. In other words, the proposed algorithm just divides one generated cluster into several ones, treating it as a collection of clusters. Therefore, there is no misclassification of objects for the datasets of a low cluster intermix. Yet, the computation time increases when moving to a higher data dimension.

Table 8. The fuzzy clustering accuracy for the large scale synthetic datasets of a low cluster intermix

# of objects	# of features	Fuzzy clustering accuracy	Clustering accuracy of defuzzified version	Time
600	4	98.9%	100%	15.5 sec
900	4	99.1%		29.8 sec
3000	10	99.3%		6 min
3000	20	98.5%		10 min

Summarizing, on synthetic datasets of a low cluster intermix, the value of fuzzy clustering accuracy is more than 96% on average, rising to around 99% when moving to larger data sizes. These results show that the method is the reliable, since it is much harder to reach 100% recovery within a fuzzy case. The clustering accuracy of the defuzzified version on synthetic datasets of a low cluster intermix is always equal to 100% for both small and large scale datasets. Moreover, these results, in general, are replicated when datasets are supplied with noisy features. On synthetic datasets of a high cluster intermix the value of fuzzy clustering accuracy tends to be lower. However, the clustering accuracy of the defuzzified version still exceeds 98% for this type of data.

5. Summary

Overall, the experiment results show that the proposed Minkowski metric Fuzzy Weighted *K*-Means versions are quite competitive when compared to the other versions of *K*-Means. In general, *MFWK*-Means method provides a wider range of the exponent values of α and β, corresponding to the highest accuracy result, and it is less sensitive to the initial conditions. It is worth mentioning that the best accuracy in our experiments is achieved at $\alpha = \beta$. The feature weighting procedure assigns considerably higher weights to informative features, causing a successful clustering result for the original datasets, both real and synthetic ones. In the case of experiments on the datasets with added noise features, the algorithm assigns close-to-zero values to weights of the noise features in all the datasets, so they do not affect the

accuracy of clustering result significantly. Furthermore, the method is much superior to the versions of *K*-Means under consideration on datasets supplied with noisy features. In our follow-up work, we plan to evaluate and validate our method by applying it to a broader range of datasets, including the online handwritten data provided by the International UNIPEN Foundation research community and large astronomical datasets generated by multi-sensor networks. To improve the algorithm's capacity, we are going to develop: (a) faster procedures for computing Minkowski centers, (b) use a semi-supervised procedure for finding best values of Minkowski exponent, and (c) explore the possibilities of using multiple data samples and ensemble methods for setting cluster centers at iterations of the method.

6. References

[1] B. Mirkin, Clustering: A Data Recovery Approach. Chapman and Hall/CRC, Francis and Taylor, Roca Baton, Fl., 2012.

[2] L. Kaufman, P.J. Rousseeuw, Finding Groups in Data: An Introduction to Cluster Analysis. Wiley-Interscience, 342 p., 1990.

[3] J. Hartigan, M. Wong, A k-means clustering algorithm. Yale University, New Haven, Connecticut, USA, 1979.

[4] J. Bezdek, Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum, NY, 1981.

[5] Jain, Data clustering: 50 years beyond K-means. Pattern Recognition Letters 31 (8), 651–666 pp., 2010.

[6] G. De Soete, OVWTRE: A Program for Optimal Variable Weighting for Ultrametric and Additive Tree Fitting. Journal of Classification 5, 101-104 pp., 1988.

[7] V. Makarenkov, P. Legendre, Optimal variable weighting for ultrametric and additive trees and K-Means partitioning. Journal of Classification 18, 245-271 pp., 2001.

[8] E. Mueller, S. Guennemann, I. Assent, T. Seidl, Evaluating clustering in subspace projections of high dimensional data. Proceedings of the VLDB Endowment 2 (1), 1270–1281 pp, 2009.

[9] J.Z. Huang, J. Xu, M. Ng, and Y. Ye, Weighting Method for Feature Selection in K-Means. In: H.Liu and H. Motoda (Ed.) Computational methods of Feature Selection, Chapman & Hall/CRC, 193-209 pp., 2008.

[10] Y. Chan, W. Ching, M. Ng, and J. Huang, An optimization algorithm for clustering using weighted dissimilarity measures. Pattern Recognition 37:5, 943-952 pp., 2004.

[11] J. Huang, M. Ng, H. Rong, Z. Li, Automated variable weighting in k-means type clustering. IEEE Transactions on Pattern Analysis and Machine Learning 27 (5), 657–668 pp., 2005.

[12] R. Cordeiro de Amorim, B. Mirkin, Minkowski Metric, Feature Weighting and Anomalous Cluster Initializing in K-Means Clustering. Pattern Recognition 45, 1061-1075 pp., 2012.

[13] D. Pelleg, A. Moore, Xmeans: extending kmeans with efficient estimation of the number of clusters. Proceeding of the Seventeenth International Conference on Machine Learning, 727–734 pp., 2000.

[14] G. Hamerly, C. Elkan. Learning the k in k-means. Advances in Neural Information Processing Systems, v.17, 2003.

[15] H. Lei, L. Tang, J. Iglesias, S. Mukherjee, S. Mohanty, S-means: similarity driven clustering and its application in gravitational-wave astronomy data mining. Proceedings of the Int. Workshop on Knowledge Discovery from Ubiquitous Data Streams (IWKDUDS 2007), Warsaw, Poland, 2007.

[16] M. Chiang, B. Mirkin, Intelligent choice of the number of clusters in k-means clustering: An experimental study with different cluster spreads. Journal of Classification, 27:1, 1-38 pp., 2010.

[17] R. Krishnapuramd, J. Keller, A possibilistic approach to clustering. IEEE Transactions on Fuzzy Systems, Vol. 1, No. 2, 98-110 pp., 1993.

[18] N. Pal, K. Pal, J. Bezdek, A mixed c-mean clustering model. In IEEE International Conference on Fuzzy Systems, 11–21 pp., 1997.

[19] H. Liu, D. Wu, J. Yih, S. Liu, Fuzzy Possibility C-Mean Based on Complete Mahalanobis Distance and Separable Criterion". Eighth International Conference on Intelligent Systems Design and Applications, 89-94 pp., 2008.

[20] Z. Li, J. Yuan, W. Zhang, Fuzzy C-Mean Algorithm with Morphology Similarity Distance. Proceedings of the Sixth International Conference on Fuzzy Systems and Knowledge Discovery 3, 90–94 pp., 2009.

[21] S. Nascimento, R. Felizardo, B. Mirkin, Laplacian normalization for deriving semantic fuzzy clusters with an additive spectral approach. Expert systems, 2012.

[22] R. Hathaway, J. Bezdek, Nerf c-means: non-Euclidean relational fuzzy clustering. Pattern Recognition, 27 (3), 429–437 pp., 1994.

[23] R. Brouwer, A method of relational fuzzy clustering based on producing feature vectors using FastMap. Information Sciences: an International Journal, v.179 n.20, 3561-3582 pp., 2009.

[24] A. Asuncion, D. Newman, UCI Machine Learning Repository/http://www.ics.uci.edu/mlearn/MLRepository.htmlS. Irvine, CA: University of California, School of Information and Computer Science, 2007.

[25] E. Kovaleva, B. Mirkin, Bisecting K-Means and 1D Projection divisive clustering: a unified framework and experimental comparisons (submitted for publication).

[26] H. Bock, Clustering methods: a history of K-means algorithms. Selected Contributions in Data Analysis and Classification, 161-172 pp., 2007.

[27] D. Steinley, K-means clustering: a half-century synthesis. British Journal of Mathematical and Statistical Psychology, Vol. 59, Nr. 1Blackwell Publishing Ltd, 1-34 pp., 2006.

[1]The first author is partially supported by Laboratory for Structural Methods of Data Analysis in Predictive Modeling, MIPT, RF government grant, ag. 11.G34.31.0073.