VAWKUM Transactions on Computer Sciences
http://vfast.org/journals/index.php/VTCS@ 2025, ISSN(e): 2308-8168, ISSN(p): 2411-6335
Volume 13, Number 2, July-December 2025 pp:250-262

Optimization of Feature Selection using Firework Algorithm for Machine Learning Algorithm

Sakin Jan[1*], **Fazli Amin Khalil**[2], **Syed Bakhtawar Shah Abid**[3], **Izaz Ullah**[4], **Inayat Ullah**[1], **Inzimam ul Haq**[5]

[1]Government Abdul Ali Khan Degree College, Charsadda, Khyber Pakhtunkhwa, Pakistan; [2]University of Peshawar, Peshawar, Pakistan; [3]The Agriculture University Peshawar, Pakistan; [4]Islamia College University, Peshawar, Pakistan; [5]Govt. Postgraduate College Charsadda, Pakistan

Keywords: *Machine learning, Optimization, Feature selection, Firework algorithm*

Journal Info:
Submitted:
April 17, 2025
Accepted:
December 23, 2025
Published:
December 31, 2025

Abstract

In machine learning and deep learning, the optimal feature selection plays an important role in enhancing performance, decreasing computational cost, and improving the interpretability of the algorithm. The performance of the machine learning algorithm is impacted by the noisy, redundant, and meaningless features found in the majority of classification problem datasets. Feature selection is the process of selecting a feature subset and a search method for finding the optimal subset of features from many features using a fitness function to improve the accuracy and execution time. This research focuses on the application of the fireworks algorithm as a useful tool for optimizing feature selection. By navigating the feature space, the recommended technique finds the optimum subset of features that maximize a model's performance. By analyzing the fitness function, which combines the complexity of the model with the predictive ability of the chosen features, the program repeatedly improves the feature subset. The paper makes use of widely known datasets on breast cancer that include a limited number of characteristics. The classification performance of selected feature subsets is evaluated using classification techniques: Support vector machine, logistic regression, and bagging classifier. The proposed algorithms are better than the particle swarm optimization algorithm, ant colony optimization algorithm, principal component analysis, and so on. The results indicate that a certain feature subset may be chosen with higher accuracy through the use of the recommended techniques, as opposed to using all characteristics. An optimal subset of feature selection techniques enhances the accuracy and decreases the number of features. Enhancements in classification accuracy are supported by a substantial decrease in the number of features with a higher weight on the fast reduction of the fitness function.

***Correspondence author email address:** sakinjan100@gmail.com
DOI: 10.21015/vtcs.v13i2.2128

1 Introduction

Most classification problem datasets contain redundant, noisy, and useless features that affect the machine learning algorithm's performance [1][2][3]. As a result, feature selection is among the preceding steps needed to reduce the size of the data sets. Feature selection techniques are widely used for the selection of informative features and the removal of irrelevant features to reduce the execution cost and enhance the accuracy of predictions [4] and classifications. Optimization techniques [5] are crucial in efficiently navigating the feature space to identify the most informative subset [6]. In the wider context of machine learning, datasets generally contain multiple features, many of which may not contribute meaningfully to the prediction performance of models. By selecting the feature that is not important and useless might increase the computational cost of the models and even reduce the performance of the model to be fitted for the new data [7]. To overcome these issues the techniques for the selection of optimal feature, such as optimization algorithm implicitly identify and chose the features that are most relevant for a specific task [8]. A meta-heuristic optimization algorithm optimizes the optimal selection of features from the dataset as feature selection is an NP-hard problem [9]. Various technique are proposed by different researcher for the reduction of dimensionality of dataset or selection of an optimal feature subset. Among them, one feature selection technique is Principal Component Analysis (PCA) that is commonly used in statistics and machine learning [10] . The primary aim is to convert high-dimensional data into low dimensional, where the variance of the data along the new axes is maximum. By retaining the most important features of the data, PCA can help simplify complex datasets and facilitate various analyses. Some authors have used Particle swarm optimization [11], Artificial Bee Colony [12], Cuckoo Search [13], and Bat Algorithm [14] for feature selection optimization.

In this paper, a novel meta-heuristic algorithm firework algorithm [15] is proposed. The algorithm generates an optimal subset of features by evaluating an objective function. It selects the optimal number of features and the optimal subset from the data set which decreases the number of features and increases the accuracy of the algorithm [16].

The rest of the paper is structured: Section 2 contains a literature review. Section 3 provides the research technique is briefly explained, and in section, the results are evaluated. The paper's conclusion is found in section 5, finally.

2 Literature review

Feature selection is a critical problem in machine learning, as the inclusion of irrelevant or redundant features can significantly degrade algorithm performance. The selection of optimal feature is inherently challenging because it based on complex fitness functions, making it an NP-hard optimization problem. Over the years, several heuristic and meta-heuristic technique have been proposed to encounter this issues, including Principal Component Analysis (PCA), Artificial Bee Colony (ABC), Particle Swarm Optimization (PSO), Bat Algorithm (BA) and Cuckoo Search Algorithm (CSA). While these techniques perform better than conventional methods, they still have limitations such as premature convergence, sensitivity to local optima, and inadequate exploration-exploitation balance which encourage the development of more effective algorithms.

In [17] Adamu applied a hybrid wrapper-based technique by integrating PSO with a modified CSA, where the random component of CSA was replaced with a chaotic map to increase diversity. This Enhanced Chaotic Crow Search Particle Swarm Optimization (ECCSPSOA) also incorporated Opposition-Based Learning (OBL) to mitigate local optima trapping, demonstrating improved convergence speed and exploitation capabilities. Similarly, Kumar [18] proposed a three-step image analysis method—preprocessing, feature extraction using Gray-Level Co-occurrence Matrix (GLCM), and PCA-based selection for reducing 44 initial features to 12 highly appropriate features. While effective, such deterministic approaches may not generalize well due to limited exploration across all datasets.

Meta-heuristic algorithm have also shown performance in feature selection. In [19], the author used ABC algorithm for wrapper-based feature selection with SVM classification and 10-fold cross-validation, enhancing the system speed and classification accuracy by removing repeated features. Likewise, [20] the author employed the Bat Algorithm (BA) with different transfer functions to map solutions from continuous to discrete spaces, where the V-shaped transfer function (BA-V) give improved classification performance. These approach illustrate the capability of meta-heuristics but also highlight the need for algorithms capable of preserving the diversity while exploring large feature spaces efficiently.

Hybrid approach tried to balance search efficiency and prediction accuracy. For instance, [21] the author applied a two-layer wrapper-embedded approach combining a Genetic Algorithm (GA) for initial feature subset selection with Elastic Net (EN) for final refinement. This approach decrease computation cost while attaining near-optimal solutions, yet in highly complex feature spaces the dependency on GA can still lead to suboptimal exploration.

[22] applied a binary Horse Herd Optimization Algorithm integrate with MRMR filtering for high-dimensional feature selection, proving improved classification accuracy with reduced number of features than PSO and GA-based methods. [23] (2023) introduced a binary chaotic Horse Herd Optimization algorithm for feature selection, showing that chaos-improved metaheuristics can significantly improve solution diversity and classification performance.

Multi-objective and evolutionary algorithms have been employed to reduce the search space and enhance classification efficiency, but their computational complexity remains high for large number of feature [24] [25]. To increase interpretability, SHAP-based feature selection approach have been applied, offering transparent and better selection for tabular data, though they lack strong global optimization ability in complex feature spaces [26]. Bayesian optimization algorithm has also been employed for tuning feature selection pipelines, illustrating notable improvements in feature subset quality at the high computational overhead [27].

Despite these developments, a clear research gap is still present: existing meta-heuristic and hybrid approaches often try to maintain a robust trade-off between exploitation and exploration, leading to either suboptimal feature subsets or slow convergence.

[28] introduced FRAME, a hybrid forward recursive and RFE method that balances exploration and exploitation, proving robust performance on high-dimensional and noisy datasets compared to other methods. Multi-objective differential evolution has been used to feature selection, attaining significant improvement in predictive performance and solution diversity by combining feature weights and redundancy indices [29]. [30] applied metaheuristic-driven feature selection for human activity recognition using Golden Jackal and war approach optimization combined with SHAP interpretability, proving strong performance on sensor datasets.

To address this gap, the Fireworks Algorithm (FWA) [31] offers a new solution with better performance. By simulating the explosion process of fireworks to explore the solution space adaptively, FWA can identify optimal feature subsets efficiently while avoiding local optima and maintaining diversity. Our proposed approach leverages FWA for feature selection to reduce the number of features and enhance the accuracy and performance of machine learning models, offering a promising alternative to existing heuristic and hybrid methods.

3 Research Design and Methods

3.1 Fireworks Algorithm (FWA) Overview

The Fireworks Algorithm (FWA) is a swarm base Meta-heuristic optimization algorithm inspired by fireworks explosions. It balances exploration (searching new areas of the solution space) and exploitation (refining promising solutions) by generating sparks around current solutions (fireworks) and selecting the most relevant sparks for next iterations. FWA is especially suitable for feature selection, where the aim is to select an optimal subset of relevant features while reducing redundancy.

FWA operates as follows:

Initialization: A population of size N fireworks (candidate solutions) is initialized. Each firework represents a binary vector corresponding to selected features:

$$\mathbf{x} = [x_1, x_2, \ldots, x_d], \quad x_i \in \{0, 1\}, \quad i = 1, \ldots, d$$

Here, d is the total number of features. A threshold (e.g., 0.5–0.2) is applied to convert continuous vectors into binary selections.

Fitness Evaluation: Each firework's fitness is computed using a multi-objective function that balances classification accuracy and feature subset size:

$$F(x) = \alpha \cdot (1 - p) + (1 - \alpha) \frac{N_{\text{selected}}}{N_{\text{feature}}}$$

- p denotes the classification accuracy for the selected feature subset.

- N_{selected} = number of selected features
- N_{features} = total number of features
- $\alpha \in [0, 1]$ = weight factor controlling the trade-off between accuracy and subset size

Explosion and Spark Generation: Each firework generates sparks based on its fitness value. These sparks represent new candidate solutions in the local neighborhood of the firework.

Number of Sparks:

$$n_i = n_{\min} + \frac{f_{\max} - f_i}{\sum_{i=1}^{N}(f_{\max} - f_i)} \cdot (n_{\max} - n_{\min})$$

where n_i denotes the number of sparks generated by the i-th firework, f_i is the fitness of the i-th firework, $f_{\max}$ represents the worst fitness in the current population, and $n_{\min}$ and $n_{\max}$ are user-defined lower and upper bounds on the number of sparks, respectively.

Explosion Amplitude:

$$d_i = A_i \cdot \text{rand}(-1, 1)$$

where A_i is the explosion amplitude of the i-th firework and rand(–1, 1) generates a uniformly distributed random number in the range [–1, 1].

Gaussian Sparks: In addition to regular sparks, special Gaussian sparks are generated around the best-performing fireworks using a Gaussian distribution to enhance local exploration and exploitation.

Next Generation Selection: The next generation is formed by selecting the best fireworks along with top-performing sparks. A small number of randomly selected sparks are also retained to maintain population diversity.

Termination: The algorithm iterates for a predefined number of iterations or until a convergence criterion is met. The firework with the highest fitness value is returned as the optimal feature subset.

3.2 Methodology Flowchart

Figure 1 show the overall flow methodology of the Optimization of Feature Selection using Firework Algorithm for Machine Learning Algorithm approach. Initially, the dataset is preprocessed and an initial population of feature subsets is generated. The Fireworks Algorithm iteratively optimizes the feature subsets using a fitness function based on classifier performance. The final optimal feature subset is then used to train the machine learning model.

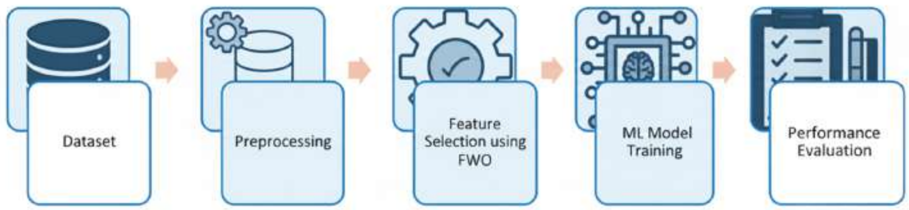

Figure 1. Methodology Flowchart

The specific flowchart of the Fireworks Algorithm with the initialisation, the creation of sparks and their mutation, fitness assessment, and the selection of the most suitable feature subsets over iterations are presented in Figure 2.

3.3 Pseudo-code of Fireworks Algorithm

A pseudo-code of the Fireworks Algorithm that is used in the selection of the features is shown in Algorithm 1. The algorithm starts with a population of binary fireworks, fitness, and continuously enhances the solutions by use of explosion and mutation processes although the population diversity is maintained.

Algorithm 1. Fireworks Algorithm for Feature Selection

Require: Dataset with d features, number of fireworks N, minimum and maximum sparks ($n_{\min}, n_{\max}$), maximum iterations T
Ensure: Optimal subset of features
1: Initialize N fireworks as binary vectors
2: Evaluate fitness $f(x)$ for each firework
3: **for** $t = 1$ to T **do**
4:     **for** each firework F_i **do**
5:         Generate S_i sparks around F_i using explosion amplitude A_i
6:         Apply Gaussian mutation to selected sparks for diversity
7:     **end for**
8:     Evaluate fitness of all generated sparks
9:     Select next generation from best fireworks and best sparks
10:     Remove duplicate solutions to maintain population diversity
11: **end for**
12: **return** Best firework representing the optimal feature subset

3.4 Feature Selection Problem Definition

Given a dataset with d features, the goal is to find a subset of features that maximizes classifier accuracy while minimizing the number of features. This multi-objective optimization problem is NP-hard because the search

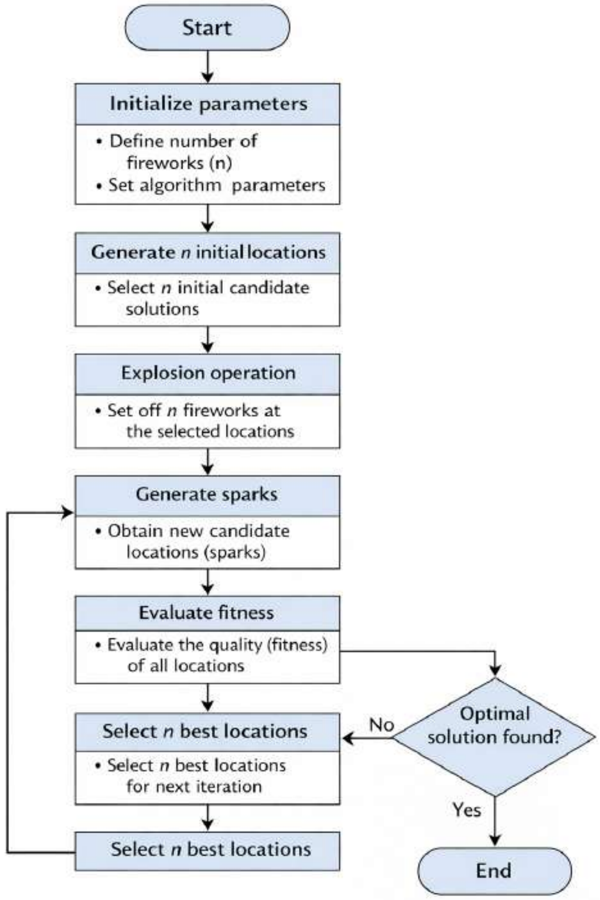

Figure 2. Flowchart of Algorithm

space grows exponentially with d. FWA provides an efficient approach to explore this space while maintaining solution diversity.

3.5 Dataset Description

- **Source:** Kaggle standard cancer dataset
- **Number of Features:** 35
- **Number of Instances:** 755
- **Usage:** The dataset is used to evaluate the feature selection performance of FWA and to compare it against baseline methods.

3.6 Data Preprocessing

- **Missing Values:** Imputed using mean/mode for numerical/categorical features
- **Feature Encoding:** Categorical features encoded using one-hot encoding
- **Normalization:** Numerical features scaled to [0,1] range.
- **Cross-validation:** 10-fold cross-validation applied to estimate classifier performance.

4 Result and discussion

The best performance of the Fireworks Algorithm is due to its adaptive explosion manners, which gradually balances exploration and exploitation. The PCA, performs linear transformation without considering classifier efficiency, FWA directly optimizes classification accuracy as a wrapper-based feature selection technique. The FWA as compared to PSO and ABC, maintains greater population diversity through controlled spark generation, decreasing premature convergence that lead to more compact and discriminative feature subsets.

A computer system system with an Intel Core i7 processor, 16 GB RAM, running Windows 10 (64-bit) is used for all experiments. The models were implemented in Python 3.x using Scikit-learn and XGBoost libraries.

4.1 Parameters setting and performance evaluation for optimization algorithms

In the given Table 1 are the Hyperparameters for the fireworks algorithm.

Table 1. Hyperparameters for Fireworks Algorithm

Parameter	Value	Description / Role
Maximum Iterations	100	Maximum number of iterations. Ensures convergence with low computational cost.
Population Magnitude	20	Number of fireworks (candidate solutions). Preserves diversity with reasonable efficiency.
Dimension	[0, 1]	Normalized search space threshold. Ensures numerical stability and uniform scaling.
Number of Runs	20	Independent runs to decrease randomness and ensure statistical reliability.
$W_{\max}$	0.9	Controls exploration ability at early iterations by allowing high jumps.
$W_{\min}$	0.4	Encourages exploitation and fine-tuning during later iterations.
$V_{\max}$	6	Maximum step size limit to avoid unstable movements.
C_1	2	Avoids local search behavior.
C_2	2	Reduces influence of global best solution.
f_l	2	Number of sparks per firework. Improves local exploration around best solutions.
AP	0.2	Normalizes spatial spread of sparks; lower values encourage local exploitation.
β	0.01	Limits randomness to reduce premature convergence and avoid local optima.
α	0.99	Gradually decreases explosion value over iterations, enabling smooth transition.

4.2 Faster Execution time

The execution time of a machine learning algorithms vary significantly. It depends on several variables, such as the space complexity, the size and dimensionality of the dataset, the available computation power, and the particular task . Faster execution time means models can be trained and deployed more quickly, improving overall efficiency in machine learning workflows. The selection of optimal subset of features enhances the accuracy of the machine learning algorithm in terms of execution time. In this approach, we have applied a firework optimization algorithm for optimal subsets from the dataset which reduces the training and testing time of the algorithm execution. In Table 2, machine learning algorithms are selected and tested on both all features and on the optimal subset of features that are selected by the firework optimization algorithm. It can be analyzed from Figure 3 that the execution time of the algorithm is greatly reduced. As the execution time AdaBoostClassifier on all features is from 0.148178101 to 0. 097572. similarly, the execution time of BaggingClassifieris reduced from 0.065692186 to 0.02789. Similarly, for all Models, the execution time is reduced to some extent.

Table 2. Performance analysis of various feature subsets via benchmark algorithm

S.NO	Model	Time Taken on all features	Time Taken on optimal subset
1	AdaBoostClassifier	0.148 178 101	0.097 572
2	BaggingClassifier	0.065 692 186	0.027 890
3	CalibratedClassifierCV	0.043 933 153	0.034 798
4	DecisionTreeClassifier	0.018 039 703	0.008 760
5	LinearSVC	0.021 111 012	0.017 615
6	LogisticRegression	0.029 359 102	0.014 251
7	SGDClassifier	0.024 862 289	0.011 940
8	SVC	0.023 377 657	0.017 337
9	XGBClassifier	0.148 443 937	0.092 367

Figure 3. Execution time

4.3 Number of features and accuracy-based comparison

Depending on a variety of factors, a machine learning model's performance may change when its feature number is increased or decreased. The relationship is not simple: depending on the situation and how it's handled, adding more features might either increase or decrease performance. The number of features can affect the effectiveness of a machine-learning algorithm. A more complete representation of the data can be achieved with more features, which enables the model to decide on deeper relationships and patterns that might be essential for producing predictions that are accurately. Models with more features, particularly those that can represent complex relationships, may be able to use the additional data to achieve greater accuracy. It can be generalized that by increasing the number of features, the accuracy of the machine learning model increases but the model becomes more complex. In this paper, the firework algorithm is used to select the optimal subset of features that have higher accuracy than other approaches like PCA, Bat algorithm, cuckoo search algorithm, etc. Table 3 gives the comparative analysis of the number of selected features and the obtained classification accuracy of various optimization techniques. In this experiment, we have taken only three machine learning algorithms, SVM, Logistic Regression, and Bagging classifier.

4.4 Performance of Logistic Regression with the Optimal Subset

Logistic regression is a machine learning algorithm that can be used to adapt the binary classification method. It simulates the likelihood that an input falls into a specific category. Because the logistic function is the foundation

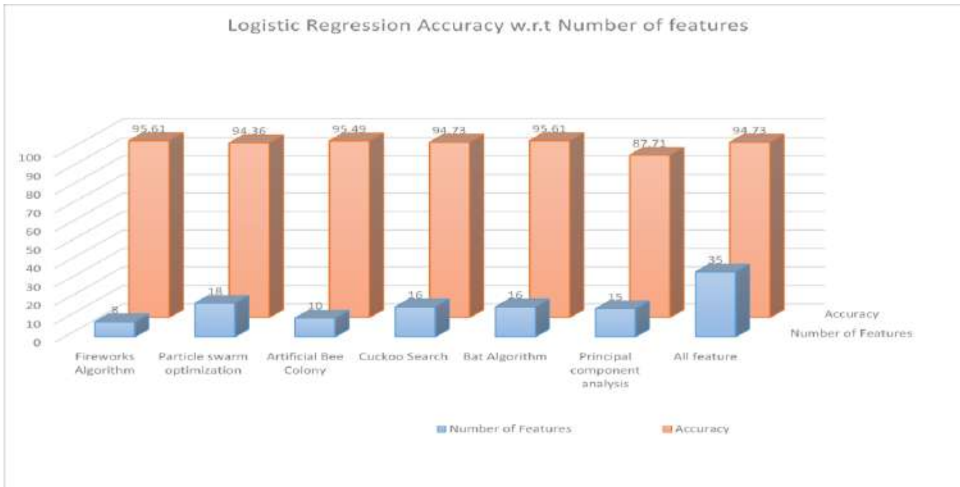

Figure 4. Logistic Regression

of this model, the output values are guaranteed to fall between 0 and 1, which allows them to be interpreted as probabilities. The logistic regression algorithm is applied on both the optimal subset returned by the firework algorithm and all number of features. From Figure 4, it can be analyzed that the performance of logistic regression in terms of the number of features and accuracy is higher than the other algorithm which returns a higher number of features and lower accuracy than the fireworks algorithm. The accuracy of logistic regression on fireworks is 95.1 with only 8 features, which is better than algorithms. PSO has an accuracy of 94.3 with 18 features, ABC has an accuracy of 95.36 with 10 features, PCA accuracy is 87.71 with 15 features, and so on. The default accuracy of logistic regression without dimensionality reduction is 94.73 with all 35 features.

Table 3. Accuracy Comparison of Feature Selection Methods across Different Machine Learning Algorithms

Optimization Algorithm	Logistic Regression		Bagging Classifier		SVM	
	Features	Accuracy (%)	Features	Accuracy (%)	Features	Accuracy (%)
Fireworks Algorithm	8	95.61	6	95.61	4	95.61
Particle Swarm Optimization [11, 17]	18	94.36	12	95.61	9	95.61
Artificial Bee Colony [12, 19]	10	95.49	12	93.86	12	94.73
Cuckoo Search [13]	16	94.73	8	94.74	9	94.73
Bat Algorithm [20]	16	95.61	10	95.61	7	92.98
Principal Component Analysis [10]	15	87.71	15	90.35	15	93.85
All Features	35	94.73	35	92.1053	35	91.22

4.5 Performance of Bagging Classifier with the optimal subset

A bagging classifier is a machine-learning algorithm that is well-known for its efficiency and simplicity. "Bagging" refers to the process of Bootstrap Aggregating. Using combinations with repetitions to create several sets of the original data, bagging reduces the variance of your prediction by creating new data for training from your original dataset. To generate a more accurate prediction, it aggregates the output of several models. In Figure 5 the bagging classifier is tested on the optimal set of features returned by various algorithms. It can be analyzed from the figure that the accuracy of the fireworks algorithm is better than all algorithms. The accuracy of the firework algorithm is 95.1 with only 6 features, the accuracy of PSO is 95.61 with 12 features, the accuracy of ABC is 93.86 with 12 features, and PCA accuracy is 90.35 with 15, but the accuracy of the Bagging classifier is 92.10 with all 35 features. So the firework returns the optimal subset of features that have higher accuracy than the other approaches.

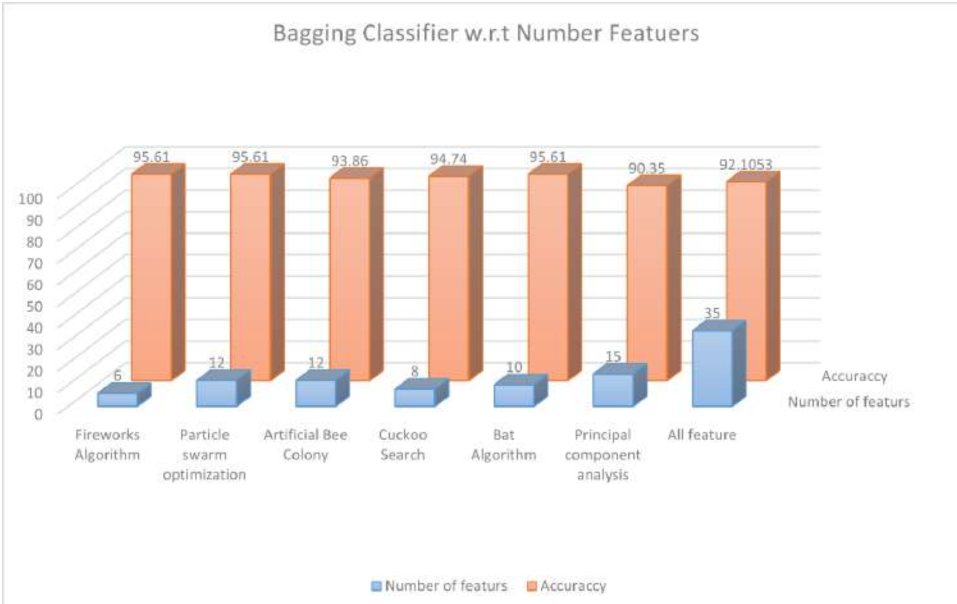

Figure 5. Bagging Classifier Accuracy

4.6 Performance of SVM with the optimal subset

The Support Vector Machine (SVM) is a reliable and flexible machine learning model that may be used for regression, outlier detection, and linear or nonlinear classification. Classifying complex but small- to medium-sized datasets is one area in which it performs. Figure 6 shows analyzed the performance of the optimal subset of the features returned by the firework algorithm is better than another algorithm. The figure shows that the accuracy of SVM with fireworks is 95.61 with 4 features, PSO accuracy is 95.1 with 9 features, PCA has an accuracy of 93.85 with 15 features, and all 35 features have 91.22 accuracy. It can be summarized that the fireworks algorithm returns the optimal subset of features, with which its performance in terms of the number of features and accuracy is better than another algorithm.

The observed improvement in accuracy with fewer features can be attributed to the reduction of irrelevant and redundant attributes, which minimizes overfitting and enhances generalization performance. By selecting only the most discriminative features, the Fireworks Algorithm reduces model complexity while preserving essential information, leading to improved classification accuracy. The Fireworks Algorithm achieved an accuracy of 95.61 while using fewer features.

4.7 Comparison with Existing Published Studies

A direct Quantitative comparison with existing feature selection technique is not feasible as our datasets is not same, feature extraction manners, different hyperparameter and evaluation method used in existing studies [17–21]. Most related studies evaluate their method on specific benchmarks or domain-specific datasets, making fair one-to-one quantitative benchmarking inappropriate.

However, the existing meta-heuristic and hybrid algorithms such as PSO, ABC, BA, and PCA based methods at a methodological level, often face issues just as premature convergence or classifier-independent feature selection. On the other hand, the proposed wrapper-based Fireworks Algorithm used an adaptive explosion technique that gradually stabilize exploration and exploitation while maintain population diversity. The results of the exper-

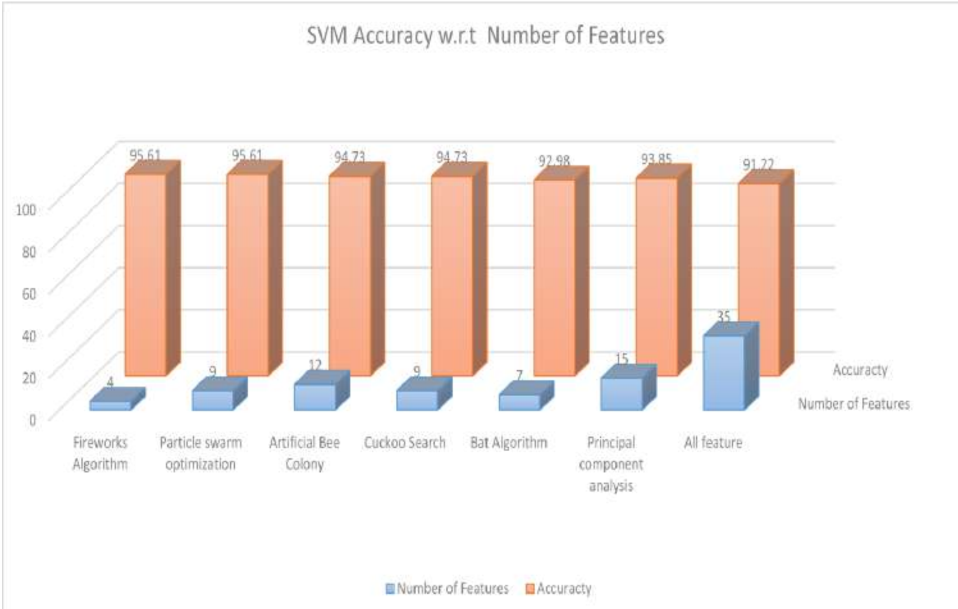

Figure 6. SVM Accuracy

iment on the employed dataset illustrate that FWA consistently select best feature subsets while attaining higher classification accuracy, show its efficiency compared to existing approaches reported in the literature.

5 Conclusion

Optimizing feature selection is a crucial step in the creation of machine learning models since it has a direct effect on their effectiveness, efficiency, and interpretability. To maximize model accuracy, minimize overfitting, and expedite training timeframes, feature selection aims to find and choose the most pertinent characteristics for inclusion in the training process. In this paper, we applied a novel approach of a firework algorithm for a feature selection problem. The firework algorithm returns the optimal subset of features by which the faster execution time and high accuracy are achieved as compared to other approaches like PCA, PSO, ABC, and other algorithms.

Author Contributions

Sakin Jan: Conceptualization, Methodology, Visualization, Writing Original draft **Fazli Amin Khalil**: Plagiarism removing, Data curation, Writing, Editing **Syed Bakhtawar Ahah Abid**: Supervision , Investigation. **Izaz Ullah**: Supervision, and investigation **Inayat Ullah**: Validation. **Inzimam Ul Haq**: Visualization and Editing

Compliance with Ethical Standards

It is declared that all authors don't have any conflict of interest. It is also declare that this article does not contain any studies with human participants or animals performed by any of the authors. Furthermore, informed consent was obtained from all individual participants included in the study.

Funding Information

The authors thank the Higher Education Department, Khyber Pakhtunkhwa, Pakistan for supporting this work.

References

[1] L. Wang, M. Han, X. Li, N. Zhang, and H. Cheng, "Review of classification methods on unbalanced data sets," *Ieee Access*, vol. 9, pp. 64606–64628, 2021.

[2] I. H. Sarker, “Machine learning: Algorithms, real-world applications and research directions,” *SN computer science*, vol. 2, no. 3, p. 160, 2021.

[3] V. Gupta, V. K. Mishra, P. Singhal, and A. Kumar, “An overview of supervised machine learning algorithm,” in *2022 11th International Conference on System Modeling & Advancement in Research Trends (SMART)*, pp. 87–92, IEEE, 2022.

[4] P. C. Sen, M. Hajra, and M. Ghosh, “Supervised classification algorithms in machine learning: A survey and review,” in *Emerging technology in modelling and graphics: Proceedings of IEM graph 2018*, pp. 99–111, Springer, 2019.

[5] W. Li, G.-G. Wang, and A. H. Gandomi, “A survey of learning-based intelligent optimization algorithms.,” *Archives of Computational Methods in Engineering*, vol. 28, no. 5, 2021.

[6] R. Zebari, A. Abdulazeez, D. Zeebaree, D. Zebari, and J. Saeed, “A comprehensive review of dimensionality reduction techniques for feature selection and feature extraction,” *Journal of Applied Science and Technology Trends*, vol. 1, no. 1, pp. 56–70, 2020.

[7] P. Dhal and C. Azad, “A comprehensive survey on feature selection in the various fields of machine learning,” *Applied intelligence*, vol. 52, no. 4, pp. 4543–4581, 2022.

[8] O. O. Akinola, A. E. Ezugwu, J. O. Agushaka, R. A. Zitar, and L. Abualigah, “Multiclass feature selection with metaheuristic optimization algorithms: a review,” *Neural Computing and Applications*, vol. 34, no. 22, pp. 19751–19790, 2022.

[9] S. Tkatek, O. Bahti, Y. Lmzouari, and J. Abouchabaka, “Artificial intelligence for improving the optimization of np-hard problems: a review,” *International Journal of Advanced Trends Computer Science and Applications*, vol. 9, no. 5, 2020.

[10] B. M. S. Hasan and A. M. Abdulazeez, “A review of principal component analysis algorithm for dimensionality reduction,” *Journal of Soft Computing and Data Mining*, vol. 2, no. 1, pp. 20–30, 2021.

[11] 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.

[12] Y. Lin, J. Wang, X. Li, Y. Zhang, and S. Huang, “An improved artificial bee colony for feature selection in qsar,” *Algorithms*, vol. 14, no. 4, p. 120, 2021.

[13] A. C. Pandey, D. S. Rajpoot, and M. Saraswat, “Feature selection method based on hybrid data transformation and binary binomial cuckoo search,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, no. 2, pp. 719–738, 2020.

[14] S. Eskandari and M. Seifaddini, “Online and offline streaming feature selection methods with bat algorithm for redundancy analysis,” *Pattern Recognition*, vol. 133, p. 109007, 2023.

[15] J. Li and Y. Tan, “A comprehensive review of the fireworks algorithm,” *ACM Computing Surveys (CSUR)*, vol. 52, no. 6, pp. 1–28, 2019.

[16] Y. Zhang, D.-w. Gong, X.-z. Gao, T. Tian, and X.-y. Sun, “Binary differential evolution with self-learning for multi-objective feature selection,” *Information Sciences*, vol. 507, pp. 67–85, 2020.

[17] A. Adamu, M. Abdullahi, S. B. Junaidu, and I. H. Hassan, “An hybrid particle swarm optimization with crow search algorithm for feature selection,” *Machine Learning with Applications*, vol. 6, p. 100108, 2021.

[18] D. Kumar *et al.*, “Feature extraction and selection of kidney ultrasound images using glcm and pca,” *Procedia Computer Science*, vol. 167, pp. 1722–1731, 2020.

[19] J. Balakumar and S. V. Mohan, “Artificial bee colony algorithm for feature selection and improved support vector machine for text classification,” *Information Discovery and Delivery*, vol. 47, no. 3, pp. 154–170, 2019.

[20] O. S. Qasim and Z. Y. Algamal, “Feature selection using different transfer functions for binary bat algorithm,” *International Journal of Mathematical, Engineering and Management Sciences*, vol. 5, no. 4, p. 697, 2020.

[21] F. Amini and G. Hu, “A two-layer feature selection method using genetic algorithm and elastic net,” *Expert Systems with Applications*, vol. 166, p. 114072, 2021.

[22] N. Mehrabi, S. P. Haeri Boroujeni, and E. Pashaei, “An efficient high-dimensional gene selection approach based on the binary horse herd optimization algorithm for biologicaldata classification,” *Iran Journal of Computer Science*, vol. 7, no. 2, pp. 279–309, 2024.

[23] E. A. Zaimoğlu, N. Yurtay, H. Demirci, and Y. Yurtay, “A binary chaotic horse herd optimization algorithm for feature selection,” *Engineering Science and Technology, an International Journal*, vol. 44, p. 101453, 2023.

[24] A. A. Bidgoli and S. Rahnamayan, “Large-scale multi-objective feature selection: A multi-phase search space shrinking approach,” *arXiv preprint arXiv:2410.21293*, 2024.

[25] X. Wang, H. Shangguan, F. Huang, S. Wu, and W. Jia, “Mel: Efficient multi-task evolutionary learning for high-dimensional feature selection,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 36, no. 8, pp. 4020–4033, 2024.

[26] E. Kraev, B. Koseoglu, L. Traverso, and M. Topiwalla, “Shap-select: Lightweight feature selection using shap values and regression,” *arXiv preprint arXiv:2410.06815*, 2024.

[27] K. Yang, L. Liu, and Y. Wen, “The impact of bayesian optimization on feature selection,” *Scientific Reports*, vol. 14, no. 1, p. 3948, 2024.

[28] N. Kapure, H. Joshi, P. Kumari, R. Mistri, and M. Mali, “Frame: Forward recursive adaptive model extraction-a technique for advance feature selection,” *arXiv preprint arXiv:2501.11972*, 2025.

[29] Z. Zhang, Q. An, Y. Wang, C. Wu, B. Dong, and C. Zhou, “A high-dimensional feature selection algorithm based on multi-objective differential evolution,” *arXiv preprint arXiv:2505.05727*, 2025.

[30] P. Sarker, J.-J. Tiang, and A.-A. Nahid, “Metaheuristic-driven feature selection for human activity recognition on ku-har dataset using xgboost classifier,” *Sensors*, vol. 25, no. 17, p. 5303, 2025.

[31] E. Ehsaeyan and A. Zolghadrasli, “Foa: fireworks optimization algorithm,” *Multimedia Tools and Applications*, vol. 81, no. 23, pp. 33151–33170, 2022.