Frequency-based Multi-objective Feature Selection to Enhance the Generalization of Evolutionary Algorithms

Abstract— **Feature selection is a critical preprocessing task in machine learning, particularly with high-dimensional datasets and decision-making while handling big data which presents significant challenges. This paper introduces an innovative approach for multi-objective feature selection, aiming to minimize the number of features and classification error simultaneously. The method mitigates the generalization issues commonly faced when relying on the results of a single run of evolutionary algorithms. Our approach leverages the frequency of each feature across multiple runs of the optimization algorithm, applied to different portions of the data, as a key metric for ranking the features. This can reduce the risk of overfitting and enhances generalization by capturing more reliable features through repeated runs and different data subsets. To enhance the robustness of the selection, we incorporate the correlation between features and labels to determine the final feature set. To evaluate the proposed method, we selected fourteen datasets with varying numbers of features and instances. Experimental results demonstrate that this post-optimization processing technique significantly enhances generalization and consistently delivers superior performance across various datasets compared to the raw optimization results.**

I. INTRODUCTION

In the era of big data, the rapid increase in the volume and complexity of information has led to high-dimensional datasets, posing significant challenges for machine learning applications. These datasets, often characterized by numerous features, frequently contain redundant, irrelevant, or noisy data, which can degrade model performance. As dimensionality increases, data becomes sparser in decision space, exacerbating challenges in effective classification, commonly referred to as "the curse of dimensionality" [1].

Feature selection has emerged as a critical preprocessing step to address these issues. It is a combinatorial optimization problem that attempts to find the best subset of $2^n - 1$ possible feature subsets from a given dataset with n features [2]. It aims to identify and retain the most relevant features, which can enhance model accuracy, reduce computational complexity, and improve interpretability. This process is particularly important in real-world datasets that can contain thousands or even millions of features [3], making the extraction of meaningful insights essential for informed decision-making [4].

Feature selection is inherently a multi-objective problem, requiring a balance between maximizing classification accuracy and minimizing the number of features [5]. By reducing the feature set, it not only alleviates computational burden but also helps prevent model overfitting [6]. However, traditional feature selection methods, which involve evaluating vast numbers of feature subsets, are computationally expensive and suffer from the NP-hard nature of the problem [7]. Additionally, complex interactions between features, such as redundancy, further complicate the selection process [8].The reason is that there exists a huge search space and complex feature interactions, especially for high-dimensional datasets [9]. Evolutionary Multi-objective optimization (MOO) techniques offer a solution to these challenges by simultaneously exploring multiple trade-offs, allowing the discovery of Pareto-optimal solutions that balance feature count and classification accuracy [10]. MOO is particularly valuable in feature selection as it avoids convergence to a single suboptimal solution and facilitates exploration of diverse, high-quality solutions.

Evolutionary algorithms are well-suited for addressing the multi-objective nature of feature selection. These algorithms leverage principles of natural selection and genetic evolution to efficiently search through the high-dimensional space, identifying high-quality feature subsets. However, evolutionary algorithms face inherent challenges related to stochasticity, often resulting in variability across different runs, which can limit their generalization ability and lead to inconsistent results [11]. This is a common challenge in data-driven evolutionary computation [12], [13], where the optimization process is performed on the training set, but the results on the test set can differ significantly. Relying on a single run of the algorithm to select an optimal set of features based solely on the training set can lead to overfitting, resulting in poor performance on the test set and diminishing the generalization capability of evolutionary algorithms.

To address this, we propose a robust feature selection method that aggregates results from multiple runs of evolutionary algorithms using a voting mechanism. By selecting the most frequently occurring features across runs and further refining the selection based on validation accuracy and feature-label correlation, our method improves generalization and stabilizes feature selection outcomes. This approach mitigates the stochastic variability of evolutionary algorithms and provides a more reliable set of features that enhances model performance across diverse datasets. The integration of a voting scheme significantly boosts generalizability and robustness, overcoming the limitations of relying on a single run. Previous studies have demonstrated the effectiveness of voting mechanisms in stabilizing feature selection, and our results confirm its potential in achieving consistent and improved performance across a variety of benchmarks.

The organization of this paper is as follows: Section II provides a detailed background review, Section III outlines our proposed method, illuminating the steps and techniques employed, Section IV presents results and analysis from our study, and Section V concludes with remarks.

II. Background Review

A. Multi-objective Feature Selection

Multi-objective optimization involves addressing problems with multiple, often conflicting objectives. In short, the best results cannot be achieved for all objectives simultaneously, which defines a class of problems known as multi-objective optimization [14]. In feature selection, this typically involves balancing the improvement of predictive accuracy with the reduction in the number of selected features, leading to better model performance [15]. For instance, let $f_1(X)$ represent the classification error and $f_2(X)$ denote the number of features in a subset X. The objective is to identify a subset X^* that minimizes both f_1 and f_2. Mathematically, this can be formulated as:

$$X^* = \arg\min\{f_1(X), f_2(X)\}$$

Since f_1 and f_2 are often conflicting, finding the optimal solution requires a compromise between the two. In multi-objective problems, the comparison of candidate solutions is commonly based on the principle of dominance. In a minimization problem, given two vectors $\boldsymbol{x} = (x_1, x_2, \ldots, x_d)$ and $\boldsymbol{\acute{x}} = (\acute{x}_1, \acute{x}_2, \ldots, \acute{x}_d)$ in the search space, $\boldsymbol{x}$ is said to dominate $\boldsymbol{\acute{x}}$ ($\boldsymbol{x} \prec \boldsymbol{\acute{x}}$) if and only if:

$$\begin{aligned} &\forall i \in \{1, 2, \ldots, M\}, \quad f_i(\boldsymbol{x}) \leq f_i(\boldsymbol{\acute{x}}) \quad \text{and} \\ &\exists j \in \{1, 2, \ldots, M\} \text{ such that } f_j(\boldsymbol{x}) < f_j(\boldsymbol{\acute{x}}) \end{aligned} \tag{1}$$

This definition captures the relative superiority of a solution $\boldsymbol{x}$ over another solution $\boldsymbol{\acute{x}}$. A solution is considered non-dominated if it is not worse than any other solution in all objectives and is better in at least one. The set of all non-dominated solutions forms the Pareto front, representing optimal trade-offs across the objectives [16].

NSGA-II is one of the most widely used evolutionary algorithms for solving multi-objective optimization problems, including feature selection. It utilizes the dominance principle, where one solution dominates another if it performs better in at least one objective without being worse in any others. NSGA-II employs Pareto-based and diversity-based selection criteria, evaluating solutions based on their dominance relationships[17]. Non-dominated Sorting (NDS) ranks solutions into multiple Pareto fronts. The best solutions form the first Pareto front, while subsequent fronts consist of non-dominated solutions relative to the remaining population.

In the context of feature selection, NSGA-II treats the problem as a high-dimensional binary optimization task, where each individual is represented as a binary vector. A value of 1 indicates the selection of a feature, while 0 indicates its exclusion [16]. To evaluate each solution's performance, features marked with a 1 are used to form a reduced dataset, and the classification error is then computed.

The feature selection process minimizes two primary objectives:

- **Minimizing the Number of Features**: This objective counts the total number of selected features in each solution, aiming to select fewer features for more efficient models.
- **Minimizing the Classification Error**: This objective calculates the classification error using a specified classification model, based on the selected feature subset. The error is calculated as 1 minus the model's accuracy on the training dataset.

By optimizing these two objectives simultaneously, optimizer effectively balances feature selection and predictive accuracy[18], leading to more robust models with reduced dimensionality.

B. Maximal Information Coefficient

Mutual Information (MI) [19] is a widely used technique to measure the relationship between two variables, such as X and Y. It is mathematically defined by the following expression:

$$\text{MI}(X;Y) = \sum_{x \in X} \sum_{y \in Y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \tag{2}$$

Here, $p(x)$, $p(y)$, and $p(x,y)$ denote the marginal probabilities of x and y, as well as their joint probability. When applied to two features, $\text{MI}(X;Y)$ captures the amount of shared information between them. In cases where X is a feature and Y represents class labels, $\text{MI}(X;Y)$ reflects the level of dependence between the feature X and the class labels Y. To calculate the correlation of feature-to-feature and feature-to-label, we introduce MI.

MIC, or Maximal Information Coefficient, is a recently developed measure that applies MI to continuous variables using a binning approach. By dividing the variable space into discrete grids and normalizing MI, MIC provides a more precise measure of the relationships between continuous variables.

The computation of MIC [20] is summarized as follows: The original variable space G, defined by the values of X and Y, is divided into various $p \times q$ grids. The characteristic matrix $\text{M}(G)_{p,q}$ captures the maximum normalized MI within the $p \times q$ partitions, as shown in the equation:

$$\text{M(G)}_{p,q} = \frac{\max(\text{MI})}{\log \min(p,q)} \tag{3}$$

Here, max(MI) is the highest MI value across all possible $p \times q$ grids. MIC is then defined as:

$$\text{MIC} = \max_{0<p\times q<B(n)}\{\text{M(G)}_{p,q}\}, \quad B(n) = n^{0.6} \tag{4}$$

where n is the number of data instances, and $B(n)$ serves as a constraint on the grid size $p \times q$. For a more detailed explanation and parameter sensitivity analysis, refer to the referenced studies.

MIC values approach 1 for highly correlated variables and near 0 for independent variables. Thus, a higher MIC indicates a stronger relationship between the two variables.

III. PROPOSED METHOD

In this paper, we demonstrate that aggregating the outputs of multiple runs of an evolutionary process on different set of data can significantly improve performance compared to a single run. We explore this idea in the context of multi-objective feature selection, where the goals are to simultaneously minimize the number of features and classification error. Since feature selection is modeled as a binary optimization problem, applying a voting mechanism on the final binary vectors from the Pareto front leads to a more robust set of features. Additionally, a correlation metric between the selected features and class labels is used for further refinement—adding highly correlated features and removing those with low correlation. The following sections outline the details of each step in the process.

A. Initialization

The initialization of the algorithm is based on the number of decision variables, which corresponds to the total number of features in the dataset. For each individual, we generate a value m between 1 and n, where n represents the total number of features. Subsequently, m random cells in the individual's binary vector are set to 1, indicating the selected features, while the rest are set to 0.

The algorithm includes two objective functions and one constraint. The constraint ensures that at least one feature is selected in each solution, preventing the algorithm from evaluating solutions with no selected features, which would be meaningless for the classification task.

The evaluation function processes the population by calculating the values for both objective functions and checking the constraint for each solution. Specifically, it computes the number of selected features, the classification error, and ensures that the constraint is met. This evaluation method is applied to assess the fitness of each solution in the population. The fitness assessment involves measuring the number of selected features, the classification error, and applying the constraint to guarantee valid solutions.

The algorithm iterates through multiple generations, continuously evaluating and evolving the population to converge toward optimal solutions.

B. Evolutionary Process

In this study, we employ an evolutionary algorithm, specifically NSGA-II, to identify optimal feature subsets through multiple iterations using K-fold cross-validation. The dataset is first divided into separate training and test sets, with the test set reserved for final evaluation. The training set is further partitioned into K folds for cross-validation, ensuring that different portions of the data are used in each run to enhance robustness and generalization.

For each fold, NSGA-II is executed as follows: the algorithm begins by initializing a population of candidate solutions, where each individual represents a potential feature subset. These solutions are then evaluated based on two objectives: minimizing the number of selected features and minimizing classification error on the training data. NSGA-II employs non-dominated sorting to organize solutions into Pareto fronts, identifying non-dominated solutions that balance both objectives. The selection process also utilizes crowding distance to maintain diversity within the population, preventing premature convergence to suboptimal solutions. Genetic operators, such as crossover and mutation, are then applied to generate new offspring, and the process is iterated for a set number of generations.

Once the PF is generated from the training fold, the solutions on the PF are evaluated on the corresponding validation fold. This step allows us to calculate the validation accuracy for each solution, ensuring that the selected features generalize well beyond the training data. The evolutionary process is repeated n times, once for each fold in the cross-validation setup. In each run, NSGA-II produces a training PF, which is subsequently evaluated on the validation fold, yielding multiple Pareto fronts with corresponding validation accuracy scores.

This iterative procedure ensures that the algorithm explores various feature subsets across different data splits, reducing the likelihood of overfitting. In the subsequent section, we introduce a voting mechanism that aggregates the results from all iterations. By analyzing the frequency of selected features across multiple runs and considering their validation performance, we identify the most reliable features for the final feature set, striking a balance between accuracy and dimensionality reduction.

C. Voting-Correlation Mechanism

To further refine the selection of non-dominated feature subsets obtained from the validation set, we implement a voting mechanism. This method helps identify a robust set of features that consistently perform well across different data splits, enhancing the generalizability and stability of the feature selection process. By aggregating results from multiple runs of the optimization process, we ensure that frequently selected features are those that consistently contribute to minimizing classification error across varying portions of the dataset. To refine the selected feature subsets obtained from multiple runs of the evolutionary process, we use a voting mechanism based on the frequency of each feature's selection. For each run, a set of non-dominated binary vectors representing feature subsets is generated. We tally the frequency of each feature's occurrence across all runs, and the features that appear most frequently are selected for further analysis.

From the m most frequently selected features, we incrementally build feature subsets. We start by selecting the two most frequent features, then in each subsequent iteration, we add the next most frequent feature to the subset. For example, the first subset contains the top 2 features, the second subset includes the top 3 features, and so on, until all m features are included in the final subset. Each of these subsets is evaluated on the validation set, and the subset that achieves the highest validation accuracy is selected for further analysis.

To further enhance the robustness of the selected feature subset, a correlation analysis is conducted. We calculate the correlation between the selected features and the target labels. Features with a mutual correlation below a predefined threshold (e.g., 0.1) are eliminated, as they are likely to introduce redundancy. Conversely, features with correlations greater than a higher threshold (e.g., 0.2) are added back to the feature set if they contribute meaningful information. This additional step ensures that the final subset contains features that are not only frequently selected but also have strong predictive relationships with the target variable.

The final set of features is then evaluated on the previously unseen test data to assess the model's generalization capability. This evaluation typically results in accuracy that is comparable to, or better than, the average performance across multiple rounds of the evolutionary process. By combining voting with correlation analysis, we achieve a robust and reliable set of features that effectively balance accuracy and dimensionality reduction. The voting mechanism, combined with correlation refinement, results in a robust feature set that surpasses the benchmark in terms of generalization and performance, demonstrating the effectiveness of the proposed method.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

A. Datasets

This study utilized a diverse set of biological datasets to evaluate the feature selection process. The datasets vary in sample size, number of features, and classification tasks, providing a comprehensive assessment of the proposed method's performance. The datasets primarily represent biological data, with sample sizes ranging from 50 to 174 and feature counts ranging from 2,000 to 24,481. This broad range in both instances and features offers a solid foundation for evaluating the feature selection method across different biological contexts. The classification tasks include both binary and multi-class problems.

By covering a wide range of sample sizes, feature counts, and classification types, these datasets provide a robust framework for assessing the effectiveness and generalizability of the proposed feature selection method in various biological scenarios. The properties of the employed datasets are outlined in Table I.

B. Experimental Settings

We conducted multiple runs of the algorithm as detailed in Algorithm 1, repeating the process 10 times with different seeds to mitigate the impact of stochasticity. Recognizing the computational expense associated with feature selection on large-scale datasets, we maintained a fixed number of function calls at 15,000 for the mentioned algorithms, ensuring a fair comparison. The parameter n of the proposed method is set to 10. Both the population size and number of iterations of the evolutionary algorithm (i.e., NSGA-II) are set to 100. SPX and bit-flip are utilized as the crossover and mutation, respectively. The process involves splitting the dataset into training and test sets using a 70-30 split ratio, with stratified sampling to maintain class distribution. The training set is further divided into training and validation sets using 10-fold

Algorithm 1: Evolutionary Feature Selection with Voting and Correlation Refinement

```
Input  : Dataset, $K$: number of folds, $m$: number of
         top frequent features
Output: Final feature subset, test accuracy

Step 1: Evolutionary Process ;
for each fold $k$ in $K$ do
    Initialize population of feature subsets;
    Run the evolutionary process and return the
      Pareto front on tainting set;
    Evaluate Pareto front solutions on validation set;
    Store non-dominated solutions;
end

Step 2: Voting Mechanism ;
Initialize feature_frequency array;
for each solution in Pareto fronts across all folds do
    for each selected feature in solution do
        Increment corresponding feature_frequency;
    end
end

Step 3: Subset Selection with Correlation
  Refinement ;
for each $i$ from 2 to $m$ do
    Select top $i$ most frequent features;
    Evaluate subset on validation set;
    Select subset with the best validation accuracy;
    for each feature in the selected subset do
        Calculate correlation with the target label;
        if correlation $<$ threshold_low then
            Remove feature;
        end
        else if correlation $>$ threshold_high then
            Add feature;
        end
    end
end

Step 4: Final Evaluation ;
Evaluate the refined subset on test set;
Return accuracy and performance metrics;
```

TABLE I: Dataset Descriptions

Number	Dataset	#Samples	# Features	# Classes
1	Colon	62	2,000	2
2	Prostate	102	5,966	2
3	ALLAML	72	7,129	2
4	Lymphoma	96	4,026	9
5	Leukemia	72	7,070	2
6	GLI	85	22,283	2
7	Lung	203	3,312	5
8	Glioma	50	4,434	4
9	CLL_SUB	111	11,340	3
10	11_Tumor	174	12,533	11
11	SRBCT	83	2,308	4
12	CARCINOM	174	9,182	2
13	Breast	97	24,481	2
14	CNS	60	7,128	2

cross-validation (70-30 split), ensuring that various portions of data are used for training and validation to promote robustness and generalizability. NSGA-II optimization is applied to the training set, and solutions from the final Pareto front are evaluated on the validation set for feature selection. The frequency of feature selection is tracked across multiple runs (i.e., 10), and the top m (i.e., 50) frequent features are identified for further analysis. To refine the feature subset, features with low mutual correlation (< 0.1) are eliminated, and features with high correlation (> 0.2) are added. The final robust subset is then evaluated on the test set, with performance compared to the benchmark solution based on minimum validation error. Additionally, to compute the classification error, we employ the kNN classifier with parameter k set to 5 for all datasets.

The proposed method is compared with three additional approaches to evaluate the feature selection performance. The methods are as follows:

Best Validation Set (Best Val): The best-performing solution from the validation set (based on the Pareto front) is selected, and its performance is reported on the test set. This serves as a baseline for comparison.

Best Frequent Features (Best Freq): The most frequently selected features, in a number equal to the size of the best solution from the validation Pareto front, are used to form a feature subset. This subset is applied to the test set, and its performance is compared with the other methods.

Incremental Selection from Top m Frequent Features (Incr Freq): Starting with the top 2 most frequent features, feature subsets are incrementally built from the top m frequent features, and the performance of each subset is evaluated on the validation set. The best-performing subset is then applied to the test set, and the results are reported.

C. Numerical Results and Analysis

In this section, we analyze the classification error rates across four feature selection methods. Before analyzing the numerical results, Figure 1 shows some samples from the Pareto front generated by the evolutionary process. As can be seen, the evolutionary process was able to identify very small sets of features with high accuracy on the datasets. However, these subsets may degrade in performance on the test set. Aggregating results over multiple runs can help improve the test performance. The numerical results on the classification errors are presented in Table II. The results reflect the effectiveness of each method in minimizing classification error across a range of biological datasets. The most notable finding is the superior performance of the Proposed Method, which consistently achieves the lowest average classification error of 0.185, outperforming all other methods.

The error rates for Best Val and Best Freq are fairly close, with average errors of 0.254 and 0.271, respectively. These methods exhibit similar behavior in many datasets, likely due to their reliance on selecting solutions from the validation set without substantial refinement. Although these methods can find effective feature subsets, their relatively high error rates, especially in complex datasets like CLL_SUB and Breast, indicate they may struggle to generalize as effectively as desired, particularly when dealing with high-dimensional data.

Inc Freq, which incrementally selects features based on frequency, performs slightly better than Best Val and Best Freq, with an average error of 0.249. However, while it reduces dimensionality in some datasets and achieves moderate improvements, it still fails to consistently match the performance of the Proposed Method. For instance, in datasets like CNS and Glioma, Inc Freq produces higher error rates compared to the Proposed Method, indicating its limitations in complex feature spaces.

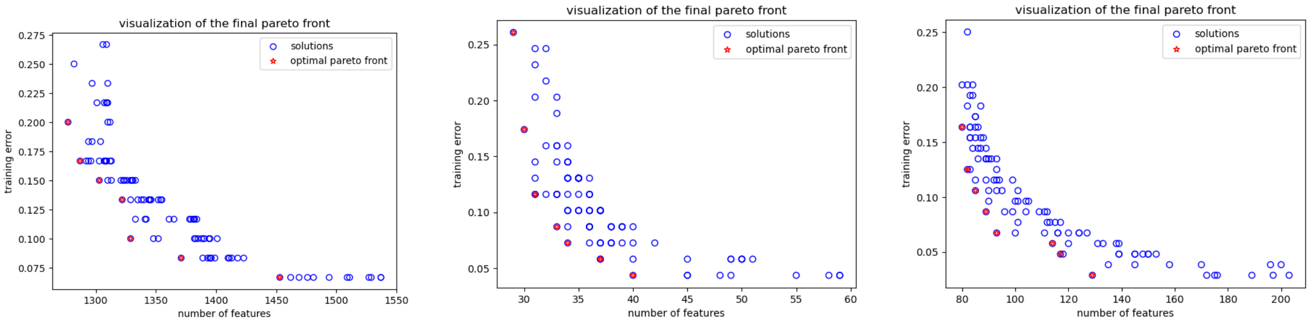
The Proposed Method, on the other hand, stands out by achieving the lowest classification errors across a majority of datasets. For example, in the Prostate dataset, it reduces the error to 0.039, significantly lower than the other methods. This indicates that the Proposed Method not only effectively reduces the number of features but also selects more informative features, leading to improved classification performance. The method's robust performance is particularly evident in datasets with complex feature interactions, such as ALLAML and Lymphoma, where it achieves error rates of 0.009 and 0.093, respectively, far lower than the other methods.

The effectiveness of the Proposed Method in minimizing classification error can be attributed to its combined approach of selecting the most frequent features and refining them using a correlation-based mechanism. This process ensures that only the most relevant and non-redundant features are retained, leading to superior generalization across datasets. In datasets such as Breast and CNS, where dimensionality is high, the method maintains competitive error rates despite the inherent complexity, highlighting its scalability and adaptability.

In conclusion, the Proposed Method consistently outperforms Best Val, Best Freq, and Inc Freq in terms of classification error, achieving the lowest average error across a variety of datasets. Its ability to select a minimal yet informative subset of features results in more accurate models, especially in high-dimensional and complex datasets. This makes the Proposed Method the most reliable and effective approach for feature selection and classification accuracy in biological datasets. To ensure the robustness and statistical soundness of our Proposed Method, we have performed the Wilcoxon statistical test to validate the significance of our results.

Table III shows the number of features for each set resulted from each metho. The methods display varying capabilities in reducing the dimensionality of the datasets. Both Best Val and Best Freq perform identically, selecting an average of 137.314 features across all datasets. These methods tend to retain a large number of features, especially in high-dimensional datasets like Breast and GLI, where they select 1331 and 107 features, respectively. While these methods can effectively capture relevant features, they often result in models that are more complex, computationally expensive, and harder to interpret due to the large feature sets.

The Inc Freq method improves on this by incrementally selecting features based on their frequency of occurrence,



(a) Breast dataset

(b) Colon dataset

(c) 11Tumor dataset

Fig. 1: Pareto front on some sample datasets

leading to a reduction in the number of features selected. On average, it selects 39.621 features, significantly fewer than Best Val and Best Freq. In simpler datasets like Colon and Prostate, Inc Freq selects fewer features (19 and 15.4, respectively), providing a more compact solution. However, it struggles in more complex datasets like Glioma and Lung, where it selects 69 and 45.6 features, respectively, showing that it does not always offer significant dimensionality reduction in challenging cases.

The standout method in this comparison is the Proposed Method, which consistently selects the fewest features across all datasets, with an average of only 25.757 features. This demonstrates the method's superior ability to identify the most relevant features while discarding redundant or non-informative ones, even in high-dimensional datasets. For example, in the Breast dataset, which has 24,481 features, the Proposed Method reduces the selected features to just 1.5, compared to 1331 features chosen by Best Val and Best Freq. Similarly, in the GLI dataset, which has 22,283 features, the Proposed Method selects only 3.33 features, in stark contrast to the 107 features selected by the other methods.

This substantial reduction in feature count, even in complex datasets, highlights the Proposed Method's strength. In simpler datasets such as Colon and Prostate, the method also performs exceptionally well, selecting only 8.3 and 4.3 features, respectively, compared to much higher numbers chosen by the other methods. In these cases, the Proposed Method ensures a more efficient and interpretable model without compromising performance.

V. Conclusion Remarks

This study introduced a feature selection method that effectively combines frequency-based voting with correlation refinement to tackle the complexity of high-dimensional datasets. The proposed method stands out by consistently selecting fewer features while maintaining or improving classification accuracy. By running a multi-objective evolutionary algorithm multiple times and tallying feature frequencies, the method identifies the most relevant features across various folds. The frequent selection of features across different optimization runs on varying data subsets ensures that the final chosen features generalize well. By focusing on

TABLE II: Comparison of methods in terms of the classification error. The methods include best validation set (Best Val), incremental selection from top 50 frequent features (Incr Freq), best frequent features (Best Freq), and proposed method.

	Best Val	**Best Freq**	**Incr Freq**	**Proposed Method**
Colon	0.267	0.326	0.258	**0.205**
Prostate	0.078	0.087	0.055	**0.039**
ALLAML	0.135	0.091	0.073	**0.009**
Lymphoma	0.300	0.252	0.148	**0.093**
Leukemia	0.264	0.373	0.291	**0.014**
GLI	0.238	0.231	0.292	**0.227**
Lung	0.102	0.105	0.090	**0.075**
Glioma	0.346	0.400	0.347	**0.180**
CLL_SUB	0.441	0.488	0.391	**0.362**
11_Tumor	**0.320**	0.374	0.367	0.359
SRBCT	0.115	0.104	0.115	**0.037**
CARCINOM	**0.236**	0.284	0.319	0.274
Breast	0.336	**0.313**	0.413	0.389
CNS	0.378	0.368	**0.321**	0.333
Avgerage	0.254	0.271	0.249	**0.185**

TABLE III: Comparison of methods in terms of the number of features. The methods include best validation set (Best Val), incremental selection from top 50 frequent features (Incr Freq), best frequent features (Best Freq), and proposed method.

	Best Val	**Best Freq**	**Incr Freq**	**Proposed Method**
Colon	**6**	6	19	8
Prostate	19	19	15	**4**
ALLAML	**4**	4	5	13
Lymphoma	**11**	11	43	73
Leukemia	21	21	57	**6**
GLI	107	107	34	**3**
Lung	**7**	7	21	46
Glioma	**20**	20	37	69
CLL_SUB	60	60	66	**10**
11_Tumor	81	81	82	**52**
SRBCT	**6**	6	6	24
CARCINOM	176	176	82	**49**
Breast	1331	1331	47	**2**
CNS	74	74	41	**3**
Average	137	137	39	**25**

features that consistently appear in high-performing solutions across multiple iterations, the method reduces the risk of overfitting to any specific training subset. This voting-based approach captures the most stable and reliable features, thereby enhancing the robustness of the final model.

The addition of correlation refinement further enhances the feature selection by removing redundant features with low correlation to the target, while adding highly correlated ones. This dual approach ensures that the final feature set is not only smaller but also highly informative, leading to models that are more accurate and less prone to overfitting. Compared to other methods, the proposed method achieves the smallest feature sets and the lowest average classification error across a wide range of datasets. This reduction in error and feature count highlights the method's ability to generalize better across diverse datasets, including those with complex feature spaces.

The dataset used in this study is publicly available from [source]. Additionally, the code for the algorithm is open-source and can be accessed at [repository link]. Both the dataset and code are freely available for replication and further research.

References

[1] Z. Wang, S. Gao, M. Zhou, S. Sato, J. Cheng, and J. Wang, "Information-theory-based nondominated sorting ant colony optimization for multiobjective feature selection in classification," *IEEE Transactions on Cybernetics*, vol. 53, no. 8, pp. 5276–5289, 2022.

[2] X.-h. Wang, Y. Zhang, X.-y. Sun, Y.-l. Wang, and C.-h. Du, "Multi-objective feature selection based on artificial bee colony: An acceleration approach with variable sample size," *Applied Soft Computing*, vol. 88, p. 106041, 2020.

[3] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," *Computers & electrical engineering*, vol. 40, no. 1, pp. 16–28, 2014.

[4] J. B. Jane and E. Ganesh, "A review on big data with machine learning and fuzzy logic for better decision making," *Int. J. Sci. Technol. Res*, vol. 8, no. 10, pp. 1221–1225, 2019.

[5] G. Kou, Y. Xu, Y. Peng, F. Shen, Y. Chen, K. Chang, and S. Kou, "Bankruptcy prediction for smes using transactional data and two-stage multiobjective feature selection," *Decision Support Systems*, vol. 140, p. 113429, 2021.

[6] B. Venkatesh and J. Anuradha, "A review of feature selection and its methods," *Cybernetics and information technologies*, vol. 19, no. 1, pp. 3–26, 2019.

[7] A.-D. Li, B. Xue, and M. Zhang, "Improved binary particle swarm optimization for feature selection with new initialization and search space reduction strategies," *Applied Soft Computing*, vol. 106, p. 107302, 2021.

[8] R. Jiao, B. Xue, and M. Zhang, "Solving multi-objective feature selection problems in classification via problem reformulation and duplication handling," *IEEE Transactions on Evolutionary Computation*, 2022.

[9] K. Chen, B. Xue, M. Zhang, and F. Zhou, "Correlation-guided updating strategy for feature selection in classification with surrogate-assisted particle swarm optimization," *IEEE Transactions on Evolutionary Computation*, vol. 26, no. 5, pp. 1015–1029, 2021.

[10] G. Goos, J. Hartmanis, and J. van Leeuwen, "Advanced research in computing and software science."

[11] H. Ishibuchi, N. Tsukamoto, Y. Hitotsuyanagi, and Y. Nojima, "Effectiveness of scalability improvement attempts on the performance of nsga-ii for many-objective problems," in *Proceedings of the 10th annual conference on Genetic and evolutionary computation*, 2008, pp. 649–656.

[12] Y. Jin, H. Wang, and C. Sun, *Data-driven evolutionary optimization.* Springer, 2021.

[13] Y. Jin, H. Wang, T. Chugh, D. Guo, and K. Miettinen, "Data-driven evolutionary optimization: An overview and case studies," *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 3, pp. 442–458, 2018.

[14] Z. Fan, W. Li, X. Cai, H. Li, C. Wei, Q. Zhang, K. Deb, and E. Goodman, "Push and pull search for solving constrained multi-objective optimization problems," *Swarm and evolutionary computation*, vol. 44, pp. 665–679, 2019.

[15] D. Liang, C.-F. Tsai, and H.-T. Wu, "The effect of feature selection on financial distress prediction," *Knowledge-Based Systems*, vol. 73, pp. 289–297, 2015.

[16] A. A. Bidgoli, H. Ebrahimpour-Komleh, and S. Rahnamayan, "Reference-point-based multi-objective optimization algorithm with opposition-based voting scheme for multi-label feature selection," *Information Sciences*, vol. 547, pp. 1–17, 2021.

[17] N. Gunantara, "A review of multi-objective optimization: Methods and its applications," *Cogent Engineering*, vol. 5, no. 1, p. 1502242, 2018.

[18] B. Xue, M. Zhang, and W. N. Browne, "Particle swarm optimization for feature selection in classification: A multi-objective approach," *IEEE transactions on cybernetics*, vol. 43, no. 6, pp. 1656–1671, 2012.

[19] P. Viola and W. M. Wells III, "Alignment by maximization of mutual information," *International journal of computer vision*, vol. 24, no. 2, pp. 137–154, 1997.

[20] A. Rafie, P. Moradi, and A. Ghaderzadeh, "A multi-objective online streaming multi-label feature selection using mutual information," *Expert Systems with Applications*, vol. 216, p. 119428, 2023.