



RESEARCH ARTICLE

OPEN ACCESS

OPTIMIZING MACHINE LEARNING MODELS PERFORMANCE USING SWARM-BASED FEATURE SELECTION: METHODS AND EXPERIMENTAL EVALUATION.

Ravi Bhushan¹, Vineeta Khemchandani²

^{1,2}School of Computing Science and Engineering, Galgotias University, Gr.Noida

¹School of Computer Application and Technology, Galgotias University, Gr.Noida

¹<https://orcid.org/0009-0001-7224-9169>, ²<https://orcid.org/0000-0002-7934-8493>

Email: ravidelhi2524@gmail.com, Vineetakh05@gmail.com.

ARTICLE INFO

Article History

Received: November 17, 2025

Reviewed: January 20, 2026

Accepted: May 15, 2026

Published: June 30, 2026

Keywords:

Swarm Intelligence,
Evolutionary Algorithms,
Feature Selection,
Machine Learning,
Dimensionality Reduction.

ABSTRACT

Large-scale datasets have grown rapidly in recent decades due to the rapid advancement of database and computer technology. However, the number of data mining applications that use high-dimensional datasets and need great speed and accuracy is growing quickly. The curse of dimensionality, in which there are substantially more characteristics than patterns, is a key problem with these applications. Feature selection is one of the dimensionality reduction strategies that can improve the data mining task's accuracy and lower its computational cost. The goal of the feature selection process is to choose a subset of features that are most relevant to the target class and have the lowest inner similarity. By removing unnecessary, redundant, or noisy data, it lowers the dimensionality of the data. This study compares and contrasts several feature selection techniques and provides a general classification of them. This research also examines the latest swarm intelligence techniques and reviews feature selection techniques based on these algorithms using the credit card fraud detection dataset. Additionally, we assess the advantages and disadvantages of the various swarm intelligence-based feature selection techniques that were examined.



Copyright ©2026 by authors and Galileo Institute of Technology and Education of the Amazon (ITEGAM). This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

I. INTRODUCTION

One of the most crucial applications of machine learning is the detection of credit card theft or pattern recognition. Several fields use machine learning techniques, including medical diagnosis [1],[2], marketing[3], image processing [4],[5], text mining [6],[7], information retrieval, and identification [8]. Dimensionality reduction is one of the key strategies to get rid of features that are redundant or unnecessary once there are a number of them in the original features[9]. Two methods for dimensional reduction were introduced in the past: feature selection and feature extraction. Dimensionality reduction can enhance the machine's performance by eliminating redundant and irrelevant characteristics [6], [10]. The main feature space is mapped to a smaller space during feature extraction. Actually, by integrating pre-existing features, this strategy creates fewer features that contain all or most of the information found in the primary features. Furthermore, in feature selection, redundant and unnecessary features are eliminated to choose a subset of the original features with n being the number of original features, the entire search space to discover the most pertinent and non-redundant features, including all potential subsets, is 2^n . Finding the best features requires a thorough search, which is typically not computationally possible, even for medium-sized datasets. Finding a solution that is both computationally practical and high-quality is necessary because searching through all potential subgroups is exceedingly expensive. To prevent growing computational complexity, a lot of feature selection techniques include metaheuristic algorithms [11]. In a reasonable amount of time, these algorithms will be able to optimise the feature selection problem with the right level of precision. Swarm intelligence (SI) and evolutionary algorithms (EA) are the two primary categories into which metaheuristic algorithms fall. Compared to EAs and other single-solution-based methods, SI is a relatively recent type of evolutionary computation. Swarm-based techniques are the most widely used nature-inspired meta-heuristic group.

SI algorithms employ approximate and non-deterministic strategies to effectively and efficiently explore and exploit the search space in order to obtain near-optimal answers. Artificial intelligence based on collective behaviours in decentralised, self-organised systems is known as swarm intelligence (SI). A population of simple actors that interact locally and with their surroundings often makes up these systems. In the fields of data mining, pattern recognition, statistics, and other engineering applications, feature selection has been a topic of intense investigation[12]. Reviewing several feature selection techniques will help you determine the best strategy for that application because feature selection is used in most engineering problems. To put it another way, different engineering application areas use different feature selection techniques. For instance, the precision of the feature selection technique is significantly more crucial than its computing complexity in many engineering tasks, such as trial engineering and biomedical engineering. In fact, even if a feature selection approach has a high computational complexity, we prefer to use it in certain situations in order to maximise accuracy.

However, the computational complexity of the feature selection method is also crucial in many engineering problems, like image processing and machine vision, and computational complexity and execution time must be taken into account when selecting the feature selection method in this field. In light of this, it is crucial to examine several feature selection techniques in order to select the best one for that particular application. To feature sections, numerous SI-based methods have been used in recent decades. Only a small number of studies examine feature selection techniques, despite their extremely satisfactory performance compare feature selection methods based on swarm intelligence. The well-known and conventional SI algorithm and the feature selection techniques derived from it were the exclusive attention of this paper's writers. Additionally, a thorough literature evaluation of SI algorithms for feature selection is conducted. This paper's absence of experimental results and solely listing the advantages and disadvantages of various SI-based feature selection techniques is one of its drawbacks.

Additionally, swarm intelligence methods for feature selection in data mining are reviewed by [13]. This paper's focus on PSO, ABC, and ACO-based feature selection techniques is one of its drawbacks; many other SI algorithms that have been put forth recently—such as the Salp Swarm Algorithm, Grey Wolf Optimisation, Whale Optimisation Algorithm, and Gravitational Search Algorithm—have not been covered. The fact that this research merely examined the various approaches theoretically and did not do any numerical analysis to compare them is another flaw. Additionally,[14] examined the challenges that arise during the feature selection process and the application of swarm intelligence to feature selection. Additionally, this paper offers a succinct summary of several algorithms, such as ABC, ACO, and PSO. This paper's flaw is that it only compares the various approaches based on classification accuracy criteria; it makes no comparisons based on computational complexity, execution time, or a few other chosen features.

The absence of a review of new SI algorithms, such as the Salp Swarm Algorithm, Whale Optimisation Algorithm, and Grey Wolf Optimisation, is another flaw in this work. Since there hasn't been a thorough analysis of SI-based feature selection techniques before, this paper's primary goal is to close the coverage gap of SI algorithms for feature selection. The goal of this study is to present a thorough analysis of SI-based feature selection techniques and their classification. Additionally, this work attempts to review the most popular and cutting-edge SI-based feature selection technique. Additionally, in order to provide a more accurate assessment of these algorithms, a number of experiments have been designed in this study to compare the performance of various SI-based approaches. The rest of the paper is structured as follows: Section 2 introduces the feature selection problem, and Section 3 examines several feature selection techniques and categories from earlier ODs. SI-based feature selection techniques are reviewed in Section 4. The experimental outcomes of several SI-based techniques are shown in Section 5. The conclusion is finally presented in Section 6.

II. THE FEATURE SELECTION PROBLEM IS INTRODUCED

The dimensionality problem of a dataset where the feature subset size is significantly larger than the pattern size is a fundamental problem with machine learning techniques [15]. For instance, in medical applications that involve very high-dimensional datasets, the classification parameters are also increased, which results in a significant decline in the classifier's performance. A general rule states that at least $10 \times n \times C$ training data is needed for a classification problem with n dimension and C class. If this amount of training data is not practically available, the number of training data needed is decreased by decreasing the feature subset size. Consequently, the classification algorithm's performance improves [16]; [17]. One common method for eliminating duplicate and noise-producing features is dimensionality reduction. According to [6] and [10], it is an effective way to increase accuracy performance, reduce computational complexity, create more generalised models, and use less storage. Two main methods for dimension reduction have been put forth in recent years: feature extraction and feature selection. In feature extraction, the original features are transformed into a new, lower-dimensional feature, either linearly or nonlinearly.

By choosing a subset of the original characteristics that still has enough information for classification, feature selection, on the other hand, directly lowers the number of features. In the fields of statistics, pattern recognition, and data mining, feature selection has been a topic of intense research. The basic principle behind feature selection is to pick a subset of accessible characteristics by removing redundant features that have a high correlation as well as features with little to no predictive value [12]. The main idea of feature selection is to choose a subset of available features by removing features with little or no predictive information and also redundant features that are strongly correlated[12]. Feature selection is a fundamental research topic in statistical pattern recognition, machine learning, and data mining that dates back to the 1970s. Several attempts have been made to improve the performance of the feature selection methods. Subset generation, subset evaluation, stopping criteria, and result validation are the four primary steps of the feature selection process.

A subset of the candidate feature set is created from the original features in each search iteration, and an evaluation criterion is used to gauge how acceptable the subset is. Until a predefined stop requirement is met, the subset generation and evaluation procedures are repeated. Following this procedure, the test dataset is used to validate the optimal subset of the chosen feature [18]. Generally speaking, feature selection is important [19]. Feature selection methods generally perform more reliably and efficiently in the supervised mode because of the use of class labels. In supervised methods, a set of train data is available, each of which is described by taking feature values along with the class label, whereas in unsupervised methods, training data lacks class tags. Consequently, choosing a feature in the unsupervised mode is more challenging, and this area has been taken into consideration in numerous studies [20]. Google Scholar gathers the papers for this review from a variety of databases, such as Elsevier, Springer, IEEE, and others.

To determine the present state of research, papers published between 2016 and 2020 were given the most weight. Review, dimensionality reduction, feature selection, evolutionary algorithm, optimisation, nature-inspired meta-heuristic, swarm-based techniques, metaheuristic optimisation algorithm, population-based algorithms, and evolutionary computation are the keywords used in the first search to find review papers. In the second search, the keywords are used to find research papers that focus on swarm intelligence-based feature selection methods. Also included are references to the featured papers as well as additional papers from Science Direct, Springer, and IEEE Xplore recommender systems. Subsequently, articles that concentrated on feature extraction and feature selection techniques not based on swarm intelligence were eliminated. Stated differently, only pertinent works that selected features only using swarm intelligence methods were included. As a result, 85 papers in all—including six review studies—were carefully examined for feature selection. The remaining references dealt with different approaches to the feature selection problem.

III. CONTEXTS

In general, feature selection techniques explore the solution space in order to maximise two competing goals: minimising the redundancy of chosen features and increasing their relevance to the target class [21]. Numerous search tactics are used to maximise these goals. According [22], these techniques can be broadly divided into single-objective and multi-objective approaches. Only one fitness function goal is used to optimise the population in single-objective approaches. The accuracy of the optimisation process will therefore be significantly impacted by the objective selection and fitness function definition. Additionally, many optimisation problems have several targets, and optimisation speed is decreased when a fitness function is defined with only one aim. Taking into account several objectives in the fitness function of the feature selection issue is one way to get over these obstacles. A collection of non-dominated feature subsets can be obtained by modelling feature selection as a multi-objective problem in order to satisfy various real-world application criteria.

Only a few multi-objective function approaches, including MOFSEF [23], MOFSEEG [23], and MOFSGADM [24], are used in the majority of the previously described feature selection methods. Furthermore, a number of earlier feature selection techniques aim to identify the fewest duplicated features with the greatest reliance on output labels. A new fuzzy multiple kernel learning model for classification is created by Wang. A novel criterion that chooses a subset of important and dissimilar features is put forward in this model. Furthermore, machine learning approaches have recently employed graph-based methods to uncover similarity associations among the data [25]. Graph-based techniques for feature selection offer an underlying manifold structure as a general framework to represent the connections among features. Graph-based techniques were used in a number of studies to address the feature selection issue.

For instance, a dense subgraph discovery method is used to solve the unsupervised feature selection problem present an additional clustering-based feature subset selection approach for large dimensional data. For similar grouping features, this work uses a graph-theoretic clustering technique suggest a feature selection technique based on hypergraphs. By taking into account each sample's associated class label, this study employs an information-theoretic criterion to assess the suitability of various features. The unsupervised feature selection procedure incorporates the idea of graph clustering with the node centrality metric. real-world applications. While the wrapping model could choose a superior feature subset, it can deteriorate with high-dimensional medical datasets and is costly to execute. This is because the feature subset quality computation makes use of learning methods.

III.1 MODEL OF FILTER

The filter model uses no learning mechanism to assess the features' importance. As a result, this approach's approaches are usually quick. The features with the highest ranks are chosen in this model after they have been assessed and rated using information-theoretic metrics [26]. According to one perspective, there are two types of filter-based methods: univariate and multivariate. These approaches, however, produce weak and complex learning models, neglect feature dependencies, and display comparable features in the final feature set. Multivariate approaches, a subset of the filter approach, address both redundant and irrelevant features in their ranking strategies in order to address this problem. Stated differently, multivariate filtering techniques were put forth with the goal of taking feature dependencies into account to a certain extent. Minimal redundancy maximal relevance (mRMR) , Relevance redundancy feature selection (RRFS), MIFS, Normalised Mutual Information Feature Selection (NMIFS), MIFS-U, Unsupervised feature selection based on Ant Colony Optimisation (UFSACO), and Hilbert–Schmidt independence criterion (HSIC) are a few multivariate techniques.

III.2 MODEL OF A WRAPPER

The wrapper approach uses a classifier that has been trained to assess a number of salient attributes. The wrapper model selects a collection of features with the maximum classification accuracy by evaluating a subset of features in their search operations using a specific learning model. The majority of wrapper approaches use iterative search procedures, in which the population of solutions is guided towards the optimal solution with each iteration of the learning model. However, the wrapper approach's high computing cost and loss of generality are caused by the learning model that is used in the search process. Filter methods are suitable for populations with random solutions and aim to increase the optimality of these solutions at each iteration step since they are quick enough and their output is independent of a particular classifier. In real-world applications, a fitness function is used to determine the optimality of the individual solutions of the created decision-makers after a number of initial solutions are generated at random at the start of most meta-heuristic algorithms.

For gene selection and cancer classification, [27] integrate correlation feature selection with a modified binary PSO method. By removing the unnecessary and duplicated features, this technique chooses a subset of highly significant attributes. The Shuffled Frog Leaping Algorithm and enhanced mRMR are combined in [28] to increase the accuracy of acoustic defect identification. This method uses the MRMR criterion to build a single-objective function, which is then optimised using the PSO algorithm to reduce the dimensions of the original features. The neural network classifier is then used to detect the final auditory defect after the final feature subset has been chosen. [29] offer a new PSO-based feature selection technique that combines illness diagnosis performance. To choose a subset of pertinent and non-redundant features in DNA microarray datasets, developed a recursive particle swarm optimisation strategy. The authors of this paper integrated the different filter-based feature selection measures with the developed wrapper-based recursive PSO method use the modified BPSO algorithm with the binary black hole algorithm to construct a feature selection technique for cancer classification.

This method introduces a hybrid feature selection method by combining the benefits of the wrapper and filter approaches. To increase effectiveness and make it easier to explore and exploit the PSO to further improve performance, this hybrid feature selection method incorporates the binary black hole algorithm into the PSO algorithm. Furthermore, an enhanced BPSO approach for feature selection was created. This technique employs a novel Win-Win approach to improve feature selection accuracy in the detection of renal and liver diseases. In addition, [30] presented a new hybrid PSO-based feature selection technique for laser-induced breakdown spectroscopy investigation.

An effort has been made to combine the benefits of coating and filtering techniques in this procedure. A PSO-based feature selection method with several classifiers is suggested in [13] to enhance the population. A new generation of production will start if none of the termination requirements are fulfilled. Until one of the termination criteria is satisfied, this cycle is repeated. Evolutionary Algorithms (EA) and Swarm Intelligence (SI) are two types of meta-heuristic techniques. EAs employ mechanisms including reproduction, mutation, recombination, and selection that are modelled after biological evolution. Individuals in a population are represented by candidate solutions to the optimisation problem, and the quality of these solutions is determined by the fitness function. The starting population evolves and advances towards global optimisation following iterations of the evolutionary algorithm [31].

Furthermore, the optimisation approach based on swarm intelligence takes into account a lot of complex artificial aspects. Nature is the source of the idea for SI optimisation algorithms. Each component performs a basic task, but together, these components produce global intelligent behaviour that is impossible for any one component to accomplish on its own. Particle Swarm Optimisation (PSO), Ant Colony Optimisation (ACO), Artificial Bee Colony Optimisation (ABC), Differential Evolution (DE), Gravitational Search Algorithm (GSA), Firefly Algorithm (FA), Bat Algorithm (BA), Cuckoo Optimisation Algorithm (COA), Grey Wolf Optimisation (GWO), Whale Optimisation Algorithm (WOA), and Salp Swarm Algorithm (SSA) are among the SI-based feature selection techniques that are reviewed and described in Table 2 in the sections that follow.

III.3 MODEL OF HYBRID

By combining the filter and wrapper models, the hybrid strategy aims to benefit from both of them. Like filter-based approaches, this strategy primarily focusses on maximising performance using a specific learning algorithm and time complexity.

III.4 THE EMBEDDED MODEL

The feature selection issue is taken into account by the embedded approach as a component of the machine learning technique. Actually, a machine learning technique is employed in the embedded approach to find the last feature subset.

The key traits of the various feature selection techniques are compiled in Table 1. Six characteristics of various feature selection techniques are listed in this table: the number of objectives, the type of approach, the search strategy, the application, the weakness, and the strength.

Table1: The main characteristics of different feature selection methods. The synonyms used in this Table are: MOP: Multi-Objective Optimization, SOP: Single-Objective Optimization, RB: Ranking-Based, SSB: Subset selection-based, ACO: Ant Colony Optimization, GA: Genetic Algorithm, ABC: Artificial Bee Colony, DE: Differential Evolution, NSGA: Nondominated Sorting Genetic Algorithm.

Methods	Numberofobjectives	Type	Searchstrategy	Application	Weakness	Strength
RDC	SOP	Filter-RB	Univariate	Textual		
DFS	SOP	Filter-RB	Univariate	Textual		
NDM	SOP	Filter-RB	Univariate	Textual		
FS	SOP	Filter-RB	Univariate	Textual/Numeric		
GI	SOP	Filter-RB	Univariate	Textual/Numeric	Ignoring feature dependency,	Fast, Independent of the classifier
MI	SOP	Filter-RB	Univariate	Textual/Numeric	Low performance	
LS	SOP	Filter-RB	Univariate	Numeric		
IG	SOP	Filter-RB	Univariate	Textual/Numeric		
CHI	SOP	Filter-RB	Univariate	Textual/Numeric		
GR	SOP	Filter-RB	Univariate	Numeric		
TV	SOP	Filter-RB	Univariate	Numeric		
RRFS	SOP	Filter	Multivariate Greedy	Numeric	Usingagreedyssearch	
					Strategy and easily	Having higher accuracy
mRMR	SOP	Filter	Multivariate Greedy	Numeric	Trappinginto thelocal	Thanunivariate methods
RSM	SOP	Filter	Multivariate Greedy	Numeric	optima.	
GUFS	SOP	Filter	MultivariateGreedy	Numeric		
MECY_FS	MOP	Filter	GA	Numeric	OnlyApplicabletonum ericdatasets, Havinglo werperformance, In-text datasets.	Usinginformation-theoreticMethodsto evaluate the Objective function that has Made them fast.
RRFSACO	SOP	Filter/SSB	MultivariateACO	Numeric	Being single-objective, Forcing the evolvingPopulationto form Particular features due To the use of a single Qualityfunction.	Consideringthe feature dependency

GCACO	SOP	Filter/SSB	MultivariateACO	Numeric		
MGCACO	SOP	Filter/SSB	MultivariateACO	Numeric		
FASTFS	SOP	Filter/SSB	MultivariateGraph-based	Textual/Numeric		
GCNC	SOP	Filter/SSB	MultivariateGraph-based	Numeric		
FSGA	SOP	Filter/SSB	MultivariateGA	ForeignFiber		
MICGSOFS	SOP	Filter/SSB	MultivariateSequential	BiomedicalDatabases		
MBFFS	SOP	Filter/SSB	MarkovBlanket	Numeric		
BT-SFS	SOP	Wrapper	Sequential	FaultDetection	Depending on the initial solutions and easily falling into the local optima	Having low computational complexity
MOABC	MOP	Wrapper	ABC	Numeric		
TMABC-FS	MOP	Wrapper	ABC	Numeric	In efficient for high-dimensional datasets Due to high computational complexity.	Low probability of trapping into the local optimum, High classification accuracy
MOPSOFS	MOP	Wrapper	PSO	Numeric		
MOACO	MOP	Wrapper	ACO	Numeric		
MOWGA	MOP	Wrapper	GA	FaceRecognition		
MODEFS	MOP	Wrapper	DE	FaceRecognition		
MONSGA	MOP	Wrapper	NSGA-II	Motorimagery		
NestedGA	MOP		NSGA-II	Microarray		

Source: Authors, (2026).

Table 2: Outlining the reviewed swarm intelligence-based feature selection methods.

Reference	Simethod	Numberofobjectives	Type	Application
Zhangetal.(2017a)	PSO	Singleobjective	Wrapper	Numerical
Jainetal.(2018)	PSO	Singleobjective	Hybrid	DNA microarray
Zhangetal.(2018a)	PSO	Singleobjective	Filter	Signal
QasimandAlgamal(2018)	PSO	Singleobjective	Wrapper	Medical
Prasadetal.(2018)	PSO	Singleobjective	Wrapper	DNA microarray
Pashaeietal.(2019)	PSO	Singleobjective	Wrapper	Medical
Gunasundarietal.(2018)	PSO	Singleobjective	Wrapper	Medical
Yanetal.(2019)	PSO	Singleobjective	Wrapper	Spectroscopy
Xueetal.(2020)	PSO	Singleobjective	Wrapper	Numerical
Rostamietal.(2020b)	PSO	Multiobjective	Filter	Medical
Liuetal.(2019a)	ACO	Multiobjective	Wrapper	Numerical
ShunmugapriyaandKanmani(2017)	ABC	Singleobjective	Wrapper	Numerical/Medical
Hanceretal.(2018b)	ABC	Singleobjective	Wrapper	Numerical
ArslanandOzturk(2019)	ABC	Singleobjective	Wrapper	Numerical
Zhangetal.(2019a)	ABC	Multi-Objective	Wrapper	Numerical
Wangetal.(2020b)	ABC	Multi-Objective	Wrapper	Numerical
Hanceretal.(2018a)	DE	Multiobjective	Filter	Numerical/Medical
Zhangetal.(2020)	DE	Multiobjective	Wrapper	Numerical/Medical
Hancer(2020)	DE	Singleobjective	Wrapper	Numerical/Medical
Xiangetal.(2015)	GSA	Singleobjective	Wrapper	Numerical/Medical
Taradehetal.(2019)	GSA	Singleobjective	Wrapper	Numerical/Medical
Zhangetal.(2017b)	FA	Multi-Objective	Wrapper	Numerical
Zhangetal.(2018b)	FA	Singleobjective	Wrapper	Facial expression
LarabiMarie-SainteandAlalyani(2020)	FA	Singleobjective	Wrapper	Text
SelvakumarandMuneeswaran(2019)	FA	Singleobjective	Hybrid	Network Intrusion
TawhidandDsouza(2018)	BA	Singleobjective	Wrapper	Numerical/Medical
Liuetal.(2020)	BA	Singleobjective	Wrapper	Image Steganalysis
Al-Betaretal.(2020)	BA	Singleobjective	Hybrid	Medical
Elyasigomarietal.(2017)	COA	Singleobjective	Wrapper	Numerical/Medical
JayaramanandSultana(2019)	COA	Singleobjective	Wrapper	Medical
Prabukumaretal.(2019)	COA	Singleobjective	Wrapper	Gene expression
Tuetal.(2019)	GWO	Singleobjective	Wrapper	Numerical/Medical
Abdel-Bassetetal.(2020)	GWO	Singleobjective	Wrapper	Numerical/Medical
MafarjaandMirjalili(2017)	WOA	Singleobjective	Wrapper	Numerical/Medical
MafarjaandMirjalili(2018)	WOA	Singleobjective	Wrapper	Numerical/Medical
Nematzadehetal.(2019)	WOA	Singleobjective	Filter	Medical
Farisetal.(2018)	SSA	Singleobjective	Wrapper	Numerical/Medical
Ibrahimet al.(2019)	SSA	Singleobjective	Wrapper	Numerical
Tubishatetal.(2020)	SSA	Singleobjective	Wrapper	Numerical/Medical

Al-Zoubietal.(2020)	SSA	Singleobjective	Wrapper	Medical
Hegazyetal.(2020)	SSA	Multi-Objective	Wrapper	Numerical/Medical
Neggazetal.(2020)	SSA	Singleobjective	Wrapper	Numerical/Medical

Source: Authors, (2026).

IV.SWARM INTELLIGENCE-BASED FEATURE SELECTION.

Feature selection methods based on how to evaluate the features are classified into two categories: feature ranking and subset selection methods. Feature ranking methods, based on a specified criterion, assign a score to each feature. Then the features that did not get scores are removed. But in subset selection methods, the space of possible subsets is searched to find the optimal subset. If the number of initial features is n , the search space for the optimal subset contains all the features, which is equal to 2^n different states. In other words, in the property ranking methods, the value of each property is evaluated independently, and the relationship between features is not considered. In these methods, it is assumed that the features are independent of each other and that a possible dependence between the features is not considered. Although this simplistic assumption reduces the computational complexity of the feature selection method, in many cases, it may reduce the performance of the feature selection method. Feature subset selection is an NP-Hard problem. In the simplest way, the best subset can be found by evaluating all possible subsets with an exhaustive search strategy.

Although this method guarantees an optimal feature subset, finding the optimal solution is very time-consuming and even impractical for medium-sized datasets. Since it is very costly to evaluate all possible subsets, a feature subset must be searched, that, it is acceptable both in terms of computational complexity and in terms of appropriateness [32] [33]. One approach to solving complex optimization and NP- Hard problems is meta-heuristic algorithms. Meta-heuristic algorithms are approximate approaches that can find satisfactory solutions over acceptable time instead of finding the optimal solution. These algorithms are one of the categories of approximate optimization algorithms that have strategies to escape from local optima and can be used in a wide range of optimization problems. Many feature selection methods use meta-heuristics to avoid increasing computational complexity in high-dimensional dataset. These algorithms use primitive mechanisms and operations to solve an optimization problem and search for the optimal solution over a number of iterations. These algorithms are often extended by the authors of [34] to choose more informative features.

In Henni et al.(2018), the authors employed Google's PageRank centrality measure to rank features based on their importance[34] propose another graph-based feature selection method for the multi-label high-dimensional dataset. The authors of this paper utilize the PageRank centrality measure to rank the features based on their value in the graph. Also, in this paper, the correlation distance criterion is used to remove irrelevant features. In [35], an unsupervised graph-based feature selection method for high-dimensional data is proposed. In this paper, the Laplacian graph and local geometrical structure are used for better representation of the feature space. In a subspace clustering guided unsupervised feature selection method is proposed. This work uses the subspace clustering to learning of the clustering, and then those features with good preservation ability of the cluster labels are selected. Generally, previous methods for selecting optimal feature sets can be categorized into four groups consist of filter, wrapper, embedded, and hybrid models. These categories are explained in the following subsections.

IV.1 PSO-BASED METHODS

Particle Swarm Optimization is a powerful SI-based optimization method, developed by Kennedy and Eberhart in 1995. This optimization method, inspired by the collective conduct of birds and fish, has recently been used in a number of researches to optimize the selection of the final feature subset. Unler et al.(2011) introduced a new search strategy to select the final feature set, which integrated the filter approach. Moreover, developed a hybrid feature selection method for the medical application. The authors of this paper, proposed a hybrid feature selection approach by combining the wrapper PSO and rough sets theory to improve the classification accuracy of disease diagnosis. Furthermore, in Huang and Dun(2008) another hybrid approach for feature subset selection and parameter optimization is proposed. In this method the PSO optimization method, and SVM is integrated to improve the performance of feature selection. The authors of this paper combined PSO-SVM with a distributed parallel architecture to overcome the high computational complexity for the high-dimensional datasets.

Moreover, the authors of developed novel initialization methods and best particle updating strategies to improve the performance of optimization tasks in feature subset selection. This method aims to select a subset of relevant and non-redundant features for computational complexity reduction. Proposed an integration model of binary PSO (BPSO) algorithm and the Hamming distance to select the final feature set in the classification task. In this method, a hamming distance is used to improve the velocity update process in a BPSO algorithm and then this improved BPSO is utilized for feature selection problem. An improved multi-objective PSO algorithm is proposed for unreliable data classification. In this paper, two new operators of the reinforced memory and the hybrid mutation are introduced to improve the searchability of the PSO algorithm. In efficient PSO-based feature selection method is proposed by the integration of filter and wrapper approaches.

The proposed method, called HPSO-LS, introduced a new local search to select the subset of non-redundant and relevant features. In [36] a multi-objective particle swarm optimization approach is developed for cost-based feature selection in classification. The main goal of this method is to produce a Pareto front of non-dominated solutions, that is, feature set, to meet different prerequisites for increasing the classification accuracy and reducing computational complexity. In this paper, a new Self-Adaptive Parameter and Strategy a high-dimensional dataset. The reported results showed that the use of these mechanisms greatly increased the search ability of particle optimization algorithms for high-dimensional datasets. Moreover, in [38], a novel graph-based feature selection method is developed to increase disease diagnosis accuracy. In this method, using the node centrality criterion, a new mechanism for initializing the particles is proposed. Then, by defining a multi-objective fitness function, a subset of the final features that are least similar to each other and most relevant to the target class are selected. Finally, based on the selected features, the disease is diagnosed, disease diagnosis accuracy. In this method, using the node centrality criterion, a new mechanism for initializing the particles is proposed.

Then, by defining a multi-objective fitness function, a subset of the final features that are least similar to each other and most relevant to the target class are selected. Finally, based on the selected features, the disease is diagnosed.

IV.2 ACO-BASED METHODS

About three decades ago, a new heuristic-based optimization method called the Ant System (AS) is proposed by Dorigo and his colleagues to solve complicated optimization problems. AS was first utilized for the traveling salesman problem. Then, Dorigo and Caro(1999) developed the Ant Colony Optimization (ACO) algorithm to improve the AS algorithm. This improved algorithm can be able to search the global optimal by representing the problem as a graph model. Moreover, ACO has been successfully utilized in many types of research to select the final feature set. In Kabir et al.(2012) an ACO-based feature selection approach is developed by combining the neural network and IG method to remove redundant and irrelevant features. Moreover, the authors of Li et al.(2013) proposed a new ACO-based gene selection method in the DNA microarray dataset. This proposed method includes two phases of essential features by improved AS algorithm and then final features are selected using a modified ACO algorithm.

Furthermore, in a new ACO-based feature selection approach is developed. In this approach, unlike the previous ant feature selection methods based and ACO, the feature selection problem is modeled as a directed graph. In Forsati et al.(2014), a new version of ACO is proposed to remove the irrelevant and redundant features. This method selects the final feature set by utilizing a local search strategy to avoid getting stuck in the local optimal. Moreover, in Keetal.(2008), a rough set theoretic-based ACO feature selection method is developed. In Tabakhi et al.(2014) a novel ACO-based feature selection method is proposed for unsupervised mode. The authors of this paper selected the most non-redundant features that have the least similarity with each other. Moreover, Moradi and Moradi and Rostami, developed a filter-based feature selection approach utilizing the ACO algorithm and graph clustering.

This approach represented the feature space as a clustered graph. Then, according to the similarity between the features and by defining a filter criterion, it selects a dissimilar and related subset of the features. In Dadaneh et al.(2016) proposed an unsupervised ACO-based feature selection method to remove redundant and irrelevant features. This method tries to select an optimal subset of features in a hierarchical process, by considering the similarity between features. In [2], a combination of feature selection and ant colony optimization is proposed to improve the classification accuracy of imbalanced data. In this method, instead of using a single-objective fitness function, a multi-objective ant colony optimization algorithm is used to improve the performance of feature selection. The reported results showed acceptable performance of the proposed method in classifying imbalanced and high-dimensional datasets.

IV.3 ABC-BASED METHODS

The Artificial Bee Colony algorithm (ABC) is an SI-based optimization method inspired by the lifestyle of the bee population. This algorithm tries to imitate the food search behavior of the bee population. This optimization technique integrated a local search strategy with a random search strategy that can be utilized for complex optimization problems. This swarm intelligence-based optimization technique has been applied in many references to select a final feature set. In Schiezero and Pedrini (2013), a novel feature selection method based on an artificial bee colony is developed to improve the machine learning task. In this method, the feature space is modeled as a binary vector and the classification function. The authors of, developed an improved ABC-based feature selection approach. This approach combined the similarity search strategy and a binary version of ABC to improve the performance of feature selection. Shunmugapriya and Kanmani[39], proposed a novel feature selection method by integrating. This method tries to utilize the advantages of an artificial bee colony and an ant colony optimization algorithm, simultaneously.

In[40] a new feature selection approach is developed by the combination of the ABC algorithm and the Pareto-optimal front surface. The authors of this paper used a multi-objective fitness function and genetic operators to improve the accuracy of the feature selection method and the convergence of the ABC algorithm. In Arslan and Ozturk(2019) a Multi Hive ABC Programming is developed to select the final features in high-dimensional datasets. This approach utilized the ability of an automatic programming algorithm to remove irrelevant and redundant features. The authors of [41], developed a multi-objective ABC-based feature selection approach. In this method, two new operators are used to improve its search capability and convergence of the ABC search strategy. In [42], an ABC-based feature selection is proposed by integrating of multi- multi-objective optimization algorithm with a sample reduction strategy. This proposed method has both increased classification accuracy and reduced computational complexity.

IV.4 DE-BASED METHODS

The Differential Evolution (DE) algorithm is an SI-based search strategy technique that has been developed to solve complex optimization problems. This optimization method is introduced to overcome the main weakness of the genetic algorithm, namely its lack of local search capability. The main difference between DE algorithm and the genetic algorithms is in the genetic selection operators. DE algorithm is used in different applications of machine learning tasks and feature selection. For example, Al-Anietal developed a wrapper-based feature subset selection using DE algorithm and a wheel-based search mechanism.[43] proposed a multi-objective DE-based approach to remove the irrelevant and redundant features and improve classification accuracy. In [44], a multi-objective DE-based feature selection approach is developed. In this approach, a novel mutation operator is defined to escape the local optimum. Furthermore, in [45], another new multi-objective DE-based feature selection approach is developed for feature selection and to improve the performance of the clustering algorithm simultaneously.

IV.5 GSA-BASED METHODS

Moreover, some SI-based optimisation methods have been inspired by physical laws. Gravitational Search Algorithm (GSA) is a physics-based optimisation technique inspired by Newton's law of universal gravitation. GSA, a popular SI-based algorithm, has been widely applied in machine learning, and recently, many feature selection methods have been developed using this optimisation algorithm. proposed a GSA-based feature selection approach using a linear chaotic map to improve classification accuracy. In Xiang , a hybrid system for feature selection based on an enhanced GSA and K-nearest neighbour classifier is developed.

In this method, a piecewise linear chaotic map for exploration and sequential quadratic programming for exploitation are employed. In [46], an efficient hybrid feature selection method is proposed to utilize the advantages of the SI and the genetic algorithm to improve the performance of the GSA algorithm.

IV.6 FA-BASED METHODS

The Firefly Algorithm (FA) was introduced, the main idea was inspired by the optical connection between fireflies. The Firefly Algorithm is a sensible example of swarm intelligence in which low-performance agents can work together to achieve great results with high performance. In [47] a novel FA-based feature selection method, called return-cost-based binary FFA is developed. The authors of this paper provide a variety of strategies to prevent premature convergence of the FA algorithm and thus improve the accuracy of feature selection. In [48], a feature selection method is developed using the firefly optimization algorithm to increase the accuracy of classification and regression models. In this method, to improve convergence and prevent trapping in local optimization, some variation is added to standard FA, which improves the accuracy of final feature selection. Moreover, in [49] FA-based feature selection method is proposed. In this method, after selecting the final features, these features are utilized to classify Arabic texts using an SVM classifier. Furthermore, in Sel-vakumar and Muneeswaran(2019), another FA-based feature selection method is presented for network intrusion detection. In this method, a combination of filter-based feature selection method (i.e., Mutual Information) and wrapper-based feature selection method (i.e., C4.5 and Bayesian network) has been utilized to select the final features.

IV.7 BA-BASED METHOD

The Bat Algorithm (BA) is an algorithm inspired by the collective behavior of bats in the natural environment, introduced. The bat algorithm is a kind of swarm intelligence-based algorithm that is inspired by the echolocation behavior of bats. Bats find the exact path and location of their prey by sending sound waves and receiving reflections. When the sound waves return to the transmitter of the bat waves, the bat can draw an audio image of the obstacles in front of its surroundings and see its surroundings well. In [50], a hybrid variant of- rithm and improved PSO algorithm is to improve the feature selection performance. In this proposed method, the PSO algorithm is used to improve the convergence power of the hybrid algorithm. Moreover, in [49], a binary BA-based feature selection method ion accuracy. Furthermore, [54] used the bat algorithm to search optimal features subset to increase the accuracy of cancer classification. In this method, a combination of filter and wrapper approaches was used to improve the performance of feature selection. In this method, robust mRMR as a filter to select the most relevant features and an improved BA algorithm as a search strategy in the wrapper approach is presented to select the final feature subset.

IV.8 COA-BASED METHOD

The Cuckoo Optimization Algorithm (COA) is another algorithm based on swarm intelligence, inspired by the special lifestyle of a bird called the cuckoo. The specific habitude of laying eggs and the reproduction of this bird has been the basis for the formation of this optimization algorithm. Like other evolutionary algorithms, the cuckoo optimization algorithm begins with an initial population of cuckoos. This early population of cuckoos has a number of eggs that are placed in the nest of the host bird. Some eggs, which are more similar to host bird eggs, have a better chance of growing into adult birds. The other eggs are identified and destroyed by the host butterfly. Grown eggs show that the nest is a better place in the search space, and the usefulness of that area is higher. The goal of the cuckoo optimization algorithm, which is the optimization function, isto find the place where most eggs have a more chance to survive. In a COA-based feature selection method is developed to improve the cancer classification accuracy.

In this method, first, the irrelevant features are removed using a simple and fast filter-based feature selection. Then, from these relevant features, the final features are selected by integration wrapper-based feature selection and COA algorithm. In Jayaraman and Sultana(2019), a combination method of cuckoo search algorithm and neural network is developed for feature selection. In this method, after selecting a feature subset of non-redundant and relevant features, the final chosen features are sent to the classifier, and heart disease is classified. Moreover, in [37], another cuckoo search-based feature selection method is proposed for improving the disease diagnosis accuracy. In this method, the process of feature subset selection is optimized using the cuckoo search optimization, and then these selected features are sent to the SVM classifier to identify lung cancer.

IV.9 GWO-BASED METHOD

Gray Wolf Optimization (GWO) is a new evolutionary algorithm-based optimization technique inspired by gray wolves. This optimization method is one of the latest bio-inspired techniques, which imitates the hunting process of a pack of gray wolves in nature. Recently, some GWO-based methods has been utilized as a tool for feature selection in machine learning applications. Ia Multi-Objective GWO was applied to search for the most relevant and non-redundant features. This proposed method utilized the low computational complexity of the filter model to improve the accuracy of the wrapper model. In a binary version of the GWO is developed to select the optimal feature subset for classification tasks. In the wrapper-based feature selection method, classification accuracy and the number of selected features are utilized for the fitness function. In to enhance the previous GWO-based feature selection approach, a multi-strategy ensemble GWO is developed for feature selection. Moreover, [51] developed a GWO-based wrapper feature selection approach that integrated with a mutation operator for data classification. The mutation operator proposed in this paper tries to remove the redundant and irrelevant features.

IV.10 WOA-BASED METHODS

Optimization Algorithm (WOA) is another SI-based optimization method inspired by the hunting behaviour of humpback whales. This optimization method consists of three operators to imitate the search for prey, encircling prey, and bubble-net foraging behaviour of humpback whales.

Recently, the WOA algorithm is successfully applied in many different optimization problems and feature selection. In a hybrid WOA with a simulated annealing algorithm is developed for feature selection. The aim of using simulated annealing in this hybrid method is to improve the exploitation by searching the most promising regions located by the WOA algorithm. In a WOA-based wrapper feature selection algorithm is developed. In this study, tournament and roulette wheel selection mechanism and also crossover and mutation operators, are used to improve the exploration and exploitation of the search process of WOA algorithm. Introduced a frequency-based filter feature selection method using a whale algorithm on high-dimensional medical datasets. In this method, a filter criterion is utilized to discard the irrelevant features using the WOA. Then, the reminder features are ranked based on another filtering method, namely, Mutual Congestion.

IV.11 SSA-BASED METHODS

Salp Swarm Algorithm (SSA), another bio-inspired algorithm based on swarm intelligence, is proposed for solving optimization problems. This algorithm is inspired by the swarming behavior of salps when moving and for in the oceans. In an efficient binary Salp Swarm Algorithm with a crossover scheme is proposed to improve the accuracy of feature selection. Moreover, in a hybrid optimization method for the feature selection problem is proposed by the integration of the slap swarm algorithm with particle swarm optimization. This combination of both swarm optimization algorithms improved the efficacy of the exploration and the exploitation steps. In the integration of Improved SSA and novel local search algorithm is developed for the feature selection problem. In this method, the accuracy of the classification accuracy of KNN classifier on the training data is used as a fitness function, and the final features are selected in repetitive process. Also, in this method, to create the initial population, a method based on Opposition-Based Learning is presented, which has caused diversity in the initial solutions. Furthermore, in [52], a Salp Swarm Algorithm-based feature weighting method is developed to predict the presence of liver is order, heart, and Parkinson's disease.

Also, in solution accuracy, reliability, and convergence speed of basic SSA is improved by adding a new control parameter and inertial weight. Then this improved algorithm was tested in the feature selection problem. Moreover, in [53], anew variant of SSA optimization algorithm is proposed for feature selection. In this method, an additional phase is utilized to overcome the problem of being stuck in local optima by encouraging the exploration of the search. With the advancement of data collection technologies and the increasing capacity of data storage over the last decades, high-dimensional datasets have grown significantly. Usually, many features of these datasets are irrelevant or redundant, which reduces the performance of the prediction model. Feature selection plays an essential role in machine learning and, more specifically, in high-dimensional datasets. Reducing the size of the medical dataset, on the one hand, reduces the computational complexity and, on the other hand, decreases the parameters of the classification algorithm. As a result, the accuracy of the prediction model will be increased. In the past decades, rapid growth of computer and database technologies has led to the rapid growth of large-scale datasets.

On the other hand, data mining applications with high-dimensional datasets that require high speed and accuracy are rapidly increasing. An important tissue with these applications is the curse of dimensionality, where the number of features is much greater than the number of patterns. One of the dimensionality reduction approaches is feature selection which can increase the accuracy of the data mining task and reduce its computational complexity. The feature selection method aims at selecting a subset of features with the lowest target class. It reduces the dimensionality of the data by eliminating irrelevant, redundant, or noisy data. In this paper, comparative analysis and categorization of different feature selection methods are presented. Moreover, in this paper, wrapper and filter SI-based methods (i.e., PSO, ACO, ABC, DE, GSA, FA, BA, COA, GWO, WOA, and SSA) and their application in feature selection are studied. Furthermore, the strengths and weaknesses of the different studied SI-based feature selection methods are evaluated, and the factors that can lead to the superiority of these methods are analyzed.

V. EXPERIMENTAL RESULTS

In this section, the performances of the different swarms- evaluated. The results are shown in terms of the number of selected features and the classification accuracy (ACC). The classification accuracy was calculated as follows:

$$ACC = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

Where TP, TN, FP, and FN stand for the number of true positives, true negatives, false positives, and false negatives, respectively. In each experiment, each feature selection method is run ten times, and the mean and standard deviation (square root of its variance) often different runs is often used to compare different methods.

V.1 DATASETS

All simulations were executed against the credit card fraud dataset, which is freely available on the Kaggle repository via the following link : <https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud>, accessed on 20 May 2022. This dataset consists of transactions generated by credit cards in Europe in September 2013 during the time span of two days. The dataset represents a binary classification challenge composed of only two target variables (classes)—the positive class, which denotes fraudulent transactions, and the negative class that represents regular transactions. Moreover, the dataset is extremely asymmetrical (imbalanced), containing only 492 fraud instances out of 284,807 total transactions. Therefore, the positive class (frauds) represent only 0.172% of the dataset. The dataset is composed of 30 numerical features, where attributes F1, F2, F28 are obtained by applying the principal component analysis (PCA), while F29 and F30, which represent time and amount, respectively, were not transformed with the PCA.

The time refers to the number of seconds elapsed between the first and each other transaction in the dataset, while the amount is the value of every transaction. In the first set of experiments, the original credit card fraud dataset, as it is hosted on Kaggle, is used, and the goal of these experiments was to establish how tuned ML models perform on highly imbalanced data. However, it is also important to validate the performance of ML models for balanced datasets. In the proposed study, the minority class (class 1 in this example) was oversampled to the number of instances of the majority category (class 0 in this example); therefore, the dataset, which in datapoints, is almost twice as large as the original one.

It needs to be noted that the experimental setup for SVM simulations had to be different, due to the specific properties of the model. In the case of SVM, a smaller dataset has been used, as the model operates very slowly. This was tackled by creating a reduced dataset with the size of only 0.5% of the original dataset that was fed to the models while keeping the original class distribution (by using the stratification strategy). In the case of SVM and ELM experiments, the normalisation technique was also used. However, since the XGBoost operates on a decision tree basis, normalization has not been performed in that case. A total of 70% of each dataset was used for training, while the remaining 30% was utilized for testing of all models.

V.2 EXPERIMENTAL SETUP AND PROPOSED ENCODING SCHEME

As noted previously, all three ML models were tuned by the proposed GSFA meta-heuristic. In the case of the first utilised model, namely the SVM, hyperparameters were subjected to optimisation; two of them are continuous, while one is of the integer data type. The optimized SVM parameters, along with respective lower and upper bounds, are as follows:

C, boundaries:[2–5,215], type: continuous,
 γ , boundaries:[2–15,23], type: continuous, and kernel type, boundaries :[0,3], type: integer, where value 0 denotes polynomial(poly),
 1 marks radial basis function (rbf),
 2 represents sigmoid, and finally
 3 represents linear kernel type.

For the ELM model, both the number of neurons (nn) in the hidden layer and the values of the weights and biases between the input and hidden layers were subjected to the optimisation process. The lower and upper bounds for weights and biases were set as -1 and 1, respectively, in the interval [30,150]. The nn is an integer, while the weights and biases may take any continuous value from the specified range. Moreover, hyperparameters, while the optimisation of weights and biases is the ELM's training process, in the case of ELM, metaheuristics were used for both hyperparameters' optimisation and training. Finally, the set of XGBoost hyperparameters that were subjected to optimisation consists of the following:

learning rate (η), boundaries:[0.1,0.9], type: continuous,
 min_child_weight, boundaries:[0,10], type: continuous,
 sub sample, boundaries:[0.01,1], type: continuous,
 colsample, by tree, boundaries:[0.01,1], type: continuous,
 max_depth, boundaries:[3,10], type: integer and gamma,
 boundaries:[0,0.5], type: continuous

The number of parameters for the softprob objective function ('num_class': self. no classes) is also passed as a parameter to XGBoost. All other parameters are fixed and take default XGBoost values. All observed models were implemented in the Python programming language by employing standard machine learning libraries: scikit-learn, scipy, numpy, and pandas. For other models' hyperparameters, the default values from the scikit-learn Python library were used. The SVM and XGBoost models were adopted from scikit-learn; however, the ELM was coded from scratch because this model is not available in this package. In all implementations, a standard solution's encoding scheme was used. Every metaheuristic solution is represented as an array (vector) of size l , where l is the number of hyperparameters that were optimised. Therefore, the l for the SVM and XGBoost solutions are 3. However, in the case of ELM, l depends on the determined value for the nnhyper-parameter, and, if the fs denotes the size of the input feature vector, it can be derived as: $1 + nn \cdot fs + nn$. The first component of the ELM metaheuristic solution represents the number of neurons (integer); subsequent nn components are biases (continuous), while remaining $nn \cdot fs$ parameters are weights.

V.3 COMPARATIVE ANALYSIS AND DISCUSSION

In all experiments, the obtained outcomes of all three models optimised by the suggested GSFA metaheuristic were compared to the results generated by other well-known swarm algorithms that were also implemented for the SVM, ELM, and XGBoost model tuning under the same conditions, as described in Section 4.2. The competitive algorithms include the original FA [49], BA [2], ABC[42], since cosine algorithm SCA, WOA (Nematzadehetal.(2019)), and SNS etc. All metaheuristics were implemented independently in this research and adjusted (in terms of controls' parameters setup) as in the original publications that were mentioned beforehand. For the easier summary of results presented in tables, the machine learning model prefix is placed in front of the methods' names used for hyperparameter optimization, e.g., SVM-GSFA denotes the results obtained by SVM classifier tuned with the proposed GSFA method.

These simulations were executed with 20 solutions in the population ($N=20$) and 15 iterations in each run ($T=15$) for each metaheuristic method, except for the GSFA and FA. Since the FA in each iteration in the average case performs $2N$ solutions' evaluations, in this case N was set to 10. Additionally, due to the slightly higher complexity of the proposed GSFA over the basic FA, the GSFA was tested with only nine solutions. This reduction of the population for the FA and GSFA methods was required to set firm grounds for fair comparisons. Specific GSFA parameters (gsp, gssand cmt) were set according to Table 1. All swarm algorithms were also implemented in Python, and Intel®Core™i9-11900K Processor with 64GB of RAM and Windows 11 O.S. was used as a simulation platform. All employed datasets were relatively large, and the cache argument for SVM and XGBoost models in scikit-learn environment was set to 32GB to improve the computation speed.

On the other hand, the ELM is implemented by using cupy instead of the numpy library for operations with matrices, because cupy supports execution on GPU, and, in this case, NVIDIA GeForce GTX1080 GPU with 8 GB of memory is employed for such computations. Due to the stochastic nature of swarm approaches, all methods were executed in 50 independent runs, and best, worst, mean, and median accuracies, along with standard deviation, are recorded. However, accuracy may not be an objective metric, especially for imbalanced datasets; therefore, precision, recall, and f1-score per class and micro-averaged are also shown along with the receiver operating characteristic area under the curve (ROC AUC). It is noted that, in all result tables, the best-achieved metrics are denoted in bold style. Table 3 depicts the outcomes of the experiments with the SVM model employed.

As was already mentioned, the SVM model is specific due to its slow operation, and the reduced dataset was used. In this particular scenario, the SVM-GSFA achieved the best accuracy result, while the SVM-ABC obtained slightly better mean and median values. However, neither SVM-GSFA nor SVM-ABC exhibited the best robustness, which can be noticed from std metrics, while SVM-WOA, SVM-HHO and SVM-SCA did not show results variation over different runs. On the other hand, Table 4 shows the outcomes of the simulations with the ELM model against the dataset. In this scenario, the novel ELM-GSFA model outperformed all other hybrid ELM frameworks for all performance indicators, including best, worst, mean, and median classification accuracy, as well as the stability of results expressed with standard deviation. Table 5 puts forward the outcomes of the experiments with the XGBoost classifier on a real dataset. Similarly to the previous scenario, the XGBoost-GSFA achieved the best accuracy results for all utilised metrics and outperformed all other competitor methods Table6 presents the outcomes of the experiments with SVM model on the synthetic dataset. In this case, the SVM-GSFA and SVM WOA were tied for the best accuracy result (96.9519%), while the SVM-GSFA achieved better mean and median values, as well as better stability than opponent algorithms. For then extscenario, The XGBoost-GSFA and XGBoost-ABC were tied for the best and median accuracy results, while XGBoost-GSFA finished first for worst, mean and standard deviation indicators.

Table 3: General Metrics of SVM Credit Card Fraud.

Metrics	SVM-GSFA	SVM-FA	SVM-BA	SVM-ABC	SVM-SCA	SVM-MBO	SVM-HHO	SVM-EHO	SVM-WOA	SVM-SNS
best (%)	99.9552	99.9064	99.883	99.9532	99.9064	99.8361	99.9064	99.9064	99.9064	99.9064
worst (%)	99.9064	99.8596	99.8596	99.9064	99.9064	99.8127	99.9064	99.8127	99.9064	99.8596
mean (%)	99.922	99.8752	99.8674	99.9298	99.9064	99.8283	99.9064	99.8752	99.9064	99.8908
median (%)	99.9064	99.8596	99.8596	99.9298	99.9064	99.8361	99.9064	99.9064	99.9064	99.9064
std	0.00027	0.00027	0.000135	0.000234	0	0.000135	0	0.000541	0	0.00027

Source: Authors, (2026).

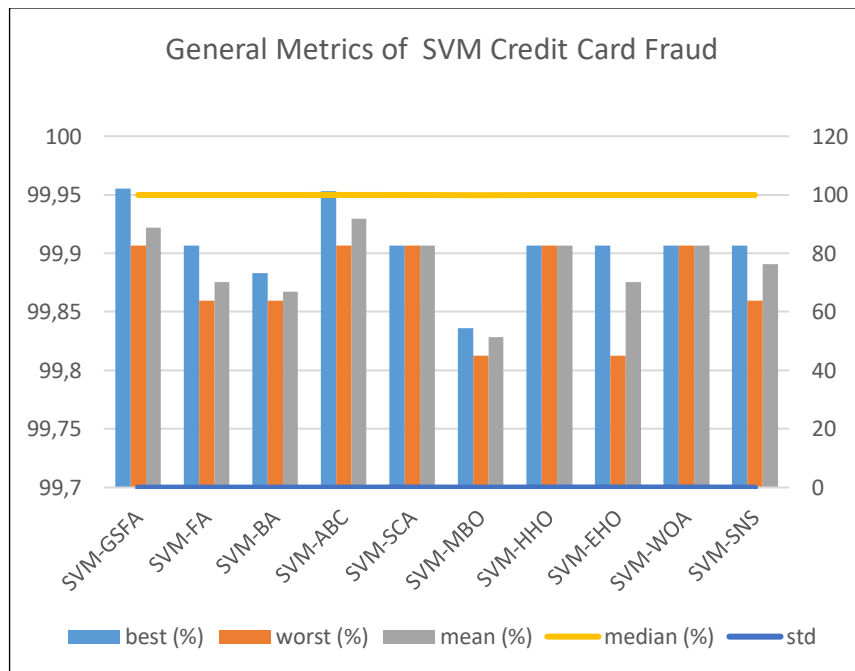


Figure 1: General Metrics of SVM Credit Card Fraud.

Source: Authors, (2026).

Table 4: General Metrics of ELM Credit Card Fraud.

Metrics	ELM-GSFA	ELM-FA	ELM-BA	ELM-ABC	ELM-SCA	ELM-MBO	ELM-HHO	ELM-EHO	ELM-WOA	ELM-SNS
best (%)	99.9462	99.9111	99.9345	99.9345	99.9169	99.9333	99.9181	99.9263	99.9192	99.9111
worst (%)	99.9427	99.8841	99.9134	99.8947	99.9029	99.8982	99.8947	99.9075	99.9075	99.8947
mean (%)	99.9442	99.8947	99.9207	99.9058	99.9105	99.9099	99.9081	99.916	99.916	99.9029
median (%)	99.9438	99.8917	99.9175	99.897	99.9111	99.904	99.9099	99.9151	99.9157	99.9029
std	0.000018	0.000123	0.000095	0.000192	0.000059	0.000163	0.000109	0.000077	0.000136	0.00007

Source: Authors, (2026).

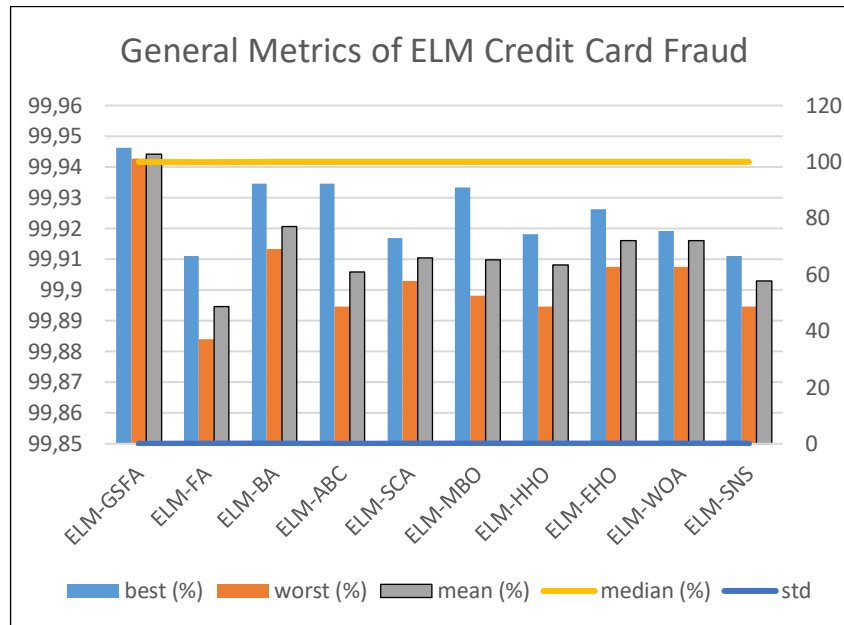


Figure 2: General Metrics of ELM Credit Card Fraud.

Source: Authors, (2026).

Table 5: General Metric of XGBoost Credit Card Fraud.

Metrics	XGBoost-GSFA	XGBoost-FA	XGBoost-BA	XGBoost-ABC	XGBoost-SCA	XGBoost-MBO	XGBoost-HHO	XGBoost-EHO	XGBoost-WOA	XGBoost-SNS
best (%)	99.9707	99.9684	99.9672	99.9684	99.9661	99.9672	99.9661	99.9672	99.9661	99.9661
worst (%)	99.9696	99.9649	99.9625	99.9661	99.9649	99.9649	99.9649	99.9649	99.9661	99.9637
mean (%)	99.9704	99.9668	99.9649	99.9668	99.9653	99.9664	99.9657	99.9657	99.9661	99.9649
median (%)	99.9707	99.9672	99.9649	99.9661	99.9649	99.9672	99.9661	99.9649	99.9661	99.9649
std	0.000007	0.000018	0.000023	0.000014	0.000007	0.000014	0.000007	0.000014	0	0.000012

Source: Authors, (2026).

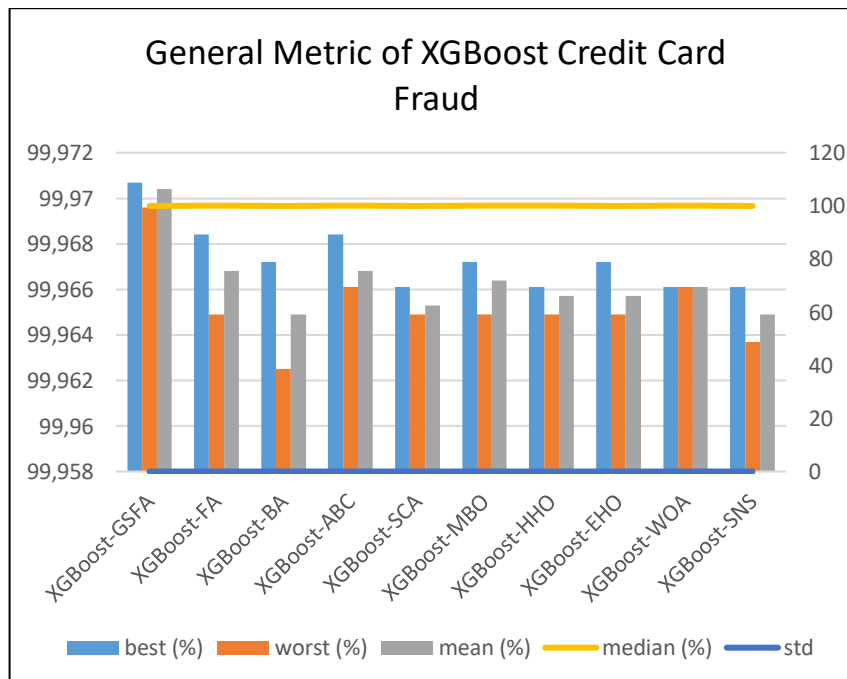


Figure 3: General Metric of XGBoost Credit Carrd.

Source: Authors, (2026).

Tables 3–5 provide additional detailed results for other significant machine learning performance metrics, including per-class and micro-averaged precision, recall, and F1Score, along with ROCAUC for the experiments where models were employed with a general metric technique. On the other hand, Tables 6–8 present the same metrics for the simulations where models were tested against the dataset with detailed metrics.

Table 6: SVM Credit Card Fraud detailed metrics.

Metaheuristic	Accuracy (%)	Precision 0	Precision 1	M.Avg. Precision	Recall 0	Recall 1	M.Avg. Recall	F1 Score 0	F1 Score 1	M.Avg. F1 Score	M.Avg. ROC AUC	M.Avg. PR AUC
SVM-GSFA	99.9532	0.999765	0.875	0.999532	0.999765	0.875	0.999532	0.999765	0.875	0.999532	1	1
SVM-FA	99.9064	0.999531	0.75	0.999064	0.999531	0.75	0.999064	0.999531	0.75	0.999064	1	1
SVM-BA	99.883	0.999063	0.8	0.99869	0.999765	0.5	0.99883	0.999414	0.615385	0.998695	1	1
SVM-ABC	99.9532	0.998251	0.002015	0.996385	0.535413	0.5	0.535346	0.696993	0.004014	0.695695	0.52	0.56
SVM-SCA	99.9064	0.999531	0.75	0.999064	0.999531	0.75	0.999064	0.999531	0.75	0.999064	1	1
SVM-MBO	99.8361	0.998127	0	0.996258	0.999765	0	0.997893	0.998946	0	0.997075	0.5	0.5
SVM-HHO	99.9064	0.999531	0.75	0.999064	0.999531	0.75	0.999064	0.999531	0.75	0.999064	1	1
SVM-EHO	99.9064	0.999297	0.833333	0.998986	0.999765	0.625	0.999064	0.999531	0.714286	0.998997	1	1
SVM-WOA	99.9064	0.999531	0.75	0.999064	0.999531	0.75	0.999064	0.999531	0.75	0.999064	1	1
SVM-SNS	99.9064	0.999531	0.75	0.999064	0.999531	0.75	0.999064	0.999531	0.75	0.999064	1	1

Source: Authors, (2026).

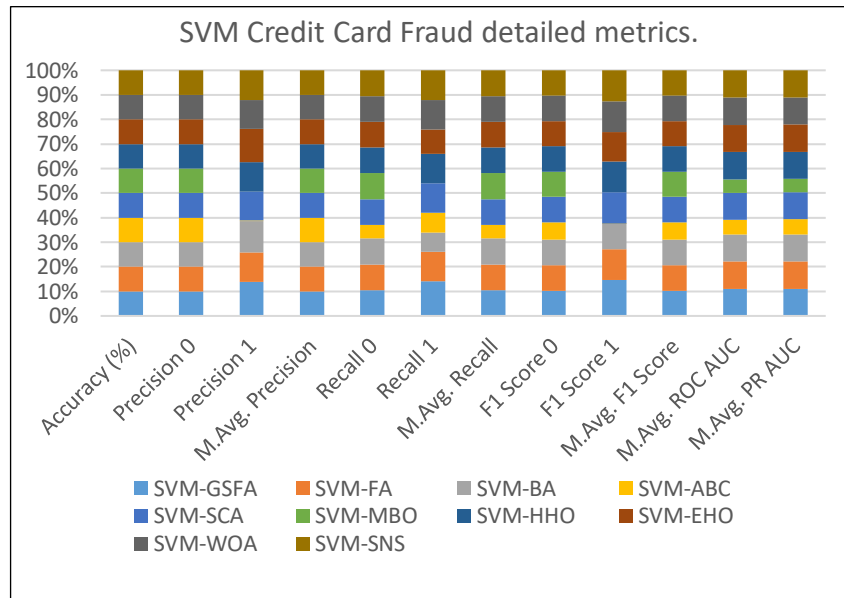


Figure 4: SVM Credit Card Fraud Detailed Metrics.

Source: Authors, (2026).

Table 7: ELM Credit Card Fraud detailed metrics.

Metaheuristic	Accuracy (%)	Precision 0	Precision 1	M.Avg. Precision	Recall 0	Recall 1	M.Avg. Recall	F1 Score 0	F1 Score 1	M.Avg. F1 Score	M.Avg. ROC AUC	M.Avg. PR AUC
ELM-GSFA	99.9462	0.999648	0.875	0.999441	0.999812	0.788732	0.999462	0.99973	0.82963	0.999448	1	1
ELM-FA	99.9111	0.999274	0.851064	0.999027	0.999836	0.56338	0.999111	0.999555	0.677966	0.99902	1	1
ELM-BA	99.9345	0.999637	0.816176	0.999332	0.999707	0.78169	0.999345	0.999672	0.798561	0.999338	1	1
ELM-ABC	99.9345	0.999555	0.852459	0.99931	0.999789	0.732394	0.999345	0.999672	0.787879	0.99932	1	1
ELM-SCA	99.9169	0.999332	0.858586	0.999098	0.999836	0.598592	0.999169	0.999584	0.705394	0.999095	1	1
ELM-MBO	99.9333	0.999578	0.834646	0.999304	0.999754	0.746479	0.999333	0.999666	0.788104	0.999314	1	1
ELM-HHO	99.9181	0.999402	0.827273	0.999116	0.999777	0.640845	0.999181	0.99959	0.722222	0.999129	1	1
ELM-EHO	99.9263	0.998338	0	0.996679	1	0	0.998338	0.999168	0	0.997508	0.5	0.5
ELM-WOA	99.9192	0.999391	0.841121	0.999128	0.999801	#####	0.999192	0.999596	0.722892	0.999136	1	1
ELM-SNS	99.9111	0.999274	0.851064	0.999027	0.999836	0.56338	0.999111	0.999555	0.677966	0.99902	1	1

Source: Authors, (2026).

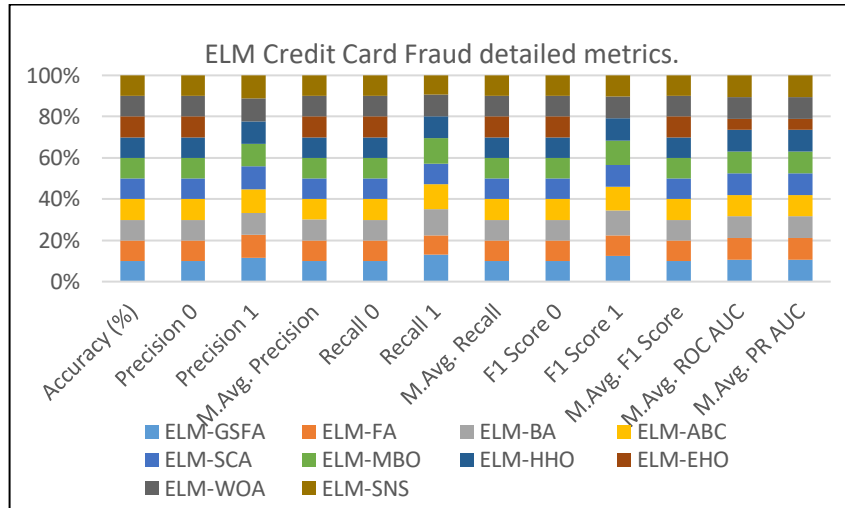


Figure 5: ELM Credit Card Fraud detailed metrics.

Source: Authors, (2026).

Table 8: XGBoostCreditCardFrauddetailedmetrics.

Metaheuristic	Accuracy (%)	Precision 0	Precision 1	M.Avg. Precision	Recall 0	Recall 1	M.Avg. Recall	F1 Score 0	F1 Score 1	M.Avg. F1 Score	M.Avg. ROC AUC
XGBoostGSFA	99.9707	0.999754	0.968	0.999701	0.999953	0.852113	0.999707	0.999853	0.906367	0.999698	1
XGBoost-FA	99.9684	0.999742	0.96	0.999676	0.999941	0.84507	0.999684	0.999842	0.898876	0.999674	1
XGBoost-BA	99.9672	0.99973	0.959677	0.999664	0.999941	0.838028	0.999672	0.999836	0.894737	0.999661	1
XGBoost-ABC	99.9684	0.999742	0.96	0.999676	0.999941	0.84507	0.999684	0.999842	0.898876	0.999674	1
XGBoost-SCA	99.9661	0.999719	0.95935	0.999652	0.999941	0.830986	0.999661	0.99983	0.890566	0.999648	1
XGBoost-MBO	99.9672	0.99973	0.959677	0.999664	0.999941	0.838028	0.999672	0.999836	0.894737	0.999661	1
XGBoost-HHO	99.9661	0.999719	0.95935	0.999652	0.999941	0.830986	0.999661	0.99983	0.890566	0.999648	1
XGBoost-EHO	99.9672	0.999742	0.952381	0.999663	0.99993	0.84507	0.999672	0.999836	0.895522	0.999663	1
XGBoost-WOA	99.9661	0.99973	0.952	0.999651	0.99993	0.838028	0.999661	0.99983	0.891386	0.99965	1
Boost-SNS	99.9661	0.999719	0.95935	0.999652	0.999941	0.830986	0.999661	0.99983	0.890566	0.999648	1

Source: Authors, (2026).

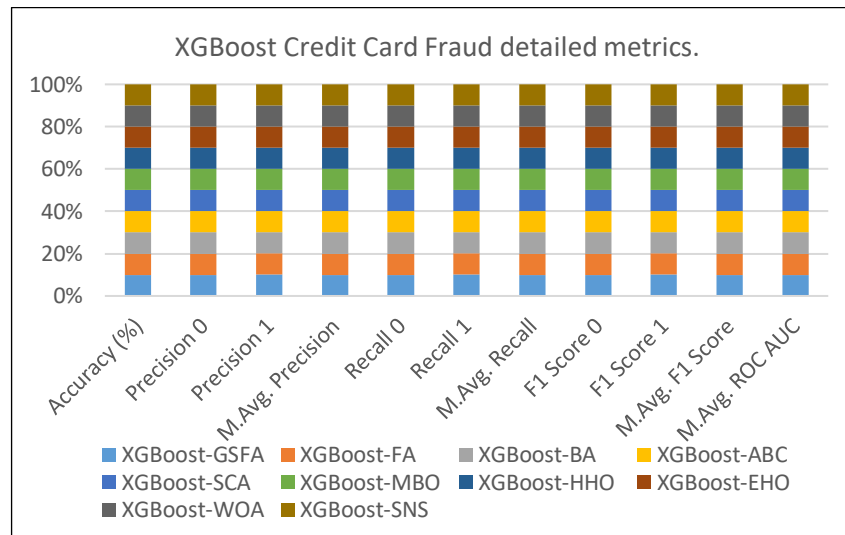


Figure 6: XGBoost Credit Card Fraud detailed metrics.

Source: Authors, (2026).

The yielded detailed metrics provided significant insights into the algorithm's performance, especially for imbalanced datasets such as credit card fraud. From the reported metrics, it can be unequivocally confirmed that, on average, when all models are taken into account, the proposed GSFA algorithm proved that it is able to achieve the best classification performance for a minority class (class 1 in this example). The generated confusion matrices for the original credit card fraud dataset by tuned SVM, ELM, and XGBoost models with GSFA, FA, and SNS algorithms are shown in Figure 5, while the precision-recall (PR) curve graphs for GSFA and arbitrary chosen approaches for the synthetic dataset are depicted in Figure. Lastly, the hyperparameters' values for the tuned SVM, ELM, and XGBoost models, generated in the best run for each applied optimization algorithm, are shown in Tables 3–5, respectively. It should be noted that Table 15 depicts just the number of neurons nn parameter that obtained the best results, as showing all weights and biases parameters would not be feasible.

V.4 STATISTICAL TESTS

Since the experimental outcomes are typically insufficient to establish that one algorithm outperforms its competitors, contemporary computer science practice requires researchers to determine whether the obtained improvements are statistically significant. In the research suggested in this manuscript, 10 methods (including proposed GSFA) were validated for SVM, ELM, and XGBoost tuning against the original (highly imbalanced) and synthetic, generated using the SMOTE technique (balanced) on credit card fraud datasets. Therefore, 10 methods were compared against 6 problem instances, which fall into the domain of multiple approaches for multi-problem analysis. According to the recommendations from the literature [65–67], statistical tests in such scenarios may be conducted by constructing a results sample for each approach by taking averages of the measured objectives over multiple independent runs for each problem. However, this approach may have disadvantages when the measured variable has outliers that do not follow a normal distribution, which may lead to deceptive conclusions. According to a literature survey, whether the average objective function value should be taken for statistical tests when comparing stochastic methods still remains an open question [55]. For the purpose of comparing 10 methods for 6 problem instances, despite the above-noted potential disadvantages, the objective function (classification error rate) was averaged over 50 independent runs and used in statistical tests. However, this decision was rendered based on the conducted Shapiro–Wilk [39] test for single-problem analysis [55] in the following way: for each algorithm and every problem, a data sample is constructed by taking the results obtained in each run, and respective p-values are calculated for every method–problem pair. The obtained p-values for this test are shown in the Table.

Table 9: Shapiro–results for single-problem analysis.

Methods										
Problem	GSFA	FA	BA	ABC	SCA	MBO	HHO	EHO	WOA	SNS
SVM	6.15×10^{-1}	4.24×10^{-1}	7.05×10^{-1}	4.37×10^{-1}	8.29×10^{-1}	6.02×10^{-1}	7.40×10^{-1}	4.40×10^{-1}	1.23×10^{-1}	4.23×10^{-1}
SVM Smote	8.06×10^{-1}	7.49×10^{-1}	5.82×10^{-1}	4.92×10^{-1}	3.05×10^{-1}	5.93×10^{-1}	8.08×10^{-1}	3.63×10^{-1}	3.33×10^{-1}	4.51×10^{-1}
ELM	6.37×10^{-1}	5.32×10^{-1}	6.52×10^{-1}	5.30×10^{-1}	8.42×10^{-2}	6.05×10^{-1}	5.55×10^{-1}	2.84×10^{-1}	2.85×10^{-1}	8.52×10^{-2}
ELM Smote	1.56×10^{-1}	9.46×10^{-2}	4.52×10^{-1}	6.95×10^{-1}	1.34×10^{-1}	1.12×10^{-1}	8.58×10^{-2}	3.50×10^{-1}	7.92×10^{-1}	3.10×10^{-1}
XGB	7.98×10^{-1}	4.24×10^{-1}	9.52×10^{-2}	7.35×10^{-1}	7.29×10^{-1}	7.49×10^{-1}	5.32×10^{-1}	6.93×10^{-1}	6.07×10^{-1}	7.43×10^{-1}
XGB Smote	3.94×10^{-1}	3.84×10^{-1}	1.75×10^{-1}	5.23×10^{-1}	4.01×10^{-1}	5.69×10^{-1}	6.66×10^{-1}	7.24×10^{-1}	5.21×10^{-1}	5.62×10^{-1}

Source: Authors, (2026).

The results from Table 9 indicate that all p-values are higher than the threshold significance level $\alpha = 0.05$, yielding the conclusion that the null hypothesis cannot be rejected; therefore, the data samples for all method–problem pairs originate from a normal distribution, and it is safe to use the average objective in the statistical tests. From this point, we proceeded with multi-problem multiple methods statistical analysis, and the data sample for each method was constructed by taking the average objective function value over 50 independent runs for each problem instance. First, the safe use of the parametric tests conditions, which include independence, normality, and homoscedasticity of the variances of the data, was checked [11]. The condition of independence was satisfied because each run was executed independently, starting with a unique pseudo-random number seed. To check normality, the Shapiro–Wilk test [39] was used again. The results for every method are reported in Table 10.

Table 10: Shapiro–Analysis.

Methods										
	GSFA	FA	BA	ABC	SCA	MBO	HHO	EHO	WOA	SNS
p-value	3.20×10^{-3}	8.34×10^{-3}	9.71×10^{-3}	6.81×10^{-3}	3.28×10^{-3}	9.19×10^{-3}	8.90×10^{-3}	2.32×10^{-3}	4.58×10^{-3}	8.34×10^{-3}

Source: Authors, (2026).

To check homoscedasticity based on means, Levene’s test [10] is employed, and the p-value of 0.64 is obtained, which follows that the homoscedasticity is satisfied. However, the rendered p-values from the Shapiro–Wilk test for all methods are smaller than $\alpha = 0.05$ (Table 18), yielding the conclusion that the safe use of parametric tests is not satisfied, and we proceeded with non-parametric tests. In all non-parametric tests, the proposed GSFA was established as the control method.

VI. CONCLUSIONS

The research presented in this paper proposed a novel variant of the famous FA algorithm, named GSFA metaheuristic, with the aim of addressing the notable drawbacks of the original implementation. The introduced GSFA approach adopts a disputation operator from the recently proposed SNS metaheuristic. The suggested GSFA was paired with three standard ML models, namely the SVM, ELM, and XGBoost models, to perform hyper-parameter optimisation. This challenge is still an open question in the ML domain, because every ML model has to be tuned for a specific dataset. The suggested hybrid model’s performance was verified against the credit card fraud detection dataset, which includes transactions gathered across Europe in 2013. The employed dataset is highly disproportionate, as the majority of entries represent valid transactions, and just a small portion mark the fraudulent actions.

The performances of the proposed GSFA-optimised SVM, ELM, and XGBoost were compared to nine other metaheuristic-optimised variants of ML models. The competitor metaheuristics encompassed well-known algorithms, such as the original implementation of FA, BA, ABC, SCA, MBO, HHO, EHO, WOA, and SNS. The experiments were conducted in two phases: first, with the original imbalanced dataset. The outcomes of the executed simulations indicate the superior performance of proposed GSFA for most of the test instances. Finally, rigorous statistical tests were performed to confirm the significance of the obtained test results. Therefore, in the proposed research, it was shown that the FA metaheuristic can be further improved and that the tuned SVM, ELM, and XGBoost models achieve decent performance for a highly challenging and important credit card fraud dataset. However, the proposed study also has some limitations.

First, the GSFA metaheuristic employs three additional control parameters that need to be adjusted for each particular NP-hard challenge. Moreover, the performance of the algorithm needs to be further evaluated for other NP-hard challenges. The future trials of the suggested GSFA algorithm will include further testing on supplementary real-life credit card datasets. Another direction for validating the proposed GSFA is to apply it and test it on other NP-hard problems, falling into the domains of cloud computing, cryptocurrency forecasting, and image processing and classification.

VII. AUTHOR'S CONTRIBUTION

Conceptualisation: Ravi Bhushan.

Methodology: Ravi Bhushan.

Investigation: Ravi Bhushan.

Discussion of results: Ravi Bhushan.

Writing: Ravi Bhushan.

Writing Review and Editing: Ravi Bhushan.

Resources: Ravi Bhushan.

Supervision: Vineeta Khemchandani.

Approval of the final text: Vineeta Khemchandani, Ravi Bhusha.

VIII. REFERENCES

- [1] Liu, Q., Gu, Q., Wu, Z., 2017. Feature selection method based on support vector machine and shape analysis for high-throughput medical data. *Comput. Biol. Med.* 91, 103–111.
- [2] Liu, Y., et al., 2019a. A classification method based on feature selection for imbalanced data. *IEEE Access* 7, 81794–81807.
- [3] Yuan, X., et al., 2020. Integrated long-term stock selection models based on feature selection and machine learning algorithms for China Stock Market. *IEEE Access* 8, 22672–22685.
- [4] Liang, Y., Zhang, M., Browne, W. N., 2017. Image feature selection using genetic programming for figure-ground segmentation. *Eng. Appl. Artif. Intell.* 62, 96–108.
- [5] Zhou, X., et al., 2017. Eye tracking data guided feature selection for image classification. *Pattern Recognit.* 63, 56–70.
- [6] Wang, H., Hong, M., 2019. Supervised hebb rule based feature selection for text classification. *Inf. Process. Manage.* 56 (1), 167–191.
- [7] Kou, G., et al., 2020. Evaluation of feature selection methods for text classification with small datasets using multiple criteria decision-making methods. *Appl. Soft Comput.* 86, 105836.
- [8] Koide, K., Miura, J., Menegatti, E., 2020. Monocular person tracking and identification with on-line deep feature selection for person following robots. *Robot. Auton. Syst.* 124, 103348.
- [9] Tu, Q., Chen, X., Liu, X., 2019. Multi-strategy ensemble grey wolf optimizer and its application to feature selection. *Appl. Soft Comput.* 76, 16–30.
- [10] Tang, X., Dai, Y., Xiang, Y., 2019. Feature selection based on feature interactions with application to text categorization. *Expert Syst. Appl.* 120, 207–216.
- [11] Singh, U., Singh, S. N., 2019. A new optimal feature selection scheme for classification of power quality disturbances based on ant colony framework. *Appl. Soft Comput.* 74, 216–225.
- [12] Liu, Y., et al., 2019b. Flexible unsupervised feature extraction for image classification. *Neural Netw.* 115, 65–71.
- [13] Nguyen, B. H., Xue, B., Zhang, M., 2020. A survey on swarm intelligence approaches to feature selection in data mining. *Swarm Evol. Comput.* 54, 100663.
- [14] Nayar, N., Ahuja, S., Jain, S., 2019. Swarm intelligence for feature selection: A review of literature and reflection on future challenges. In: *Advances in Data and Information Sciences*. Springer Singapore, Singapore.
- [15] Rostami, M., Berahmand, K., Forouzandeh, S., 2020a. A novel method of constrained feature selection by the measurement of pairwise constraints uncertainty. *J. Big Data* 7 (1), 83.
- [16] Shu, W., Qian, W., Xie, Y., 2020. Incremental feature selection for dynamic hybrid data using neighbourhood rough set. *Knowl.-Based Syst.* 105516.
- [17] Gokalp, O., Tasci, E., Ugur, A., 2020. A novel wrapper feature selection algorithm based on iterated greedy metaheuristic for sentiment classification. *Expert Syst. Appl.* 146, 113176.
- [18] Renuka Devi, D., Sasikala, S., 2019. Online Feature Selection (OFS) with Accelerated Bat Algorithm (ABA) and Ensemble Incremental Deep Multiple Layer Perceptron (EIDMLP) for big data streams. *J. Big Data* 6 (1), 103.
- [19] Tang, C., et al., 2018. Robust unsupervised feature selection via dual self-representation and manifold regularization. *Knowl.-Based Syst.* 145, 109–120.
- [20] Zhang, H., et al., 2019b. An efficient framework for unsupervised feature selection. *Neurocomputing* 366, 194–207.
- [21] Chen, R.-C., et al., 2020. Selecting critical features for data classification based on machine learning methods. *J. Big Data* 7 (1), 52.
- [22] Senawi, A., Wei, H.-L., Billings, S. A., 2017. A new maximum relevance-minimum multicollinearity (MRmMC) method for feature selection and ranking. *Pattern Recognit.* 67, 47–61.
- [23] Martín-Smith, P., et al., 2017. A supervised filter method for multi-objective feature selection in EEG classification based on multi-resolution analysis for BCI. *Neurocomputing* 250, 45–56.
- [24] Li, A.-D., Xue, B., Zhang, M., 2020a. Multi-objective feature selection using hybridization of a genetic algorithm and direct multi-search for key quality characteristic selection. *Inform. Sci.* 523, 245–265.

- [25] Forouzandeh,S., Berahmand, K., Rostami, M., 2020. Presentation of a recommendersystem with ensemble learning and graph embedding: A case on MovieLens.Multimedia Tools Appl..
- [26] Labani,M., et al., 2018. A novel multivariate filter method for feature selection in textclassification problems. Eng. Appl. Artif. Intell. 70, 25–37.
- [27] Jain,I., Jain, V.K., Jain, R., 2018. Correlation feature selection based improved-binaryparticle swarm optimization for gene selection and cancer classification. Appl. SoftComput. 62, 203–215.
- [28] Zhang,T.,etal.,2018a.Afastfeatureselectionalgorithmbasedonswarmintelligencein acoustic defect detection. IEEE Access 6, 28848–28858.
- [29] Qasim,O.S., Algamal, Z.Y., 2018. Feature selection using particle swarm optimization-basedlogisticregressionmodel.Chemometr.Intell.Lab.Syst.182,41–46.
- [30] Yan,C.,etal.,2019.Anovelhybridfeatureselectionstrategyinquantitativeanalysisoflaser-inducedbreakdownspectroscopy.Anal.Chim.Acta1080,35–42.
- [31] Gong,D., et al., 2020. A similarity-based cooperative co-evolutionary algorithm fordynamic interval multiobjective optimization problems. IEEE Trans. Evol. Comput.24 (1), 142–156.
- [32] Li,J., et al., 2020b. An enhanced-indicator based many-objective evolutionary algorithmwith adaptive reference point. Swarm Evol. Comput. 55, 100669.
- [33] Santucci,V., Bairoletti, M., Milani, A., 2020. An algebraic framework for swarm andevolutionary algorithms in combinatorial optimization. Swarm Evol. Comput. 55,100673.
- [34] Ghaemi,M., Feizi-Derakhshi, M.-R., 2016. Feature selection using forest optimizationalgorithm. Pattern Recognit. 60, 121–129.
- [35] Hashemi,A., Dowlatshahi, M.B., Nezamabadi-pour, H., 2020. MGFS: A multi-labelgraph-based feature selection algorithm via PageRank centrality. Expert Syst. Appl.142, 113024.
- [36] Li,S., et al., 2019a. Dual graph regularized compact feature representation forunsupervisedfeatureselection.Neurocomputing331,77–96.
- [37] Prabukumar,M., Agilandeewari, L., Ganesan, K., 2019. An intelligent lung cancerdiagnosis system using cuckoo search optimization and support vector machineclassifier.J.AmbientIntell.Humaniz.Comput.10(1),267–293.
- [38] Rostami,M.,etal.,2020b. Integration of multi-objective PSO-based feature selection and node centrality for medical datasets.Genomics112(6),4370–4384.
- [39] Shunmugapriya,P., Kanmani, S., 2017. A hybrid algorithm using ant and bee colony optimisation for feature selection and classification (AC-ABC Hybrid). Swarm Evol.Comput. 36, 27–36.
- [40] Hancer,E., et al., 2018b. Pareto front feature selection based on artificial bee colonyoptimization. Inform. Sci. 422, 462–479.
- [41] Zhang,Y.,etal.,2019a.Cost-sensitive feature selection using two-archival multi-Objective artificial bee colony algorithm. Expert Syst.Appl.137,46–58.
- [42] Wang,X.-h.,etal.,2020b. Multi-objective feature selection based on artificial bee colony:Anaccelerationapproachwithvariablesamplesize.Appl.SoftComput.88, 106041.
- [43] Hancer,E., Xue, B., Zhang, M., 2018a. Differential evolution for filter feature selection based on information theory and feature ranking. Knowl.-Based Syst. 140, 103–119.
- [44] Zhang,Y.,etal., 2020. Binary differentiable evolution with self-learning for multi-objective feature selection.Inform.Sci.507,67–85.
- [45] Hancer,E., 2020. A new multi-objective differential evolution approach for simultaneous clustering and feature selection.Eng.Appl.Artif.Intell.87,103307.
- [46] Taradeh,M.,etal.,2019.Anevolutionarygravitationalsearch-basedfeatureselection.Inform.Sci.497,219–239.
- [47] Zhang,Y., Song, X.-f., Gong, D.-w., 2017b. A return-cost-based binary firefly algorithmforfeatureselection.Inform.Sci.418–419,561–574.
- [48] Zhang,L.,etal.,2018b. Feature selection using firefly optimisation for classification and regression models.Decis.SupportSyst.106,64–85.
- [49] Liu,F., Yan, X., Lu, Y., 2020. Feature selection for image steganalysis using binary batalgorithm. IEEE Access 8, 4244–4249.
- [50] Tawhid,M.A., Dsouza, K.B., 2018. Hybrid binary bat enhanced particle swarmoptimizationalgorithmforsolvingfeatureselectionproblems.Appl.Comput.Inform..
- [51] Prabu kumar, M., Agilandeewari, L., Ganesan, K., 2019. An intelligent lung cancerdiagnosis system using cuckoo search optimization and support vector machineclassifier.J.AmbientIntell.Humaniz.Comput.10(1),267–293.
- [52] AbdelBasset,M.,etal.,2020.Anewfusionofgreylwolfoptimizeralgorithmwithatwo-phasemutationforfeatureselection.Expert Syst.Appl.139,112824.
- [53] Al-Zoubi,A.M., et al., 2020. Salp chain-based optimization of support vector machines and feature weighting for medical diagnostic information systems. In: Mirjalili, S.,Faris, H., Aljarah, I. (Eds.), Evolutionary Machine Learning Techniques: Algorithms and Applications.SpringerSingapore, Singapore,pp.11–34.
- [54] Neggaz,N., et al., 2020. Boosting salp swarm algorithm by sine cosine algorithm anddisruptoperatorforfeatureselection.ExpertSyst.Appl.145,113103.
- [55] Al-Betar,M.A., Alomari, O.A., Abu-Romman, S.M.,2020.ATRIZ-inspiredbatalgorithmforgenselectionincancerclassification. Genomics112(1),114–126.
- [56] Selva kumar,B., Muneeswaran, K., 2019. Firefly algorithm based feature selection for network intrusion detection.Comput.Secur.81,148–155.