

IMPROVING SVM CLASSIFICATION PERFORMANCE WITH DRACO LIZARD FEATURE SELECTION OPTIMIZER

İbrahim Berkan Aydılek^{1✉}, İsmail Açar²

Received on April 18, 2025

Presented by S. Hadjitodorov, Corresponding Member of BAS, on June 24, 2025

Abstract

Feature selection plays a vital role in the machine learning process by eliminating redundant and irrelevant attributes from datasets. This not only enhances model accuracy but also reduces training time. When applied to high-dimensional datasets, feature selection helps mitigate the risk of overfitting, thereby enabling the development of models with stronger generalization capabilities. In this study, the recently proposed Draco Lizard Optimizer algorithm is employed for feature selection in conjunction with Support Vector Machines to improve classification performance. The optimization process utilizes an effective fitness function, and experimental evaluations are conducted accordingly. The proposed method is assessed using six different datasets obtained from the UCI Machine Learning Repository. Average number of selected features and classification performance metrics are analyzed in terms of best, worst, average, and standard deviation values. The results demonstrate that the proposed approach yields effective and robust outcomes in feature selection tasks.

Key words: feature selection, classification, draco lizard optimizer, SVM

Introduction. Dimensionality reduction is a fundamental preprocessing technique frequently employed in high-dimensional data analysis, visualization, and modelling tasks. A high number of dimensions in a dataset can increase computational complexity, reduce processing speed, and adversely affect the performance of machine learning algorithms. This issue becomes particularly pronounced when working with datasets that contain a large number of features.

<https://doi.org/10.7546/CRABS.2025.07.08>

Among the most common approaches to dimensionality reduction is feature selection, which focuses on identifying and retaining only the relevant features that contain the necessary information for solving a given problem [1]. Feature selection improves the performance of machine learning algorithms by eliminating redundant and irrelevant features, thereby reducing training time and enhancing model interpretability. Furthermore, it mitigates the risk of overfitting, enabling the development of models with better generalization capabilities. This process can be formulated as an optimization problem, where the goal is to identify the smallest possible subset of relevant features that contributes most significantly to learning accuracy [2].

In this study, experimental investigations were conducted on the use of the Draco Lizard Optimization algorithm – which simulates the survival behaviour of Draco lizards – for feature selection. Support Vector Machines were employed to evaluate classification performance. During the optimization process, an efficient fitness function was utilized to assess the performance of the selected feature subsets [3]. Experimental evaluations were carried out on six datasets obtained from the UCI Machine Learning Repository using 5-fold cross-validation, and the results were systematically compared.

Literature review. In the study conducted by ONAY [4], the Shark Algorithm was employed for feature selection using S, V, and U shaped transfer functions. The proposed approach was applied to eight datasets obtained from the UCI Machine Learning Repository, utilizing the k-Nearest Neighbours classifier. The method was evaluated in terms of classification accuracy, fitness values, and the number of selected features. Comparative experimental analyses were performed by benchmarking the proposed methods against various other metaheuristic algorithms. The results of these experiments indicate that the proposed approach achieved effective performance in feature selection and contributed to improved classification accuracy. ALKANHEL et al. [5] combined the Grey Wolf Optimizer and the Dipper Throated Optimization algorithms to propose a new hybrid method called GWDTO. The proposed approach was applied to the RPL-NIDDS17 dataset and evaluated using the k-Nearest Neighbours (k-NN) classifier. Experimental comparisons were conducted against eight different metaheuristic algorithms. The results demonstrated that the proposed GWDTO method outperformed the compared approaches, yielding superior results in the feature selection task.

In the study conducted by KARLUPIA and ABROL [6], a wrapper-based feature selection approach was proposed to reduce the impact of redundant and irrelevant features in high-dimensional data, utilizing five different nature-inspired algorithms. The method was evaluated on eleven datasets obtained from the UCI Machine Learning Repository, using the k-Nearest Neighbours classifier to compute classification accuracy. The experimental results involved comparisons based on metrics such as accuracy, the number of selected features, and feature reduc-

tion rate. Statistical analyses, including the Friedman ranking and Holm test, revealed that the Whale Optimization Algorithm (WOA) outperformed the other methods. In particular, WOA demonstrated the highest feature reduction rates across multiple datasets while simultaneously enhancing classification accuracy. ZHU et al. [7] proposed a hybrid feature selection method based on artificial immune algorithm (HFSIA) optimization for high-dimensional datasets. The HFSIA algorithm was tested on 22 high-dimensional datasets and compared against 23 different feature selection methods. The experimental results demonstrated that the HFSIA algorithm is both effective and computationally efficient in addressing the feature selection problem in high-dimensional data.

In the study conducted by KWAKYE et al. [8], a novel method called Particle Swarm-guided Bald Eagle Search (PS-BES) was proposed by combining the exploration capabilities of the Particle Swarm Optimization (PSO) algorithm with the exploitation strengths of the Bald Eagle Search algorithm. The performance of PS-BES was first evaluated using 26 mathematical benchmark functions. Subsequently, the proposed method was applied to 27 different datasets from the UCI Machine Learning Repository and compared against 10 other metaheuristic algorithms. The experimental results indicated that the proposed PS-BES method outperformed the compared approaches, achieving superior results in feature selection tasks. MAHAPATRA et al. [9] offered a hybrid algorithm named the Quantized Salp Swarm Algorithm (QSSA), which is an enhanced version of the Salp Swarm Algorithm (SSA). The proposed method was applied to eight different datasets and compared with five other metaheuristic algorithms. The experimental results demonstrated that QSSA achieved superior performance compared to the other methods, selecting fewer features while simultaneously attaining higher classification accuracy.

KHODADADI et al. [10] studied a binary version of the Arithmetic Optimization Algorithm (AOA) to address the feature selection problem. The proposed method, referred to as BAOA, was used in conjunction with the k-Nearest Neighbours (k-NN) classifier. It was applied to 18 different datasets and compared against four other metaheuristic algorithms. The experimental results showed that BAOA achieved the highest classification accuracy on 12 out of the 18 datasets and consistently selected the smallest number of features across all experimental cases. In the study conducted by CİNAR [11], a novel Adaptive Memetic Binary Optimization algorithm (AMBO) was proposed to address the feature selection problem. AMBO was designed to adaptively balance exploration and exploitation, incorporating logic gate-based local search mechanisms to enhance its performance. The algorithm was evaluated on 21 different datasets from the UCI Machine Learning Repository and compared against eight other metaheuristic algorithms. Experimental results demonstrated that AMBO outperformed the competing methods in terms of classification accuracy, fitness function values, and the number of selected features.

Support Vector Machines (SVM). Support Vector Machines (SVM) are machine learning algorithms originally developed to solve binary classification problems. SVM maps input vectors into a high-dimensional feature space through a nonlinear transformation and constructs a linear decision surface within this space. This decision surface is defined as a hyperplane that separates the classes from each other. The support vectors are the data points that define this hyperplane and determine the decision boundary [12].

To separate the training set with minimal error, equation (1) is utilized. Equation (1) is the objective function that SVM tries to minimize. This function is minimized subject to the constraints given in equations (2) and (3) [12].

$$(1) \quad \frac{1}{2} \|w\|^2 + C \sum_{i=1}^l \xi$$

$$(2) \quad y_i (w \cdot x_i + b) \geq 1 - \xi, \quad i = 1, \dots, l$$

$$(3) \quad \xi \geq 0, \quad i = 1, \dots, l,$$

where w represents the weight vector. The term $\|w\|^2$ represents the margin that SVM aims to maximize. The value C is the regularization parameter. A smaller C encourages a larger margin by allowing some misclassifications (more tolerance for errors). A larger C penalizes misclassifications more heavily, leading to a smaller margin. The expression ξ (epsilon) denotes slack variables, which allow for some data points to be misclassified. l is the number of data points. Equation (2) ensures data points are classified correctly. y_i is the class label, x_i is the feature vector of i -th data point and b is the bias term. Equation (3) states that slack variables should not be negative.

Support Vector Machines can yield effective results, particularly in small and high-dimensional datasets. Moreover, thanks to kernel functions, SVMs can be successfully applied to nonlinear classification problems. However, the training process can be time-consuming for large datasets, and careful tuning of the hyperparameter C is essential for optimal performance.

Draco Lizard Optimizer (DLO). The Draco Lizard Optimizer (DLO) is a metaheuristic optimization algorithm developed by Xiaowei Wang in 2025. It is inspired by the defensive and hunting behaviours of Draco lizards in nature. They are capable of gliding between trees, which allows them to conserve energy and adopt effective strategies during hunting. By modelling these natural behaviours, the DLO algorithm aims to search the solution space efficiently while balancing global and local optimization capabilities [13].

In the DLO algorithm, each individual represents a Draco lizard. In the initial step, the individuals' starting positions in the solution space are randomly

assigned. These individuals are then evaluated using a predefined fitness function. In the second step, the fitness values are calculated, and the best-performing individuals are identified. These individuals are recorded as both personal best and global best solutions. In the third phase, the positions and directions of individuals are updated. This update process enables the individuals to move toward both their personal best and the global best solutions. Through this mechanism, the algorithm effectively balances exploration and exploitation, improving its search performance. The movement and position updates of the individuals are modelled using mathematical equations that simulate the gliding behaviour of Draco lizards. Additionally, stochastic components are introduced to enhance population diversity and avoid premature convergence to local optima. Finally, once the stopping criterion is met, the algorithm terminates and returns the best solution found [13].

Proposed SVM and Draco Lizard Feature Selection Optimizer (SVM-DLOFS) classification model. The proposed method in this study aims to enhance classification performance by eliminating redundant and irrelevant features in high-dimensional datasets. The strong global and local search capabilities of the Draco Lizard Optimizer (DLO) are integrated with Support Vector Machine (SVM) to provide an effective classification. The flowchart of the proposed method is presented in Fig. 1. The first step of the proposed SVM-DLOFS algorithm involves the initialization of the population. In this step, the population size is set to 50, and each individual is generated to represent a random subset of features. Individuals consist of values ranging between $[0, 1]$, with each value corresponding to a specific feature. These values are initially assigned randomly within the range $[0, 1]$. Based on the optimization outcome, a threshold of 0.5 is applied to determine whether a feature is selected: values greater than 0.5 indicate feature selection, while values below this threshold indicate exclusion.

During the feature selection process, the fitness function used for the DLO algorithm is designed to optimize both classification accuracy and the number of selected features. The fitness function is defined in Equation (4) [3].

$$(4) \quad \text{Fitness} = y1.err + y2. \frac{|R|}{|N|}$$

In this context, *err* represents the classification error obtained using 5-fold cross-validation with the SVM classifier, $|R|$ denotes the number of selected features, and $|N|$ indicates the total number of available features. The coefficients $y1$ and $y2$ are weighting parameters used to balance classification accuracy and feature subset size. In this study, the values of $y1$ and $y2$ are set to 0.9 and 0.1, respectively. In the subsequent step, the best solutions in the population are identified based on their fitness values. Each individual updates its own personal best solution, and the overall global best solution is also updated accordingly. Then, the core mechanism of the DLO algorithm – position and direction updates – is

executed. In this step, individuals' feature subsets are updated using Lévy flight and exploration-exploitation strategies, allowing individuals to search for better solutions within the solution space and thereby enhancing the optimization capability of the algorithm. Following this, a new population is formed based on the updated individuals. This new population is created using the updated positions and fitness values of the individuals. During this process, each individual's feature subset is re-evaluated based on its newly updated position derived from the Lévy flight and exploration-exploitation mechanisms. The fitness values of the updated individuals are recalculated, and the best solutions are once again identified. This updated population serves as the input for the next iteration, allowing the optimization process to continue. The algorithm runs until 250 maximum iterations are reached, at which the process terminates and the optimal feature subset is obtained. To ensure the robustness of the results, the proposed method was executed 10 times for each dataset, and the outcomes were compared accordingly.

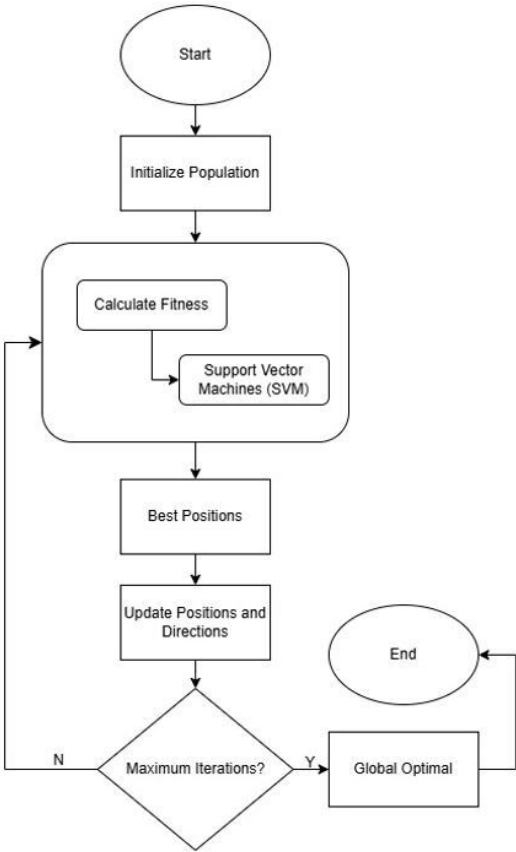


Fig. 1. Schematic flowchart of proposed classification model

Datasets. In this study, experimental studies were conducted on six different datasets obtained from the UCI Machine Learning Repository. Sonar has 61 features, 208 records, and 2 classes. Turkish Music Emotion has 51 features, 400 records, and 4 classes. Spectf Heart has 45 features, 80 records, and 2 classes. Spect Heart has 23 features, 79 records, and 2 classes. Libras Movement has 91 features, 359 records, and 15 classes. Qsar Biodegradation has 45 features, 1055 records, and 2 classes.

Results and discussions. In the experimental studies, the performance of the proposed SVM-Draco Lizard Optimizer Feature Selection (SVM-DLOFS) method was compared with Support Vector Machines (SVM). Classification accuracy, recall, precision, and F1-score performance metrics were calculated for each dataset. The results are presented in Table 1, which includes the best, mean, worst, standard deviation (std.) and average number of selected features values which are obtained with 10 independent runs. Performance metrics are scaled between zero and one.

For the Sonar dataset, the proposed method increased the average accuracy to 83.19%, compared to 79.64% with SVM. Similarly, improvements were observed in both recall and F1-score. Additionally, the average number of selected features was significantly reduced from 61 to 12.9, leading to a more compact model. In the Turkish Music Emotion dataset, the proposed method achieved improvements across all four performance metrics. The average number of selected features was reduced from 51 to 13.2, significantly simplifying the model. For the Spectf Heart dataset, while the proposed method improved best values performance in all four metrics, it demonstrated superior performance in the average F1-score only. However, the average number of selected features was significantly decreased from 45 to 8.1. In both the Spect Heart and Qsar Biodegradation datasets, the proposed approach achieved improvements across all four performance metrics. Furthermore, the average number of selected features was substantially reduced, contributing to a simpler and more efficient model structure. For the Libras Movement dataset, the proposed method enhanced best values in all metrics, while in terms of average values, it outperformed SVM in F1-score and precision. Moreover, the average number of selected features dropped from 91 to 18.8, significantly reducing the model's dimensionality.

Considering the best and worst values of Table 1, it is seen that the SVM-DLOFS method generally performs better than the classical SVM. Across all datasets, SVM-DLOFS best results achieved higher Accuracy, Recall and F1-Score values compared to SVM. The mean worst performance metric values of SVM-DLOFS for Turkish Music Emotion dataset is 54.79%, while for SVM it is approximately 41.99%. In the Sonar dataset, the SVM-DLOFS method outperformed SVM's by 83.64% with an accuracy of 90.48% for the best, while the worst, SVM-DLOFS showed a close performance to SVM's 77.58% with 76.19%. Sim-

T a b l e 1

Classification results of proposed SVM-DLOFS and SVM

		Accuracy		Recall		Precision		F1-Score	
		SVM	SVM-DLOFS	SVM	SVM-DLOFS	SVM	SVM-DLOFS	SVM	SVM-DLOFS
Sonar	Best	0.8364	0.9048	0.7801	0.9048	0.9230	0.9213	0.8076	0.9062
	Mean	0.7964	0.8319	0.7457	0.8319	0.8724	0.8403	0.7964	0.8280
	Worst	0.7758	0.7619	0.6879	0.7619	0.8512	0.7619	0.6879	0.7418
	Std.	0.0585	0.0531	0.0347	0.0531	0.0208	0.0510	0.0190	0.0531
	Avg. Features	61	12.9						
Turkish Music Emotion	Best	0.5281	0.7000	0.5281	0.7000	0.6096	0.7022	0.5042	0.6991
	Mean	0.4731	0.6098	0.4731	0.6098	0.5301	0.6101	0.4466	0.6067
	Worst	0.4219	0.5500	0.4219	0.5500	0.4491	0.5463	0.3865	0.5453
	Std.	0.0318	0.0538	0.0318	0.0538	0.0391	0.0486	0.0391	0.0538
	Avg. Features	51	13.2						
Spectf Heart	Best	0.7987	0.8750	0.7987	0.8750	0.8447	0.8750	0.7925	0.8750
	Mean	0.7655	0.7625	0.7655	0.7625	0.8088	0.7858	0.7535	0.7553
	Worst	0.7199	0.6875	0.7199	0.6875	0.7215	0.7312	0.6895	0.6875
	Std.	0.0267	0.0515	0.0515	0.0515	0.0337	0.0534	0.0305	0.0533
	Avg. Features	45	8.1						
Spect Heart	Best	0.7744	0.9375	0.8095	0.9375	0.8171	0.9453	0.7781	0.9377
	Mean	0.7297	0.8063	0.7125	0.8063	0.7423	0.8165	0.7163	0.7957
	Worst	0.6821	0.6875	0.6000	0.6875	0.6878	0.7004	0.6526	0.6838
	Std.	0.0292	0.0744	0.0784	0.0744	0.0396	0.0674	0.0480	0.0730
	Avg. Features	23	5.6						
Libras Movement	Best	0.7667	0.8333	0.7667	0.8333	0.7900	0.8770	0.7459	0.8295
	Mean	0.7415	0.7256	0.7415	0.7256	0.7630	0.7854	0.7243	0.7245
	Worst	0.7108	0.5556	0.7108	0.5556	0.7207	0.6090	0.6895	0.5411
	Std.	0.0153	0.0844	0.0153	0.0844	0.0233	0.0801	0.0172	0.0852
	Avg. Features	91	18.8						
Qsar Biodegradation	Best	0.8151	0.8957	0.8151	0.8957	0.8139	0.8959	0.8091	0.8921
	Mean	0.8039	0.8600	0.8039	0.8600	0.8019	0.8590	0.7951	0.8577
	Worst	0.7867	0.8199	0.7867	0.8199	0.7845	0.8197	0.7743	0.8158
	Std.	0.0087	0.0207	0.0087	0.0207	0.0088	0.0210	0.0105	0.0210
	Avg. Features	42	8.1						

ilarly, in the Spect Heart dataset, SVM-DLOFS has a best accuracy of 93.75%, well above SVM's 77.44%.

In Fig. 2, the average runtime in milliseconds (ms) is presented as a graphical comparison between using all features and the optimal feature subsets (i.e., those yielding the best results), evaluated through ten runs using SVM. According to the results, the proposed method significantly reduced the runtime for several datasets: from 402.74 ms to 250.92 ms for the Qsar Biodegradation dataset, from 175.25 ms to 134.39 ms for the Turkish Music Emotion dataset, and from 185.2 ms to 137.47 ms for the Libras Movement dataset. Similarly, a reduction in runtime was observed for the Sonar and Spect Heart datasets, while a slight increase was noted for the Spectf Heart dataset.

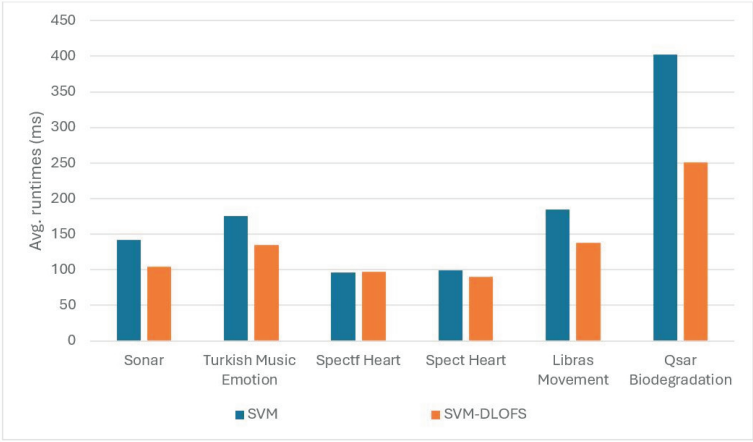


Fig. 2. Comparison of average runtimes of proposed Support Vector Machines with Draco Lizard Feature Selection Optimizer and Support Vector Machines

Conclusions. In this study, a novel classifier model was proposed by integrating Draco Lizard Optimizer Feature Selection (DLOFS) and Support Vector Machines (SVM). In experimental studies, six different datasets from the UCI Machine Learning Repository were used. The experimental results demonstrated that optimizing the feature selection process mostly led to improved classification performance and proposed SVM-DLOFS method also effectively eliminates irrelevant features and leads to also reducing classifier training time. This contributes to reducing the risk of overfitting or underfitting in high-dimensional datasets and improves the generalization capability of the SVM classifier.

REFERENCES

[1] KHALID S., T. KHALIL, S. NASREEN (2014) A survey of feature selection and

- feature extraction techniques in machine learning. In: 2014 Science and Information Conference, IEEE, 372–378.
- [2] CHUMERIN N., M. VAN HULLE (2006) Comparison of two feature extraction methods based on maximization of mutual information. In: 2006 16th IEEE Signal Processing Society Workshop on Machine Learning for Signal Processing, IEEE, 343–348.
 - [3] WANG X., H. LIU, A. G. HUSSIEEN et al. (2025) Enhanced Particle Swarm Optimization Algorithm Based on SVM Classifier for Feature Selection, *Comput. Model. Eng. Sci. (CMES)*, **142**(3), 1–20.
 - [4] ONAY F. K. (2023) Binary white shark optimization algorithm for feature selection problems, *Gümüşhane Univ. Science Journal*, **13**(2), 281–298, (in Turkish).
 - [5] ALKANHEL R., E. S. M. EL-KENAWY, A. A. ABDELHAMID et al. (2023) Network Intrusion Detection Based on Feature Selection and Hybrid Metaheuristic Optimization, *Comput. Mater. Continua*, **74**(2), 1–15.
 - [6] KARLUPIA N., P. ABROL (2023) Wrapper-based optimized feature selection using nature-inspired algorithms, *Neural Comput. Appl.*, **35**(17), 12675–12689.
 - [7] ZHU Y., W. LI, T. LI (2023) A hybrid artificial immune optimization for high-dimensional feature selection, *Knowl.-Based Syst.*, **260**, 110111.
 - [8] KWAKYE B. D., Y. LI, H. H. MOHAMED et al. (2024) Particle guided metaheuristic algorithm for global optimization and feature selection problems, *Expert Syst. Appl.*, **248**(3), 123362.
 - [9] MAHAPATRA A. K., N. PANDA, B. K. PATTANAYAK (2023) Quantized Salp Swarm Algorithm (QSSA) for optimal feature selection, *Int. J. Inf. Technol.*, **15**(2), 725–734.
 - [10] KHODADADI N., E. KHODADADI, Q. AL-TASHI et al. (2023) BAOA: binary arithmetic optimization algorithm with K-nearest neighbor classifier for feature selection, *IEEE Access*, **11**, 94094–94115.
 - [11] CINAR A. C. (2023) A novel adaptive memetic binary optimization algorithm for feature selection, *Artif. Intell. Rev.*, **56**(11), 13463–13520.
 - [12] CORTES C., V. VAPNIK (1995) Support-vector networks, *Mach. Learn.*, **20**, 273–297.
 - [13] WANG X. (2025) Draco lizard optimizer: a novel metaheuristic algorithm for global optimization problems, *Evol. Intell.*, **18**(1), 1–20.

¹*Department of Computer Engineering, Faculty of Engineering,
Harran University, Sanliurfa, Turkey
e-mail: berkanaydilek@harran.edu.tr, <https://orcid.org/0000-0001-8037-8625>*

²*Department of Computer Engineering,
Graduate School of Natural and Applied Sciences,
Harran University, Sanliurfa, Turkey
e-mail: ismailacr63@gmail.com, <https://orcid.org/0009-0007-9888-7429>*