Data Science and Management 7 (2024) 64–73

Contents lists available at ScienceDirect

Data Science and Management

journal homepage: www.keaipublishing.com/en/journals/data-science-and-management

Research article

# Hybrid distributed feature selection using particle swarm optimization-mutual information

Khumukcham Robindro, Sanasam Surjalata Devi, Urikhimbam Boby Clinton, Linthoingambi Takhellambam, Yambem Ranjan Singh, Nazrul Hoque *

*Department of Computer Science, Manipur University, Imphal, Manipur, 795003, India*

ARTICLE INFO

*Keywords:*
Feature selection
Particle swarm optimization (PSO)
Classification
Accuracy

ABSTRACT

Feature selection (FS) is a data preprocessing step in machine learning (ML) that selects a subset of relevant and informative features from a large feature pool. FS helps ML models improve their predictive accuracy at lower computational costs. Moreover, FS can handle the model overfitting problem on a high-dimensional dataset. A major problem with the filter and wrapper FS methods is that they consume a significant amount of time during FS on high-dimensional datasets. The proposed "HDFS(PSO-MI): hybrid distribute feature selection using particle swarm optimization-mutual information (PSO-MI)", is a PSO-based hybrid method that can overcome the problem mentioned above. This method hybridizes the filter and wrapper techniques in a distributed manner. A new combiner is also introduced to merge the effective features selected from multiple data distributions. The effectiveness of the proposed HDFS(PSO-MI) method is evaluated using five ML classifiers, i.e., logistic regression (LR), k-NN, support vector machine (SVM), decision tree (DT), and random forest (RF), on various datasets in terms of accuracy and Matthew's correlation coefficient (MCC). From the experimental analysis, we observed that HDFS(PSO-MI) method yielded more than 98%, 95%, 92%, 90%, and 85% accuracy for the unbalanced, kidney disease, emotions, wafer manufacturing, and breast cancer datasets, respectively. Our method shows promising results comapred to other methods, such as mutual information, gain ratio, Spearman correlation, analysis of variance (ANOVA), Pearson correlation, and an ensemble feature selection with ranking method (EFSRank).

## 1. Introduction

Feature selection (FS) is an integral preprocessing step of machine learning (ML) that evaluates the original feature set to find the most informative and nonredundant features. The FS method evaluates features by using a selection criterion or objective function to select the most informative features (Li et al., 2017). Owing to the tremendous increase in data volume and complexity, the FS is considered an important ML step for dimensionality reduction. The primary goal of FS is to obtain a subset of relevant and informative features that can enhance the performance of an ML model, reduce time and space complexity, and prevent model overfitting (Venkatesh and Anuradha, 2019). Based on their selection mechanism, FS methods can be classified into four types: filters, wrappers, embeddings, and hybrids (Hoque et al., 2014). The filter method uses a feature evaluation measure to determine the effectiveness of each feature. It computes a score and ranks each feature based on the calculated score. Most filter-based FS methods employ a greedy approach to iteratively determine the best features. To evaluate high-dimensional data, the filter method is considered cost-effective in terms of time. In the wrapper, an explicitly used ML method functions as an evaluator to validate all possible subsets of features generated from the original feature group. The wrapper method uses an exhaustive search to assess all possible subsets of features using a learning algorithm and selects the subset that gives the best performance. Because the wrapper method uses an exhaustive search to evaluate all possible subsets of the original feature group, its computational time is very long compared to that of the filter method. However, wrapper methods performed better than filter methods. An embedded method considers FS as an integral part of the training process. Features are selected when training a learning model. The hybrid method combines any of the three previously mentioned

Peer review under responsibility of Xi'an Jiaotong University.
* Corresponding author.
*E-mail address:* tonazrul@gmail.com (N. Hoque).

https://doi.org/10.1016/j.dsm.2023.10.003
Received 17 December 2022; Received in revised form 5 October 2023; Accepted 7 October 2023
Available online 14 October 2023

methods. Hybrid methods are widely used in many ML applications because of their excellent discriminative behavior in pattern recognition and data analysis. In this paper, we discuss our proposed FS method called the "hybrid distributed feature selection using particle swarm optimization-mutual information (PSO-MI)" abbreviated as HDFS(PSO-MI), to select the most informative features from a large feature pool.

Feature selection plays a significant role in selecting informative, relevant, and non-redundant features from a large feature space during data preprocessing. After a comprehensive study of existing feature selection techniques, we observed that many wrapper-based FS methods ignore feature redundancy, and filter-based methods select features with some redundancy among the selected features. Moreover, the computational cost of wrapper techniques is high, and the selection of an optimal subset of features from a high-dimensional dataset is a major problem for ML researchers. Among existing FS methods, incorporating a single objective function into the standalone FS method does not yield satisfactory results for high-dimensional data. To overcome these problems, we developed a hybrid feature-selection method combining the concepts of PSO optimization as well as MI. The PSO optimization is applied to a distributed dataset to select optimal features from each distribution and then combine the solutions of all distributions using MI, which yields the best subset of features. The proposed hybrid model integrates both the filter and wrapper methods. As the wrapper method consumes a significant amount of time, we applied it in a distributed manner. The wrapper methods are executed in parallel with multiple objective functions during the feature subset evaluation. This step significantly reduces the computational cost. Moreover, for each partition of the original data, the proposed method employs a filter FS using MI and combines the features from each partition to obtain the best set of optimal features. The main advantage of the proposed hybrid model is that it reduces the computational cost of feature selection by the parallel execution of the method on multiple cores.

The major problems of FS selection in high-dimensional datasets are the computational cost and performance. For a given dataset D with $n$ numbers of features, an FS method needs to select a subset of features, $m$, such that $m < n$. The subset of $m$ features should yield better performance with reduced computational cost during prediction. This method should work on distributed datasets to select the best feature subset using PSO and MI.

The main contributions of this study are as follows:

- We developed an effective hybrid feature selection method called HDFS(PSO-MI) using PSO-MI.
- A new objective function is defined to help the PSO optimization algorithm.
- A new combiner method is proposed to combine the subset of features selected from each dataset distribution.
- The proposed hybrid feature selection method is evaluated on high-dimensional datasets.
- A parallel programming is applied to execute the method on multiple cores.
- The effectiveness of the HDFS(PSO-MI) is compared with some existing feature selection methods.

The remainder of this paper is organized as follows. Related studies and existing FS methods are discussed in Section 2. The PSO algorithm is described in Section 3. The proposed HDFS(PSO-MI) method and the corresponding algorithm with a working example are described in Section 4. Experimental results and analyses are presented in Section 5. Finally, Section 6 makes the conclusion and suggests the future work.

## 2. Related work

In the literature, we found several articles on FS (Baruah et al., 2020; Chandrashekar and Sahin, 2014; Kumar and Minz, 2014; Li et al., 2017; Zhu et al., 2023). Most FS methods follow the filter approach and incorporate information-theoretic measures as feature evaluators during the feature selection (Hoque et al., 2016, 2018). In addition, evolutionary-based wrapper FS methods have shown excellent results for high-dimensional datasets (Moslehi and Haeri, 2020; Wang and Huang, 2009). To tackle the unmanageable challenges of computational costs in mining high-dimensional data, an effective FS method using PSO is developed by Fong et al. (2015). In high-dimensional datasets, most FS methods have intractable computational demands because the size of the search space to find the best optimal subset is exponential. Therefore, the authors developed a lightweight FS method incorporating accelerated PSO, which yielded an enhanced performance with reduced computational cost. This method selects the most informative features on big data in an incremental manner, and the selected subsets of features are evaluated on multiple test case sets on big datasets. Researchers have developed variants of the PSO method for feature selection because evolutionary computation approaches have been found to be effective in exploring the confounding effects of feature interactions. A modified binary PSO-based FS method was proposed by Vieira et al. (2013) and used for mortality prediction in patients with sepsis. The enhanced binary particle swarm optimization (BPSO) method optimizes the SVM kernel parameters to manage the premature convergence of the PSO. The BPSO-based FS method works as a wrapper method and evaluates feature subsets using an SVM classifier. The selected subset of features yielded a high accuracy in mortality prediction in patients with sepsis. The PSO technique is combined with other methods, such as genetic algorithms, rough-set theory, and information gain, to develop hybrid FS methods, as PSO-based hybrid methods show promising results for FS. An effective hybrid FS method was proposed by Li et al. (2023) that combines three techniques, i.e., GA-Kmeans, GA-PSO-K-means, and harmony-K-means. This method was applied to enhance the accuracy of diabetes diagnosis applications, achieving an accuracy of 91.65%. The subset of features selected from the diabetic dataset was evaluated using a metaheuristic harmony search, and the performance was improved using the K-means clustering algorithm.

As reported by Moradi and Gholampour (2016), most existing PSO-based FS methods do not consider the correlation or feature-feature interactions during the search for the optimal features. As a result, the probability of selecting redundant features is high, which affects performance. To overcome this problem, Moradi and Gholampour (2016) developed a novel hybrid PSO-FS method called HPSO-LS, which incorporates a new local search strategy. This method uses a correlation-based search strategy that guides the PSO to identify less-correlated features. This method selects less correlated features, known as dissimilar features, with a higher probability than more correlated features. An FS is an optimization problem that considers either single or multiple objectives for feature subset evaluation. If multiple objectives are used during feature selection, this method requires a significant amount of processing time. To address this issue, Bansal et al. (2022) developed a hybrid method with multiobjective optimization using PSO. This method, known as mRMR-PSO, removes redundant and irrelevant features from sign language data. Initially, the method applies a histogram of oriented gradient (HOG) technique for feature extraction and fits the SVM classifier into the PSO. We performed an experimental comparison of the proposed mRMR-PSO with the HOG (without FS), PSO, and mRMR in terms of accuracy and computation time.

Another challenging problem of FS is inadequate handling of very high-dimensional data with a smaller number of samples, and most ML methods face the model overfitting problem on those datasets. In such a situation, neither the filter nor wrapper approach can effectively select the best feature subset. In addition, feature subset instability is a common problem in small sample size data, which has not been properly addressed by many existing FS methods (Brahim and Limam, 2016). Therefore, Brahim and Limam (2016) developed an effective hybrid FS method that incorporated a cooperative subset search technique for

instance learning. They restructured the problem of a small sample size into a filter-based FS tool that can select only a few subsets of informative features. They used cancer datasets to establish the efficacy of their method and proved that their method selects the most stable feature subsets that provide high detection accuracy.

From the above discussion, it is very clear that FS is a generic data preprocessing step of ML that selects the most informative features useful for the respective learning models. Most FS methods employ a feature evaluator or objective function to assess the importance and relevance of features in an optimized manner. As discussed above, many FS methods have been developed to select features from diverse applications such as network data, gene expression, language, disease, and chemical data. We observed that hybrid FS methods are widely used on high-dimensional datasets owing to their calibration in searching for the best feature set with reduced computational cost. Moreover, evolution-based hybrid feature selection methods have gained popularity for providing the best performance by selecting a stable feature set. Considering all the benefits of the hybrid FS methods, we developed the proposed HDFS(PSO-MI) method, which works in a distributed manner with multiple fitness functions operated on multiple partitions of the original dataset.

## 3. PSO

PSO is a popular optimization method developed by Eberhart and Kennedy (1995) that uses a common metaheuristics searching algorithm for optimization. The method was developed based on inspiration from the natural process of social behavior and the dynamic movement of animals. In PSO, solutions known as a group of random particles are initialized first, and the particles are updated iteratively to search for the optimal solution. The position vector and velocity are updated during the optimal search process. The position vector is known as "pbest" or local best for the individual particle's best solution (fitness) so far. The next parameter is "gbest" which represents the best position achieved so far among all the particles. The best position is set as the current global position of the particles, called gbest. PSO uses 1 and 2 to update the positions and velocities of the particles, respectively.

$$v_i^{(k+1)} = \omega v_i^k + C_1 r_1 (pbest_i^k - x_i^k) + C_2 r_2 (gbest_i^k - x_i^k) \tag{1}$$

$$x_i^{(k+1)} = x_i^k + v_i^{(k+1)} \tag{2}$$

The velocity and current position at the $k^{th}$ iteration are denoted by $v_i^k$ and $x_i^k$, respectively. PSO uses two random variables, $r_1$, $r_2$ where $C_1$, $C_2$ are positive constants. The inertia weight, represented by $\omega$, is used to balance the tradeoff between exploration and exploitation. The different parameters and their meaning used in PSO are listed in Table 1. $v_{\max}$ is the upper bound of the velocity in all dimensions, and controls the particle rush movement during the search. PSO techniques have been successfully used in various applications such as data mining, design and modeling, prediction and forecasting, and networking. Although PSO was developed to solve unconstrained single-objective problems, people use PSO variants to solve constrained optimization problems. Our proposed method uses PSO to determine the relevant subset of features using three objective functions. The equations for the objective functions are given by Eqs. (3)–(5).

**Table 1**
Particle swarm optimization (PSO) parameters and objective setting on all datasets of experimental.

| Parameter name | Parameter value/meaning |
|---|---|
| PSO learning factors ($C_1$, $C_2$) | (2, 2) |
| PSO inertia weight ($\omega$) | (0.9) |
| Maximum iteration | 100 |
| *A* | 0.88 |
| *Acc* | Accuracy |
| *Rel_avg* | Average of relevant |
| *Red_avg* | Average of redundancy |
| *SF* | Number of selected features |
| *TF* | Total number of features |

### 3.1. Objective functions used in PSO for feature selection

As previously mentioned, a single objective function may be biased in selecting the best subset of features from a distribution. Hence, we used multiple objective functions for an effective analysis of features in different data distributions. To define the objective function, we used the accuracy value obtained from the SVM classifier using the corresponding features the PSO swarm considers. This accuracy value plays a significant role in selecting the best subset of features, along with other parameters such as relevance (Rel), redundancy (red) of the features, and the ratio between the number of selected features and the total number of features. The major disadvantage of applying multiple objective functions in HDFS (PSO-MI) is that it takes an exponential time on large datasets to evaluate features with the three objective functions incorporated into the PSO.

$$Objective1(obj_1) = \alpha \times (1 - acc) + (1 - \alpha)^* \left(1 - \frac{SF}{TF}\right) \tag{3}$$

$$Objective2(obj_2) = acc \times Rel_{avg} - (1 - acc) \times Red_{avg} \times \frac{SF}{TF} \tag{4}$$

$$Objective3(obj_3) = acc + (1 - acc) \times \frac{SF}{TF} \times (Rel_{avg} - Red_{avg}) \tag{5}$$

### 3.2. Parameter settings in PSO

The PSO algorithm uses several parameters during execution. In our experimental analysis, we initially set the parameters described by Shi and Eberhart (1998) and executed the PSO method for our problem. Next, we experimentally tuned the parameters and selected those that fit the best. The swarm size is an important parameter that influences PSO in determining the optimal solution. If we set a large swarm size, the PSO complexity for finding the best solution from a large search space is high. We found a general heuristic for swarm size in the literature (Moradi and Gholampour, 2016; Zhang et al., 2016). Similarly, the number of iterations was set based on the specific problem. Significantly fewer iterations led to premature termination without obtaining the best solution, and a large number of iterations required computation time to converge. The PSO learning factors $C_1$ and $C_2$ are set empirically, and incorrect values of $C_1$ and $C_2$ may exhibit cyclic behavior in the PSO. The inertia weight $\omega$ should always be less than one to handle divergence or explosion. The parameters and their corresponding values are shown in Table 1.

### 3.3. Complexity analysis of PSO

The computational time of the PSO algorithm depends on multiple parameters, particularly the swarm size, number of iterations, and acceleration coefficients. If the swarm size is $\mu$, the number of iterations is $n$, and the dimension of each particle is $d$, then for $d$ dimension, the PSO takes $O(d \times v)$ and $O(d \times p)$ times to update the particle's velocity and positions, respectively. Because the position and velocity are updated for each particle until they converge or the maximum number of iterations is reached, the total time complexity of PSO is $O(n \times \mu \times O(d \times v) + O(d \times p)$, excluding the swarm initialization time.

## 4. Proposed method

This section discusses the proposed HDFS(PSO-MI) method in detail. The method consists of two phases: (i) PSO feature subset selection and (ii) a combination of feature subsets using MI. In the first phase, we divided the original dataset into $k$ random partitions. We applied the PSO method to each partition as a wrapper to determine the best subset of

features. PSO uses a feature subset evaluation function, known as an objective function. In our proposed method, instead of applying a single objective function, we apply three different objective functions to overcome the possible biases of a single objective function in any distribution of data objects. The outcome of the PSO algorithm is a subset of the relevant features for a particular partition, $d_i$. This method generates three subsets of selected features for each partition. The UNION operation combines features selected from the three subsets. This process executes all partitions of the datasets. If there are $k$ partitions, the method generates $k$ feature subsets. In the next phase, the proposed method combines the three subsets of features and ranks the individual features at the time of combination. This method uses the MI measure to compute the rank of each feature over the entire dataset. Finally, a threshold was calculated from the MI scores of the combined features, and the method selected features with MI scores greater than the threshold value. These are considered the best subsets of features selected by the proposed HDFS(PSO-MI) method. Because the proposed method applies PSO for the initial feature subset selection in a distributed manner, all feature subsets are combined using the MI score. Hence, our method is called HDFS(PSO-MI), i.e., hybrid distributed feature selection using PSO-MI. The workflow of the proposed method is shown in Fig. 1. Next, the MI approaches used in the proposed method are discussed (Fig. 2).

The primary learning factors of PSO, such as social learning factor ($C_1$) and cognitive learning factor ($C_2$), are set as 2. Harb and Desuky (2014) suggested that the values of $C_1$ and $C_2$ can take any random value, but the added values of $C_1$ and $C_2$ should not exceed 4. The maximum number of iterations was set to 100, and the fitness function or objective of the PSO algorithm was set as $obj_1$, $obj_2$, $obj_3$. After initializing the required parameters, the features of the distributed data were

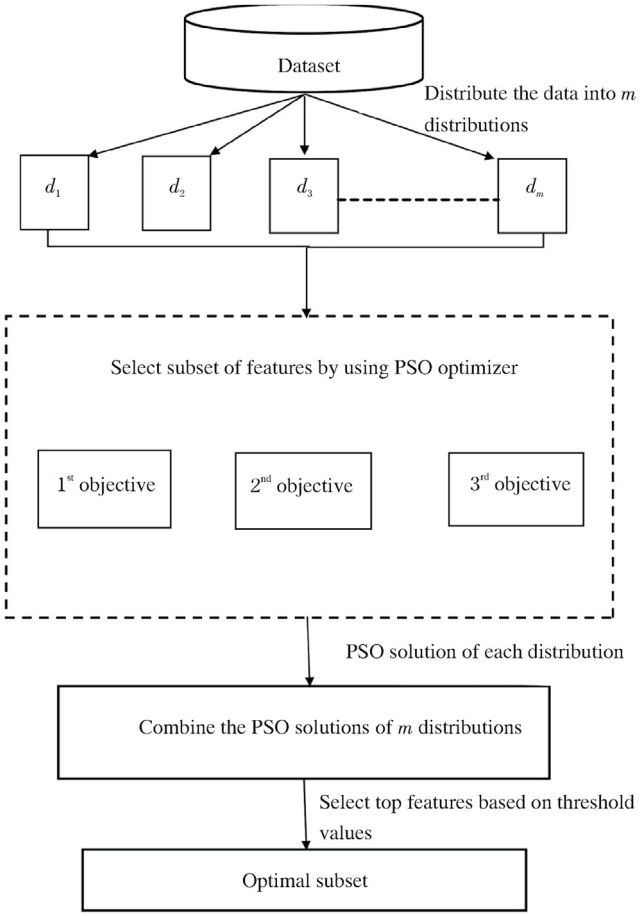


**Fig. 1.** Flowchart of the proposed method.

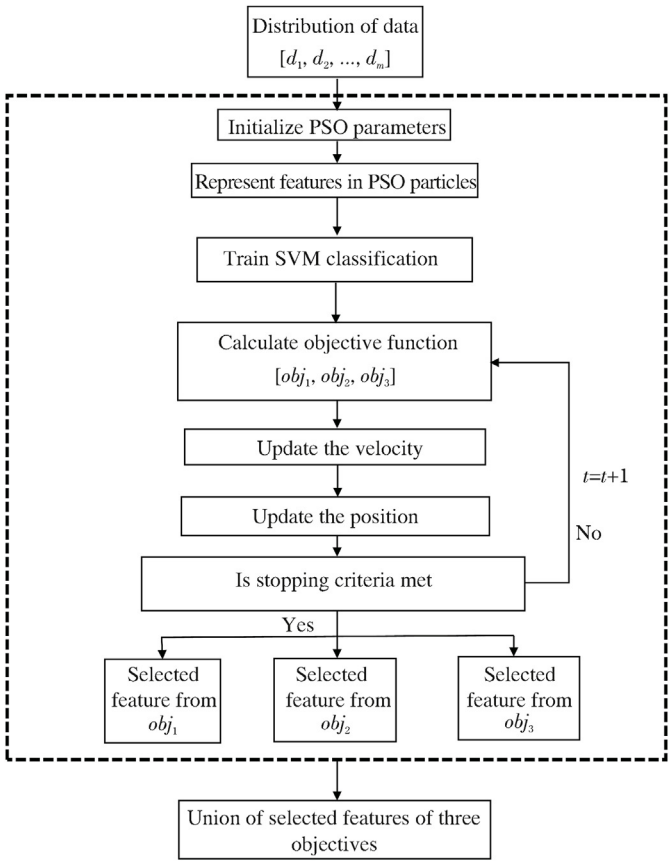


**Fig. 2.** Process of particle swarm optimization (PSO) with three objective functions.

represented as PSO particles. The objective function $obj_1$ is evaluated by setting the initial velocity of each particle to zero and iteration $t$ to $0 + 1$. Then, we find the personal or local best for each particle and the global best for the swarm. Now, consider the random numbers $r_1$ and $r_2$ in the range (0,1), and update each particle's velocity and position using Eqs. (1) and (2). Next, we calculate the objective function $obj_1$ for each distribution and check whether the stopping criteria are satisfied. Otherwise, the number of iterations is increased, and the same process is repeated until the stopping criteria are satisfied. The same process of PSO is performed for the other two objective functions, $obj_2$ and $obj_3$, using the same data distribution. After executing all the objective functions, the UNION operation is performed on the three selected subsets of features generated by the three objective functions to generate one subset of features for evaluation.

The symbols used to describe the proposed algorithm are listed in Table 2.

The algorithm begins its execution on dataset $D$ with $n$ features: Initially, we divide the entire dataset into $m$ distributions; subsequently, for each distribution $d_i \in D$.

- Initialize the PSO parameters and defined the objective $obj_{fn}$;
- Compute the binary PSO and the union of *Pso_SF* where $obj_i \in obj_{fn}$;
- and compute the correlation between feature $f_i$ and class $C$ using MI where $f_i \in$ *Union_Pso_SF*.

Then, we combine the *MI_score* for each feature $f_i \in$ *MI_score* for all distributions $d_1$, $d_2$, …, $d_m$. The threshold value was set to half the maximum score. Finally, features with a value greater than or equal to the threshold value are selected.

**Table 2**
Symbols used and their description.

| Symbols | Meaning |
|---|---|
| $D$ | Dataset |
| $N$ | Number of features in dataset D |
| $C$ | Class label |
| $M$ | Number of distribution to be split |
| $d_i$ | $i^{th}$ distribution |
| $MI$ | Mutual information |
| Binary_PSO | Binary particle swarm optimization (BPSO) |
| Pso_SF | Feature subset generated by BPSO |
| ∪ | UNION |
| $obj_{f_n}$ | Variable that contains all three objectives function |
| $obj_1$ | The first objective of PSO |
| $obj_2$ | The second objective of PSO |
| $obj_3$ | The third objective of PSO |
| $TH$ | Threshold value |

### 4.1. Proposed algorithm

The steps of the proposed HDFS(PSO-MI) method are presented in Algorithm 1. To understand the algorithm better, we use the various symbols shown in Table 2 with their meanings. The algorithm consists of two phases: PSO feature selection and feature ranking using MI. The steps in both phases are presented using the same algorithm.

**Algorithm 1:** HDFS(PSO-MI) Feature Selection

Input: Dataset $D$ with $n$ features, class label $C$, and number of distribution $m$.
Output: $F'$: The subset of $k$ optimal features.

**foreach** $d_i \in D$ **do**
  Initialize Binary_PSO parameters;
  $obj_{fn} = (obj_1, obj_2, obj_3)$;
  **foreach** *objective function* $obj_i \in obj_{fn}$ **do**
    selected feature $Pso\_SF = Binary\_PSO(d_i, C)$;
    $Union\_Pso\_SF = \cup_{i=1}^{3}(Pso\_SF)$;
  **end**
  **foreach** *feature* $f_i \in Union\_Pso\_SF$ **do**
    calculate $MI\_Score = MI(f_i, C)$;
  **end**
**end**
**foreach** *feature* $f_j \in MI\_Score$ **do**
  calculate $Score = combiner(MI\_Score)$;
**end**
$TH = \frac{1}{2}\max(Score)$;
**foreach** *objective feature* $f_j \in MI\_Score$ **do**
  **if** $MI\_Score(f_j) > TH$ **then**
    Select $f_j$ and put it into the set $F'$;
  **end**
**end**
**return** features set $F'$;

### 4.2. Working example

To better understand the proposed method, we discuss a working example. Let us consider a label dataset $D$ with features $F = (f_1, f_2, f_3, f_4, f_5, f_6, f_7)$ and a class label $C$ as listed in Table 3. First, we split the dataset into three categories: distribution 1, distribution 2, and distribution 3, as listed in Table 4. For each distribution, we apply the basic binary PSO algorithm with three different objective functions to select the initial subset of features from the distribution. The accuracy obtained from the SVM classifier is used to define the objective functions. The three objective functions select three subsets from each distribution, which is combined using the UNION operator to generate a subset of the selected features. This process is repeated for the other two distributions (Table 5).

As shown in Table 4, dataset $D$ is partitioned into three distributions, each of which contains six rows as instances, seven columns as features, and $C$ is a label.

This method applies PSO to the first distribution and selects three feature subsets using $obj_1$, $obj_2$, and $obj_3$. Using the UNION operation, the three subsets are merged to obtain a single feature set.

Definition 1. (relevant feature) A feature is called a relevant feature to the target variable if it can discriminate samples correctly w.r.t the given feature. The relevance of individual features can be evaluated using mathematical, statistical, and information-theoretical measures.

**Table 3**
Example dataset.

| Number | $f_1$ | $f_2$ | $f_3$ | $f_4$ | $f_5$ | $f_6$ | $f_7$ | C |
|---|---|---|---|---|---|---|---|---|
| 0 | 4 | 0 | 4 | 4 | 0 | 1 | 2 | 1 |
| 1 | 1 | 2 | 3 | 0 | 1 | 3 | 3 | 1 |
| 2 | 4 | 0 | 2 | 0 | 1 | 3 | 3 | 1 |
| 3 | 3 | 0 | 4 | 5 | 1 | 3 | 2 | 1 |
| 4 | 1 | 2 | 4 | 5 | 2 | 2 | 3 | 1 |
| 5 | 1 | 2 | 3 | 4 | 2 | 2 | 3 | 1 |
| 6 | 4 | 0 | 1 | 0 | 1 | 2 | 3 | 1 |
| 7 | 1 | 2 | 7 | 0 | 1 | 2 | 2 | 1 |
| 8 | 2 | 2 | 8 | 0 | 1 | 2 | 2 | 1 |
| 9 | 2 | 2 | 1 | 0 | 1 | 1 | 3 | 0 |
| 10 | 4 | 0 | 5 | 0 | 1 | 1 | 3 | 0 |
| 11 | 1 | 2 | 3 | 4 | 2 | 2 | 2 | 0 |
| 12 | 2 | 2 | 5 | 0 | 1 | 1 | 3 | 0 |
| 13 | 2 | 2 | 3 | 0 | 1 | 1 | 5 | 0 |
| 14 | 2 | 2 | 1 | 0 | 1 | 2 | 3 | 0 |
| 15 | 4 | 0 | 5 | 0 | 1 | 3 | 1 | 0 |
| 16 | 2 | 2 | 3 | 0 | 1 | 2 | 2 | 0 |
| 17 | 3 | 0 | 3 | 0 | 1 | 3 | 3 | 0 |

Definition 2. (optimal feature set) A subset of feature, say $S = \{f_1, f_2, \cdots, f_k\}$ is called optimal if the performance measured on $S$ by an ML algorithm is always higher than any other subset of features say $S'$.

Proposition 1. The subset of features selected by HDFS(PSO-MI) is relevant and optimal

Proof. HDFS(PSO-MI) first evaluates possible subsets of features using PSO techniques, and then selects the best subset, say $S = \{f_1, f_2, \cdots, f_k\}$. The method evaluates each feature $f_i \in S$ in terms of the MI between the feature $f_i$ and the target variable $C$ and chooses the features highly relevant to $C$. Hence, HDFS(PSO-MI) always selects the relevant and optimal values.

## 5. Experimental analysis

We implemented our proposed HDFS(PSO-MI) hybrid feature selection method on a computer with 8 GB primary memory, i5 $11^{th}$ Gen intel processor, and a 64-bit Windows 11 Operating System using the Python programming language in a Jupiter notebook. During the implementation, we used various Python packages such as NumPy, Pandas, sci-kit learn, and Keras.

### 5.1. Datasets used

Fifteen datasets were used to validate the effectiveness of the proposed method. The datasets are summarized in Table 6. Most of the datasets contained both numerical and categorical features without missing values. All the datasets were imbalanced in terms of class distributions.

### 5.2. Performance measures used

We used various performance analysis measures, as shown in Table 7 to effectively analyze the proposed method. The accuracy and Mathew's correlation coefficients were used to analyze and compare our method with other competing methods.

### 5.3. Parallel execution of the proposed HDFS(PSO-MI) method

We evaluated the HDFS(PSO-MI) method using five large datasets with over 1,000 features. The proposed method takes an exponential amount of time to select an optimal subset of features because the

**Table 4**
Distribution of the example dataset.

| $f_1$ | $f_2$ | $f_3$ | $f_4$ | $f_5$ | $f_6$ | $f_7$ | $C$ |
|---|---|---|---|---|---|---|---|
| *1st distribution* | | | | | | | |
| 4 | 0 | 1 | 0 | 1 | 2 | 3 | 1 |
| 1 | 2 | 4 | 5 | 2 | 2 | 3 | 1 |
| 1 | 2 | 3 | 0 | 1 | 3 | 3 | 1 |
| 4 | 0 | 5 | 0 | 1 | 3 | 1 | 0 |
| 2 | 2 | 1 | 0 | 1 | 1 | 3 | 0 |
| 3 | 0 | 4 | 5 | 1 | 3 | 2 | 1 |
| *2nd distribution* | | | | | | | |
| 2 | 2 | 3 | 0 | 1 | 2 | 2 | 0 |
| 2 | 2 | 5 | 0 | 1 | 1 | 3 | 0 |
| 1 | 2 | 3 | 4 | 2 | 2 | 2 | 0 |
| 1 | 2 | 7 | 0 | 1 | 2 | 2 | 1 |
| 4 | 0 | 5 | 0 | 1 | 1 | 3 | 0 |
| 4 | 0 | 4 | 4 | 0 | 1 | 2 | 1 |
| *3rd distribution* | | | | | | | |
| 1 | 2 | 3 | 4 | 2 | 2 | 3 | 1 |
| 2 | 2 | 3 | 0 | 1 | 1 | 5 | 0 |
| 3 | 0 | 3 | 0 | 1 | 3 | 3 | 0 |
| 4 | 0 | 2 | 0 | 1 | 3 | 3 | 1 |
| 2 | 2 | 1 | 0 | 1 | 2 | 3 | 0 |
| 2 | 2 | 8 | 0 | 1 | 2 | 2 | 1 |

**Table 5**
Selected features using particle swarm optimization (PSO).

| Distribution | Objective | Feature mask | Features | UNION |
|---|---|---|---|---|
| 1 | 1 | [1 1 1 1 1 1 1] | [$f_1f_2f_3f_4f_5f_6f_7$] | [$f_1f_2f_3f_4f_5f_6f_7$] |
| | 2 | [1 1 0 0 0 0 0 ] | [$f_1f_2$ ] | |
| | 3 | [1 0 0 0 0 0 1] | [$f_1f_7$] | |
| 2 | 1 | [1 0 1 1 1 1 1] | [$f_1f_3f_4f_5f_6f_7$] | [$f_1f_3f_4f_5f_6f_7$] |
| | 2 | [1 0 1 0 1 0 0] | [$f_1f_3f_5$] | |
| | 3 | [0 0 1 0 0 0 1] | [$f_3f_7$] | |
| 3 | 1 | [1 1 1 1 1 1 1] | [$f_1f_2f_3f_4f_5f_6f_7$] | [$f_1f_2f_3f_4f_5f_6f_7$] |
| | 2 | [1 0 1 0 0 1 1] | [$f_1f_3f_6f_7$] | |
| | 3 | [1 0 1 0 0 1 0] | [$f_1f_3f_6$ ] | |
| Computed mutual information (MI) for each feature | | | | |
| Feature | *Distribution*$_1$ | *Distribution*$_2$ | *Distribution*$_3$ | Total MI |
| $f_1$ | 0.58 | 0.25 | 0.54 | 1.37 |
| $f_2$ | 0.0 | 0.0 | 0.0 | 0.0 |
| $f_3$ | 0.58 | 0.91 | 0.54 | 20.3 |
| $f_4$ | 0.25 | 0.04 | 0.19 | 0.48 |
| $f_5$ | 0.10 | 0.37 | 0.19 | 0.66 |
| $f_6$ | 0.45 | 0.22 | 0.20 | 0.87 |
| $f_7$ | 0.37 | 0.25 | 0.33 | 0.95 |

dimensions of the dataset are very high, and the PSO method executes multiple objective functions. Because of the exponential execution time, the proposed method may fail to operate in many applications of ML methods in real-time decision-making processes. Therefore, to improve the execution performance of the HDFS(PSO-MI) method, we used parallel programming in a high-performance computing environment. We applied the parallel computing concept to reduce the computations using Python's multiprocessing module. Initially, our proposed algorithm could take any number of distributions as a parameter; however, here, we consider only two because we have limited processor cores. The flow of the execution process is shown in Fig. 3. Using the core of the processor, the process of executing the algorithm begins, and waits until the entire process is completed. After dividing the entire input dataset into two distributions, another processor core executes each distribution. For each distribution, one processor core began the execution process and waited until the entire process was completed. Because we applied three objective functions to each distribution, we used the core of one processor to compute each objective function. We adopted the shared memory concept to group the resulting outputs. The results of each objective function were grouped to provide one result for the distribution. Subsequently, the results of each distribution were combined, and the feature set was selected accordingly. During the program execution, we used nine processor cores.

### 5.4. List of 10 high-ranked features selected by our proposed HDFS(PSO-MI) method

The proposed HDFS(PSO-MI) method ranks all features in descending order of their scores and selects only the high-ranked features. Next, the selected features are fed into the ML model to classify the instances. It is important to know the names of the features selected as the best set from a single dataset. In Table 8, we list only the first 10 high-ranked features from all selected features, owing to space constraints. For some datasets, the method selected fewer than 10 features.

**Table 7**
Performance measures.

| Symbol & metrics | Meaning |
|---|---|
| *TP* | Number of actual positive instances |
| *TN* | Number of actual negative instances |
| *FP* | Number of false positive instances instead of actual negative |
| *FN* | Number of false negative instances instead of actual positive |
| Accuracy | $\text{Accuracy} = \frac{(TP + TN)}{(TP + FP + TN + FN)}$ |
| MCC | $\text{MCC} = \frac{(TN \times TP - FN \times FP)}{\sqrt{((TP + FP)(TP + FN)(TN + FP)(TN + FN))}}$ |

**Table 6**
Dataset description.

| Dataset | Data type | Number of class labels | Number of instances | Number of attributes | Number of selected attributes |
|---|---|---|---|---|---|
| Fetal heath | Real | 3 | 2,126 | 22 | 11 |
| Kidney disease | Categorical, real, integer | 3 | 400 | 26 | 12 |
| ECG | Integer, real | 2 | 1,000 | 28 | 3 |
| Hypothyroid | Categorical | 2 | 3,772 | 30 | 2 |
| Unbalanced | Categorical, real, integer | 2 | 856 | 33 | 27 |
| MOFP | Categorical, real, integer | 2 | 61 | 197 | 156 |
| Bioassay | Categorical, real, integer | 2 | 827 | 915 | 27 |
| Parkinson disease | Integer, real | 2 | 756 | 754 | 687 |
| Cancer normal gene | Real | 2 | 133 | 1,928 | 850 |
| Colon cancer gene | Categorical, real, integer | 2 | 62 | 2,002 | 1,830 |
| Wafer manufacturing | Integer, real | 2 | 1,763 | 1,159 | 3 |
| Breast cancer | Categorical, real, integer | 2 | 705 | 1,941 | 754 |
| Emotions | Categorical, real | 3 | 2,132 | 2,549 | 2,056 |
| Malaria | Categorical, integer | 2 | 5,512 | 2,501 | 2,072 |
| Skin cancer | Integer | 7 | 10,015 | 2,353 | 1,326 |

**Table 8**
Names of the selected high-ranked features.

| Dataset | 10 high-ranked features selected by HDFS(PSO-MI) | | | | | | | | | | Maximum accuracy |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | Rank 1 | Rank 2 | Rank 3 | Rank 4 | Rank 5 | Rank 6 | Rank 7 | Rank 8 | Rank 9 | Rank 10 | |
| Fetal health | histogram _width | mean _value _of _short _term _variability | percentage _of _time _with _abnormal _long _term _variability | abnormal _short _term _variability | histogram _variance | histogram _mean | histogram _min | - | - | - | 0.84 |
| Kidney disease | hemo | pcv | sc | rc | bgr | sg | al | bu | wc | sod | 0.98 |
| ECG | ecgpatt | ldl | weight | - | - | - | - | - | - | - | 0.96 |
| Unbalanced | WBN_EN _H_1.00 | WBN_EN _L_1.00 | WBN_GC _H_0.75 | WBN_GC _H_1.00 | WBN_EN _H_0.75 | WBN_LP _H_0.25 | WBN_LP _H_0.50 | WBN_LP _H_0.75 | WBN_LP _L_1.00 | XLogP | 0.98 |
| MOFP | PS14 | NEUT | LYMPH | CD62L | CD88 | CD16 | CD11B | CD11B + FMLF | CD62L | CD88 | 0.86 |
| Wafer manufacturing | feature_3 | feature_2 | feature_1 | - | - | - | - | - | - | - | 0.92 |
| Breast cancer | pp_IRS1 | pp_Jak2 | pp_Rad50 | pp_Rad51 | pp_Src | pp_YB.1 | pp_YB.1.pS102 | pp_NF2 | pp_PCNA | pp_PRDX1 | 0.88 |
| Emotions | moments _12_a | moments _12_b | moments _2_a | moments _2_b | covmat _0_a | moments _13_b | covmat _96_b | moments _10_a | moments _10_b | moments _13_a | 0.99 |
| Malaria | pixel_0 | pixel_1126 | pixel_1480 | pixel_787 | pixel_1222 | pixel_1561 | pixel_1642 | pixel_961 | pixel_1378 | pixel_1123 | 0.75 |
| Skin cancer | pixel1818 | pixel1822 | pixel1867 | pixel1868 | pixel1890 | pixel1907 | pixel1908 | pixel1840 | pixel1846 | pixel1860 | 0.58 |

### 5.5. Result analysis on big datasets

In our experimental analysis, we used five large datasets: wafer manufacturing, malaria, breast cancer, skin cancer, and emotion. The description of the datasets is given in Table 6. To better generalize the performance of the HDFS(PSO-MI) method, we statistically validated it using a *t*-test. We compared the statistical significance of the HDFS(PSO-MI) method with those of other competing FS methods by considering the *p*-values. If the *p*-value of a pair of FS methods using a classifier is greater than 0.05, there are no significant differences in the performances of the two FS methods. Otherwise, the methods used may have differed significantly (Tables 9–12). In addition to the *p*-value, we validated and analyzed the proposed HDFS(PSO-MI) using the accuracy and MCC scores on five high-dimensional datasets, as shown in Figs. S1–S13.

### 5.6. Statistical analysis of the proposed method using p-values

We applied a *t*-test to validate the statistical significance of the proposed HDFS(PSO-MI) method and compared it with other FS methods. From the computed *p*-values, we observed that the proposed method behaved similarly to the FS methods. As shown in Table 9, the *p*-value between the proposed method and EFS_Rank method is smaller than the threshold of 0.05. Hence, the *t*-test reflects the significant differences between the two classifiers evaluated using the k-NN classifier. Similarly, there was a significant difference between HDFS(PSO-MI) and ANOVA using the random forest classifier. As shown in Table 10, on the breast cancer dataset, the proposed HDFS(PSO-MI) shows results similar to those of other FS methods, as the *p*-values of all competing methods are greater than 0.05. Hence, there were no significant differences between the FS methods. On the emotion dataset, the proposed method yielded a *p*-value of 1 when compared with other FS methods, as shown in Table 11. Therefore, there was no significant difference between HDFS-MI and other competing FS methods. Finally, Table 12 shows all *p*-values greater than 0.05, which indicates that there were no significant differences between HDFS-MI and other FS methods for the skin cancer dataset.

### 5.7. Result analysis on big datasets in terms of accuracy

From the experimental results on HDFS(PSO-MI) on five real big datasets, we observed that on malaria, skin cancer, breast cancer, and emotion datasets, the proposed method gives very good results using all the classifiers as shown in Fig. S6-S12. However, the proposed method shows a slightly lower accuracy than other methods using DT, SVM, and k-NN classifiers, as shown in Fig. S4.

### 5.8. Result analysis

We used the accuracy and MCC measures to compare the HDFS(PSO-MI) method with six other FS methods. The performance metrics were computed by applying five machine learning classifiers, i.e., LR, k-NN, SVM, DT, and RF.

The HDFS(PSO-MI) method was compared with other FS methods, such as the ensemble feature selection with ranking method (EFS-Rank), Spearman correlation, ANOVA, Pearson correlation, mutual information, and gain ratio using five classifiers: LR, k-NN, SVM, DT, and RF. As shown in Fig. S14, the proposed method provides better accuracy on the fetal health dataset than the EFS_Rank, Spearman correlation, ANOVA, Pearson correlation, and mutual information using LR, SVM, DT, and RF. In the kidney disease dataset, our method outperformed the Spearman correlation, ANOVA, Pearson correlation, and mutual information in terms of accuracy using all five classifiers, as shown in Fig. S15. Similarly, as shown in Fig. S16, our method outperformed the Spearman correlation, ANOVA, Pearson correlation, and mutual information using k-NN, SVM, DT, and RF on the ECG dataset. On the hypothyroid dataset, as shown in Fig. S17, the HDFS(PSO-MI) method outperformed the

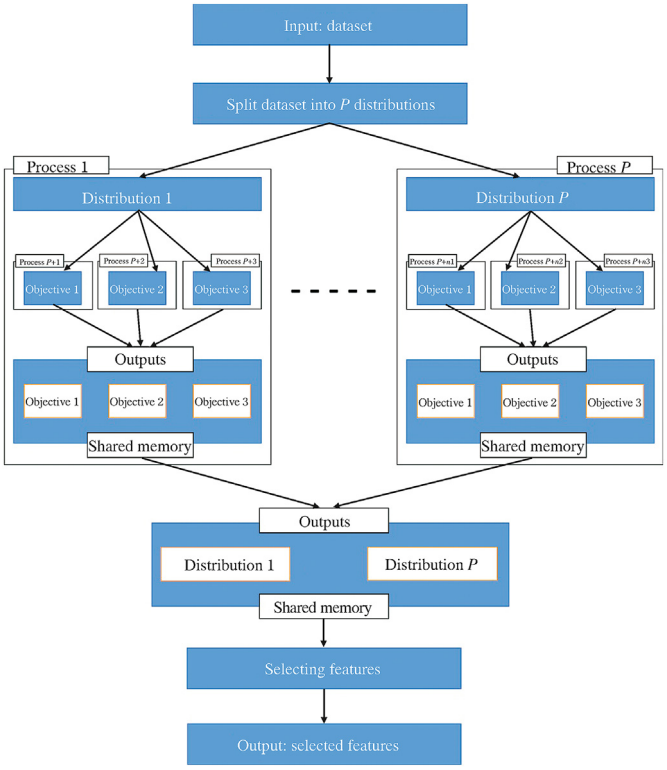


**Fig. 3.** Architecture of the parallel execution of the HDFS(PSO-MI).

EFS_Rank, Spearman correlation, ANOVA, Pearson correlation, and mutual information using LR and SVM, but the EFS_Rank, Spearman correlation, ANOVA, and Pearson correlation measures yielded better accuracy using k-NN, DT, and RF. Using the classifiers LR, k-NN, SVM, and RM, our method showed similar accuracy to all competing feature selection methods on the unbalanced dataset. However, as shown in Fig. S18, our method outperformed the EFS_Rank, Spearman correlation, Pearson correlation, and gain ratio using the DT classifier. On the MOFP and bioassay datasets, as shown in Figs. S19 and S20, the proposed method provided better accuracy than all the competing methods using LR, k-NN, SVM, and RF. However, the DT showed a lower accuracy for our method on both datasets. As shown in Fig. S21, using the LR, SVM, and RF classifiers, the proposed FS method outperformed competing methods. However, using k-NN and DT classifiers, our method yielded similar accuracy on the Parkinson's disease dataset. On the cancer gene dataset, our method performed very well compared with other methods using all classifiers, as shown in Fig. S22. However, on the colon cancer dataset, our method provides better accuracy using LR only; with other classifiers, it showed slightly lower accuracy, as shown in Fig. S23. The MCC performance measure plays a significant role in validating the FS method for unbalanced datasets. A high MCC score indicated good prediction results (Chicco and Jurman, 2020). MCC scores computed on all datasets using different classifiers for our proposed method were compared with other methods, as shown in Figs. S24, S25, S26, S27, S28, S29, S30, S31, S32

### 5.9. Discussion

We developed an efficient hybrid feature selection method, known as

**Table 11**
Comparision of HDFS(PSO-MI) with other methods using *p*-value on emotion dataset.

| Methods | Logistic regression (LR) | k-NN | Support vector machine (SVM) | Decision tree (DT) | Random forest (RF) |
|---|---|---|---|---|---|
| EFS_Rank | 1 | 1 | 1 | 1 | 1 |
| Spearman correlation | 1 | 1 | 1 | 1 | 1 |
| Analysis of variance (ANOVA) | 1 | 1 | 1 | 1 | 1 |
| Pearson correlation | 1 | 1 | 1 | 1 | 1 |
| Mutual information | 1 | 1 | 1 | 1 | 1 |
| Gain ratio | 1 | 1 | 1 | 1 | 1 |

**Table 9**
Comparision of HDFS(PSO-MI) with other methods using *p*-value on malaria dataset.

| Methods | Logistic regression (LR) | k-NN | Support vector machine (SVM) | Decision tree (DT) | Random forest (RF) |
|---|---|---|---|---|---|
| EFS_Rank | 0.72328 | **0.01987** | 0.95319 | 0.8473 | 1 |
| Spearman correlation | 0.18751 | 0.40029 | 0.72047 | 0.8484 | 0.37546 |
| Analysis of variance (ANOVA) | 0.18751 | 0.40029 | 0.72047 | 0.84639 | **0.03031** |
| Pearson correlation | 0.18751 | 0.40029 | 0.72047 | 1 | 0.45648 |
| Mutual information | 0.18751 | 0.40029 | 0.72047 | 0.87766 | 0.75119 |
| Gain ratio | 0.18751 | 0.40029 | 0.72047 | 0.64071 | 1 |

**Table 10**
Comparision of HDFS(PSO-MI) with other methods using *p*-value on breast cancer dataset.

| Methods | Logistic regression (LR) | k-NN | Support vector machine (SVM) | Decision tree (DT) | Random forest (RF) |
|---|---|---|---|---|---|
| EFS_Rank | 0.75391 | 0.07031 | 0.39153 | 0.87761 | 0.5 |
| Spearman correlation | 0.30176 | 0.21875 | 1 | 0.21533 | 0.5 |
| Analysis of variance (ANOVA) | 0.30176 | 0.21875 | 1 | 0.20049 | 1 |
| Pearson correlation | 0.30176 | 0.21875 | 1 | 0.31050 | 1 |
| Mutual information | 0.30176 | 0.21875 | 1 | 1 | 1 |
| Gain ratio | 0.30176 | 0.21875 | 1 | 1 | 0.5 |

**Table 12**
Comparision of HDFS(PSO-MI) with other methods using *p*-value on skin cancer dataset.

| Methods | Logistic regression (LR) | k-NN | Support vector machine (SVM) | Decision tree (DT) | Random forest (RF) |
|---|---|---|---|---|---|
| EFS_Rank | 1 | 0.38331 | 1 | 0.73588 | 0.32401 |
| Spearman correlation | 1 | 1 | 1 | 0.84502 | 0.67764 |
| Analysis of variance (ANOVA) | 1 | 1 | 1 | 0.82380 | 1 |
| Pearson correlation | 1 | 1 | 1 | 0.54126 | 0.22952 |
| Mutual information | 1 | 1 | 1 | 0.42436 | 1 |
| Gain ratio | 1 | 1 | 1 | 0.85055 | 0.05224 |

HDFS(PSO-MI) using PSO and MI. This method is highly effective in selecting an optimal subset of features that can yield high classification accuracy on different datasets. Although this method does not consider the class imbalance problem during feature selection, MCC values ensure that the selected features on various datasets provide a significantly high classification accuracy. The main advantage of the proposed hybrid method is that it can select optimal subset features from a high-dimensional dataset by evaluating possible subsets generated by PSO, where SVM is used as an evaluator of the subset. Hence, PSO ensures that the best subset of features can be selected. In the next phase, the method evaluates each feature of the selected subset to compute their ranks in terms of their MI scores. Thus, the hybrid method always selects the best features from the entire feature subset. The experimental results demonstrate that the proposed method significantly reduces the dimensions and redundancies of high-dimensional datasets. In addition, the feature subset selected by the proposed method yields better results than the existing filter-based FS methods, that is, mutual information, gain ratio, Spearman correlation, ANOVA, Pearson correlation, and the ensemble feature selection with ranking method using five ML classifiers: LR, k-NN, SVM, DT, and RF. Although the proposed objective function enhances the PSO algorithm, the computational cost is slightly higher than those of the two existing objective functions used in our HDFS(PSO-MI) method.

## 6. Conclusion and future work

This study introduces an effective hybrid FS method called HDFS(PSO-MI) using PSO and MI, which selects a subset of optimal features from a high-dimensional dataset. This method considers distributed data, and from each distribution, it selects three subsets of features using three objective functions in the PSO optimization techniques. The UNION of the three subsets is evaluated again for each distribution to select the high-ranked features as the most optimal. The computational cost of the hybrid method is high because the evaluation of the feature subset with PSO using three objective functions requires a significant amount of time. We applied three objective functions in PSO to evaluate a possible subset of features in each distribution to reduce the chance of bias in a single objective function. From the experimental results, we observed that the proposed HDFS(PSO-MI) selected an optimal feature subset that provided high classification accuracy on various datasets. Although the proposed method works well on most datasets taken from multiple application domains, including five large real datasets, the major drawback of the method is its exponential computational time. The computational time increases asymptotically as the dimension of the dataset increases and when using three objective functions on the PSO optimizer. Therefore, an effective mechanism is required for addressing this issue. In future work, we plan to implement the proposed method in a distributed parallel programming environment to reduce computational costs.

## CRediT author statement

**Khumukcham Robindro**: Methodology, Writing-Reviewing and Editing. **Sanasam Surjalata Devi**: Implementation, Writing-Original draft preparation. **Urikhimbam Boby Clinton**: Validation, Testing, Investigation. **Linthoingambi Takhellambam**: Testing and Debugging, Editing. **Yambem Ranjan Singh**: Testing, Validation, Editing. **Nazrul Hoque**: Conceptualization, Supervision.

## Declaration of competing interest

The authors declare that there are no conflicts of interest.

## Acknowledgments

The work is funded by the University Grant Commission (UGC) under (Start-up-Grant No.: F 30-592/2021(BSR)).

## Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.dsm.2023.10.003.

## References

Bansal, S.R., Wadhawan, S., Goel, R., 2022. mrmr-pso: a hybrid feature selection technique with a multiobjective approach for sign language recognition. Arabian J. Sci. Eng. 47 (8), 10365–10380.
Baruah, H.S., Thakur, J., Sarmah, S., et al., 2020. A feature selection method using pso-mi. In: 2020 International Conference on Computational Performance Evaluation (ComPE). IEEE, pp. 280–284.
Brahim, A.B., Limam, M., 2016. A hybrid feature selection method based on instance learning and cooperative subset search. Pattern Recogn. Lett. 69 (Jan.), 28–34.
Chandrashekar, G., Sahin, F., 2014. A survey on feature selection methods. Comput. Electr. Eng. 40 (1), 16–28.
Chicco, D., Jurman, G., 2020. The advantages of the matthews correlation coefficient (mcc) over f1 score and accuracy in binary classification evaluation. BMC Genom. 21 (1), 1–13.
Eberhart, R., Kennedy, J., 1995. Particle swarm optimization. In: International Conference on Neural Networks, 4, pp. 1942–1948.
Fong, S., Wong, R., Vasilakos, A., et al., 2015. Accelerated pso swarm search feature selection for data stream mining big data. IEEE Trans. Serv. Comput. 9 (1), 33–45.
Harb, H.M., Desuky, A.S., 2014. Feature selection on classification of medical datasets based on particle swarm optimization. Int. J. Comput. Appl. 104 (5), 14–17.
Hoque, N., Ahmed, H., Bhattacharyya, D., et al., 2016. A fuzzy mutual information-based feature selection method for classification. Fuzzy Inf. Eng 8 (3), 355–384.
Hoque, N., Bhattacharyya, D.K., Kalita, J.K., 2014. Mifs-nd: a mutual information-based feature selection method. Expert Syst. Appl. 41 (14), 6371–6385.
Hoque, N., Singh, M., Bhattacharyya, D.K., 2018. Efs-mi: an ensemble feature selection method for classification: an ensemble feature selection method. Complex Intell. Syst. 4 (Oct.), 105–118.
Kumar, V., Minz, S., 2014. Feature selection: a literature review. Smart Comput. Rev. 4 (3), 211–229.
Li, J., Cheng, K., Wang, S., et al., 2017. Feature selection: a data perspective. ACM Comput. Surv. 50 (6), 1–45.
Li, X., Zhang, J., Safara, F., 2023. Improving the accuracy of diabetes diagnosis applications through a hybrid feature selection algorithm. Neural Process. Lett. 55 (1), 153–169.
Moradi, P., Gholampour, M., 2016. A hybrid particle swarm optimization for feature subset selection by integrating a novel local search strategy. Appl. Soft Comput. 43 (Jun.), 117–130.
Moslehi, F., Haeri, A., 2020. An evolutionary computation-based approach for feature selection. J. Ambient Intell. Hum. Comput. 11 (9), 3757–3769.

Shi, Y., Eberhart, R.C., 1998. Parameter selection in particle swarm optimization. In: Evolutionary Programming VII. EP 1998. Springer, pp. 591–600.
Venkatesh, B., Anuradha, J., 2019. A review of feature selection and its methods. Cybern. Inf. Technol. 19 (1), 3–26.
Vieira, S.M., Mendonça, L.F., Farinha, G.J., et al., 2013. Modified binary pso for feature selection using svm applied to mortality prediction of septic patients. Appl. Soft Comput. 13 (8), 3494–3504.
Wang, C.M., Huang, Y.F., 2009. Evolutionary-based feature selection approaches with new criteria for data mining: a case study of credit approval data. Expert Syst. Appl. 36 (3), 5900–5908.
Zhang, Y., Gong, D., Zhang, W., 2016. Feature selection of unreliable data using an improved multi-objective PSO algorithm. Neurocomputing, 171 (Jan.), 1281–1290.
Zhu, Y., Li, W., Li, T., 2023. A hybrid artificial immune optimization for high-dimensional feature selection. Knowl-based Syst. 260 (Jan.), 110111.