Shahid Bahonar University of Kerman

Journal of Mahani Mathematical Research

Print ISSN: 2251-7952 Online ISSN: 2645-4505

A MULTI-OBJECTIVE OPTIMIZATION APPROACH FOR ONLINE STREAMING FEATURE SELECTION USING FUZZY PARETO DOMINANCE

A. Hashemi, M.R. Pajoohan ✉, and M.B. Dowlatshahi

Article type: Research Article

(Received: 12 February 2023, Received in revised form 12 August 2023)

(Accepted: 08 October 2023, Published Online: 20 October 2023)

Abstract. Feature selection is one of the most important tasks in machine learning. Traditional feature selection methods are inadequate for reducing the dimensionality of online data streams because they assume that the feature space is fixed and every time a feature is added, the algorithm must be executed from the beginning, which in addition to not performing real-time processing, causes many unnecessary calculations and resource consumption. In many real-world applications such as weather forecasting, stock markets, clinical research, natural disasters, and vital-sign monitoring, the feature space changes dynamically, and feature streams are added to the data over time. Existing online streaming feature selection (OSFS) methods suffer from problems such as high computational complexity, long processing time, sensitivity to parameters, and failure to account for redundancy between features. In this paper, the process of OSFS is modeled as a multi-objective optimization problem for the first time. When a feature stream arrives, it is evaluated in the multi-objective space using fuzzy Pareto dominance, where three feature selection methods are considered as our objectives. Features are ranked according to their degree of dominance in the multi-objective space over other features. We proposed an effective method to select a minimum subset of features in a short time. Experiments were conducted using two classifiers and eight OSFS algorithms with real-world datasets. The results show that the proposed method selects a minimal subset of features in a reasonable time for all datasets.

Keywords: Online streaming feature selection, Fuzzy Pareto Dominance, High-Dimensional data, Multi-objective Optimization.
2020 MSC: 58E17.

1. Introduction

Feature selection is one of the most influential and well-known preprocessing techniques to minimize the effects of the curse of dimensionality on high-dimensional data. The curse of dimensionality causes overfitting of algorithms, decreases the Accuracy of learning algorithms, and increases learning time and

✉ pajoohan@yazd.ac.ir, ORCID: 0000-0002-7848-2025
DOI: 10.22103/jmmr.2023.21044.1402
Publisher: Shahid Bahonar University of Kerman
How to cite: A. Hashemi, M.R. Pajoohan, and M.B. Dowlatshahi, *A multi-objective optimization approach for online streaming feature selection using fuzzy Pareto dominance*, J. Mahani Math. Res. 2024; 13(1): 467 - 490.

computational complexity. Feature selection can reduce the above effects by removing irrelevant and redundant features from the data. Reducing feature dimensionality leads to better model interpretability, higher prediction accuracy, and reduced computational time [3], [4], [5], [9], [32], [34].

In recent decades, feature selection has been one of the most exciting topics for machine learning researchers, leading to the presentation of numerous feature selection methods in various applications [1], [11], [12], [13], [22]. In most feature selection methods, all the features are assumed to be initially available, which is unrealistic in many real-world applications [6], [25], [39].

Features can be dynamically expanded for specific observational purposes in environmental monitoring as new sensors are deployed. Another example is that relevant keywords are continuously generated and rapidly propagated by certain types when a popular event occurs in social networks. The dimensions of the dynamic feature space are very large or even infinite. Therefore, it is costly and time-consuming to wait for all features. On the other hand, due to the increasing data size and dimensions in some large datasets, it is impossible to load the entire data into memory. Therefore, for large data sets, it is better to process the data as rows (samples) or columns (features) [16], [30], [31], [38].

Online feature selection algorithms are provided to find the optimal subset of features in cases where traditional feature selection algorithms are not applicable. These algorithms evaluate features based on data streams with different time stamps and update the subset of features with each new incoming data stream. Therefore, the optimal subset of features is always available. In online feature selection methods where features are delivered in streams, the number of training samples is assumed to be constant, and features are added to the data one at a time. This task is called OSFS [6], [14], [34].

Scalability, computational cost, and parameter sensitivity are the three main challenges in OSFS methods. Scalability means that the performance of the feature selection algorithm does not decrease as the data dimensions increase. Since Big Data in real-world applications is mostly generated in streaming form, it is important to provide OSFS methods so that the number of selected features does not increase significantly when the number of input features is increased [34].

On the other hand, it is necessary to provide techniques with low computational complexity and high speed to make online decisions. Moreover, in many OSFS methods presented so far, some parameters affect the performance of the algorithm. Most of these parameters must first be fixed. In a dynamic feature space and in a situation where we do not know the future feature streams, the selected parameter value may decrease the performance of the algorithm.

Therefore, we must present effective methods for the OSFS problem that balance classification accuracy, number of selected features, and execution time. Moreover, the parameters of the algorithm should be updated with the incoming features, or an algorithm without parameters should be provided [34].

Multi-objective optimization is an approach that researchers have become interested in for machine learning applications. This approach is used when the best solution is sought based on multiple objectives [8], [9], [10]. Feature selection is also one of the machine learning applications where multi-objective optimization is applied and yields promising results. The concept of Pareto dominance is one of the practical approaches to solve multi-objective problems. In this approach, different alternatives are placed in a multi-objective space, and the best solution is selected based on the Pareto dominance concept [20]. If we intend to evaluate the alternatives based on the degree of dominance in the multi-objective space, a fuzzy version of the Pareto dominance concept can also be used. In this fuzzy version, we can determine the degree of dominance of each alternative over the other alternatives [19].

In this paper, we propose a new online streaming feature selection method based on the concept of fuzzy Pareto dominance called OFS-FPS (Online Feature Selection using Fuzzy Pareto Dominance). This method uses a filtering strategy and is proposed for single-stream features. In this method, the correlation of each new feature with the class label is first calculated to filter out the irrelevant features. The relevant features are added to the feature set. The feature set is updated when it reaches a limit to select a minimum number of features. This limit is determined by a non-linear relationship of the number of feature streams added to the dataset so far. Then, the feature set is updated using the concept of fuzzy Pareto dominance. A council of relevance-based and redundancy-based metrics is used for the decision-making process to obtain the most relevant and least redundant feature set. Features are ranked based on their degree of dominance in this multi-objective space over other features.

Previously proposed OSFS methods based on single-stream features have weaknesses, such as low classification accuracy, high computational complexity, high number of selected features, sensitivity to parameter values, different number of selected features by different orders of samples, and high processing time. Therefore, this work aims to develop a parameter-free OSFS method that achieves better classification performance with a minimum subset of features in a reasonable time.

The main contributions of the proposed method can be discussed as follows:

(1) Most OSFS methods presented so far have used only one criterion to evaluate the features. We believe that a decision-making council can do more than just one decision-making indicator. For this reason, we use multiple objectives for decision-making in OSFS to achieve better performance.
(2) The ensemble of feature selection metrics for online streaming features is modeled as a multi-objective optimization problem. The concept of fuzzy Pareto dominance is used for the first time to solve this optimization and rank the features.

(3) The number of selected features is limited and determined by a non-linear relationship of the number of feature streams added to the dataset so far to select a minimal subset of features and ensure the scalability of the proposed method.
(4) Some proposed online OSFS methods suffer from high computational costs. Therefore, we have developed a simple and effective method to process streaming features in a short period of time.
(5) Some proposed online OSFS methods are sensitive to parameter values that may affect method performance when dealing with unknown feature streams. We have developed a parameter-free method to deal with this challenge.
(6) The current OSFS methods select a different number of features by a different order of training examples. Therefore, we made the number of features selected dependent on the number of feature streams so that a fixed number of features are selected at each training sample order.

Based on the experiments, we have shown the superiority of the proposed algorithm in classification performance using two well-known classifiers. K-Nearest Neighbor and CART classifiers are used to compare the results of all the comparison algorithms and the proposed algorithm based on classification accuracy, F-score, Precision, and Sensitivity. Also, the algorithms are compared based on the final feature subset and running time.

The structure of this article is as follows: In Section 2, we review some related work in the literature. Section 3 gives the fundamental concepts. Section 4 describes the proposed algorithm in detail, and Section 5 presents the experimental results. Finally, the article is concluded in Section 6.

2. **Related Works**

This section discusses some related articles on online streaming feature selection considering single-stream features. OSFSMI [26] is an online feature selection algorithm that evaluates single feature streams based on the concept of mutual information. In this algorithm, each time a new feature arrives, the value of mutual information between that feature and the class label is calculated. If this value is greater than zero, the feature is temporarily selected. Features that have passed this step are analyzed in the redundancy phase. The redundancy of the features is calculated by the interaction gain criterion, which calculates the overall correlation of each feature with the current feature set and the class label. For this purpose, the interaction gain is calculated for all features. If the lowest interaction gain is related to the newest feature, the algorithm proceeds to the next step and waits until the new feature arrives. Otherwise, the feature with the lowest value is removed.

Alpha-investing [35] algorithm does not work on the basis of a global model, but its performance is based on a statistical approach. This algorithm is proposed to evaluate the single-stream features. When a new feature is added to the feature space, it computes a P-value that indicates the probability of a decision to select or reject a feature. The algorithm is proposed to control the threshold value during the feature selection process by selecting new features in the model. The threshold value increases as new features are selected so that more features are selected in the future. Also, the threshold is lowered when a feature does not meet the threshold. One of the advantages of the alpha-investing algorithm is its ability to manage a set of features of unknown size, even to infinity.

OFS-3AM [37] is an online feature selection algorithm that selects the most relevant features with the least redundancy based on a rough set of adapted neighbors. This algorithm uses three evaluation indicators to find a subset of features with the highest dependency. At the same time, a minimum subset of features is selected by eliminating redundancies from the feature set.

OFS-Density [36] also uses rough set theory to solve the online feature selection problem based on individual feature streams. The operation of this algorithm is based on determining the number of neighbors during the dependency calculation by a new density-neighbor relationship, which is automatically performed by the density information of the surrounding samples. With this new neighborhood relationship, you do not need to specify any parameters in advance. At the same time, a fuzzy equality constraint is used for redundancy analysis, which minimizes and differentiates the selected feature subset. This algorithm aims to select new features with maximum correlation, maximum dependence, and minimum redundancy.

OSFV [15] uses a voting strategy to perform rank aggregation in OSFS. In this method, the potentially relevant features are added to the feature set. Then, each time the number of features in the feature set exceeds the maximum capacity, a voting process is performed to keep the most compelling features and ignore the others.

OFSFI [23] performs a fuzzy integral method to solve the OSFS problem. In this method, the features that exceed a predetermined threshold are added to the feature set. A fuzzy integral procedure is performed based on three redundancy metrics to update the feature set to keep the most redundant features.

RHOFS [21] is an OSFS method that attempts to identify practical features by considering relevance and redundancy constraints. In this method, the processing time is very low due to parallel computations. A new feature evaluation metric is also presented, which uses useful implicit patterns from the edge region plus the explicit patterns from the positive region.

In Table 1, a comparison is made between the above OSFS methods based on several terms. These terms are the main feature selection technique, parameter sensitivity, number of selected features based on changing data samples, and whether the method requires prior knowledge before initialization.

Table 1. The comparison between benchmark OSFS methods.

Method	Technique	Parameter Sensitivity	Different number of selected features by changing data samples	Require prior knowledge
OSFSMI	Information theory	Yes	Yes	No
OFS-3AM	Neighborhood rough set	Yes	Yes	No
OFS-Density	Neighborhood rough set	Yes	Yes	No
Alpha-investing	Information theory	Yes	Yes	Yes
OFSV	Voting	No	Yes	No
OFSFI	Fuzzy Integral	Yes	Yes	No
RHOFS	Rough Hypercuboid	Yes	Yes	No

3. Fundamental Concepts

In this section, we first introduce the notations used in the article and the concept of fuzzy logic and then describe the problem formulation and fuzzy Pareto dominance.

3.1. **Notations.** Table 2 summarizes the adopted symbols of this article.

Table 2. Symbol Annotations.

Symbol	Explanations	Symbol	Explanations
X	Dataset	$\mu(X)$	Fuzzy membership degree of X on μ
$G(x)$	Objective vector	$X_t(f_t, C)$	Data stream
S	Optimal feature set	n	Number of samples
d	Number of features	F	Feature space
C	Class Label	f_t	Streaming feature at timestamp t
E^*	Euclidean distance matrix	M	Decision matrix
t	Timestamp	E_1, E_2, E_3	Objective vectors

3.2. **Fuzzy logic.** Many uncertain factors play a role in decision-making to solve real-world problems. These uncertain factors may include indeterminacy, randomness, and incompleteness of some information in decision-making. In this situation, reaching a conclusion about a phenomenon is impossible. This problem also exists in machine learning. For example, if we want to decide on several options based on several criteria, none of the options can be superior to the other options based on all the criteria. Fuzzy logic is used in mathematics and artificial intelligence to solve such problems [27].

Fuzzy logic allows us to represent real-world problems more realistically and to process imprecise real-world information. Moreover, methods based on fuzzy logic use simpler and fewer rules than exact methods, which makes them easier to work with [17].

The theory of fuzzy sets is based on the fact that in the real world, many things can be expressed in a non-deterministic and membership degree. Zadeh

[33] introduced the theory of fuzzy sets in 1965. According to this theory, the degree of certainty of a phenomenon is expressed by the degree of membership. A fuzzy set μ on the random variable X defines as: $\mu : X \to [0, 1]$ where $\mu(X)$ indicates the membership degree of X on μ. Fuzzy logic based methods are used for problems that cannot be answered by exact methods.

3.3. **Problem Formulation and Fuzzy Pareto Dominance.** A multi-objective optimization problem (MOP) is a technique for finding optimal solutions to NP-hard problems. Instead of finding an optimal solution, this technique obtains a set of compromise solutions. These optimal solutions are called a Pareto-optimal set [29]. This work uses a Pareto-based model to find the best solutions based on a minimization approach. Therefore, a maximization MOP is defined as follows:

$$\min G(x) = (G_1(x), G_2(x), \dots, G_n(x)),\ s.c.x \in S \tag{1}$$

where $x = (x_1, \dots.., x_k\ )$ refers to a vector containing the decision variables, and S indicates the set of feasible solutions. Finally, $G(x) = (G_1(x), G_2(x), \dots, G_n(x))$ shows the vector of objectives which n $(n \geq 2)$ is the length of this vector (number of objectives). The vector G should be considered a beneficial function that shows solution quality because, in this paper, the MOP is considered a maximization problem.

The following concepts are the main definitions of Pareto-based MOPs:

Definition 3.1. Pareto Dominance. Let us consider two objective vectors $v = (v_1, \dots.., v_n)$ and $u = (u_1, \dots.., u_n)$. If no element of u is smaller than the corresponding element of v, and at least one component of v is strictly smaller, we can say v dominates u (*shown by* $u < v$). The Pareto dominance relation is formulated as follows:

$$\forall j \in \{1, \dots, n\} : v_j \leq u_j \ \bigwedge \exists j \ \{1, \dots, n\} : v_j < u_j. \tag{2}$$

The efficiency of a MOP is measured by Pareto dominance and is used to indicate the superiority of one solution over the others. It is essential for MOP because different solutions may have advantages over others depending on their objectives [9].

Definition 3.2. Fuzzy Pareto Dominance. Let us consider two objective vectors $v = (v_1, \dots\dots.., v_n)$ and $u = (u_1, \dots\dots.., u_n)$. We say that vector u dominates vector v with degree μ_u by the following equation:

$$\mu_u(u, v) = \frac{\prod_i \min(u_i, v_i)}{\prod_i u_i} \tag{3}$$

We say that vector u is dominated by vector v with degree μ_v by the following equation:

$$\mu_v(u, v) = \frac{\prod_i min(u_i, v_i)}{\prod_i v_i} \tag{4}$$

These dominance degrees can be used to rank alternatives in a multi-objective space. We can assign each alternative the maximum degree of being dominated by any other alternatives in the multi-objective space and sort them in increasing order [19], [24].

In this paper, we used multi-objective optimization to solve the OSFS problem. In this model, we have used an ensemble of feature selection methods to score the feature streams. Thus, our objectives are the scores assigned by the feature selection methods. The features are evaluated in multi-objective space using the following scores. The features are generated incrementally in OSFS.

Figure 1 shows the OSFS framework. This framework is the basis of most OSFS methods. It means that the features in OSFS are evaluated in two levels. The incoming features are evaluated at the first level to determine whether they should be selected or ignored. In the second level, the feature set is also updated each time new features are added to select a minimum subset of features.

For example, in our model, consider $u = (0.1, 0.8, 0.3)$ and $v = (0.7, 0.2, 0.3)$ are two objective vectors for two features in the multi-objective space, feature u dominates feature v by the following degree:

$$\mu_u\left(u, v\right) = \frac{0.1 \times 0.2}{0.1 \times 0.8 \times 0.3} = 0.8333$$

Also, feature u is dominated by feature v by the following degree:

$$\mu_v\left(u, v\right) = \frac{0.1 \times 0.2}{0.7 \times 0.2 \times 0.3} = 0.4761$$

It means that u is a better feature than v because it dominates v with a higher degree of dominance.

4. **Proposed Algorithm**

In this paper, we attempt to develop an OSFS method for single-stream features. In this paper, we use an ensemble strategy for the OSFS problem. The OSFS problem is modeled as a multi-objective optimization process. In this algorithm, three feature evaluation objectives based on filter relevance and redundancy are considered as our experts in the optimization process. Figure 2 and Figure 3 show a graphical summary and step-by-step procedure of the proposed algorithm. These steps are explained in detail below:

The features are generated incrementally in the online streaming feature selection. In Figure 2, the framework of online streaming feature selection is demonstrated. Let us consider $X = [x_1, x_2, \ldots, x_n]^T \in \mathbb{R}^{n \times d}$ as the dataset, the dataset contains n instances in a feature space $F = [f_1, f_2, \ldots, f_d]^T \in \mathbb{R}^d$ and a class label $C = [c_1, c_2, \ldots, c_n]^T \in \mathbb{R}^{n \times 1}$, where d is the number of features. In this problem, in each time-stamp t, f_t is arrived at the feature space, and the value of d is unknown. Thus, a candidate set S_t is achieved in each time-stamp based on arrived features. In other words, by increasing the number of feature streams, more features are accepted. This function, however, is chosen non-linearly to avoid increasing the arbitrary number of features.

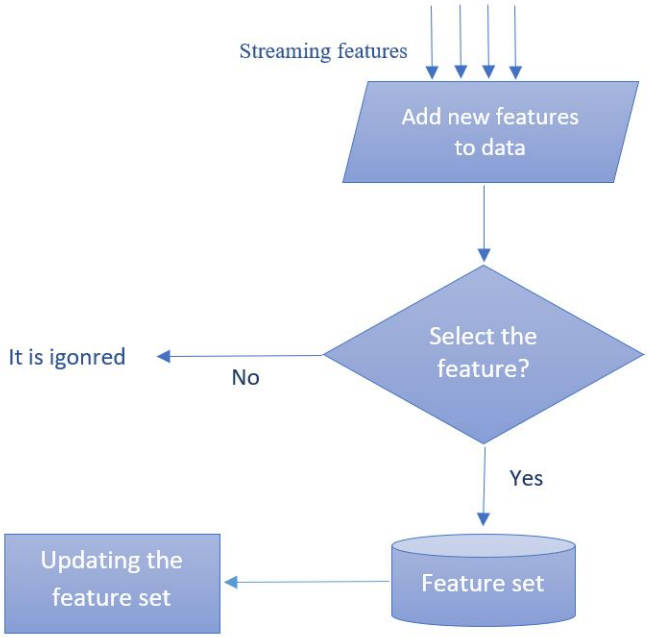

FIGURE 1. The structure of online streaming feature selection

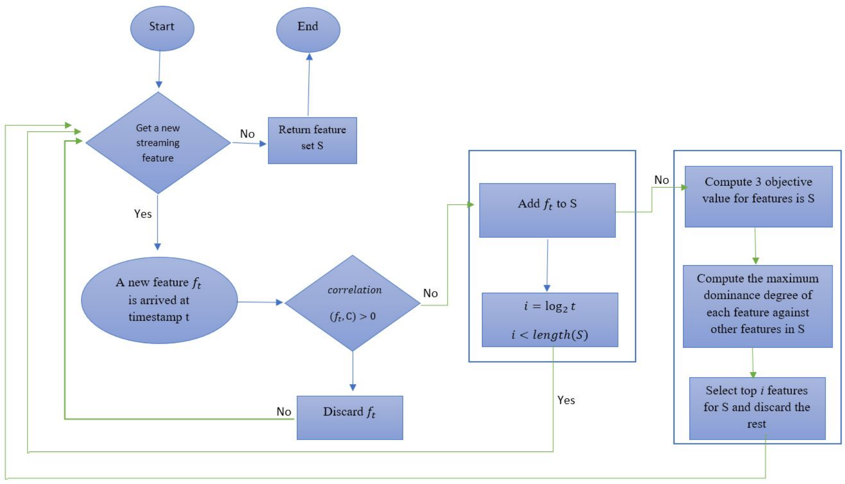

FIGURE 2. Graphical abstract of the proposed method

Algorithm 1: OFS-FPD algorithm

Input: Data stream $X_t(f_t, C)$ $where\ f_t$: new arriving feature at time t, C: class label

Output: Optimal feature set S_t till time t, $S_0 = \emptyset$

1. **Repeat**
2. $f_t \leftarrow$ newly arrived feature at time t
3. $i = \log_2 t$
4. $if\ corr(f_t, C) > 0$
5. $S_t = S_t \cup f_t$, $X = X \cup X_t$
6. **end if**
7. $if\ length(S_t) > i$
8. Compute the Correlation matrix (E) between the features in S_t
9. E1 = Calculate the minimum correlation for each feature in matrix E
10. E2 = Calculate the correlation distance of each feature against the class label in matrix E
11. E3 = Calculate the cosine distance of each feature against the class label in matrix E
12. Construct the decision matrix (M) by E1, E2, and E3 vectors as $M = [E1, E2, E3]$
13. Compute the fuzzy dominance degree of features based on Eq.3
14. Obtain maximum dominance degree based on Eq.4 for features and sort them in ascending order
15. Perform algorithm 2 by delivering matrix R
16. Select top i features based on the obtained ranking and eliminate the values of removed features in and X
17. **end if**
18. **Until** no new feature is available
19. **Return** S_t

Figure 3. Pseudo code of the proposed method

In current OSFS methods, the number of selected features depends on the approach and thresholds used in these methods. There is no control over the number of selected features in different datasets. Moreover, these methods select different numbers of features in a dataset with different subsets of training examples. To avoid this and control the number of selected features in different datasets, we make the number of selected features depend on a non-linear relationship with the number of received features.

In step 2 of the proposed method, the feature stream f_t arrives in the feature space. In step 3, the capacity of the feature spaces is determined based on the feature count. The logarithm of the number of features added to the dataset is considered.

In this algorithm, there are two stages of decision-making. In the first stage, the relevance of each new feature with the class label is calculated. This value is determined using Pearson's correlation coefficient [18] to avoid including irrelevant features in the calculations. In steps 4 to 6, the correlation between the new feature and the class label is calculated. The formula for calculating the correlation coefficient is as follows:

$$corr\ (X,Y) = \frac{Cov(X,Y)}{\sigma_X \sigma_Y} \tag{5}$$

where $Cov(X,Y)$ refers to the covariance between variable X and Y. Also σ_X and σ_X are the standard deviation of X and Y, respectively.

In the proposed algorithm, the feature is added to S_t when $corr > 0$. Otherwise, it is ignored. Also, X is the dataset with the desirable feature set so far. The reason for using a value of zero for feature acceptance or rejection is that features that have even a small relevance to the class label have a chance of being selected. When the number of selected features exceeds the maximum number of features (i represents the number of features received so far), the second phase of decision- making is performed. This means that the feature set is updated.

As we discussed earlier, the maximum number of selected features is a function of all the features that have arrived so far. We considered a non-linear function to control the number of features and avoid it becoming too large. The algorithm checks whether the number of features in the set exceeds the maximum capacity or not. If the number of features is within the allowed size, the algorithm waits for the arrival of a new feature and returns to step 2.

When the number of features exceeds the maximum capacity, the feature set is updated, and a certain number of features are removed to achieve the desired set of features. Steps 7 to 17 of the algorithm refer to this part. We go through these steps one by one.

After establishing the condition in the seventh step, we determine three feature evaluation criteria. These criteria are our objectives in the multi-objective optimization process. The feature set updating process is conducted using fuzzy Pareto dominance ranking. The purpose of feature selection is to select a subset of features associated with the class label with the least redundancy. Therefore, the selected objectives are a combination of methods that measure the degree of relevance and redundancy of features. One redundancy and two relevancy check method are considered.

We used the Pearson correlation coefficient to capture the relevance and redundancy of features. This metric is commonly used in feature selection and has shown promising results. It indicates the degree of linear dependence between two features or a feature and the class label. Since the degree of relevance of features is more important than non-redundancy, we also considered the cosine distance criterion to calculate the degree of relevance of a feature and the class label. The purpose of using two criteria to calculate relevance was to review from the perspective of two experts. Therefore, we created a feature evaluation council where two experts comment on the relevance of features and one expert comment on their redundancy.

We obtain the correlation between the features in S_t using Equation 5 to capture the redundancy between the features. By calculating the correlation between all the features in the set, the correlation matrix is obtained as follows:

$$E^* = \begin{bmatrix} corr\ (F_1, F_1) & corr\ (F_1, F_2) & \cdots & corr\ (F_1, F_h) \\ corr\ (F_2, F_1) & corr\ (F_2, F_2) & \cdots & corr\ (F_2, F_h) \\ \vdots & \vdots & \ddots & \vdots \\ corr\ (F_d, F_1) & corr\ (F_d, F_2) & \ldots & corr\ (F_d, F_h) \end{bmatrix} \tag{6}$$

As our redundancy-based objective, each feature's minimum correlation against other features is considered. It means that the minimum value of each row of E^* is computed to construct vectors $E1$. This procedure is conducted in steps 8 to 9. Since we modeled the OSFS problem as a minimization approach, we considered the minimum correlation value of each feature with respect to other features as a redundancy value.

In addition to this objective, the correlation and cosine distance values are considered the second and third objectives. We used the distance to capture the correlation and cosine objectives because we considered our problem as a minimization process. The correlation distance value (E2) between features F_i and F_j is computed as follows:

$$E2\ \ (F_i, F_j) = 1 - corr\ \ (F_i, F_j) \tag{7}$$

The cosine distance metric between two random variables is also defined as follows:

$$E3\ \ (A,\ B) = 1 - \left| \frac{\sum_{i=1}^{n} (A_i B_i)}{\left(\sqrt{\sum_{i=1}^{n} {A_i}^2}\right)\left(\sqrt{\sum_{i=1}^{n} {B_i}^2}\right)} \right|, \tag{8}$$

Thus, the decision matrix is constructed as follows:

$$M = \begin{bmatrix} E1\,(1) & E2\,(1) & E3\,(1) \\ E1\,(2) & E2\,(2) & E3\,(2) \\ \vdots & \vdots & \vdots \\ E1\,(d) & E2\,(d) & E3\,(d) \end{bmatrix} \tag{9}$$

where $E2(2)$ indicates the value of our second objective (correlation distance) for feature 2. Steps 10 to 12 show these calculations.

Now we have our decision matrix to perform the multi-objective optimization process. Based on Equation 4, the degree of dominance is calculated based on all features, and the maximum value is recorded. The features are sorted in ascending order to determine their ranking. Steps 13 to 15 refer to this part.

As mentioned in the basic concepts section, there are two deterministic and fuzzy approaches in modeling based on the concept of Pareto dominance. In the deterministic approach of Pareto dominance, the features are placed in the objective space based on multiple objectives and only the features that are in the Pareto front are selected as the final feature set. In this approach, the number of selected features cannot be controlled and a different number is selected for each dataset. For example, in one dataset, one feature may dominate all other features, while in another dataset, many non-dominated features may remain in the Pareto front. To address this problem, we used the fuzzy Pareto dominance approach, where the concept of dominance is considered a fuzzy measure rather than a definite one.

As mentioned in the basic concepts section, there are two deterministic and fuzzy approaches in modeling based on the concept of Pareto dominance. In the deterministic approach of Pareto dominance, the features are placed in the objective space based on multiple objectives and only the features that are in the Pareto front are selected as the final feature set. In this approach, the number of selected features cannot be controlled and a different number is selected for each dataset. For example, in one dataset, one feature may dominate all other features, while in another dataset, many non-dominated features may remain in the Pareto front. To address this problem, we used the fuzzy Pareto dominance approach, where the concept of dominance is considered a fuzzy measure rather than a definite one.

Each time a new feature is added, its relevance is first calculated using the Pearson correlation coefficient. Based on this approach, irrelevant features are ignored as soon as possible. But related features are added to the feature set for further consideration. We have considered the capacity to control the number of features so that we have a certain number of features available at any time. When the number of features exceeds the capacity, the feature space needs to be updated. For this update, we transferred all the features of the feature space into a multi-objective space. We also used the fuzzy Pareto dominance method to rank the features. In this method, we first calculated the percentage of dominance of each feature by other features and then ordered them in ascending order according to this percentage. This means that the feature that has the lowest percentage of defeat by other features gets the highest rank. Finally, the features that are in the lowest rank and are not included in the capacity of the feature space are removed.

Thus, in step 16, the best features are selected based on the maximum number of features required. The eliminated features are removed from the feature space. This process is performed each time a new feature is added to the feature space. Finally, S_t is the optimal subset of features.

For a better understanding, we use a numerical example. Suppose the feature space capacity is equal to 2 and three features are available to us. We want to remove one of these features based on the fuzzy Pareto dominance method in a multi-objective space. Assume that the objective vectors for 3 features are based on the three objectives presented below. The first and second terms, respectively, are the value of the correlation distance and the cosine distance of each feature to the class label, and the last term is the minimum correlation of each feature against other features.

$$F_1 = \{0.2, 0.4, 0.5\}, F_2 = \{0.8, 0.7, 0.1\}, F_3 = \{0.6, 0.6, 0.6\}$$

Now we calculate the dominance degree of each feature by other features based on Equation 4.

$$Feature1 : \mu_2(1, 2) = \frac{0.1\times0.2}{0.1\times0.7\times0.8} = 0.35$$
$$Feature1 : \mu_3(1, 3) = \frac{0.2\times0.6}{0.6\times0.6\times0.6} = 0.55$$
$$Feature2 : \mu_1(2, 1) = \frac{0.1\times0.2}{0.2\times0.4\times0.5} = 0.5$$
$$Feature2 : \mu_3(2, 3) = \frac{0.1\times0.6}{0.6\times0.6\times0.6} = 0.27$$
$$Feature3 : \mu_1(3, 1) = \frac{0.2\times0.6}{0.2\times0.4\times0.5} = 0.35$$
$$Feature3 : \mu_2(3, 2) = \frac{0.1\times0.6}{0.8\times0.7\times0.1} = 1.07$$

Now we extract the maximum dominance degree of each feature, which is 0.55 for feature 1, 0.5 for feature 2, and 1.07 for feature 3. Therefore, the highest dominance degree belongs to feature 3 and is removed. It can also be seen in the objective vector of this feature that it has high values in this minimization problem.

5. Numerical results

This section will introduce the datasets and evaluation metrics for performing the simulations. Also, the obtained results and a discussion over these results are provided.

5.1. **Datasets and Evaluation metrics.** In this paper, six real-world datasets are used to perform the experiments. Table3 contains the detailed characteristics of these datasets. Datasets include Prostate-GE, Colon, Leukemia [1], Sorlie [2], Subramanian [28] and NCI60 [3] from different applications including Microarray, Biological, and Image Data.

Four well-known metrics, namely Accuracy, Sensitivity, Precision, and F-measure [2], [13], are used to evaluate the performance of feature selection algorithms in the learning procedure.

5.2. **Results.** Eight online streaming feature selection algorithms are provided for comparison with the proposed algorithm in these experiments. These algorithms are OSFSMI [26], OFS-A3M [37], OFS-Density [36], Alpha-Investing [35], OFSV [15], OFSFI [23], and RHOFS [21]. The reason for choosing these methods compared to the proposed method is that our goal is to compare with the most powerful and newest methods in this field. Considering that the proposed method is based on an ensemble approach, we have chosen two methods OFSV and OFSFI that are structured based on this approach for comparison

---

[1] https://jundongl.github.io/scikit-feature/datasets.html

[2] https://search.r-project.org/CRAN

[3] https://dtp.cancer.gov/discovery$_d$$evelopment/nci - 60/$]

to measure the performance of the proposed method. RHOFS method three is the newest in this field and also uses the new idea of rough hyperbolic. OFS-A3M and OFS-Density have used rough set theory and are among the most famous and strongest methods in this field. OSFSMI and Alpha-Investing have used information theory to evaluate the streaming features which is a strong and well-known approach in the field of feature selection. Therefore, our goal has been to compare the proposed methods with methods that have a similar structure to the proposed method in terms of ideas and methods that have used different ideas to get the best evaluation of the proposed method.

For all simulations, the value of the parameter for all algorithms is set according to the recommendation of the following article. For the classification performance of all algorithms, the classifiers k-nearest neighbor (KNN) is chosen as representatives of lazy classifiers (non-parametric) and CART as representative of eager classifiers (parametric). All experiments are performed using MATLAB R2018a and a Windows Server 2013-64 bit machine with 64 GB Ram and Intel (R) Xeon (R) Gold 6254 CPU with 16 3.10 GHz processors. In addition, the fitcknn and fitctree functions run the KNN and CART algorithms, respectively. These functions are from the MATLAB Statistics and Machine Learning Toolbox with the default setting. The K value of KNN is set to 5 and was determined based on multiple experiments to obtain the best value.

The classification performance of each method in each dataset has been evaluated by hold-out validation. For this purpose, 70 percent of the samples are randomly selected for the training set and the other 30 percent for the test set. To simulate the online streaming feature selection problem, the features from the training set are added incrementally, and after adding each new feature, the algorithms update the feature set. The final selected feature set is used for the comparisons. It should also be noted that the announced results are based on an average of 30 independent runs for each algorithm.

One of the current challenges in online streaming feature selection is the sensitivity of the methods to feature ordering. When the features of a dataset are passed to the algorithms in different orders, different features may be selected. This is because redundancy is considered in online feature selection methods with the current feature set. Therefore, it may affect other types of redundancy arrangements differently. Of course, this issue is not so unreasonable in the streaming data, but it is not suitable in a scenario where our offline data set does not fit in the memory for processing, and we process it in a streaming manner. The proposed method also has this challenge. It seems that in cases where there are many training samples in the dataset, the ensemble approach takes some time, and the number of calculations increases.

The classification performance of all algorithms was examined using two different classifiers and four classification criteria, the number of final selected features, sensitivity to the order of training samples, and run-time. Thus, Tables 4 to 11 present the classification reports obtained. Also, a nonparametric Friedman test [7] is performed to compare the significance of the results based

TABLE 3. The main characteristics of the datasets.

Dataset	Number of instances	Number of features	Number of classes	Domain
Sorlie	85	457	5	Microarray Data
Leukemia	72	7070	2	Microarray Data
Colon	62	2000	2	Microarray Data
Prostate-GE	102	5967	2	Biological Data
Subramanian	50	10101	2	Image Data
NCI60	60	9703	2	Microarray Data

on a statistical approach. These comparisons are shown in Table 12. To test how sensitive all methods are to different subsets of training samples, we tested the performance of these methods using 10 different subsets of training samples. This experiment was performed on the Leukemia dataset, and the number of features selected based on each subset of training samples is shown in Figure 4. The number of final selected features and the running time of the methods are shown in Figure 5 and Figure 6, respectively. Run-times are obtained using the tic-toc function in MATLAB.

TABLE 4. Results based on the KNN classifier (Accuracy)

Dataset	OSFSMI	OFS-A3M	OFS-Density	Alpha-Investing	OFSFI	OFSV	RHOFS	OFS-FPD
Prostate-GE	0.8300	0.7475	0.8987	0.8663	0.9063	0.9063	0.8913	**0.9075**
Leukemia	0.8982	0.8232	0.9143	0.8679	0.9161	0.9161	0.8804	**0.9375**
Subramanian	0.6150	0.6300	0.6175	0.4800	0.6300	0.6300	0.6500	**0.7025**
Sorlie	0.5721	**0.7000**	0.6618	0.2662	0.6162	0.6162	0.6588	0.6706
Colon	0.7125	0.6792	0.6896	0.5292	0.7556	0.7650	0.7611	**0.8056**
NCI60	0.2380	0.3120	0.2460	0.1020	**0.4600**	0.4531	0.2620	0.3020

TABLE 5. Results based on the KNN classifier (Sensitivity)

Dataset	OSFSMI	OFS-A3M	OFS-Density	Alpha-Investing	OFSFI	OFSV	RHOFS	OFS-FPD
Prostate-GE	**1.0000**	0.4737	0.9217	0.8421	0.9014	0.9014	0.8876	0.9431
Leukemia	0.9444	0.8889	0.9444	0.9444	0.9624	0.9624	0.9033	**0.9602**
Subramanian	0.8462	0.6154	0.8462	0.6923	0.8340	0.8340	0.7809	**0.8737**
Sorlie	0.5951	0.6289	0.5083	0.2244	0.6320	0.6320	0.6477	**0.6767**
Colon	0.8571	**0.9286**	0.8571	**0.9286**	0.8388	0.8120	0.7758	0.9130
NCI60	0.3167	0.2167	0.3500	0.1000	**0.4198**	**0.4198**	0.2554	0.2668

5.3. **Discussion.** Using the results from the previous section, we can observe the optimality and efficiency of the proposed algorithm by comparing it with similar algorithms from the literature. One of our main ideas in this paper is to use multiple evaluation criteria instead of just one. The reason for this approach is that each feature evaluation algorithm somehow determines the importance of each feature and their combination can lead to a more comprehensive study. For this aggregation process, three filter-based algorithms

Table 6. Results based on the KNN classifier (Precision)

Dataset	OSFSMI	OFS-A3M	OFS-Density	Alpha-Investing	OFSFI	OFSV	RHOFS	OFS-FPD
Prostate-GE	0.8261	0.6429	**0.9217**	0.8889	0.9111	0.9111	0.8976	0.8795
Leukemia	0.9444	0.7619	0.8947	0.8947	0.9190	0.9190	0.9185	**0.9460**
Subramania	n 0.7857	0.5714	0.6875	0.6000	0.6995	0.6995	0.7238	**0.7504**
Sorlie	0.5231	**0.6976**	0.6393	0.2672	0.6370	0.6370	0.6400	0.6797
Colon	0.7059	**0.8667**	0.7059	0.6842	0.8022	0.8040	0.8514	0.8100
NCI60	0.4073	0.3177	0.2870	0.2778	**0.5161**	**0.5161**	0.3149	0.3315

Table 7. Results based on the KNN classifier (F-Measure)

Dataset	OSFSMI	OFS-A3M	OFS-Density	Alpha-Investing	OFSFI	OFSV	RHOFS	OFS-FPD
Prostate-GE	0.8327	0.7369	0.8935	0.8677	0.9007	0.9007	0.8866	**0.9071**
Leukemia	0.9203	0.8690	0.9352	0.8987	0.9359	0.9359	0.9048	**0.9517**
Subramanian	0.7123	0.7152	0.7169	0.6047	0.7479	0.7479	0.7408	**0.7962**
Sorlie	0.5510	**0.6759**	0.6477	0.3182	0.5977	0.5977	0.6344	0.6498
Colon	0.7712	0.7467	0.7382	0.5693	0.8101	0.8144	0.8022	**0.8539**
NCI60	0.4067	0.5009	0.4715	0.3019	**0.6011**	**0.6011**	0.4721	0.4598

Table 8. Results based on the CART classifier (Accuracy)

Dataset	OSFSMI	OFS-A3M	OFS-Density	Alpha-Investing	OFSFI	OFSV	RHOFS	OFS-FPD
Prostate-GE	0.8075	0.7787	**0.8650**	0.8287	0.8238	0.8238	0.8213	0.8513
Leukemia	0.8875	0.8250	0.9089	0.8964	0.8982	0.8982	0.8625	**0.9089**
Subramanian	0.5700	0.6075	0.5925	0.5550	0.5900	0.5900	0.5925	**0.6600**
Sorlie	0.5897	0.5279	0.5868	0.2971	0.5824	0.5824	0.6029	**0.6206**
Colon	0.7250	0.6271	0.6750	0.5979	0.7444	0.7488	0.7222	**0.7750**
NCI60	0.1780	0.2100	0.1880	0.1240	0.2400	0.2400	0.2300	**0.2580**

Table 9. Results based on the CART classifier (Sensitivity)

Dataset	OSFSMI	OFS-A3M	OFS-Density	Alpha-Investing	OFSFI	OFSV	RHOFS	OFS-FPD
Prostate-GE	0.8421	0.4737	0.8300	0.7368	0.8244	0.8244	0.8126	**0.8578**
Leukemia	**1.0000**	**1.0000**	0.9444	**1.0000**	0.9248	0.9248	0.8841	0.9224
Subramanian	0.6923	0.6154	0.7692	0.7692	0.6763	0.6763	0.6835	**0.7715**
Sorlie	0.5951	**0.6289**	0.5083	0.2244	0.5497	0.5497	0.5802	0.5891
Colon	0.9286	0.9286	**1.0000**	0.9286	0.8320	0.8401	0.7694	0.8696
NCI60	0.2667	**0.4333**	0.1333	0.1333	0.2369	0.2369	0.2352	0.2430

Table 10. Results based on the CART classifier (Precision)

Dataset	OSFSMI	OFS-A3M	OFS-Density	Alpha-Investing	OFSFI	OFSV	RHOFS	OFS-FPD
Prostate-GE	0.8000	0.6429	0.8481	0.7368	0.8225	0.8225	0.8317	**0.8503**
Leukemia	0.8571	0.7826	**0.9444**	0.9000	0.9214	0.9214	0.9057	0.9341
Subramanian	**0.7500**	0.6667	0.6250	0.6667	0.7124	0.7124	0.7008	0.7450
Sorlie	0.5777	0.5136	0.5888	0.2412	0.5544	0.5544	0.5897	**0.6041**
Colon	0.7647	0.6842	0.6667	0.6842	**0.7977**	0.7961	0.8027	0.7940
NCI60	0.2917	**0.4353**	0.1567	0.1917	0.2657	0.2657	0.2521	0.2924

Table 11. Results based on the CART classifier (F-Measure)

Dataset	OSFSMI	OFS-A3M	OFS-Density	Alpha-Investing	OFSFI	OFSV	RHOFS	OFS-FPD
Prostate-GE	0.7988	0.7554	**0.8617**	0.8224	0.8139	0.8139	0.8162	0.8477
Leukemia	0.9149	0.8688	**0.9296**	0.9235	0.9204	0.9204	0.8881	0.9267
Subramanian	0.6601	0.6883	0.6858	0.6685	0.6770	0.6770	0.6804	**0.7457**
Sorlie	0.5723	0.5104	0.5811	0.3353	0.5798	0.5798	0.5923	**0.6141**
Colon	0.7859	0.7017	0.7357	0.7195	0.8021	0.8091	0.7736	**0.8221**
NCI60	0.3934	0.4338	0.3592	0.3194	**0.4540**	**0.4540**	0.4277	0.4219

Table 12. Friedman ranks of methods based on 4 criteria and 2 classifiers

Dataset	OSFSMI	OFS-A3M	OFS-Density	Alpha-Investing	OFSFI	OFSV	RHOFS	OFS-FPD
KNN(Accuracy)	6.33	5.33	4.83	7.50	2.83	2.50	4.17	**1.67**
KNN(Sensitivity)	3.50	6.17	3.83	5.83	3.17	3.33	5.83	**2.50**
KNN(Precision)	4.33	5.17	5.00	7.00	3.33	3.17	3.83	**3.00**
KNN(F-Measure)	6.33	5.33	4.67	7.50	2.50	2.33	4.50	**2.00**
CART(Accuracy)	5.67	6.17	3.67	6.67	3.67	3.50	4.50	**1.17**
CART (Sensitivity)	2.17	3.50	4.00	4.50	5.17	5.00	6.17	**3.33**
CART (Precision)	4.33	6.00	5.00	6.67	3.50	3.67	3.67	**2.00**
CART (F-Measure)	6.17	6.00	3.50	6.00	3.67	3.50	4.33	**2.00**

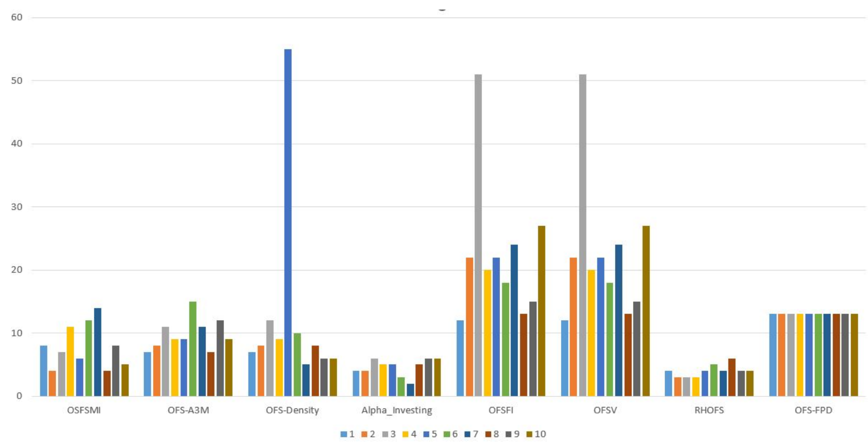

Figure 4. Number of selected features by different subsets of training samples

are used based on different scores. In most of the comparison algorithms, the importance of features is determined based on a single metric. From Tables 4 to 4, it can be seen that the proposed algorithm has better classification performance. Table 12 also shows the Friedman ranks. We can conclude that the proposed method performs statistically better than the comparison methods on

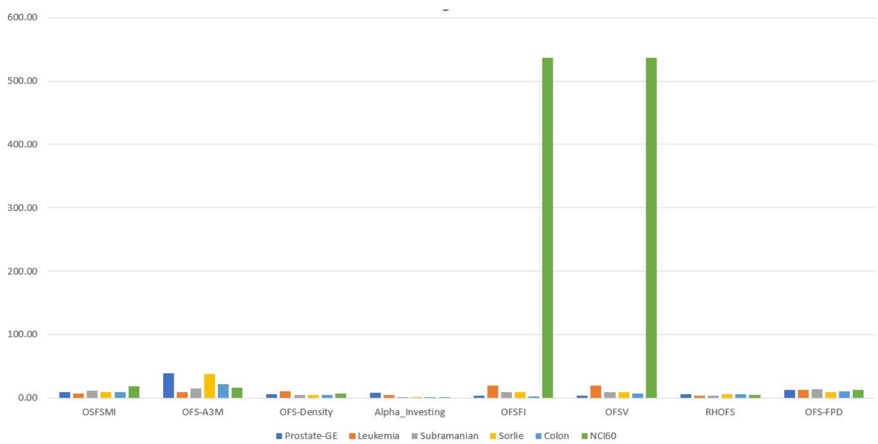

FIGURE 5. Number of final selected features

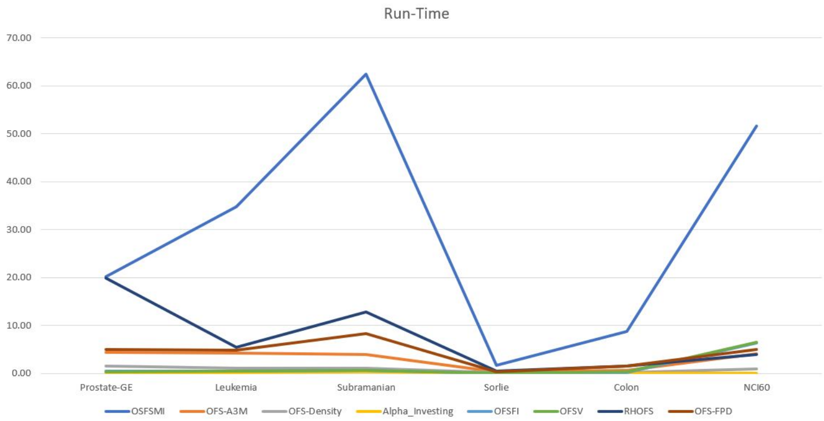

FIGURE 6. Run-times

all criteria. Thus, we can claim that the proposed algorithm has selected the most optimal feature subsets compared to similar algorithms in the literature.

The multi-objective optimization model is our main approach for selecting streaming features. The fuzzy Pareto dominance relationship ranks the features. The Pareto dominance approach has shown promising performance with two or three objectives. Based on these reasons, we have chosen this approach to achieve the optimal feature set as quickly as possible. However, computing three criteria for updating the feature space each time the number of features exceeds the maximum capacity can increase the computational cost and slow down the algorithm. As you can see in Figure 6, the execution time of the proposed algorithm is slower than some other methods. However, it can be said that the proposed method performs in a reasonable time and does not add much computational cost to the model.

If we want to calculate the computational complexity of the proposed method in the worst case, the number of features is equal to $log(d)$ each time the set of features needs to be updated. To calculate the degree of dominance of each feature over other features, this value must be calculated for each feature by the number of other features. Therefore, in general, $d \times \log(d)$ values must be calculated. Thus, we can say that the computational complexity of the proposed method is equal to $O\,(d \times \log(d)\ )$. It means that if the dataset includes 1000 (d) features, the feature set capacity is equal to $log(1000) = 10$. So, in the worst case, the maximum number of calculations to achieve degrees of dominance is 10000.

Table 1 shows that all OSFS methods compared in this study select a different number of features for different subsets of training samples in a dataset. In fact, all of these methods are sensitive to the order of the training samples. This problem is illustrated in Figure 4 for all methods in the Leukemia dataset. Ten different subsets of training samples are presented to the algorithms and the number of selected features in these samples is indicated. This figure shows that only the proposed method selected the same number of features in all 10 cases. This is because we made the number of selected features depend on the number of feature streams. On the other hand, the proposed method is not sensitive to a certain threshold value. Therefore, different arrangements of the training samples do not affect the number of selected features.

The proposed method is not sensitive to any particular parameter. Therefore, the number of selected features in different datasets has no significant variance. The feature selection capacity also increases by increasing the number of features in a dataset. Figure 5 shows that the proposed method selects a reasonable number of features, and the selection variance between different datasets is minimal. This problem occurs in most of the compared methods.

It is also evident from the results that the proposed method has selected an acceptable number of features. Of course, this number is not minimal, but the performance evaluation of a feature selection method is not based only on the selection of the lowest number of features. Instead, an acceptable and minimal number of features with the highest predictive power is determined in a reasonable amount of time. Based on the obtained results, it can be said that the proposed method was able to balance between these criteria.

6. **Conclusion**

This article presents a new online streaming feature selection based on a multi-objective optimization model. We intend to select a minimal and optimal subset of the arriving features using the ensemble of filter algorithms in this model. The fuzzy Pareto dominance concept is considered to rank the features.

We have achieved a reasonable run-time since the fuzzy dominance relationship is a simple and quick procedure. By conducting various experiments, the efficiency and optimality of the proposed algorithm have been shown compared to similar algorithms in the literature. One of the current challenges in online streaming feature selection is the sensitivity of the methods to feature ordering. When the features of a dataset are passed to the algorithms in different orders, different features may be selected. This challenge still exists in the proposed methods, and so far no method has completely solved it. Therefore, this area can be considered as the way of the future. In addition, the methods presented in this area are usually of filter type because they have the least processing time, and wrapper methods are time consuming due to the use of learning algorithms. However, embedded methods can also be used to improve performance, which is a broad area to investigate.

7. Author Contributions

8. Data Availability Statement

Data generated during the study are subject to a data sharing mandate and available in a few public repositories. All used data are cited in text.

9. Ethical considerations

The authors avoided from data fabrication and falsification.

10. Funding

This research did not receive any specific grant from the public, commercial, or not-for-profit funding agencies.

11. Conflict of interest

The authors declare no conflict of interest.

References

[1] Bayati, H., Dowlatshahi, M. B., & Hashemi, A. (2022). MSSL: A memetic-based sparse subspace learning algorithm for multi-label classification. International Journal of Machine Learning and Cybernetics, 13(11), 3607–3624. https://doi.org/10.1007/s13042-022-01616-5.

[2] Bolón-Canedo, V., & Alonso-Betanzos, A. (2018). Evaluation of Ensembles for Feature Selection. In V. Bolón-Canedo & A. Alonso-Betanzos (Eds.), Recent Advances in Ensembles for Feature Selection (pp. 97–113). Springer International Publishing. https://doi.org/10.1007/978-3-319-90080-3-6.

[3] Dhal, P., & Azad, C. (2022). A comprehensive survey on feature selection in the various fields of machine learning. Applied Intelligence, 52(4), 4543–4581. https://doi.org/10.1007/s10489-021-02550-9.

[4] Dowlatshahi, M. B., & Hashemi, A. (2023). Unsupervised feature selection: A fuzzy multi-criteria decision-making approach. Iranian Journal of Fuzzy Systems, 20(7), 55–70. https://doi.org/10.22111/IJFS.2023.7630.

[5] Dowlatshahi, M. B., Zare-Chahooki, M. A., Beiranvand, S., & Hashemi, A. (2022). GKRR: A gravitational-based kernel ridge regression for software development effort estimation. Journal of Mahani Mathematical Research, 11(3), 147–174. https://doi.org/10.22103/jmmr.2022.18988.1202.

[6] Eskandari, S., & Seifaddini, M. (2023). Online and offline streaming feature selection methods with bat algorithm for redundancy analysis. Pattern Recognition, 133, 109007. https://doi.org/10.1016/j.patcog.2022.109007.

[7] Friedman, M. (1940). A Comparison of Alternative Tests of Significance for the Problem of m Rankings. The Annals of Mathematical Statistics, 11(1), 86–92. https://doi.org/10.1214/aoms/1177731944.

[8] Hashemi, A., Bagher Dowlatshahi, M., & Nezamabadi-pour, H. (2021). A pareto-based ensemble of feature selection algorithms. Expert Systems with Applications, 180, 115130. https://doi.org/10.1016/j.eswa.2021.115130.

[9] Hashemi, A., Bagher Dowlatshahi, M., & Nezamabadi-pour, H. (2021). An efficient Pareto-based feature selection algorithm for multi-label classification. Information Sciences, 581, 428–447. https://doi.org/10.1016/j.ins.2021.09.052.

[10] Hashemi, A., Dowlatshahi, M. B., & Nezamabadi-pour, H. (2021). Minimum redundancy maximum relevance ensemble feature selection: A bi-objective Pareto-based approach. Journal of Soft Computing and Information Technology. https://jscit.nit.ac.ir/article-138958-en.html.

[11] Hashemi, A., Dowlatshahi, M. B., & Nezamabadi-pour, H. (2021). VMFS: A VIKOR-based multi-target feature selection. Expert Systems with Applications, 182, 115224. https://doi.org/10.1016/j.eswa.2021.115224.

[12] Hashemi, A., Dowlatshahi, M. B., & Nezamabadi-pour, H. (2022). Ensemble of feature selection algorithms: A multi-criteria decision-making approach. International Journal of Machine Learning and Cybernetics, 13(1), 49–69. https://doi.org/10.1007/s13042-021-01347-z.

[13] Hashemi, A., Joodaki, M., Joodaki, N. Z., & Dowlatshahi, M. B. (2022). Ant colony optimization equipped with an ensemble of heuristics through multi-criteria decision making: A case study in ensemble feature selection. Applied Soft Computing, 124, 109046. https://doi.org/10.1016/j.asoc.2022.109046.

[14] Hashemi, A., Pajoohan, M.-R., & Dowlatshahi, M. B. (2022). Online streaming feature selection based on Sugeno fuzzy integral. 2022 9th Iranian Joint Congress on Fuzzy and Intelligent Systems (CFIS), 1–6. https://doi.org/10.1109/CFIS54774.2022.9756477.

[15] Hashemi, A., Pajoohan, M.-R., & Dowlatshahi, M. B. (2023). An election strategy for online streaming feature selection. 28th International Computer Conference, Computer Society of Iran (CSICC), 01–04. https://doi.org/10.1109/CSICC58665.2023.10105319.

[16] Hu, X., Zhou, P., Li, P., Wang, J., & Wu, X. (2018). A survey on online feature selection with streaming features. Frontiers of Computer Science, 12(3), 479–493. https://doi.org/10.1007/s11704-016-5489-3.

[17] Joodaki, M., Dowlatshahi, M. B., & Joodaki, N. Z. (2021). An ensemble feature selection algorithm based on PageRank centrality and fuzzy logic. Knowledge-Based Systems, 233, 107538. https://doi.org/10.1016/j.knosys.2021.107538.

[18] Kashef, S., & Nezamabadi-pour, H. (2019). A label-specific multi-label feature selection algorithm based on the Pareto dominance concept. Pattern Recognition, 88, 654–667. https://doi.org/10.1016/j.patcog.2018.12.020.

[19] Krzeszowska-Zakrzewska, B. (2015). Fuzzy Pareto Dominance in Multiple Criteria Project Scheduling Problem. Multiple Criteria Decision Making, 10, 93–104.

[20] Li, M., Yang, S.,& Liu, X. (2015). Bi-goal evolution for many-objective optimization problems. Artificial Intelligence, 228, 45–65. https://doi.org/10.1016/j.artint.2015.06.007.

[21] Luo, C., Wang, S., Li, T., Chen, H., Lv, J., & Yi, Z. (2023). RHD-OFS: A Distributed Online Algorithm Towards Scalable Streaming Feature Selection. IEEE Transactions on Parallel and Distributed Systems, 34(6), 1830–1847. https://doi.org/10.1109/TPDS.2023.3265974.

[22] Miri, M., Dowlatshahi, M. B., Hashemi, A., Rafsanjani, M. K., Gupta, B. B., & Alhalabi, W. (2022). Ensemble feature selection for multi-label text classification: An intelligent order statistics approach. International Journal of Intelligent Systems, 37(12), 11319–11341. https://doi.org/10.1002/int.23044.

[23] Pajoohan, M.-R., Hashemi, A., & Dowlatshahi, M. B. (2022). An online streaming feature selection method based on the Choquet fuzzy integral. Fuzzy Systems and Its Applications, 5(1), 161–185. https://doi.org/10.22034/jfsa.2022.331660.1116.

[24] Prajapati, A. (2021). Two-Archive Fuzzy-Pareto-Dominance Swarm Optimization for Many-Objective Software Architecture Reconstruction. Arabian Journal for Science and Engineering, 46(4), 3503–3518. https://doi.org/10.1007/s13369-020-05147-5.

[25] Rafie, A., Moradi, P., & Ghaderzadeh, A. (2023). A Multi-Objective online streaming Multi-Label feature selection using mutual information. Expert Systems with Applications, 216, 119428. https://doi.org/10.1016/j.eswa.2022.119428.

[26] Rahmaninia, M., & Moradi, P. (2018). OSFSMI: Online stream feature selection method based on mutual information. Applied Soft Computing, 68, 733–746. https://doi.org/10.1016/j.asoc.2017.08.034.

[27] Serrano-Guerrero, J., Romero, F. P., & Olivas, J. A. (2021). Fuzzy logic applied to opinion mining: A review. Knowledge-Based Systems, 222, 107018. https://doi.org/10.1016/j.knosys.2021.107018.

[28] Suryanarayan, P., Subramanian, A., & Mandalapu, D. (2010). Dynamic Hand Pose Recognition Using Depth Data. 20th International Conference on Pattern Recognition, 3105–3108. https://doi.org/10.1109/ICPR.2010.760.

[29] Talbi, E. (2009). Metaheuristics: From design to implementation. John Wiley & Sons.

[30] Wang, J., Zhao, P., Hoi, S. C. H., & Jin, R. (2014). Online Feature Selection and Its Applications. IEEE Transactions on Knowledge and Data Engineering, 26(3), 698–710. https://doi.org/10.1109/TKDE.2013.32.

[31] Wu, D., He, Y., Luo, X., & Zhou, M. (2022). A Latent Factor Analysis-Based Approach to Online Sparse Streaming Feature Selection. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 52(11), 6744–6758. https://doi.org/10.1109/TSMC.2021.3096065.

[32] You, D., Sun, M., Liang, S., Li, R., Wang, Y., Xiao, J., Yuan, F., Shen, L., & Wu, X. (2022). Online feature selection for multi-source streaming features. Information Sciences, 590, 267–295. https://doi.org/10.1016/j.ins.2022.01.008.

[33] Zadeh, L. A. (1965). Fuzzy sets. Information and Control, 8(3), 338–353. https://doi.org/10.1016/S0019-9958(65)90241-X.

[34] Zaman, E. A. K., Mohamed, A., & Ahmad, A. (2022). Feature selection for online streaming high-dimensional data: A state-of-the-art review. Applied Soft Computing, 127, 109355. https://doi.org/10.1016/j.asoc.2022.109355.

[35] Zhou, J., P. Foster, D., A. Stine, R., & H. Ungar, L. (2006). Streamwise feature selection. Journal of Machine Learning Research, 3(2), 1532–4435. https://dl.acm.org/doi/abs/10.5555/1248547.1248614.

[36] Zhou, P., Hu, X., Li, P., & Wu, X. (2019). OFS-Density: A novel online streaming feature selection method. Pattern Recognition, 86, 48–61. https://doi.org/10.1016/j.patcog.2018.08.009.

[37] Zhou, P., Hu, X., Li, P., & Wu, X. (2019). Online streaming feature selection using adapted Neighborhood Rough Set. Information Sciences, 481, 258–279. https://doi.org/10.1016/j.ins.2018.12.074.

[38] Zhou, P., Zhang, Y., Li, P., & Wu, X. (2022). General assembly framework for online streaming feature selection via Rough Set models. Expert Systems with Applications, 204, 117520. https://doi.org/10.1016/j.eswa.2022.117520.

[39] ZhouPeng, ZhaoShu, YanYuanting, & WuXindong. (2022). Online Scalable Streaming Feature Selection via Dynamic Decision. ACM Transactions on Knowledge Discovery from Data (TKDD), 16(5), 1–20. https://doi.org/10.1145/3502737.

Amin Hashemi
Orcid number: 0000-0001-8019-1721
Department of Department of Computer Engineering
Yazd University
Yazd, Iran
Email address: hashemi.am@stu.yazd.ac.ir

Mohammad-Reza Pajoohan
Orcid number: 0000-0002-7848-2025
Department of Department of Computer Engineering
Yazd University
Yazd, Iran
Email address: pajoohan@yazd.ac.ir

Mohammad Bagher Dowlatshahi
Orcid number: 0000-0002-4862-0626
Department of Department of Computer Engineering
Lorestan University
Khorramabad, Iran
Email address: dowlatshahi.mb@lu.ac.ir