



Local means-based fuzzy k -nearest neighbor classifier with Minkowski distance and relevance-complementarity feature weighting

Mahinda Mailagaha Kumbure¹ · Pasi Luukka¹

Received: 4 June 2024 / Accepted: 10 August 2024
© The Author(s) 2024

Abstract

This paper introduces an enhanced fuzzy k -nearest neighbor (FKNN) approach called the feature-weighted Minkowski distance and local means-based fuzzy k -nearest neighbor (FWM-LMFKNN). This method improves classification accuracy by incorporating feature weights, Minkowski distance, and class representative local mean vectors. The feature weighting process is developed based on feature relevance and complementarity. We improve the distance calculations between instances by utilizing feature information-based weighting and Minkowski distance, resulting in a more precise set of nearest neighbors. Furthermore, the FWM-LMFKNN classifier considers the local structure of class subsets by using local mean vectors instead of individual neighbors, which improves its classification performance. Empirical results using twenty different real-world data sets demonstrate that the proposed method achieves statistically significantly higher classification performance than traditional KNN, FKNN, and six other related state-of-the-art methods.

Keywords Classification · Relevance · Complementarity · Fuzzy entropy · Local means · Machine learning

1 Introduction

Fuzzy k -nearest neighbor (FKNN) (Keller et al. 1985), a variation of the traditional k -nearest neighbor (KNN) (Cover and Hart 1967) classifier, is considered a more robust classifier than the traditional KNN (Derrac et al. 2015). In the traditional KNN classifier, each instance in the data set is assigned a single class label based on the majority class of its k nearest neighbors. By contrast, the FKNN classifier assigns a degree of membership to each instance in a specific class in the data set based the distances of its k nearest neighbors (Keller et al. 1985). These membership degrees are used as weights in the classification process, which makes the FKNN more robust to noise at the class boundaries (Keller et al. 1985; Maillo et al. 2020). The FKNN classification is an active topic of recent research and is being used in various applications (for

examples, see González et al. 2021; Kumar and Thakur 2021; Maillo et al. 2020). However, the classical FKNN method has several limitations, such as sensitivity to the choice of the membership function (Derrac et al. 2016), number of nearest neighbors (k) (Derrac et al. 2015), and difficulty in handling high-dimensional data (Maillo et al. 2020). To address these limitations, researchers, such as Kassani et al. (2017) and Biswas et al. (2018), have proposed several enhancements. Recently, Zeraatkar and Afsari (2021) introduced two novel extensions of the FKNN classifier by incorporating the concepts of interval-valued fuzzy (IVF) sets, intuitionistic fuzzy (IF) sets, and the resampling method known as SMOTE, with a focus on addressing class imbalance classification problems. The primary purpose of these new FKNN variants, SMOTE-IVF and SMOTE-IVIF, was to enhance the classification performance of instances in the minority class. Moreover, González et al. (2021) proposed a novel fuzzy KNN method, called MonFKNN, based on monotonic constraints to enhance classification performance by addressing the issue of class noise. Based on the concept of multiple pseudo-metrics of fuzzy parameterized fuzzy soft matrices (*fpfs*-matrices), Memis et al. (2022) introduced the fuzzy parameterized fuzzy soft KNN (FPFS-kNN) classifier. This

✉ Mahinda Mailagaha Kumbure
mahinda.mailagaha.kumbure@lut.fi

Pasi Luukka
pasi.luukka@lut.fi

¹ Business School, LUT University, Yliopistonkatu 34,
53850 Lappeenranta, Finland

classifier takes into account the impact of model parameters on classification. A distinctive feature of this method is its use of five distance measures within the *fpfs*-matrices to generate multiple k nearest neighbors. Additionally, Bian et al. (2022) proposed a new FKNN approach, the fuzzy KNN method with adaptive nearest neighbors (A-FKNN), which focuses on determining a fixed k value for each test data instance. The core idea of A-FKNN is to find the optimal k value for each training instance during the training phase and to build a decision tree, called the A-FKNN tree. During the testing phase, A-FKNN identifies the optimal k for each testing instance by searching through the A-FKNN tree and then runs FKNN with the optimal k for each testing instance. Regarding specific application context, several enhancements for the FKNN algorithm have also been proposed, for example, a boosted particle swarm optimization with FKNN (bSRWPSO-FKNN) classifier for predicting atopic dermatitis disease (Li et al. 2023) and a Harris Hawks optimization and Sobol sequence and stochastic fractal search based FKNN (SSFSHHO-FKNN) model for diagnosis of Alzheimer's disease (Zhang et al. 2023).

Furthermore, Kumbure et al. (2019) particularly focused on FKNN to address class imbalance problems. They proposed a new variant of FKNN that employs class-representative local means, which are locally representative of their respective classes, to enhance classification accuracy. This method was further improved in Kumbure et al. (2020) by introducing a Bonferroni mean-based local mean computation process that outperforms traditional FKNN and other competitive classifiers. In the present study, we focus on further improving the performance of the local means-based FKNN method.

Our research aims to design and develop a local means-based FKNN classifier that effectively addresses the noise and uncertainty in the data, thereby yielding improved performance. To achieve this goal, we employ a feature weighting process based on fuzzy entropy (De Luca and Termini 1971), similarity measures (Luukka 2011), and feature selection concepts, such as relevance and complementarity (Ma and Ju 2022; Vergara and Estevez 2014; Yu and Liu 2004), to assess feature importance. We also used Minkowski distance to calculate distance between instances and employed class representative local mean vectors from class subsets instead of individual nearest neighbors to find class memberships.

In the context of feature selection, the theoretical concepts of relevance and complementarity have been widely employed as effective methods to identify optimal feature subsets (Ma and Ju 2022). Feature relevance recognizes the features that carry out valuable information regarding the target variable (Vergara and Estevez 2014). By contrast, feature complementarity highlights that combining two or

more features, even those that may be individually insufficient, can collectively provide meaningful information about the class variable (Ma and Ju 2022; Vergara and Estevez 2014). Thus, we focus on these theoretic feature selection measures to generate weights for features across the importance of each feature by estimating relevance and complementarity information. This enables us to obtain better weights for features, which can then be utilized, in conjunction with distance calculation to identify more suitable nearest neighbors for the new instance. Furthermore, entropy is used to measure the level of uncertainty in the values of a feature (Luukka 2011; Vergara and Estevez 2014). This can be useful in cases where the data are uncertain or noisy, as it can help identify features that are more informative or relevant for classification. Therefore, we specifically focus on fuzzy entropy and similarity-based relevance and complementarity weighting methods to improve the accuracy of the proposed classifier.

The Minkowski distance, a generalized distance measure, is characterized by a specific parameter (called order parameter), making it more flexible and adaptable to various types of data distributions and feature spaces. Therefore, by using the Minkowski distance with an appropriate order parameter, can achieve better performance for nearest neighbor search in such data sets, as it can handle different types of data distributions more effectively. Accordingly, the Minkowski distance is employed in combination with fuzzy entropy and class prototypes to develop and propose the feature-weighted Minkowski distance-based local mean fuzzy k -nearest neighbor (FWM-LMFKNN) method.

The flowchart in Fig. 1 outlines the steps of the proposed FWM-LMFKNN method. The process consists of two main phases. The first phase generates feature weights by combining the effects of feature relevance and complementarity. The second phase performs the classification, which includes the following steps: calculating the Minkowski distance between training instances (X_i) and the query sample (y), finding the k nearest neighbors for each class (j), calculating the class representative local mean (V_j), determining the distance between local means and the query sample, computing class memberships, and finally classifying the query sample.

The main contributions of this paper can be summarized as follows:

- Feature weights that are generated using a combined effect of feature relevance and complementarity are used to weight the distances between testing and training instances in the learning process of the FKNN algorithm.
- Minkowski distance is adopted for distance calculation to identify the most reasonable nearest neighbors,

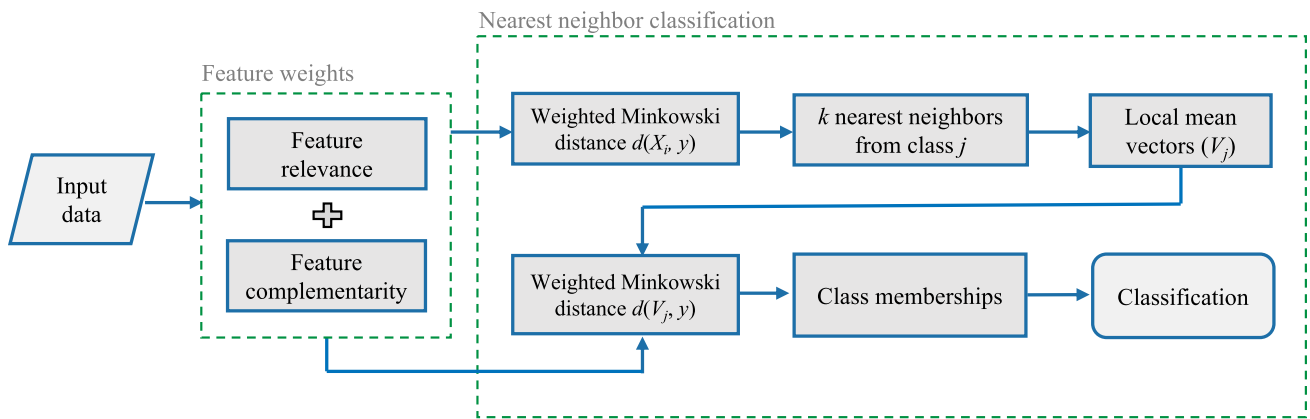


Fig. 1 Flowchart of the proposed method

thereby achieving better class representative local mean vectors (i.e., class prototypes).

- The decision rule on classification is made by considering the membership values, which are calculated using weighted Minkowski distances between the new instance and class representative local mean vectors.

The effectiveness of the proposed FWM-LMFKNN classifier is examined using various real-life data sets in both low- and high-dimensional spaces, covering binary- and multi-class problems. In the empirical analysis, the proposed method is compared with classical KNN (Cover and Hart 1967) and FKNN (Keller et al. 1985) methods and six more competitive methods, including LM-FKNN (Kumbure et al. 2019), LM-PNN (Gou et al. 2014), MLM-KHNN (Pan et al. 2017), FPFS-kNN (Memis et al. 2022), generalized mean distance-based KNN (GMD-KNN) (Gou et al. 2019), and interval-valued k-nearest neighbor (IV-KNN) (Derrac et al. 2015).

2 Preliminaries

This section briefly discusses the related KNN methods, fuzzy entropy, similarity measure, (relevance- and complementarity-based) feature weighting strategy, and the Minkowski distance.

2.1 Related KNN methods

The KNN algorithm (Cover and Hart 1967) is simple and effective, and it is based on the idea that an unseen data instance can be classified by looking at its closest “neighbors” from the training data set. It starts by calculating the distance between the unknown instance and all the instances in the training set. The distance can be computed using various distance metrics, but the Euclidean distance is the most commonly used (Derrac et al. 2015).

Once the distances are calculated, the k nearest neighbors are selected based on their nearness to the unknown data instance. Then, the class labels of the k nearest neighbors are counted, and the class with the majority number of votes is assigned to the unknown instance.

By contrast, the FKNN method (Keller et al. 1985) uses fuzzy set theory (Zadeh 1965) to assign class membership degrees to each data instance instead of crisp labels. The basic idea of the FKNN is to assign a membership degree to the unknown instance in each class based on the degree of similarity to its k nearest neighbors and their memberships to each class. The membership degree of the unknown instance (y) in each class (j) represented by its k nearest neighbors is calculated as follows:

$$u_j(y) = \frac{\sum_{i=1}^k u_{ij}(1/\|y - x_i\|^{2/(m-1)})}{\sum_{i=1}^k (1/\|y - x_i\|^{2/(m-1)})} \tag{1}$$

where x_i is the i th neighboring instance in the training set, and u_{ij} is the membership degree of i th neighboring instance in the j th class. $m > 1$ is the fuzzy strength parameter. There are two approaches to calculating u_{ij} : the crisp membership method and fuzzy membership method—detailed information about these methods can be found in Keller et al. (1985).

As previously noted, the KNN and FKNN methods are both affected by the value of parameter k (Yang and Sinaga 2021), and are particularly susceptible to the effects of outliers. To deal with these issues, particularly to the KNN classifier, the idea of class representative local mean (LM) vectors was first introduced by Mitania and Hamamotob (2006). The resulting LM-KNN classifier (Mitania and Hamamotob 2006) computes a local mean vector of nearest neighbors from each class. The unknown instance is then assigned to the class represented by the local mean vector that is closest to it. The effectiveness of this method has led to the development of several variants that aim to improve performance by addressing not only the impact of outliers

but also the sensitivity to the neighborhood size k . The local mean-based pseudo k -nearest neighbor (LM-PNN) (Gou et al. 2014) and multi-local means-based k -harmonic nearest neighbor (MLM-KHNN) (Pan et al. 2017) classifiers are examples of variants that have demonstrated improved classification performance. Based on the concept of class prototypes, recent enhancements to the classical FKNN method have been proposed, such as the multi-local power means fuzzy k -nearest neighbor (MLPM-FKNN) (Kumbure et al. 2019) and Bonferroni mean-based fuzzy k -nearest neighbor (BM-FKNN) (Kumbure et al. 2020) methods. These methods have been successful in achieving appropriate local class prototypes by incorporating various mean operators, such as generalized and Bonferroni means.

In this paper, our objective is to enhance the performance of the local means-based FKNN method by incorporating the Minkowski distance measure, class prototypes, and a feature weighting scheme.

2.2 Minkowski distance

Minkowski distance is a generalization of the Euclidean distance and Manhattan distance. It is a measure of the distance between two points in a metric space, which is defined by a norm. The Minkowski distance between two instances x_i and x_r in d -feature space is defined as:

$$d_{\text{mink_dis}}(x_i, x_r) = \left(\sum_{l=1}^d |x_i^l - x_r^l|^p \right)^{1/p} \quad \text{for } p \geq 1 \quad (2)$$

where the parameter p is called the order of the Minkowski distance. By using different values for p , we can define several different distances; for example, when $p = 2$, the Minkowski distance is equivalent to the Euclidean distance, which is the most commonly used distance measure for continuous features. Further, we can obtain the Manhattan distance by setting $p = 1$ and harmonic distance with $p = -1$ as examples. Due to this special property of Minkowski distance, it has been used in many applications, such as Bergamasco and Nunes (2019) and Gueorguieva et al. (2017). Additionally, the weighted Minkowski distances can be defined with feature weights w^l for $l = 1, \dots, d$ according to:

$$d_{\text{mink_dis}}^w(x_i, x_r) = \left(\sum_{l=1}^d w^l |x_i^l - x_r^l|^p \right)^{1/p} \quad \text{for } p \geq 1 \quad (3)$$

2.3 Fuzzy entropy

Entropy (H) is a concept used in information theory to measure the uncertainty or randomness of a system or feature (Vergara and Estevez 2014). Entropy, usually

discussed in a probability space (De Luca and Termini 1971), measures the amount of information required to describe the outcome of a random feature. The higher the entropy, the more uncertain or unpredictable the feature is. Fuzzy entropy, first defined by De Luca and Termini (1971), is an expanded version of classical entropy in the fuzzy sets theory. It is a measure that quantifies the degree of fuzziness of a fuzzy set (Al-sharhan et al. 2001). It is defined based on the concept of Shannon entropy (Shannon 1948), which is a measure of randomness in a probability distribution. However, fuzzy entropy differs from Shannon entropy, as it deals with vagueness and ambiguity uncertainties rather than probabilistic concepts (Al-sharhan et al. 2001). According to De Luca and Termini (1971) and Luukka (2011), the fuzzy entropy (h) can be defined for a given fuzzy set A defined over a universe U as:

$$h(A) = - \sum_{i=1}^n (\mu_A(x_i) \log \mu_A + (1 - \mu_A(x_i)) \log(1 - \mu_A(x_i))) \quad (4)$$

where, $\mu_A(x_i)$ represents the membership degree of an element x_i in the set A . The fuzzy entropy has been used to find feature importance concerning the target variable in the feature selection process (Luukka 2011; Lohrmann et al. 2018). It has also been applied for classification problems, especially to improve the KNN performance; for example, see Morente-Molinera et al. (2017).

2.4 Similarity measure

In feature selection, a similarity measure quantifies the degree of closeness or correlation between two features. It helps assess how similar or related two features are to each other. Łukasiewicz similarity (Łukasiewicz 1970) is a specific similarity measure based on the Łukasiewicz t-norm, which is a triangular norm used in fuzzy set theory (Zadeh 1965). For our study, the Łukasiewicz similarity is utilized together with fuzzy entropy to find feature relevance, and it can be defined according to Luukka et al. (2001) as follows:

$$S(x, y) = (1 - |x^p - y^p|)^{1/p} \quad (5)$$

where $x, y \in [0, 1]^1$ and $x, y \in \mathbb{R}^d$. This is chosen because it satisfies monotonicity, symmetricity, and transitivity properties (Luukka et al. 2001).

¹ Note that since similarity degree $s \in [0, 1]$, it can be used with fuzzy entropy measures even though it is initially defined for fuzzy sets.

2.5 Mutual information

Mutual information is a well-known method for measuring the amount of information one random feature provides compared to another feature (Vergara and Estevez 2014). This notion has been dominant and valuable in the context of feature selection, where mutual information is measured for each feature concerning the target variable. Features with higher mutual information are considered more relevant, as they contribute more information to predicting or classifying the target variable. In Meyer et al. (2008) and Vergara and Estevez (2014), for given X and Y two random features, mutual information (I) is defined as follows:

$$I(X; Y) = \begin{cases} H(X) - H(X \setminus Y) \\ H(Y) - H(Y \setminus X) \\ H(X) + H(Y) - H(X, Y) \end{cases} \quad (6)$$

where $H(X)$ and $H(Y)$ represent the entropy of features X and Y , respectively. $H(X, Y)$ denotes the joint entropy of X and Y , while $H(X \setminus Y)$ indicates the conditional entropy of X given Y . $H(Y \setminus X)$ is the conditional entropy of Y given X . $I(X; Y)$ measures the degree of correlation between features X and Y or the amount of information X covers about Y .

2.6 Feature relevance and complementarity

This subsection briefly discusses the concepts of relevance and complementarity in the context of feature selection. Before briefly describing these concepts, we first introduce some basic notations and terminologies in Table 1.

The sets mentioned in Table 1 are related as follows: $X = f_i \cup A \cup \neg\{f_i, A\}$, $\emptyset = f_i \cap A \cap \neg\{f_i, A\}$.

2.6.1 Relevance

A feature is considered relevant if individually or together with other features it provides information about class variable C . There are many definitions of “relevance” in the literature, but roughly, they can be divided into probabilistic framework (Kohavi and John 1997) and mutual information framework (Meyer et al. 2008). The

probabilistic framework defines three levels of relevance: strongly relevant, weakly relevant, and irrelevant features. Strongly relevant features give unique information about C and cannot be replaced by other features. Weakly relevant features also provide information about C , but other features can replace them without losing information about C . Irrelevant features do not give information on C , and they can be removed without losing any vital information. Similarly, a mutual information framework is defined into these three categories as given in Eq. (6).

2.6.2 Complementarity

The notion of complementarity (also called information synergy) signifies that the working of two features together could carry more information than the sum of their individual values to the target variable (Singha and Shenoy 2018). It is used to measure the degree of interaction between an individual feature f_i and feature subset A given C (Vergara and Estevez 2014). This can be again measured, for example, by using mutual information $[I(f_i; A|C)]$. One way to understand the complementarity effect is the following: when information that A has about C is greater when it interacts with feature f_i compared to when it does not interact, then complementarity information is present.

Feature selection is a critical step in preparing data for machine learning models. Most feature selection approaches are usually based on notions of relevance, redundancy, and complementarity (Sun et al. 2017; Singha and Shenoy 2018). Redundancy occurs when multiple features provide the same or similar information about the class variable (Singha and Shenoy 2018). Redundant features are often highly correlated with each other and do not offer new or additional information to the model. Therefore, in a typical feature selection task, redundant features are identified and removed. However, to create the feature weighting criterion in the proposed method, we focus solely on relevance and complementarity concepts and do not consider the redundancy measure. This is because this approach avoids the calculation of pairwise correlations and interdependencies between features, reducing computational time. Especially in cases where the number of features is very large, redundancy checks can be more computationally expensive. Moreover, in some sense, complementarity is closely related to redundancy (Sun et al. 2017), and it is also known that complementarity approach is efficient than redundancy approach (Singha and Shenoy 2018). For these reasons, relevance and complementarity are considered in our study to have an efficient and effective feature weighting strategy. In general, relevant and complementary features maximize the mutual information with the class variable, ensuring that the

Table 1 Basic notations and their meaning

Notation	Meaning
A	Current subset of selected features
f_i	Feature to be added or deleted from the subset A
$\neg f_i$	All features in X except f_i , $\neg f_i = X \setminus f_i$
$\{f_i, A\}$	Subset consisting from feature f_i and subset A

feature informativeness is observed reasonably well (Singha and Shenoy 2018).

3 The proposed FWM-LMFKNN classifier

This section introduces the feature-weighted Minkowski distance-based local mean fuzzy k-nearest neighbor (FWM-LMFKNN) method, which is based on the concepts of feature weighting, Minkowski distance, and class representative local mean vectors. The feature weighting employs fuzzy entropy and similarity measures, incorporating relevance and complementarity notions. We begin by defining the calculation of feature relevance and complementarity and, subsequently, the weighting strategy for the new classifier.

3.1 Feature weighting based on relevance and complementarity

The calculation of relevance is based on fuzzy entropy and similarity measures. Suppose a training data set $\{X_i \in \mathbb{R}^d, \omega_i\}_{i=1}^n$ consisting n number of instances in d -dimensional feature space [i.e., $X_i = (x_i^1, x_i^2, \dots, x_i^d)$] and t different classes [i.e., $\omega \in (C = (c_1, c_2, \dots, c_t))$]. Given these, relevance of features to class variable is calculated using the following steps:

- (1) Normalize feature data into unit interval, that is, $X_i \rightarrow [0, 1]^d$
- (2) Obtain ideal vectors² $v_j \in \mathbb{R}^d$ for each class j from the training set data using arithmetic mean.

$$v_j = \left(\frac{1}{n_j} \sum_{i=1}^{n_j} X_i \right), \quad j = 1, \dots, t \quad (7)$$

where n_j denotes the number of instances belonging to class j , that is, the mean calculation is restricted to only those samples that belong to class j . X_i is an instance belonging to j th class.

- (3) Compute similarity measure S from each training instance $X_i = (x_i^1, x_i^2, \dots, x_i^d)$ to corresponding ideal vector $v_j = (v_j^1, v_j^2, \dots, v_j^d)$ as follows:

$$S(x_i, v_j) = \frac{1}{d} \sum_{l=1}^d (1 - |(x_i^l)^p - (v_j^l)^p|)^{1/p} \quad (8)$$

for $x_i, v_j \in [0, 1]^d$. For the sake of simplicity, we use $p = 1$ in the proposed method. Notice that the matrix

$[S]_{n \times d \times t}$ needs to be reshaped by reducing the dimensions as $n \times t$ and d .

- (4) Compute relevance (denoted by f^{REL}) by measuring fuzzy entropy (h) for each feature $l \in [1, d]$ using the similarity values ($S(x_i, v_j)$) from the previous step and Eq. (4) as:

$$f^{REL} = h(l) = - \sum_{i=1}^n (S^l \log S^l + (1 - S^l) \log(1 - S^l)) \quad (9)$$

where S^l is the similarity for feature l of a sample X_i with ideal vector v_j of class j is summed over all samples ($i = 1, \dots, n$) and classes ($j = 1, \dots, t$).

The calculation of complementarity (f^{COMP}) is performed for $l = 1, \dots, d$ using following steps:

- (1) Compute intersection between l th feature and all the other features by using algebraic product, that is,

$$x_i^l \cap x_i^{\hat{l}} = x_i^l x_i^{\hat{l}} \quad (10)$$

where $\hat{l} = 1, \dots, d$ and this way, the intersection matrix T_1 is formed.

- (2) Add class variable C to the intersection matrix as $T_2 = \{T_1, C\}$.
- (3) Compute correlation³ between features, $Corr(T_2)$.
- (4) Subtract identity matrix I from the correlation matrix and take absolute values from it, that is, $T_3 = |Corr(T_2) - I|$.
- (5) Find the maximum correlation, $H_c = \max(T_3)$
- (6) Compute complementarity value (f^{COMP}) using the negation of correlation, such as $f^{COMP} = 1 - H_c$, and collect the complementarity information in the matrix.

In the proposed method, we generate feature weights focusing on an aggregate effect of relevance and complementarity in the learning process. Both relevance and complementarity have positive influence on feature weights, in fact, their combined effect could offer a best trade-off considering model's performance and flexibility with small data sets (Singha and Shenoy 2018). Moreover, combining relevance with complementarity may identify individually relevant features to the class variable and provide unique information when considered simultaneously. This strategy can weigh features by considering different characteristics of the class distribution and features. This ensures the identification of more suitable nearest neighbors to the query instance based on distance, ultimately enhancing the classification performance of the method.

² The ideal vectors represent the mean vectors of each class subset in the training set.

³ For correlation, Kendall (1938) rank correlation is used.

3.2 The FWM-LMFKNN algorithm

Based on the fundamental concepts of the previously discussed classifiers, the feature weighting strategy, Minkowski distance, and local means, we propose an extension of the FKNN method: the FWM-LMFKNN classifier.

The proposed classification method utilizes a two-phase approach: it calculates weights for each feature, and then it employs nearest neighbor classification to an unknown instance. In the first phase, a strategy incorporating the concepts based on relevance and complementarity is utilized to calculate the feature weights. The second phase involves finding sets of k nearest neighbors for an unknown instance (y) from each class based on the feature-weighted Minkowski distances between y and each training instance, followed by calculating local mean vectors for each set of k nearest neighbors. Next, the Minkowski distance between y and each local mean vector is calculated (feature weights are also applied), and fuzzy memberships for each class concerning y are computed. Finally, y is classified into the class with the highest membership degree. A formal definition of the proposed FWM-LMFKNN algorithm can be defined as follows. Suppose that we have a training data set $\{x_i, \omega_i\}_{i=1}^n$ that is composed of n number of instances in d -dimensional feature space [i.e., $x_i = (x_i^1, x_i^2, \dots, x_i^d)$] and t different classes [i.e., $\omega \in (c = (c_1, c_2, \dots, c_t))$]. In the FWM-LMFKNN method, the class label ω^* for a given unknown instance [$y = (y^1, y^2, \dots, y^d)$] is achieved as described below.

Phase 1:

Calculate of relevance and complementarity measures using the notions presented in Sect. 3.1 and consider a combined effect of those measures by summing them (i.e., $f^{REL} + f^{COMP}$). It is well-known that higher entropy values correspond to lower similarity and increased uncertainty in the corresponding feature, while lower entropy values indicate greater similarity and increased importance. This is reflected in the relevance value, which is based on fuzzy entropy and similarity. Therefore, the complementary values of relevance are employed as feature weights (w). This principle holds for complementarity as well. Accordingly, for a given feature l , the weight (w^l) can be defined as:

$$w^l = 1 - \frac{(f^{REL} + f^{COMP})}{\sum_{l=1}^d (f^{REL} + f^{COMP})} \tag{11}$$

Phase 2:

- (1) Compute the Minkowski distance, $d_{mink_dis}(y, x_j)$ between y and each training instance x_j according to:

$$d_{mink_dis}(y, x_i) = \left(\sum_{l=1}^d w^l |y^l - x_i^l|^p \right)^{1/p} \tag{12}$$

As shown in this computation, a feature weight w^l , which is computed in the previous phase, is allocated to distance, which allows instances that are closer to y to be given a higher weight (lower distance), while instances that are farther away are given a lower weight (higher distance). This can help to mitigate the effects of noise and outliers in the training data.

- (2) Find the set of k nearest neighbors, $\{x_{ij}^{nm}\}_{i=1}^k$ of y from each class j based on the ordered distances computed in the previous step. Here, nm stands for “nearest neighbor.”
- (3) Compute a local mean vector (z) using the set of k nearest neighbors in each class j according to:

$$z_j = \frac{1}{k} \sum_{i=1}^k x_{ij}^{nm} \tag{13}$$

- (4) Compute the Minkowski distances between y and each local mean vector as:

$$d_{mink_dis}(y, z_j) = \left(\sum_{l=1}^d w^l |y^l - z_j^l|^p \right)^{1/p} \tag{14}$$

- (5) Compute fuzzy membership (u_j) to class j concerning y using the distances $d_{mink_dis}(y, z_j)$ for $j = 1, 2, \dots, t$ according to:

$$u_j(y) = \frac{\sum_{j=1}^t u_{jj} (1/d_{mink_dis}(y, z_j))^{2/(\gamma-1)}}{\sum_{j=1}^t (1/d_{mink_dis}(y, z_j))^{2/(\gamma-1)}} \tag{15}$$

where u_{jj} is 1 for the known class and 0 is for the other classes. Notice that here u_{jj} has twice j , and it is because number of classes and the number of local mean vectors are the same.

- (6) Return the class ω^* of y , which has the highest membership degree [i.e., $\omega^* = \arg \max_{\omega_i} u_i(y)$].

By incorporating a combination of relevance and complementarity into the feature weighting process, the FWM-LMFKNN method can effectively handle uncertainty and vagueness in the data, leading to more reasonable decision boundaries. Using the Minkowski distance metric also allows for more flexible and powerful distance computations, further improving the classifier’s performance. The steps of the proposed method discussed under Phase 1 and Phase 2 are summarized as Algorithm 1 and Algorithm 2.

Algorithm 1 FWM-LMFKNN (Phase 1:Feature weights generation)

Input: training data $\{X_i \in \mathbb{R}^d, \omega_i\}_{i=1}^n$, $X_i = (x_i^1, x_i^2, \dots, x_i^d)$, $\omega \in (c_1, c_2, \dots, c_t)$
Output: feature weight, w^l
Start

- 1 data normalization: $X_i \rightarrow [0, 1]^d$
- for** $j \leftarrow 1$ **to** t **do**
- 2 obtain ideal vectors, $v_j = \left(\frac{1}{n_j} \sum_{i=1}^{n_j} X_i\right) \in \mathbb{R}^d$
- for** $i \leftarrow 1$ **to** n **do**
- 3 compute similarity measure, $S\langle x_i, v_j \rangle = \frac{1}{d} \sum_{l=1}^d (1 - |(x_i^l)^p - (v_j^l)^p|)^{1/p}$
- 4 **end**
- 5 **end**
- 6 **for** $l \leftarrow 1$ **to** d **do**
- 7 (i) compute feature relevance by using fuzzy entropy,

$$f^{REL} = h(l) = - \sum_{i=1}^n (S^l \log S^l + (1 - S^l) \log(1 - S^l))$$
- (ii) compute feature intersection, $x_i^l \cap x_i^{l'} = x_i^l x_i^{l'}$ and generate intersection matrix T_1
- (iii) add class variable C to the matrix T_1 as $T_2 = \{T_1, C\}$
- (iv) compute correlation between features, $corr(T_2)$
- (v) compute $T_3 = |Corr(T_2) - I|$
- (vi) find the maximum correlation, $H_c = \max(T_3)$
- (vii) compute complementarity value, $f^{COMP} = 1 - H_c$
- (viii) compute feature weights,

$$w^l = 1 - \frac{(f^{REL} + f^{COMP})}{\sum_{l=1}^d (f^{REL} + f^{COMP})}$$
- 8 **end**
- 9 **return** w^l

Algorithm 2 FWM-LMFKNN (Phase 2: Classification)

Input: training data $\{X_i \in \mathbb{R}^d, \omega_i\}_{i=1}^n$, unknown instance $y = (y^1, y^2, \dots, y^d)$, k , and p

Output: class label ω^* for y

Start

- 1 **for** $i \leftarrow 1$ **to** n **do**
- 2 compute $d_{mink.dis}(y, x_i) = (\sum_{l=1}^d w^l |y^l - x_i^l|^p)^{1/p}$
- 3 **end**
- 4 **for** $j \leftarrow 1$ **to** t **do**
- 5 (i) find a set of k nearest neighbors $\{x_{ij}^{nn}\}_{i=1}^k$ of y
 (ii) compute a local mean vector $z_j = \frac{1}{k} \sum_{i=1}^k x_{ij}^{nn}$
 (iii) compute the Minkowski distances between y and z_j as,

$$d_{mink.dis}(y, z_j) = (\sum_{l=1}^d w^l |y^l - z_j^l|^p)^{1/p}$$

 (iv) compute fuzzy membership (u_j) concerning y as,

$$u_j(y) = \frac{\sum_{j=1}^t u_{jj}(1/d_{mink.dis}(y, z_j)^{2/(\gamma-1)})}{\sum_{j=1}^t (1/d_{mink.dis}(y, z_j))^{2/(\gamma-1)}}$$
- 6 **end**
- 7 **return** ω^* such that $\omega^* = \arg \max_{\omega_i} u_i(y)$

To demonstrate the impact of using Minkowski distance in the proposed method, a simple experiment was conducted, as indicated by Karimi and Torabi (2022), by selecting the Vehicle data set from UCI repository (Dheeru and Taniskidou 2017). Three data instances, labeled x_1, x_2 , and x_3 , were selected from the data set. The Minkowski distances between x_1 and x_2 and between x_2 and x_3 were then calculated for varying values of p , as depicted in

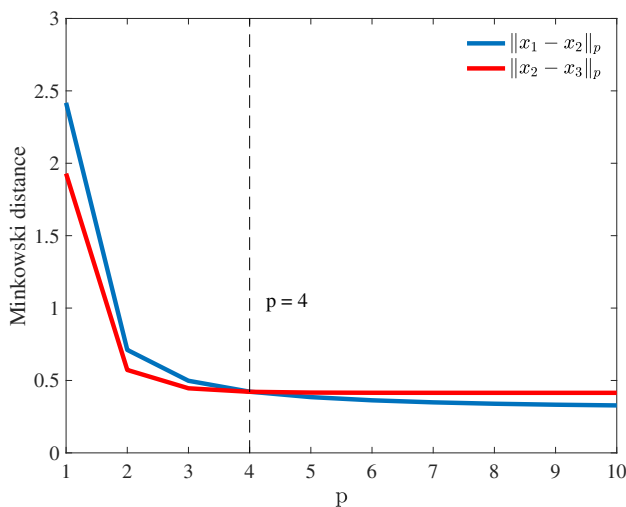


Fig. 2 Minkowski distance between x_1 and x_2 , and x_2 and x_3 with respect to different values of p

Fig. 2. The results, as shown in the figure, indicate that when p is less than 4, x_3 is closer to x_2 than x_1 . Conversely, when p is greater than 4, x_1 is closer to x_2 than x_3 . Accordingly, when x_2 is used as a test instance with x_1 and x_3 serving as training instances, the performance of FWM-LMFKNN classifier can be highly dependent on the value of the parameter p . This clearly indicates that using the Minkowski distance rather than the Euclidean distance in the proposed method allows it to find nearest neighbors in a more flexible way.

Furthermore, incorporating class representative local mean vectors in the FWM-LMFKNN enhances the classifier’s robustness and effectively addresses challenges arising from data distribution, including class imbalance, feature noise, and the impact of outliers. Previous studies in Kumbure et al. (2019, 2020) have comprehensively examined and demonstrated the significance of utilizing class prototypes in the FKNN classifier. This strategy contributes to the classifier’s ability to handle complex data sets, providing a more reliable and accurate classification results.

3.3 Computational complexity analysis

In this subsection, we briefly discuss the computational complexities of the proposed FWM-LMFKNN method. Let us consider n , which indicates the number of training

instances characterized by c classes in d -dimensional feature space, and query sample y (the sample to be classified). The KNN (Cover and Hart 1967) classifier consists of a calculation of the distances from y to all training instances to find k nearest neighbors and then observes the majority class among them. Therefore, according to Guo et al. (2014, 2019), the computational complexity of the KNN method is $O(nd + nk + k)$. The FKNN (Keller et al. 1985) algorithm extends the KNN method by adding an additional step for computing the membership of y considering each class based on the distances between y and k nearest neighbors. Therefore, its computational complexity is $O(nd + nk + ck + c)$. Compared to the FKNN, the LM-FKNN consists of an additional step of the local mean computation using the set of k nearest neighbors from each class, thus it requires the computation of $O(nd + nk + cdk + cd + c)$. According to Duarte et al. (2019), the computational complexity of the Minkowski distance between two points of dimension d is $O(d)$. But when we have n data instances, it requires $O(nd)$. Then the computational complexity of the LM-FKNN method combined with the Minkowski distance is $O(2nd + nk + cdk + cd + c)$. The computation of feature relevance includes the calculation of ideal vectors from each class, the similarity measure, and fuzzy entropy. Therefore, it requires a computational complexity of $O(3ncd)$. The calculation of the feature complementarity contains the steps of forming an intersection matrix, computation of the correlation, and complementarity values. Therefore, its computational complexity is $O(nd^2 + n + n) \approx O(nd^2)$. The proposed FWM-LMFKNN combined those steps; therefore, its computational complexity is $O(2nd + nk + cdk + cd + c + 3ncd + nd^2)$, and when the constant terms are ignored and $n > > k, c, d$, it is $\approx O(nd + ncd + nk + nd^2)$.

Based on the above analysis, it is evident that the FWM-LMFKNN method requires more computation time, particularly compared to the classical KNN, FKNN, and LM-FKNN methods. The primary reason for this is the additional computation involved in generating feature weights using relevance and complementarity, as well as the use of the Minkowski distance measure.

4 Experiment

To evaluate the performance of the proposed classifier, a series of experiments were conducted using real-world data sets and comparing the results to well-established baseline models. The following sub-sections describe the data sets used, the models compared, the evaluation metrics, and the experimental procedure.

4.1 Data sets

We used 20 real-world data sets to evaluate the performance of the proposed approach. These data sets are freely available at the UCI (Dheeru and Taniskidou 2017) and KEEL (Alcala-Fdez et al. 2011) machine learning repositories. Table 2 provides a summary of the main characteristics of the data sets, including the number of instances, features, classes, and corresponding data repository. The data sets ranged in size from 106 to 5500 instances and encompass binary and multi-class problems.

4.2 Testing methodology

This study employed a thirty-fold holdout validation procedure across all experiments with the data. In each run, utilizing the stratified random sampling technique, the data set was randomly divided into training and testing sets, while 30% of instances were allocated to the test set. The average classification accuracies with a 95% level of confidence of 30 splits of each data set were reported as the final results. This experimental setup was adopted based on the indications by Gou et al. (2014, 2019) and Pan et al. (2017). As classification accuracy alone may not be sufficient in evaluating the performance of a classifier, additional measures such as sensitivity and specificity are often

Table 2 The used data sets from UCI and KEEL repositories

Data set	# instances	# features	# classes	Database
Appendicitis	106	7	2	KEEL
Blood	748	4	2	UCI
Bupa	345	6	2	KEEL
Ionosphere	351	34	2	UCI
Horse	368	27	2	UCI
Mammography	962	5	2	UCI
Vehicle	846	18	2	KEEL
Saheart	462	9	2	KEEL
Sonar	208	60	2	UCI
Spambase	4597	57	2	KEEL
Retinopathy	1151	19	2	UCI
WDBC	569	30	2	UCI
Balance	625	4	3	UCI
Car	1728	6	4	KEEL
Dermatology	366	34	6	UCI
Cleveland	297	13	5	KEEL
Segment	2310	18	7	KEEL
Tae	151	5	3	KEEL
Texture	5500	40	11	KEEL
Thyroid	215	5	3	UCI

Table 3 The accuracy (%) results of each classifier, along with the corresponding standard deviations (in brackets), in all tested data sets

Data	FWM-LMFkNN	FPFS-kNN (Memis et al. 2022)	GMD-kNN (Gou et al. 2019)	LM-FKNN (Kumbure et al. 2019)	FKNN (Keller et al. 1985)	KNN (Cover and Hart 1967)	MLM-KHNN (Pan et al. 2017)	LM-PNN (Gou et al. 2014)	IV-KNN (Derrac et al. 2015)
Appendicitis	87.40 (4.22)	85.83 (4.91)	86.98 (4.50)	87.19 (4.45)	84.90 (3.77)	87.19 (4.21)	84.79 (4.16)	86.67 (4.34)	85.52 (4.06)
Blood	77.65 (2.37)	77.11 (1.65)	75.24 (2.09)	77.62 (2.41)	76.26 (1.97)	77.17 (1.81)	74.81 (2.70)	75.19 (2.11)	71.32 (2.86)
Bupa	62.01 (4.35)	60.19 (4.63)	61.49 (4.50)	61.84 (4.50)	61.52 (3.85)	61.20 (4.64)	61.81 (3.89)	61.42 (4.33)	61.46 (4.47)
Ionosphere	90.98 (2.07)	88.22 (3.24)	89.11 (3.28)	89.81 (3.07)	86.00 (2.52)	86.00 (2.52)	90.22 (2.65)	88.86 (2.95)	85.65 (2.53)
Horse	87.37 (2.58)	85.63 (2.96)	85.25 (2.60)	86.52 (2.10)	87.11 (2.15)	86.90 (2.00)	85.84 (2.37)	85.43 (2.51)	87.35 (1.97)
Mammogram	77.31 (4.34)	73.80 (9.98)	68.45 (5.3)	69.47 (7.94)	75.65 (7.13)	75.20 (7.07)	69.71 (5.13)	68.61 (5.35)	75.58 (7.72)
Vehicle	94.46 (1.58)	92.40 (2.25)	93.71 (1.66)	93.98 (1.79)	92.95 (1.79)	92.59 (1.78)	93.95 (1.57)	93.65 (1.81)	92.78 (1.86)
Saheart	66.71 (3.58)	68.32 (2.61)	62.66 (3.13)	66.00 (3.39)	64.89 (2.47)	67.48 (1.93)	64.68 (3.78)	62.13 (3.47)	64.08 (2.46)
Sonar	87.53 (2.99)	83.33 (5.04)	86.61 (3.42)	85.16 (3.26)	84.89 (3.29)	86.29 (4.06)	86.29 (3.30)	86.56 (3.45)	84.89 (3.29)
Spambase	90.71 (0.91)	89.80 (0.81)	90.14 (0.68)	89.88 (0.69)	89.86 (0.72)	89.10 (0.75)	90.18 (0.79)	90.03 (0.67)	89.75 (0.71)
Retinopathy	65.25 (3.41)	64.96 (4.13)	59.47 (5.11)	61.17 (4.74)	63.73 (2.58)	63.96 (2.38)	59.78 (5.15)	59.36 (5.02)	63.53 (2.58)
WDBC	95.93 (2.18)	95.32 (2.51)	95.73 (2.03)	95.85 (1.77)	95.81 (2.09)	95.65 (2.26)	95.83 (1.88)	95.52 (2.15)	95.67 (2.40)
Balance	91.25 (1.12)	89.45 (1.00)	89.39 (0.98)	91.03 (1.15)	89.54 (0.73)	88.98 (1.32)	92.09 (1.26)	89.48 (0.92)	88.89 (1.27)
Car	75.13 (1.5)	76.47 (1.42)	75.57 (1.17)	73.41 (1.45)	72.70 (1.27)	72.70 (1.27)	69.91 (0.27)	75.69 (1.24)	70.08 (0.00)
Dermatology	97.79 (1.47)	93.05 (1.84)	97.48 (1.22)	97.51 (1.31)	96.96 (1.43)	96.84 (1.54)	97.54 (1.36)	97.39 (1.29)	96.74 (1.54)
Cleveland	59.20 (2.39)	56.63 (2.43)	56.10 (3.15)	57.65 (3.39)	57.73 (3.62)	56.82 (2.67)	55.87 (4.04)	55.91 (3.16)	57.92 (3.95)
Segment	96.42 (0.74)	95.05 (1.68)	96.34 (0.66)	96.16 (0.72)	96.16 (0.72)	96.16 (0.72)	96.38 (0.76)	96.40 (0.74)	96.16 (0.72)
Tae	55.58 (5.25)	55.65 (6.91)	57.54 (6.60)	54.28 (6.52)	57.10 (7.01)	54.42 (6.50)	56.59 (5.69)	57.46 (6.37)	56.81 (6.30)
Texture	96.46 (1.24)	95.57 (1.34)	96.36 (1.39)	96.35 (1.36)	95.62 (1.63)	95.62 (1.63)	96.53 (1.33)	96.39 (1.40)	95.62 (1.63)
Thyroid	94.95 (3.20)	94.58 (3.06)	94.27 (4.05)	93.91 (4.61)	93.44 (4.47)	92.81 (3.77)	94.17 (4.28)	94.37 (3.95)	93.44 (4.48)
Average	82.50 (2.57)	81.07 (3.28)	80.89 (2.88)	81.24 (3.03)	81.14 (2.77)	81.15 (2.74)	80.85 (2.82)	80.83 (2.86)	80.66 (2.84)

Table 4 The sensitivity (%) results of each classifier with optimal k and p values in all tested data sets

Data	FWM-LMFKNN	FPFS-kNN (Memis et al. 2022)	GMD-KNN (Gou et al. 2019)	LM-FKNN (Kumbure et al. 2019)	FKNN (Keller et al. 1985)	KNN (Cover and Hart 1967)	MLM-KHNN (Pan et al. 2017)	LM-PNN (Gou et al. 2014)	IV-KNN (Derrac et al. 2015)
Appendicitis	69.19 (4, 3)	66.50 (14)	70.19 (15)	68.93 (8)	65.12 (20)	71.84 (5)	65.96 (19)	69.63 (12)	66.23 (15)
Blood	56.69 (19, 2)	54.28 (19)	47.66 (20)	56.99 (19)	50.57 (19)	54.34 (20)	46.51 (20)	47.47 (20)	39.01 (18)
Bupa	60.41 (10, 2)	58.25 (6)	57.99 (20)	60.06 (9)	59.88 (20)	57.72 (20)	57.65 (10)	57.38 (20)	60.15 (20)
Ionosphere	88.79 (5, 1)	85.94 (1)	86.50 (20)	87.72 (15)	83.36 (1)	83.36 (1)	88.02 (20)	86.18 (20)	82.97 (1)
Horse	86.45 (20, 1.5)	86.29 (6)	86.47 (20)	85.67 (18)	87.27 (17)	87.27 (13)	86.78 (14)	86.83 (20)	87.66 (14)
Mammogram	76.38 (20, 1)	82.11 (20)	66.20 (4)	72.62 (20)	78.33 (18)	77.51 (20)	69.55 (20)	65.93 (5)	76.61 (16)
Vehicle	96.10 (5, 1)	94.71 (2)	95.48 (18)	95.49 (7)	95.06 (4)	94.80 (3)	95.52 (13)	95.43 (18)	94.80 (5)
Saheart	52.96 (19, 3.5)	57.75 (19)	44.90 (20)	51.83 (20)	48.84 (20)	56.02 (15)	48.78 (20)	43.86 (20)	46.57 (20)
Sonar	90.92 (2, 1.5)	88.67 (1)	91.50 (6)	90.10 (2)	89.72 (1)	84.60 (2)	90.89 (4)	91.38 (3)	89.72 (1)
Spambase	90.20 (19, 1)	91.48 (7)	91.22 (20)	91.83 (16)	90.80 (14)	90.43 (5)	91.75 (20)	91.18 (20)	90.70 (13)
Retinopathy	65.33 (18, 1)	62.15 (15)	58.06 (20)	60.18 (20)	60.40 (20)	60.86 (19)	58.75 (20)	57.99 (20)	59.79 (20)
WDBC	94.34 (11, 1.5)	93.39 (7)	94.24 (20)	94.22 (18)	95.22 (11)	94.85 (5)	94.20 (20)	93.88 (18)	95.19 (4)
Balance	71.70 (11, 2)	64.83 (11)	64.79 (20)	70.87 (13)	64.90 (19)	65.05 (18)	75.80 (20)	64.86 (20)	64.43 (19)
Car	59.50 (7, 1.5)	70.04 (16)	69.87 (7)	40.11 (20)	52.55 (9)	52.55 (9)	25.65 (1)	70.22 (8)	25.00 (1)
Dermatology	97.59 (8, 1)	92.68 (3)	97.23 (20)	97.22 (9)	96.92 (4)	96.53 (4)	97.26 (14)	97.15 (13)	96.71 (3)
Cleveland	30.38 (20, 1)	28.27 (19)	31.93 (8)	30.21 (20)	30.25 (11)	27.65 (13)	28.96 (19)	31.58 (8)	30.15 (15)
Segment	96.42 (2, 1.5)	95.03 (1)	96.34 (5)	96.16 (1)	96.16 (1)	96.16 (1)	96.37 (5)	96.40 (8)	96.16 (1)
Tae	71.45 (20, 1.5)	69.74 (6)	71.46 (7)	70.71 (1)	72.03 (17)	70.71 (1)	71.31 (2)	72.30 (10)	70.92 (2)
Texture	96.46 (4, 2.5)	95.57 (2)	96.36 (7)	96.35 (4)	95.62 (1)	95.62 (1)	96.53 (6)	96.39 (7)	95.62 (1)
Thyroid	91.46 (12, 1)	89.96 (8)	90.93 (19)	91.43 (8)	89.58 (7)	86.62 (7)	91.33 (20)	91.07 (20)	89.23 (6)

necessary to provide a more comprehensive evaluation (Kumbure et al. 2019, 2020). For this reason, in addition to the accuracy, we calculated sensitivity and specificity values in our analysis, as they are commonly used measures in this context.

To provide a comprehensive comparison, several well-established models were chosen as baselines for the proposed classifier. The models included classical KNN

(Cover and Hart 1967), FKNN (Keller et al. 1985), and six other competitive classifiers: MLM-KHNN (Pan et al. 2017), LM-PNN (Gou et al. 2014), LM-FKNN (Kumbure et al. 2019), FPFS-kNN (Memis et al. 2022), GMD-KNN (Gou et al. 2019) and IV-KNN (Derrac et al. 2015). The number of nearest neighbors (k) and the Minkowski distance parameter (p) were systematically varied during the validation process to optimize the performance of the

Table 5 The specificity (%) results of each classifier in all tested data sets

Data	FWM-LMFKNN	FPFS-kNN (Memis et al. 2022)	GMD-KNN (Gou et al. 2019)	LM-FKNN (Kumbure et al. 2019)	FKNN (Keller et al. 1985)	KNN (Cover and Hart 1967)	MLM-KHNN (Pan et al. 2017)	LM-PNN (Gou et al. 2014)	IV-KNN (Derrac et al. 2015)
Appendicitis	91.25	90.11	90.68	91.21	88.32	90.27	88.50	90.27	89.12
Blood	81.71	79.95	81.06	81.59	80.77	81.28	81.03	80.91	80.78
Bupa	63.06	62.02	63.37	63.03	62.65	63.08	64.02	63.53	62.50
Ionosphere	96.70	95.06	96.70	95.72	94.47	94.47	96.27	96.71	94.40
Horse	89.87	84.61	83.03	89.03	86.95	86.41	84.45	82.88	86.86
Mammogram	80.10	76.19	80.65	80.37	78.73	78.46	82.37	81.01	78.81
Vehicle	89.67	85.61	88.53	89.70	86.60	85.90	89.43	88.38	86.65
Saheart	72.14	71.19	68.89	70.99	68.62	70.23	70.75	68.53	68.19
Sonar	85.49	80.52	83.64	82.34	82.05	88.36	83.58	83.69	82.05
Spambase	91.66	87.31	88.47	87.00	88.45	87.06	87.81	88.26	88.31
Retinopathy	66.32	71.95	70.67	70.90	68.84	67.86	71.37	70.45	69.68
WDBC	97.20	96.97	96.93	97.10	96.42	96.38	97.10	96.83	96.22
Balance	94.77	93.49	93.65	94.63	93.57	93.27	95.37	93.69	93.18
Car	85.00	88.73	88.34	82.16	83.07	83.07	75.09	88.51	75.00
Dermatology	99.57	98.63	99.51	99.51	99.41	99.38	99.52	99.49	99.37
Cleveland	86.80	85.46	87.42	86.85	87.31	86.80	86.94	87.42	87.29
Segment	99.40	99.17	99.39	99.36	99.36	99.36	99.39	99.40	99.36
Tae	66.71	68.57	69.30	72.06	70.68	72.06	68.12	68.54	69.13
Texture	99.65	99.56	99.64	99.63	99.56	99.56	99.65	99.64	99.56
Thyroid	95.54	94.87	95.20	95.32	94.49	93.11	95.35	95.28	94.34

proposed method and baseline models (i.e., grid search was performed). Specifically, the range of k was set from 1 to 20, and the set of p values considered were $\{1, 1.5, \dots, 4\}$ for all the data sets. In fuzzy KNN-based methods, the fuzzy strength parameter (m) was fixed at a value of 2 for all FKNN classifiers throughout the experiments, as suggested by Derrac et al. (2015). Lastly, statistical tests, including Friedman and Bonferroni-Dunn tests, were applied to evaluate the statistical significance of the performance improvement of the proposed method compared to benchmark methods.

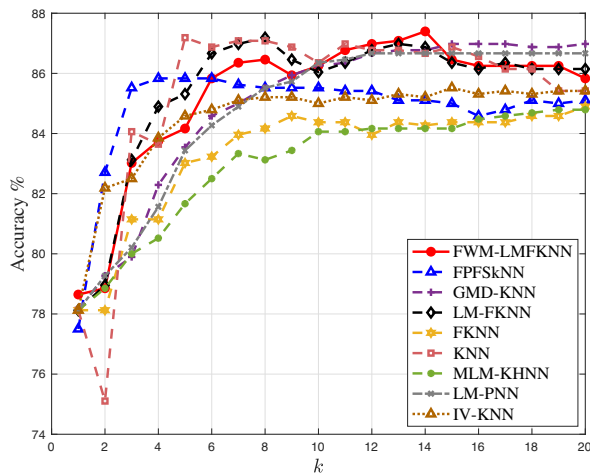
5 Results

This section presents the experimental results of the proposed method's performance against selected real-world data sets compared to the related baseline models. Optimal parameter values are also presented and discussed. Finally, a statistical analysis demonstrates that the proposed method achieved statistically significantly higher performance than the benchmark methods.

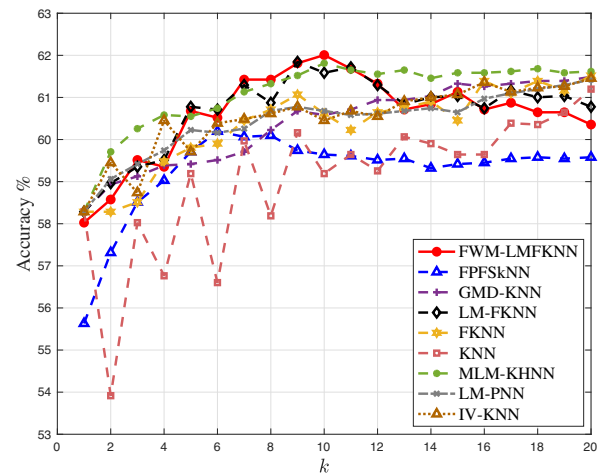
5.1 Evaluation of the proposed method

Table 3 presents a comprehensive comparison of the classification accuracy results and corresponding standard deviations of the proposed FWM-LMFKNN classifier and seven other baseline models, across 20 data sets. Note that the highest classification performance among the competing methods is highlighted in bold for each data set.

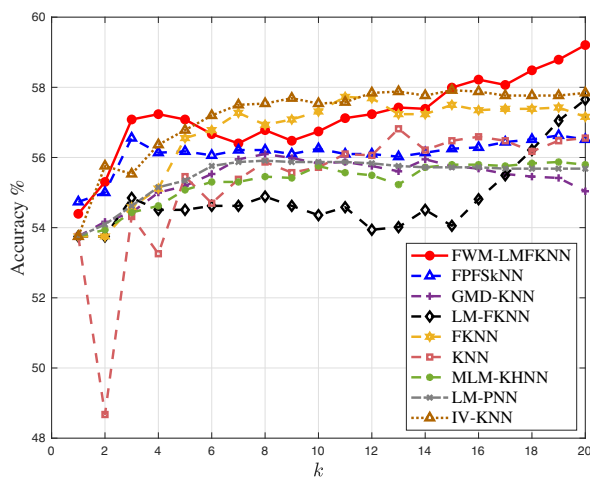
The table results show that the proposed FWM-LMFKNN method outperformed all other classifiers in terms of accuracy in 15 data sets (achieving the highest average accuracy of 82.50%). The table also shows the second-best performance in four data sets (Balance and Texture). Additionally, the proposed method had the lowest average standard deviation (of 2.57%) among all the classifiers, indicating that its performance was more consistent across the different data sets. This suggests that the proposed method was not only more accurate but also more robust than the other classifiers evaluated. In addition, corresponding sensitivity and specificity values (see Table 4 and Table 5) were reasonable and supported indications given by accuracy results. Besides, the performance of the LM-FKNN classifier appeared to be the second best (gaining average accuracy of 81.24%),



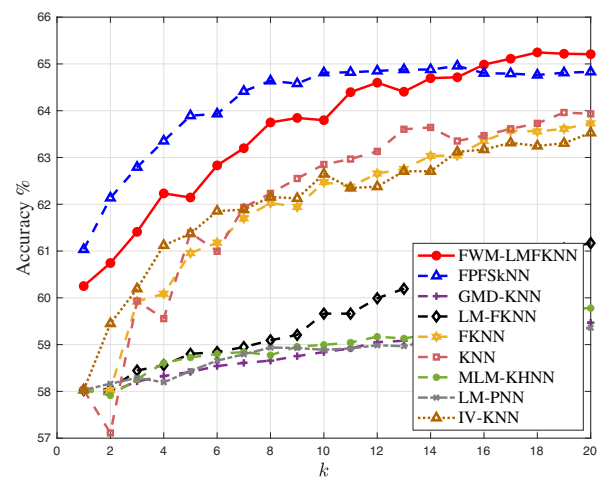
(a) Appendicitis data set



(b) Bupa data set



(c) Cleveland data set



(d) Retinopathy data set

Fig. 3 The accuracy of each method with respect to parameter k in the Appendicitis (a), Bupa (b), Cleveland (c), and Retinopathy (d) data sets

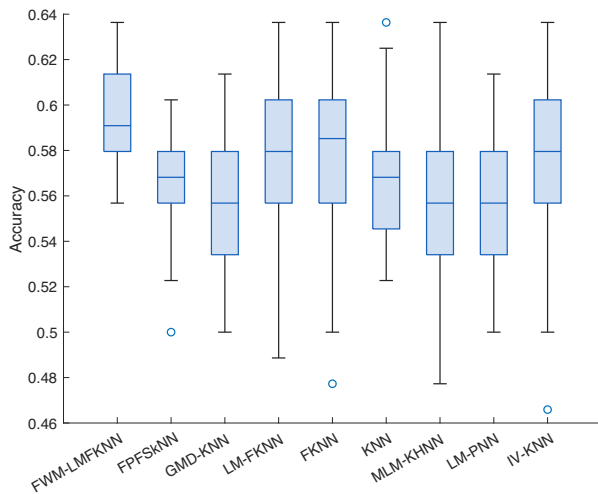
indicating the effectiveness of using class representative local means in the FKNN classifier, as indicated by related classifiers presented by Kumbure et al. (2019, 2020).

Considering the optimal parameter values (see Table 4), the proposed method achieved the highest performance with a 1–3.5 range for the Minkowski distance parameter across all data sets. Among them, the Manhattan distance ($p = 1$) appeared to have worked reasonably well in most cases, which is in line with the previous study by Kumbure and Luukka (2022). Regarding the parameter k , classification performance significantly improved with higher k values in the proposed method as well as the local means-based KNN methods. This finding aligns with previous research (Gou et al. 2014; Pan et al. 2017), demonstrating that multi-local mean vectors with nearest neighbors

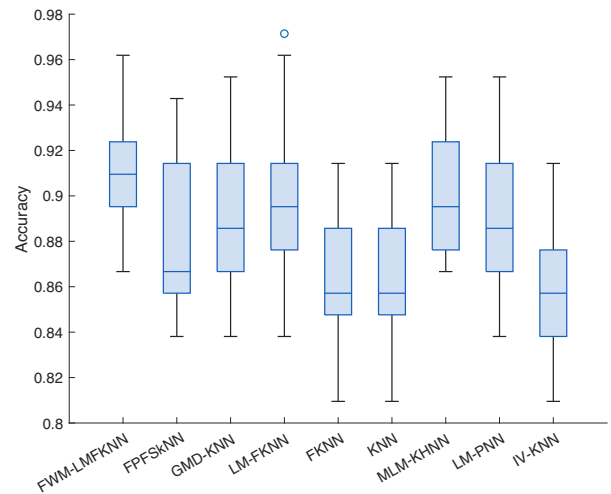
represented each class more accurately. This outcome is expected, as more data instances make local mean vectors more representative.

To further illustrate the performance of the proposed method in comparison to the baseline models, the accuracy results of all classifiers with varying values of parameter k on four selected cases, Appendicitis, Bupa, Cleveland, and Retinopathy data sets are presented in Fig. 3. As clearly shown in these sample cases, the proposed method generally outperformed the benchmark methods across a range of k values, particularly at high values of k .

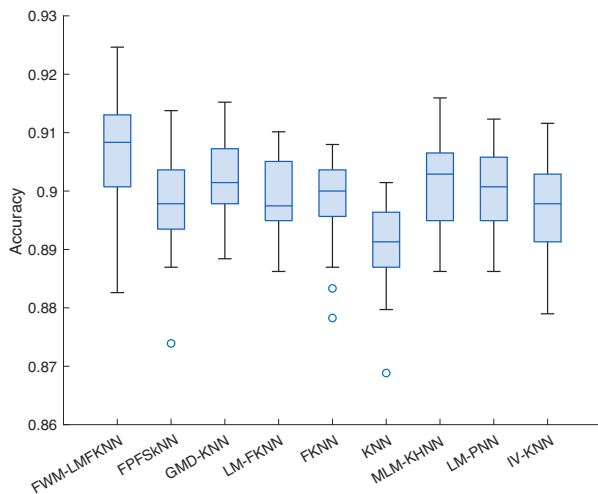
Furthermore, the classification performances of each method are depicted in box plots in Fig. 4, where each box represents the distribution of accuracy values over 30 runs during the validation on the Cleveland, Ionosphere,



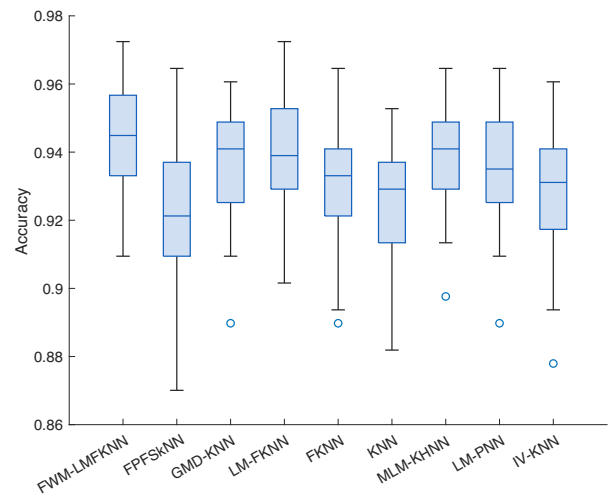
(a) Cleveland data set



(b) Ionosphere data set



(c) Spambase data set



(d) Vehicle data set

Fig. 4 Box-plot distributions of accuracy by each classifier in the Cleveland (a), Ionosphere (b), Spambase (c), and Vehicle (d) data sets

Spambase, and Vehicle data sets. A box plot analysis was conducted for the selected cases to understand the variability of each method’s performance across the cross-validation. As shown in the box plots, the proposed FWM-LMFkNN method had the highest median accuracy among all the classifiers, with a minimal interquartile range (IQR) across all cases considered, indicating that the accuracy values are relatively consistent across all runs.

5.2 Statistical analysis for benchmark comparison

To evaluate the significance of this improvement, we performed the Friedman test (Friedman 1937) and subsequently conducted the Bonferroni-Dunn test (Dunn 1961), following the methodology presented by Demšar (2006). In the Friedman test, classifiers were ranked individually for each data set—the top-performing classifier received a rank

Table 6 Ranking of each classifier according to its performance on each data set for the Friedman test

Data set	FWM-LMFKNN	FPFS-kNN (Memis et al. 2022)	GMD-KNN (Gou et al. 2019)	LM-FKNN (Kumbure et al. 2019)	FKNN (Keller et al. 1985)	KNN (Cover and Hart 1967)	MLM-KHNN (Pan et al. 2017)	LM-PNN (Gou et al. 2014)	IV-KNN (Derrac et al. 2015)
Appendicitis	1	6	4	2.5	8	2.5	9	5	7
Blood	1	4	6	2	5	3	8	7	9
Bupa	1	9	5	2	4	8	3	7	6
Ionosphere	1	6	4	3	7.5	7.5	2	5	9
Horse	1	7	9	5	3	4	6	8	2
Mammogram	1	5	9	7	2	4	6	8	3
Vehicle	1	9	4	2	6	8	3	5	7
Saheart	3	1	8	4	5	2	6	9	7
Sonar	1	9	2	6	7.5	4.5	4.5	3	7.5
Spambase	1	7	3	5	6	9	2	4	8
Retinopathy	1	2	8	6	4	3	7	9	5
WDBC	1	9	5	2	4	7	3	8	6
Balance	2	6	7	3	4	8	1	5	9
Car	4	1	3	5	6.5	6.5	9	2	8
Dermatology	1	9	4	3	6	7	2	5	8
Cleveland	1	6	7	4	3	5	9	8	2
Segment	1	9	4	6.5	6.5	6.5	3	2	6.5
Tae	7	6	1	9	3	8	5	2	4
Texture	2	9	4	5	7	7	1	3	7
Thyroid	1	2	4	6	7.5	9	5	3	7.5
R_j	1.65	6.10	5.05	4.40	5.27	5.97	4.72	5.40	6.42
R_j^2	2.72	37.21	25.50	19.36	27.82	35.70	22.32	29.16	41.28

R_j and R_j^2 indicate the average rank of classifier j and square of the average rank, respectively

Table 7 The results of the Bonferroni-Dunn test on the mean accuracy of the FWM-LMFKNN classifier against six baseline models

FWM-LMFKNN	FPFS-kNN (Memis et al. 2022)	GMD-KNN (Gou et al. 2019)	LM-FKNN (Kumbure et al. 2019)	FKNN (Keller et al. 1985)	KNN (Cover and Hart 1967)	MLM-KHNN (Pan et al. 2017)	LM-PNN (Gou et al. 2014)	IV-KNN (Derrac et al. 2015)
Average rank difference	4.45	3.40	2.75	3.62	4.32	3.07	3.75	4.77
CD value	2.401	2.401	2.401	2.401	2.401	2.401	2.401	2.401
Statistical significance	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

of 1, followed by the second-best with a rank of 2, and so forth, as detailed in Table 6. Subsequently, we calculated the Friedman statistic using the following formula:

$$\chi^2 = \frac{12N}{c_n(c_n + 1)} \left[\sum_{j=1}^c R_j^2 - \frac{c_n(c_n + 1)^2}{4} \right] \quad (16)$$

In the formula, $R_j = \frac{1}{n} \sum_{i=1}^n r_i^j$, where r_i^j represents the rank of the j th classifier on the i th data set, and there are c_n

classifiers and N data sets. In line with this, we computed $\chi^2 = 42.9$ based on the averaged ranks presented in Table 6 (here, $c_n = 9$ and $N = 20$), resulting in a corresponding p -value of 9.17×10^{-7} . This result offers sufficient evidence to reject the null hypothesis that all classifiers perform equally. In other words, this result supports the conclusion that the chosen classifiers exhibited statistically significant differences in mean accuracies at a significance level of

Table 8 The accuracy (%) results of each case with corresponding standard deviations (in brackets) in all tested data sets in the ablation study

Data	LM-FKNN (original method)	LM-FKNN + Feature weighting	LM-FKNN + Minkowski distance	Proposed FWM-LMFKNN
Appendicitis	89.17 (3.91)	89.27 (3.99)	89.17 (4.32)	89.27 (3.64)
Blood	76.90 (2.07)	76.98 (2.11)	76.90 (2.07)	76.98 (2.11)
Bupa	60.13 (5.24)	60.13 (5.12)	60.78 (4.59)	60.81 (4.95)
Ionosphere	89.94 (2.90)	89.97 (2.90)	90.03 (3.15)	90.06 (2.98)
Horse	85.84 (2.79)	85.84 (2.57)	86.25 (2.31)	86.25 (2.18)
Mammogram	73.17 (5.97)	75.45 (5.24)	78.46 (3.89)	78.88 (3.27)
Vehicle	93.90 (1.77)	93.87 (1.77)	94.90 (1.19)	94.87 (1.19)
Saheart	65.85 (4.06)	65.85 (3.76)	66.50 (3.34)	66.62 (3.01)
Sonar	86.08 (4.07)	86.02 (4.05)	87.31 (4.67)	87.31 (4.67)
Spambase	95.50 (2.81)	95.53 (2.54)	95.56 (2.31)	95.53 (2.29)
Retinopathy	61.48 (4.72)	61.17 (5.16)	65.07 (3.49)	64.98 (3.24)
WDBC	94.59 (2.20)	94.59 (2.20)	94.65 (2.05)	94.68 (2.03)
Balance	91.12 (1.21)	91.34 (1.23)	91.12 (1.21)	91.34 (1.23)
Car	71.70 (1.62)	72.38 (1.41)	72.80 (1.38)	72.95 (1.47)
Dermatology	97.60 (0.92)	97.60 (1.04)	97.90 (0.90)	97.97 (1.12)
Cleveland	56.86 (3.21)	57.01 (3.28)	58.56 (3.17)	58.67 (3.41)
Segment	95.97 (0.92)	95.92 (0.89)	96.25 (0.66)	96.26 (0.63)
Tae	55.00 (6.49)	55.00 (6.36)	56.45 (6.14)	56.59 (6.24)
Texture	96.22 (1.20)	96.21 (1.22)	96.26 (1.15)	96.26 (1.15)
Thyroid	93.75 (7.45)	93.70 (7.45)	94.06 (6.36)	94.06 (6.92)
Average	81.54 (3.21)	81.69 (2.91)	82.45 (2.88)	82.52 (2.88)

The highest accuracy values are given in bold

0.05. Since the null hypothesis was rejected, a post-hoc test can be applied now.

Accordingly, we conducted the Bonferroni-Dunn test to compare the performance of the proposed FWM-MLFKNN classifier with each other method, as indicated by Demšar (2006). In this test, the performances of two classifiers are considered significantly different if the corresponding average rank difference is greater than or equal to the critical difference (CD), which is defined as:

$$CD = q_{\alpha} \sqrt{\frac{c_n(c_n + 1)}{6N}} \quad (17)$$

where q_{α} represents the critical value from the two-tailed Bonferroni-Dunn test. After we applied the test to our analysis, we observed $CD = 2.401$ ($q_{0.05} = 2.72$, $c_n = 9$, and $N = 20$). By comparing this statistic with the difference in average rank between the FWM-LMFKNN method and each baseline method, we found that the proposed method demonstrated statistically significantly higher performance in terms of mean accuracy compared to all other methods. Table 7 presents the test results, where “Yes” denotes a significant difference between the mean accuracy of the proposed FWM-LMFKNN method and each benchmark classifier.

5.3 Ablation studies

In machine learning research, an ablation study is used to determine the significance of various components or aspects of a model and to evaluate their impact on overall performance (Meyes et al. 2019; Kwon and Lee 2024). Accordingly, we conducted ablation studies on the main components of our proposed method—namely feature weights (based on relevance and complementarity) and Minkowski distance—to demonstrate their effectiveness on classification performance. For the ablation studies, the performance of the original method, ML-FKNN (Kumbure et al. 2019), was compared with its variants: one using feature weights based on the feature relevance and complementarity and another utilizing a Minkowski distance metric-based similarity calculation. These methods were also compared with the proposed method (FWM-LMFKNN), which combines both feature weighting and the Minkowski distance. Models’ performances were compared using the experimental setup described in Sect. 4. The best average accuracy values (along with their standard deviations) across all data sets are presented in Table 8.

The results in Table 8 indicate that LM-FKNN with feature weights slightly improved performance compared

to the original method, with an increase in average accuracy from 81.54 to 81.69%. By contrast, LM-FKNN with Minkowski distance considerably improved performance, with an increase from 81.54 to 82.45%. This improvement may be due to the parameterized Minkowski distance allowing the classifier to adopt the most suitable distance metric for the data, thus achieving a more accurate set of nearest neighbors. However, the proposed FWM-LMFKNN method, which combines both feature weighting and the Minkowski distance measure, achieved the best overall results, with the highest average accuracy of 82.52%. Although the overall performance difference between the proposed method and LM-FKNN with Minkowski distance was not large, FWM-LMFKNN performed the best on many data sets, highlighting the positive impact of feature weighting. The standard deviation results of the proposed method were also low and reasonable, further supporting its robustness.

6 Conclusion

In this paper, we proposed a new fuzzy k -nearest neighbor method called FWM-LMFKNN based on feature weighting, Minkowski distance, and class representative local mean vectors. To determine the optimal feature weights in the proposed approach, we explicitly developed a feature weighting scheme considering a combined effect of relevance and complementarity. The proposed method was evaluated using a variety of real-world data sets, and the results show that it outperformed the baseline models in terms of used evaluation metrics. The use of feature weights and Minkowski distance allows for a more accurate calculation of the distances between new instance and training instances based on their nearness, which improves the accuracy of the proposed FWM-LMFKNN method. The ablation study conducted demonstrated the effectiveness of these aspects of the FMW-LMFKNN. Additionally, the proposed method takes into account the local structure of class subsets by using local mean vectors, further improving the performance of the classification. The results of this study demonstrate that the proposed method is a powerful tool for classification tasks and can be applied to a wide range of data sets.

Future work includes further testing of various data sets, evaluating the reliability of the proposed method, and investigating the possibility of incorporating other mean operators, such as generalized mean, in the local mean and ideal vector computation.

Author contributions Mahinda Mailagaha Kumbure: Conceptualization, Methodology, Software, Validation, Investigation, Writing -

Original Draft. Pasi Luukka: Conceptualization, Methodology, Writing - Review & Editing, Supervision.

Funding Open Access funding provided by LUT University (previously Lappeenranta University of Technology (LUT)).

Data availability Data used in the manuscript are freely available in the UCI and KEEL repositories.

Code availability The code of the proposed method is available at <https://github.com/MahindaMK/FWM-LMFKNN-classifier>.

Declarations

Conflict of interest On behalf of all authors, the corresponding author states that there is no Conflict of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Alcala-Fdez J, Fernandez A, Luengo J, Derrac J, García S, Sánchez L, Herrera F (2011) Keel data-mining software tool: data set repository, integration of algorithms and experimental analysis framework. *J Multiple-Valued Logic Soft Comput* 17:255–287
- Al-sharhan S, Karray F, Gueaieb W, Basir O (2001) Fuzzy entropy: a brief survey. In: 10th IEEE int. conf. on fuzzy systems, vol. 3, pp 1135–1139
- Bergamasco LCC, Nunes FLS (2019) Intelligent retrieval and classification in three-dimensional biomedical images - a systematic mapping. *Comput Sci Rev* 31:19–38
- Bian Z, Vong CM, Wong PK, Wang S (2022) Fuzzy knn method with adaptive nearest neighbors. *IEEE Trans Cybern* 52(6):5380–5393
- Biswas N, Chakraborty S, Mullick SS, Das S (2018) A parameter independent fuzzy weighted k -nearest neighbor classifier. *Pattern Recogn Lett* 101:80–87
- Cover T, Hart P (1967) Nearest neighbor pattern classification. *IEEE Trans Inf Theory* 13:21–27
- De Luca A, Termini S (1971) A definition of non-probabilistic entropy in setting of fuzzy set theory. *Inf Controls* 20:301–312
- Demšar J (2006) Statistical comparisons of classifiers over multiple data sets. *J Mach Learn Res* 7(1):1–30
- Derrac J, Chiclana F, García S, Herrera F (2016) Evolutionary fuzzy k -nearest neighbors algorithm using interval-valued fuzzy sets. *Inf Sci* 329:144–163 (**Special issue on Discovery Science**)
- Derrac J, Chiclana F, García S, Herrera F (2015) An interval valued k -nearest neighbors classifier. In: Proc. of the 2015 conf. of the int. fuzzy systems association and the European society for fuzzy logic and technology, pp 378–384. Atlantis Press

- Dheeru D, Taniskidou EK (2017) UCI machine learning repository
- Duarte FS, Rios RA, Hruschka ER, de Mello RF (2019) Decomposing time series into deterministic and stochastic influences: a survey. *Digital Signal Process* 95:102582
- Dunn OJ (1961) Multiple comparisons among means. *J Am Stat Assoc* 56:52–64
- Friedman M (1937) The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *J Am Stat Assoc* 32:675–701
- González S, García S, Li S-T, John R, Herrera F (2021) Fuzzy k-nearest neighbors with monotonicity constraints: Moving towards the robustness of monotonic noise. *Neurocomputing* 439:106–121
- Gou J, Zhan Y, Rao Y, Shen X, Wang X, He W (2014) Improved pseudo nearest neighbor classification. *Knowl-Based Syst* 70:361–375
- Gou J, Ma H, Ou W, Zeng S, Rao Y, Yang H (2019) A generalized mean distance-based k-nearest neighbor classifier. *Expert Syst Appl* 115:356–372
- Gueorguieva N, Valova I, Georgiev G (2017) M & MFCM: Fuzzy c-means clustering with mahalnobis and minkowski distance metrics. *Procedia Comput Sci* 114:224–233
- Karimi Z, Torabi Z (2022) An adaptive k-nearest neighbor classifier using differential evolution with auto-enhanced population diversity for intrusion detection. *Research Square*
- Kassani PH, Teoh ABJ, Kim E (2017) Evolutionary-modified fuzzy nearest-neighbor rule for pattern classification. *Expert Syst Appl* 88:258–269
- Keller JM, Gray MR, Givens JA (1985) A fuzzy k-nearest neighbor algorithm. *IEEE Trans Syst* 15:580–585
- Kendall M (1938) A new measure of rank correlation. *Biometrika* 30(1–2):81–89
- Kohavi R, John G (1997) Wrappers for feature subset selection. *Artif Intell* 1–2:273–324
- Kumar P, Thakur RS (2021) Liver disorder detection using variable-neighbor weighted fuzzy k nearest neighbor approach. *Multimed Tools Appl* 80:16515–16535
- Kumbure MM, Luukka P, Collan M (2019) An enhancement of fuzzy k-nearest neighbor classifier using multi-local power means. In: *Proc. of the 11th conf. of the European society for fuzzy logic and technology (eusflat 2019)*, pp. 83–90. Atlantis Press
- Kumbure MM, Luukka P (2022) A generalized fuzzy k-nearest neighbor regression model based on minkowski distance. *Granular Comput* 7:657–671
- Kumbure MM, Luukka P, Collan M (2020) A new fuzzy k-nearest neighbor classifier based on the Bonferroni mean. *Pattern Recogn Lett* 140:172–178
- Kwon Y, Lee Z (2024) A hybrid decision support system for adaptive trading strategies: combining a rule-based expert system with a deep reinforcement learning strategy. *Decis Support Syst* 177:114100
- Li Y, Zhao D, Xu Z, Heidari AA, Chen H, Jiang X, Xu S (2023) BSRWPSO-FKNN: a boosted pso with fuzzy k-nearest neighbor classifier for predicting a topic dermatitis disease. *Front Neuroinform* 16:1063048
- Lohrmann C, Luukka P, Jablonska-Sabuka M, Kauranne T (2018) A combination of fuzzy similarity measures and fuzzy entropy measures for supervised feature selection. *Expert Syst Appl* 110:216–236
- Łukasiewicz J (1970) *Selected work*. Cambridge University Press, Cambridge
- Luukka P (2011) Feature selection using fuzzy entropy measures with similarity classifier. *Expert Syst Appl* 38:4600–4607
- Luukka P, Saastamoinen K, Könönen V (2001) A classifier based on the maximal fuzzy similarity in the generalized lukasiewicz-structure. In: *Proceedings of 10th IEEE international conference on fuzzy systems*
- Ma X-A, Ju C (2022) Fuzzy information-theoretic feature selection via relevance, redundancy, and complementarity criteria. *Inf Sci* 611:564–590
- Maillo J, García S, Luengo J, Herrera F, Triguero I (2020) Fast and scalable approaches to accelerate the fuzzy k-nearest neighbors classifier for big data. *IEEE Trans Fuzzy Syst* 28(5):874–886
- Memis S, Enginoglu S, Erkan U (2022) Fuzzy parameterized fuzzy soft k-nearest neighbor classifier. *Neurocomputing* 500:351–378
- Meyer P, Schretter C, Bontempi G (2008) Information-theoretic feature selection in microarray data using variable complementarity. *IEEE J Sel Top Signal Process* 2:261–274
- Meyes R, Lu M, de Puiseau CW, Meisen T (2019) Ablation studies in artificial neural networks. <https://arxiv.org/abs/1901.08644>
- Mitania Y, Hamamoto Y (2006) A local mean-based nonparametric classifier. *Pattern Recogn Lett* 27:1151–1159
- Morente-Molinera JA, Mezei J, Carlsson C, Herrera-Viedma E (2017) Improving supervised learning classification methods using multigranular linguistic modeling and fuzzy entropy. *IEEE Trans Fuzzy Syst* 25:1078–1089
- Pan Z, Wang Y, Ku W (2017) A new k-harmonic nearest neighbor classifier based on the multi-local means. *Expert Syst Appl* 67:115–125
- Shannon CE (1948) A mathematical theory of communication. *Bell Syst Tech J* 27:623–659
- Singha S, Shenoy P (2018) An adaptive heuristic for feature selection based on complementarity. *Mach Learn* 107:2027–2071
- Sun L, Wang J, Wei J (2017) Avc: Selecting discriminative features on basis of auc by maximizing variable complementarity. *BMC Bioinformatics* 18:50
- Vergara J, Estevez P (2014) A review of feature selection methods based on mutual information. *Neural Comput Appl* 24:175–186
- Yang M-S, Sinaga KP (2021) Collaborative feature-weighted multi-view fuzzy c-means clustering. *Pattern Recogn* 119:108064
- Yu L, Liu H (2004) Efficient feature selection via analysis of relevance and redundancy. *J Mach Learn Res* 5:207–228
- Zadeh LA (1965) Fuzzy sets. *Inf Control* 8:338–353
- Zeraatkar S, Afsari F (2021) Interval-valued fuzzy and intuitionistic fuzzy-knn for imbalanced data classification. *Expert Syst Appl* 184:115510
- Zhang Q, Sheng J, Zhang Q, Wang L, Yang Z, Xin Y (2023) Enhanced Harris Hawks optimization-based fuzzy k-nearest neighbor algorithm for diagnosis of Alzheimer's disease. *Comput Biol Med* 165:107392

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.