Matrik: Jurnal Manajemen, Teknik Informatika, dan Rekayasa Komputer
Vol. 23, No. 3, July 2024, pp. 627∼640
ISSN: 2476-9843, accredited by Kemenristekdikti, Decree No: 200/M/KPT/2020
DOI: 10.30812/matrik.v23i3.3963

Dynamic Weighted Particle Swarm Optimization - Support Vector Machine Optimization in Recursive Feature Elimination Feature Selection

Irma Binti Syaidah[1], Sugiyarto Surono[1], Khang Wen Goh[2]
[1]Mathematics Study Program, Ahmad Dahlan University, Yogyakarta, Indonesia
[2]Faculty of Data Science and Information Technology, INTI International University, Malaysia

Article Info

Article history:

Received April 01, 2024
Revised May 07, 2024
Accepted June 12, 2024

Keywords:

Dynamic Weighted Particle Swarm
Feature Selection
Optimization
Recursive Feature Elimination
Support Vector Machine

ABSTRACT

Feature Selection is a crucial step in data preprocessing to enhance machine learning efficiency, reduce computational complexity, and improve classification accuracy. The main challenge in feature selection for classification is identifying the most relevant and informative subset to enhance prediction accuracy. Previous studies often resulted in suboptimal subsets, leading to poor model performance and low accuracy. **This research aimed** to enhance classification accuracy by utilizing Recursive Feature Elimination (RFE) combined with Dynamic Weighted Particle Swarm Optimization (DWPSO) and Support Vector Machine (SVM) algorithms. **The research method** involved the utilization of 12 datasets from the University of California, Irvine (UCI) repository, where features were selected via RFE and applied to the DWPSO-SVM algorithm. RFE iteratively removed the weakest features, constructing a model with the most relevant features to enhance accuracy. **The research findings** indicated that DWPSO-SVM with RFE significantly improves classification accuracy. For example, accuracy on the Breast Cancer dataset increased from 58% to 76%, and on the Heart dataset from 80% to 97%. The highest accuracy achieved was 100% on the Iris dataset. **The conclusion of these findings is** that RFE in DWPSO-SVM offers consistent and balanced results in True Positive Rate (TPR) and True Negative Rate (TNR), providing reliable and accurate predictions for various applications.

Corresponding Author:

Sugiyarto Surono,
Mathematics Study Program,
Universitas Ahmad Dahlan, Yogyakarta, Indonesia,
Email: sugiyarto@math.uad.ac.id.

How to Cite:
I. Syaidah, S. Surono, and G. Khang Wen, "Dynamic Weighted Particle Swarm Optimization - Support Vector Machine Optimization in Recursive Feature Elimination Feature Selection", *MATRIK: Jurnal Manajemen, Teknik Informatika, dan Rekayasa Komputer*, Vol. 23, No. 3, pp. 627-640, July, 2024.

1. INTRODUCTION

Feature selection is one of the most important methods in preprocessing data. Researchers in the field of statistics and machine learning have been the focus of research for many years [1]. This method aims to increase the efficiency of machine learning, minimize computational complexity, develop more generalizable models, and reduce the storage required. One of the methods that proved to be the best feature selection method is Recursive feature elimination (RFE) [2]. RFE is a wrapper feature selection method that evaluates the importance of features iteratively based on machine learning performance by recursively eliminating the least important features until the best performance is obtained or the specified number of features is reached [3]. RFE is a feature selection method that tries to select the optimal feature set based on the learned model [4]. RFE is important in improving classification performance by iteratively removing irrelevant features, ensuring that only the most informative features are used in classification model building [5]. Feature selection methods not only help simplify data representation but also contribute to improved classification accuracy [6]. Classification is an important stage in machine learning that aims to place data into certain categories or labels based on existing patterns or characteristics [7]. One of the most commonly used methods in classification is the Support Vector Machine (SVM) [8]. SVM is a machine learning algorithm that carefully finds the optimal hyperplane to separate various data classes. The SVM method uses kernel techniques to map the dataset to improve SVM performance [9]. The SVM model can adapt to both linearly and non-linearly separable data, demonstrating its effectiveness. SVM also excels at handling data from different sources, leading to increased accuracy in classification. The main advantage of SVM lies in its ability to capture non-linear relationships in the data, resulting in high classification accuracy [10]. However, SVM requires careful parameter tuning to achieve optimal performance. This tuning process can be complicated and time-consuming, especially when dealing with large or complex datasets. Therefore, parameter optimization becomes an important step to improve SVM's accuracy and overall performance [11].

Parameter optimization is the process of finding optimal values for certain parameters in a model or algorithm. Particle Swarm Optimization (PSO) is used to search for the optimal input dataset by increasing the number of features until it reaches the maximum limit. Each step involves creating a population where everyone is a randomized input feature dataset. These steps are repeated until the maximum number of input features are reached, where SVM is used to evaluate and select the best dataset for the PSO-SVM model. By applying PSO-SVM, the robust search algorithm can eliminate noise and irrelevant features in the input parameters, making it suitable for handling low-damage intensity problems [12]. Furthermore, Dynamic Weighted Particle Swarm Optimization (DWPSO) is an optimization algorithm developed by combining PSO with dynamic weight factors to improve SVM classification accuracy and shorten experiment time. In the DWPSO algorithm, the linear reduction of the inertia weight (w) is decided upon with careful consideration by the algorithm developers. DWPSO, as a variation of PSO that has proven effective over the years, utilizes this approach to focus on diversity in the previous iteration and convergence in the last iteration. The linearly decreasing inertia factor in DWPSO becomes a key parameter that greatly affects the success and evaluation of the function [13]. Several journals have discussed different types of processing methods and stages, but different datasets and case studies have been used. According to [14], The RFE algorithm was utilized to identify the most indicative features of the Chronic kidney disease (CKD) dataset. These selected features underwent classification algorithms, including SVM, KNN, decision tree, and random forest. The parameters of each classifier were adjusted to optimize classification performance, resulting in promising outcomes across all algorithms. Notably, the random forest algorithm surpassed others, achieving perfect scores of 100% for accuracy, precision, recall, and F1-score. Subsequently, the system underwent thorough evaluation through multiclass statistical analysis, revealing significant accuracy metrics of 96.67%, 98.33%, and 99.17% for SVM, KNN, and decision tree algorithms, respectively. Furthermore, research [15] on classifying patients with chronic obstructive pulmonary disease (COPD) using SVM-RFE and LibSVM produced a model with an AUC value of 0.987 and an F1 score of 0.978, distinguishing well between continuously managed and unmanaged patients. Another study [16] compared SVM and PSO-SVM for landslide vulnerability mapping in Lueyang County, China, with the result that slope-based PSO-SVM gave the best results with AUC values of 0.945 and 0.925 for training and validation datasets. Research [17] shows that the COVID-19 vaccine sentiment analysis discussed uses machine classification techniques with the aim of improving classification performance in COVID-19 vaccine sentiment analysis through the application of Information Gain Ratio (IGR) and Particle Swarm Optimization (PSO) to the Support Vector Machine (SVM) model. The results showed that the integration of IGR and PSO significantly improved the performance of the SVM model, with accuracy increasing from 0.794 (base model) to 0.837 and AUC increasing from 0.890 to 0.913.

Research related to DWPSO, among others, has been carried out [18] on developing a new feature selection method, FS-score, which is combined with the DWPSO-SVM optimization algorithm to improve SVM classification accuracy and shorten experimentation time in various application domains. The results of the DWPSO-SVM algorithm show a significant improvement in classification performance. DWPSO-SVM has better accuracy than most datasets' basic PSO algorithm. For example, on the Vowel and Diabetes datasets, the accuracy of DWPSO-SVM increased by 6.58% and 4.8%, respectively, compared to before. This algorithm also shows a significant increase in True Positive Rate (TPR) and True Negative Rate (TNR) values, such as on the Sonar dataset with a TPR value of 90% and TNR of 94.44%. Moreover, findings from multiple trials in [19] demonstrated that DWPSO achieved a higher

accuracy by several percentage points than PSO after repeated testing. Based on the description and some related research, **there is a gap that has not been overcome** by previous research, namely the evaluation of changes in DWPSO-SVM accuracy when using different feature selection methods. **The difference between this research and the previous** one is that the RFE method is used to assess its impact on the classification accuracy of DWPSO-SVM. This novel approach aims to address the research gap identified in previous studies by investigating the effectiveness of RFE in enhancing DWPSO-SVM accuracy. **This research aim**s to evaluate the impact of different feature selection methods, particularly RFE, on the accuracy of DWPSO-SVM classification. By doing so, this study seeks to contribute to advancing knowledge in machine learning by providing insights into the effectiveness of feature selection techniques in improving classification accuracy. Additionally, the findings of this research can have practical implications in various domains where accurate classification is essential, such as healthcare, finance, and marketing.

2. RESEARCH METHOD

This section discusses the proposed research method, which involves combining the RFE feature selection method with the DWPSO algorithm to improve SVM classification accuracy and reduce overall experimentation time. The RFE method is used to select key features and eliminate redundant or irrelevant features to improve data representation and increase classification algorithms' accuracy and computational efficiency. This study uses a data set derived from the University of California, Irvine (UCI) repository, which provides various data sets for machine learning research. The classification model we used with RFE and DWPSO-SVM feature selection is depicted in Figure 1.

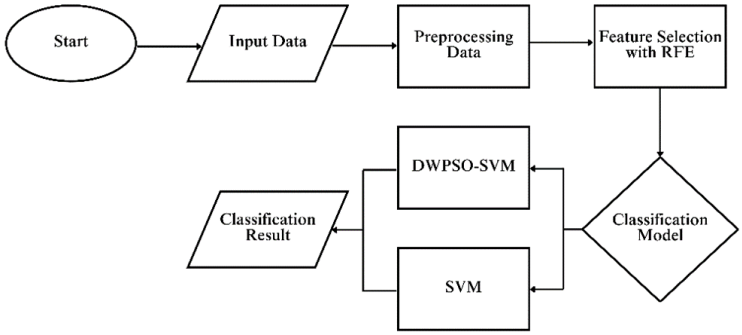

Figure 1. Research Methods

2.1. Dataset

The datasets used in this research are various datasets from the University of California, Irvine (UCI) repository. These datasets cover different feature and data set characteristics, providing various data to evaluate the proposed method. The following is an example of a dataset table from the UCI repository, as seen in Table 1.

Table 1. The Datasets from the UCI Repository

Dataset	Feature	Classes	Feature Characteristic	Data Set Characteristics
Credit Card Australia	14	2	Categorical, Real	Multivariate, Time-Series
Breast cancer	9	2	Categorical	Multivariate
CMC	9	3	Categorical, Integer	Multivariate
Diabetes	9	2	Categorical, Integer	Multivariate, Time-Series
Credit Card Germen	24	4	Integer, Real	Multivariate
Heart	14	2	Categorical, Integer, Real	Multivariate
Ionosphere	34	2	Integer, Real	Multivariate
Iris	4	2	Real	Multivariate
Sonar	60	2	Integer	Multivariate, Text
Vehicle	18	4	Real	Multivariate, Integer
Vowel	3	6	N/A	Image
Wdbc	30	2	Real	Multivariate, Data-Generator
Wine	13	3	Integer, Real	Multivariate

Next, min-max normalization is performed on the features in the dataset. Min-max normalization is a step in data preprocessing where the numerical values of the attributes in the dataset are converted into a more general format. This process aims to produce attribute values that fall within a common range, usually [0, 1] [20]. This is beneficial for improving the efficiency of the algorithm's training time and making the resulting model more stable. One commonly used normalization method is min-max normalization. Min-max normalization is defined as Equation (1) [21]. Thus, x^* is the value of x that has been normalized to be in the range of values between 0 and 1. Where x is the original variable to be normalized, while x_{min} dan x_{max} are the minimum and maximum values of variable x, respectively.

$$x^* = \frac{x - min(x)}{max(x) - min(x)} \tag{1}$$

2.2. Feature Selection with Recursive Feature Elimination

Feature selection is an information discovery tool that provides a deep understanding of the problem through analyzing the most significant features. The purpose of feature selection is to improve the quality of the classifier by identifying a list of relevant features, which at the same time can help reduce excessive computational burden [22]. The RFE method is used for feature selection. The RFE method builds a machine-learning model on the dataset and evaluates the features with certain parameters, such as logistic regression. Logistic regression is a statistical method used to understand the correlation between one or more dependent variables and binary independent variables, which have values of 0 and 1. In logistic regression, the odds ratio is often used to evaluate the strength of the relationship between dependent and independent variables [23].

With n dependent variables and binary dependent variables denoted as $(x_i, y_i)(i = 1, 2, \ldots, n)$, a logistic probability model is obtained. In logistic regression, the logistic probability Equation (2). Where n is the number of predictor variables and $\pi(x)$ is the probability of success with a value of $0 \leq \pi(x) \leq 1$. To make it easier to estimate the regression parameters, $\pi(x)$ in Equation (2) is transformed into logit form in Equation (3). The binary logistic regression model is used if the response variable or independent variable produces two categories worth 0 and 1, so it follows the Bernoulli distribution Equation (4).

$$\pi(x) = \frac{exp(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_n x_n)}{1 + exp(\beta_0 + \beta_1 x_i + \beta_2 x_2 + \ldots \beta_n x_n)} \tag{2}$$

$$logit(\pi(x)) = log\left(\frac{\pi(x)}{1 - \pi(x)}\right) = In\left(\frac{\pi(x)}{1 - \pi(x)}\right) \tag{3}$$

$$f(y_i) = \pi(x_i)^y (1 - \pi(x_i))^{1-y_i} \tag{4}$$

The solution to estimate the unknown parameters can use the Maximum Likelihood Estimation (MLE) method. This method involves optimizing a likelihood function that matches the distribution of observed data. In logistic regression, MLE provides an estimated value β, which shows how influential a feature is on the target. Systematically, the likelihood function for the binary logistic regression model is in Equation (5). where, y_i is the observation on the i-th, and $\pi(x_i)$ is the probability for the i-th prediction variable. In determining the value of the estimated coefficient (β) that maximizes L(β) the derivative of $\beta_1, \beta_2, \ldots, \beta_n$ then set to zero, resulting in the following formulas shown in Equations (6) and (7):

$$L(\beta) = In[l(\beta)] = \sum_{i=1}^{n} \{y_i In[\pi(x_i)] + (1 - y_i) In[1 - \pi(x_i)]\} \tag{5}$$

$$\sum_{i=1}^{n} [y_i - \pi(x_i)] = 0 \tag{6}$$

And

$$\sum_{i=1}^{n} x_i [y_i - \pi(x_i)] = 0 \tag{7}$$

Where β is the logistic regression coefficient, y_i is the target/dependent variable, and x_i is the corresponding feature/independent variable. The odds ratio of the regression coefficient value (β) provides a measure of how much influence the independent variable

has on the dependent variable. The odds ratio, or tendency ratio, is a figure that describes the comparison between the number of individuals experiencing a specific event and the number of individuals not experiencing that event, both in the sample and the population. The odds ratio (or(β)) is calculated by taking the exponential of the regression coefficient (β) value. To evaluate the likelihood of success (π), the odds ratio value can be calculated using the following Equation (8).

$$or(\beta) = exp(\beta) \tag{8}$$

2.3. Classification with the Support Vector Machine Method

Classification is a supervised learning method that analyzes datasets and creates models that categorize data into distinct classes. The main goal is to develop a model that optimizes performance on the training data and can effectively classify new, unknown data. One commonly used classification method in this research is the Support Vector Machine (SVM), which aims to separate multiple classes by maximizing the margin between them [24]. SVM finds the optimal hyperplane that separates two classes by maximizing the distance between the hyperplane's margin and the data points. Originally developed as a linear classifier, SVM has been extended to handle non-linear problems using the kernel trick and can now classify multiple classes [25]. SVM can be used to solve data classification problems in linear and non-linear cases in Figure 2.

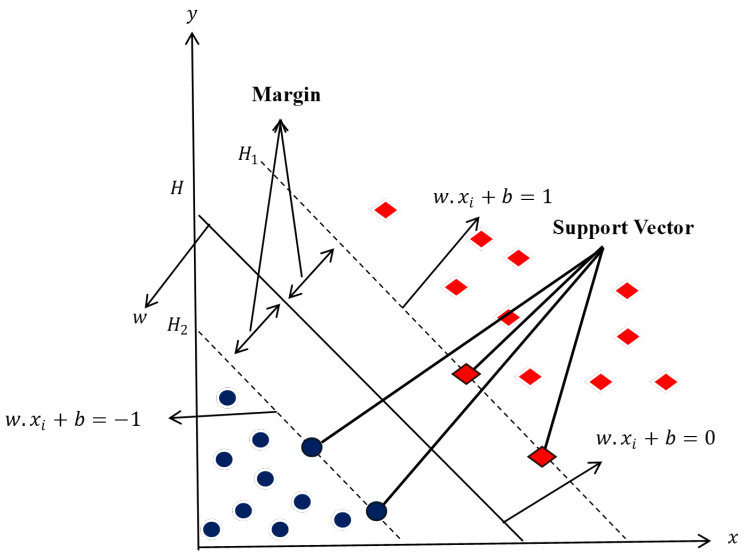

Figure 2. Hyperplane, maximum margin, and support vector machine linear

In Figure 2, SVM is considered linear if it can separate data into two different classes with at least one separator. Suppose there are two different classes with at least one separating function. Suppose there are n data points (x_i, y_i) with each input x_i dimension d denoted by $x_i \epsilon R^d$, with the label for each class denoted by $y_i \epsilon -1, +1$ for $i = 1, 2, \ldots, n$. Assuming the two classes -1 and +1 can be separated by a hyperplane, the hyperplane formula H_0 u for linearly separable data is defined in Equation (9):

$$H_0 : w^T x_i + b = 0 \tag{9}$$

Where w is the weight vector, x_i is the feature vector for the i-th data, and b is the bias. Suppose H_1 Equation (10) and H_2 (Equation 11) are the hyperplanes for the first and second class. The equation for each point on the class boundary level (support vector) is in equation (12). where x_i is the i-th data, y_i is the class label of x_i,w is the weight vector and b is a scalar or bias. So, the generalized hyperplane Equations for two positive classes and two negative classes at the margin are shown in Equations (13) and (14).

$$H_1 : w^T x_i + b = +1 \tag{10}$$

$$H_2 : w^T x_i + b = -1 \tag{11}$$

$$y_i = w^T x_i + b \tag{12}$$

$$w^T x_i + b = +1 \tag{13}$$

$$w^T x_i + b = -1 \tag{14}$$

With conditions

$$y_i(w^T x_i + b) \geq 1, \forall i = 1, 2, \ldots, n \tag{15}$$

The margins of the separating plane are d_+ and d_-. Using the distance formula between points and lines, the distance between the hyperplane H_1 and h_2 is $\frac{2}{\| w \|}$. The optimal solution finds the values of w and b that minimize the norm of vector w, in a constrained optimization problem. So, this problem can be solved by the Lagrange method by introducing the Lagrange multiplier $\alpha_i \geq 0$ or each constraint, we can formulate the Lagrangian L as Equation (16). Where α_i is the Lagr ange multiplier, w and b is a parameter of the hyperplane.

$$L(w, b, \alpha) = \frac{1}{2} \| w \|^2 - \sum_{i=1}^{n} \alpha_i [(y_i * w^T x_i + b) - 1] \tag{16}$$

In Figure 3, the SVM is modified by transforming into a new higher-dimensional vector space for data that cannot be linearly separated. Then, the data is mapped with the transformation function to a new higher-dimensional space so that a separating hyperplane separates the data. (separating hyperplane) that separates the data according to its class, and the transformation function is a kernel trick. The transformation function is the kernel trick. In the case of non-linear data, SVM uses kernel tricks by utilizing the kernel function as in Table 2. In this research, the kernel function used is the Radial Basis Function (RBF).

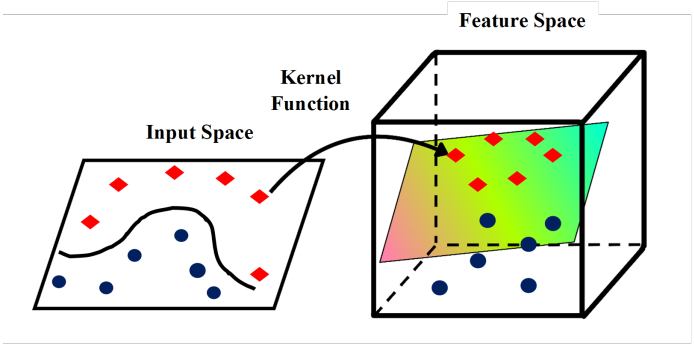

Figure 3. Kernel Function with Non-linear Transformation

Table 2. Kernel Function

Kernel Type	Formula
Kernel linear	$x_i^T x^j$
Polynomial Kernel(P-SVM)	$K(x_i, x_j) = (x_i^T x_j + 1)^d$
Radial Basis Function (RBF)	$K(x_i, x_j) = exp\left\{ - \frac{\| x_i - x_j \|^2}{2\sigma^2} \right\}$
Tangent hyperbolic (sigmoid) kernel	$K(x_j, x_j) = \beta x_i^T x_j + \beta_i$

2.4. Parameter Optimization with Dynamic Weighted Particle Swarm Optimization

To deal with instability, researchers used a feature selection method during the parameter optimization process. The concept is to select a group of parameters that actively optimize the optimization process [22]. The author researched the DWPSO model. DWPSO is one of the classic and highly effective variations of PSO because its advantages have persisted over the years. In DWPSO, the inertia factor decreases linearly and focuses on diversity in the previous iteration and convergence in the last iteration [26]. The selection of appropriate and correct inertia weights provides a balance between global and local exploitation and results in fewer iterations, on average, to find a reasonably optimal solution [27]. The inertia weight w is the most influential parameter on the success rate and function evaluation [26].

DWPSO-SVM is a more efficient approach to optimizing SVM parameters without the complexity of grid or genetic algorithms. It uses a mobile search strategy throughout the area and can dynamically adjust its search strategy according to the current situation. The evaluation is performed on various indices, with the dataset divided into training, monitoring, and testing sets to ensure the optimized model can provide accurate prediction results independently. In PSO, the particles move in the search space based on the Equation (17). Equations (17) and Equations (18) $i = 1, 2, \ldots N$, i denotes the particle index in the swarm, N is the total number of particles in the swarm, v_i is the particle velocity, $rand()$ is a function that generates a random number between 0 and 1, x_i is the particle's current position in the search space, c_1 and c_2 are learning factors that usually have fixed values. The maximum limit of v_i is defined by v_{max}. If the speed of v_i exceeds v_{max} during the iteration, then v_i will be set back to v_{max}. In the context of PSO, $pbest_i$ and $gbest_i$ refer to the best position ever achieved by particle i and the best position ever achieved by the swarm, respectively. This equation is referred to as the elementary particle swarm Equation (19) [28]. Specifically expressed as follows:

$$v_i = v_i + c_i * rand() * pbest_i) + c_2 * rand() * (gbest_i - x)_i \tag{17}$$

$$x_i = x_i + v_i \tag{18}$$

$$v_i = w * v_i + c_i * rand() * pbest_i - x_i + c_2 * rand() * (gbest_i - x)_i \tag{19}$$

Compared to Equation (17), Equation (18) only undergoes modification in the first term. Here, w is referred to as the inertia factor, and its value is non-negative. The magnitude of w affects particle movement in space. In the PSO algorithm, particles undergo two types of learning: cognitive and social. Cognitive learning occurs when particles learn from their own experiences, while social learning occurs when they learn from the experiences of other particles in the swarm. The results of cognitive learning are represented as p_{best}, while the results of social learning are symbolized as g_{best} [29]. When its value is large, global optimization capability is strong, but local optimization capability is weak, and vice versa. When its value is small, global optimization capability is weak, but local optimization capability is strong. The introduction of w has significantly improved the performance of the PSO algorithm, allowing it to flexibly adjust global and local search capabilities according to different actual situations. Equations (18) and (19) refer to the overall standard PSO algorithm [19].

Next, the improved PSO algorithm with increased inertia weight and its utilization in optimizing SVM classification parameters are highlighted. It also includes explanations of the model, algorithm, and dynamic inertia weight adjustment methods to adapt to variational trends in equations. Meanwhile, the DWPSO-SVM formula that dynamically regulates the decrease of inertia weights is expressed in Equation (20). Equation (20) reduces the inertial weight gradually through a logarithmic function $\frac{1}{2}ln(e + (kz))$ with k as the regulating factor. The inertia weight is initially large, and as z increases, the weight decreases. To maintain population diversity and prevent premature convergence, $w(t)$ is increased randomly. The use of the random function $rand()$ improves the search capability initially and avoids premature convergence, which improves the search accuracy. After conducting continuous research on the value of inertia weights, Shi and Eberhart concluded that values in the interval [0.9,1.2] positively influence solution improvement. Linearly decreasing inertia weights with $[c_1 = 2, c_2 = 2]$ and w between 0.4 and 0.9. According to their statement, w_{new} is the new inertial weight, which decreases linearly from 0.9 to 0.4, $z = \frac{t}{T}$ [19].

$$W_{new(t)} = W_{min} + \frac{W_{max} - W_{min}}{2} + \frac{1}{In(e + kz)^2} + \left(\frac{W_{max} - W_{min}}{2}\right) * rand() \tag{20}$$

2.5. Evaluation of Classification Result

In Figure 4, an illustration of the calculation is shown in the confusion matrix in Figure 4. Equations (21) to (26) illustrate the calculations performed in the confusion matrix. The TP (True Positive) parameter indicates the number of correctly positive predictions of the actual class that are also positive. FP (False Positive) refers to the number of false positive predictions of an actual class that is negative. TN (True Negative) reflects the number of negative correct predictions of actual classes that are also negative. Meanwhile, FN (False Negative) describes the number of negative incorrect predictions of an actual positive class.

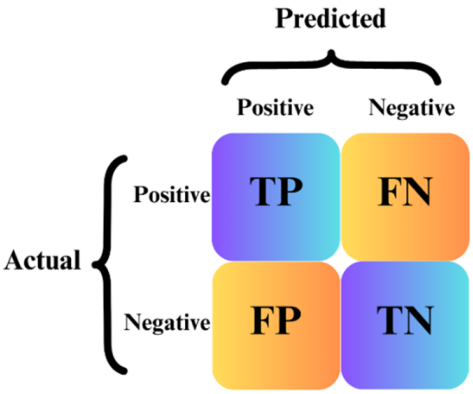

Figure 4. Confusion Matrix

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{21}$$

$$Pecision = \frac{TP}{TP + FP} \tag{22}$$

$$Recall \frac{TP}{TP + FN} \tag{23}$$

$$F1 - score = 2 \times \frac{Precision \times Recall}{Precision + recall} \tag{24}$$

$$TPR = precision = \frac{TP}{TP + FP} \tag{25}$$

$$TNR = \frac{TN}{TN + FP} \tag{26}$$

3. RESULT AND ANALYSIS

This research aims to evaluate the impact of feature selection methods, especially RFE, on the classification accuracy of DWPSO-SVM, as well as compare the results with the FS-Score method. This research also aims to overcome the limitations of previous studies that have not evaluated changes in DWPSO-SVM accuracy when using different feature selection methods [18]. The datasets used in this research come from the University of California, Irvine (UCI) repository. Some of the datasets used include Credit Card Australia, Breast Cancer, CMC, Diabetes, Credit Card Germen, Heart, Ionosphere, Iris, Sonar, Vehicle, Vowel, Wdbc, and Wine. Each dataset has its characteristics. For example, some datasets have categorical and real features, while others have a combination of integer and real. Some datasets also belong to multivariate and time-series categories.

3.1. Result of Feature Selection with RFE

The results of feature selection with RFE are shown in Table 3. Table 3 shows the change in classification accuracy along with the number of features retained. Each dataset has different characteristics that affect its feature selection. For example, while the Vowel dataset only selects 3 features. This variation shows the complexity and uniqueness of each dataset. The selection of different features emphasizes the importance of properly analyzing the characteristics of each dataset to select the most relevant and informative features. In addition, the classification accuracy varies between datasets, which means the performance of the classification model is greatly influenced by the dataset used and the features selected. Thus, RFE is an effective feature reduction algorithm in selecting the most relevant features for classification and can help improve the model's overall performance.

Table 3. Optimal feature subset

Dataset	Number of selected features	Selected features
Credit Card Australia	7	4,5,8,9,10,13,14
Breast cancer	6	1,2,3,4,6,9
CMC	3	1,2,4
Diabetes	4	1,2,6,7
Credit Card Germen	7	1,2,3,5,15,17,19
Heart	11	2,3,4,5,7,8,9,10,11,12,13
Ionosphere	19	3,4,5,6,7,8,9,11,14,18,21,22,23,26,27,29,31,32,34
Iris	4	1,2,3,4
Sonar	14	1,11,12,16,21,31,36,37,44,45,48,49,51,52
Vehicle	11	1,3,5,7,8,9,10,12,14,17,18
Vowel	3	1,2,3
Wdbc	10	2,4,9,22,23,24,25,26,28,29
Wine	5	1,7,10,12,13

3.2. Evaluation of DWPSO-SVM

This experiment shows that the DWPSO-SVM algorithm completes the classification on each dataset with varying computation time in Table 4. Table 4 shows that DWPSO-SVM exhibits diverse performance across various datasets, highlighting improvements in classification accuracy and varying computational times. The computational time of the algorithm ranges from 0.89 seconds for the Iris dataset to 50.69 seconds for the Vowel dataset. The variation in computational time is influenced not only by the complexity of the dataset but also by the size of the feature subset used. Nevertheless, experiments demonstrate that DWPSO-SVM can effectively handle various datasets, albeit with differing computational times, depending on the dataset's complexity and optimal feature subset size. Furthermore, there is a significant increase in accuracy after optimization. For example, the Breast Cancer dataset shows a notable improvement from 58% to 76%, while the Heart dataset increased from 80% to 97%. Therefore, DWPSO-SVM proves to be effective in enhancing classification accuracy and managing computational time across various datasets. The results from testing before and after optimization, as seen in Figure 5, show consistent performance improvements after optimization. In every experiment, the optimization always resulted in better performance than before.

Table 4. Evaluation on DWPSO-SVM

Dataset	Accuracy before optimization	Accuracy after optimization	Time(s)
Credit Card Australia	85%	90%	4.14
Breast cancer	58%	76%	1.69
CMC	47%	62%	31.42
Diabetes	65%	79%	5.54
Credit Card Germen	69%	81%	9.02
Heart	80%	97%	4.68
Ionosphere	85%	97%	1.65
Iris	100%	100%	0.89
Sonar	81%	95%	1.62
Vehicle	39%	83%	6.50
Vowel	12%	90%	50.69
Wdbc	92%	95%	1.53
Wine	87%	98%	2.04

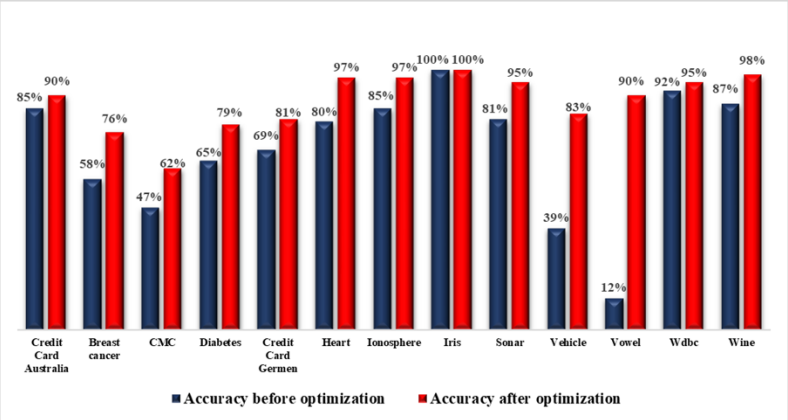

Figure 5. Model Accuracy Results

The data provided in Figure 5 illustrates significant improvements in each dataset after optimization. For instance, the accuracy of the Australian Credit Card dataset increased from 85% to 90%, marking a 5% improvement. Similar enhancements were observed across other datasets: Breast Cancer increased from 69% to 72%, CMC from 47% to 62%, Diabetes from 65% to 79%, Credit Card Germany from 69% to 81%, Heart from 80% to 97%, Ionosphere from 85% to 97%, Sonar from 81% to 95%, Vehicle from 39% to 83%, Vowel from 12% to 90%, WDBC from 92% to 95%, and Wine from 87% to 98%. These findings demonstrate that the optimizations applied to the model effectively improved classification accuracy across various datasets, indicating consistent and efficient enhancement techniques.

3.3. Evaluation

The confusion matrix is used as an evaluation to measure the model's performance. The data splitting strategy approach is done by dividing the dataset into training and testing data with a ratio of 90:10. In addition, the RBF (Radial Basis Function) kernel is used in the SVM algorithm to map the dataset and improve the performance of the classification model. In this experiment, the parameter C is 0.5, and gamma (γ) is 0.1. In Table 5, the highest results of the experiments conducted with accuracy, precision, recall, and F1-Score reached 100%, which were calculated using Equations (21), (22), (23) and (24) .

Table 5. Experiment Result

Dataset	Accuracy	Precision	Recall	F1-Score
Credit Card Australia	90%	90%	90%	90%
Breast cancer	76%	73%	72%	71%
CMC	62%	66%	62%	62%
Diabetes	79%	79%	79%	79%
Credit Card Germen	81%	80%	81%	80%
Heart	97%	97%	97%	97%
Ionosphere	97%	95%	94%	94%
Iris	100%	100%	100%	100%
Sonar	95%	90%	90%	90%
Vehicle	83%	83%	83%	83%
Vowel	90%	90%	90%	90%
Wdbc	95%	95%	95%	95%
Wine	98%	98%	98%	98%

The findings of this study were that the use of RFE methods in combination with the DWPSO and SVM algorithms proved effective in improving classification accuracy on various datasets. The results showed significant improvements in accuracy after optimization using DWPSO-SVM, with consistent improvements observed on most datasets. RFE use in DWPSO-SVM yielded consistent and balanced TPR and TNR values results. **The results of this study are** consistent with or supported by previous research that uses optimization and feature selection methods to improve the performance of classification models. For example, research conducted by [18] shows that using optimization algorithms can significantly improve the accuracy and consistency of the

classification model. However, our research shows that the RFE method in DWPSO-SVM often gives more balanced TPR and TNR values than the FS-Score method. Table 6 shows the experimental results and comparisons with previous studies to provide a clear comparison. The evaluation criteria we use are accuracy, TPR, and TNR calculated using Equations (21), (25), and (26).

Table 6. Comparison with previous research

Dataset	DWPSO-SVMRFE			DWPSO-SVM FS-Score		
	Accuracy	TPR(%)	TNR(%)	Accuracy	TPR(%)	TNR(%)
Credit Card Australia	**90%**	82%	94%	87%	88%	91%
Breast cancer	**76%**	58%	88%	74%	21%	94%
CMC	**62%**	68%	98%	54%	N/A	N/A
Diabetes	79%	63%	88%	79%	80%	87%
Credit Card Germen	**81%**	52%	93%	57%	N/A	N/A
Heart	**97%**	93%	100%	93%	88%	76%
Ionosphere	97%	91%	100%	97%	90%	100%
Iris	**100%**	100%	100%	98%	N/A	N/A
Sonar	**95%**	100%	86%	91%	90%	94%
Vehicle	**83%**	100%	100%	78%	N/A	N/A
Vowel	**90%**	92%	100%	84%	N/A	N/A
Wdbc	95%	95%	95%	95%	N/A	N/A
Wine	**98%**	100%	100%	64%	100%	100%

We also compared our experimental results with the methods discussed in [18]. Table 6 compares accuracy, TPR, and TNR results between the DWPSO-SVM method with RFE and the DWPSO-SVM method with FS-Score of multiple datasets. The results in Table 6 show that the DWPSO-SVM method with RFE tends to provide higher overall accuracy, with the highest accuracy value occurring on the Iris dataset (100%). Meanwhile, the DWPSO-SVM method with FS-Score shows greater variation in results, with the highest accuracy occurring on the Ionosphere dataset (97%). However, the DWPSO-SVM method with RFE is often more consistent in providing balanced TPR and TNR values, especially on datasets such as Heart and Ionosphere. For example, on the Heart dataset, DWPSO-SVM with RFE has a TPR of 93% and a TNR of 100%, while on the Ionosphere dataset, the TPR and TNR reached 91% and 100%, respectively. Therefore, using the RFE method in DWPSO-SVM shows that our proposed model can provide high accuracy. Furthermore, the main objective of our research, which is to improve the accuracy of SVM by optimizing the parameters using the DWPSO-SVM algorithm with RFE feature selection, has been successfully achieved.

4. CONCLUSION

This research shows that using the RFE method in combination with DWPSO and SVM algorithms has proven effective in improving classification accuracy on various datasets. The results show a significant improvement in classification accuracy after optimization using DWPSO-SVM, with consistent improvements observed on most datasets. Despite variations in computation time and accuracy results between datasets, using RFE in DWPSO-SVM provided consistent and balanced results in TPR and TNR values. The proposed model has the potential to provide more reliable and accurate predictions in various applications, such as disease detection and disaster vulnerability modeling. However, this study has limitations, including limitations in the datasets used and the use of default parameters in some cases. Experiments with various parameter configurations and a wider dataset are recommended for future work to validate the model's reliability. This research contributes to developing methods that can improve the performance of classification models, which can be useful in various real-world applications, such as healthcare and disaster mitigation.

5. ACKNOWLEDGEMENTS

The authors gratefully acknowledge FAST Ahmad Dahlan University Indonesia, especially the Mathematic Laboratory and Faculty of Data Science and Information Technology, INTI International University, Malaysia, for supporting this work.

6. DECLARATIONS

AUTHOR CONTIBUTION

Irma Binti Syaidah: Responsible for programming and coding. Sugiyarto Surono: Draft the paper and finalize the paper. Khang Wen Goh: Proofreading paper.

FUNDING STATEMENT
No funding/self-funding.

COMPETING INTEREST
All authors have no competing interests.

REFERENCES

[1] N. Maurya, N. Kumar, V. Maurya, and V. Kumar Maurya, "a Review on Machine Learning (Feature Selection, Classification and Clustering) Approaches of Big Data Mining in Different Area of Research Journal of Critical Reviews a Review on Machine Learning (Feature Selection, Classification and Clustering) Approac," *Article in Journal of Critical Reviews*, vol. 7, no. August, p. 2020, 2020, https://doi.org/10.31838/jcr.07.19.322.

[2] N. Rtayli and N. Enneya, "Enhanced credit card fraud detection based on SVM-recursive feature elimination and hyper-parameters optimization," *Journal of Information Security and Applications*, vol. 55, no. September, 2020, https://doi.org/10.1016/j.jisa.2020.102596.

[3] Y. Yin, J. Jang-Jaccard, W. Xu, A. Singh, J. Zhu, F. Sabrina, and J. Kwak, "IGRF-RFE: a hybrid feature selection method for MLP-based network intrusion detection on UNSW-NB15 dataset," *Journal of Big Data*, vol. 10, no. 1, 2023, https://doi.org/10.1186/s40537-023-00694-8.

[4] H. Jeon and S. Oh, "applied sciences Hybrid-Recursive Feature Elimination for E ffi cient Feature Selection," *Applied Sciences*, pp. 1–8, 2020.

[5] R. Zebari, A. Abdulazeez, D. Zeebaree, D. Zebari, and J. Saeed, "A Comprehensive Review of Dimensionality Reduction Techniques for Feature Selection and Feature Extraction," *Journal of Applied Science and Technology Trends*, vol. 1, no. 2, pp. 56–70, 2020, https://doi.org/10.38094/jastt1224.

[6] M. Ahsan, R. Gomes, M. M. Chowdhury, and K. E. Nygard, "Enhancing Machine Learning Prediction in Cybersecurity Using Dynamic Feature Selector," *Journal of Cybersecurity and Privacy*, vol. 1, no. 1, pp. 199–218, 2021, https://doi.org/10.3390/jcp1010011.

[7] P. Saraswat, "Supervised Machine Learning Algorithm: A Review of Classification Techniques," *Smart Innovation, Systems and Technologies*, vol. 273, pp. 477–482, 2022, https://doi.org/10.1007/978-3-030-92905-3_58.

[8] J. Cervantes, F. Garcia-Lamont, L. Rodríguez-Mazahua, and A. Lopez, "A comprehensive survey on support vector machine classification: Applications, challenges and trends," *Neurocomputing*, vol. 408, no. xxxx, pp. 189–215, 2020, https://doi.org/10.1016/j.neucom.2019.10.118.

[9] B. R. Murlidhar, H. Nguyen, J. Rostami, X. N. Bui, D. J. Armaghani, P. Ragam, and E. T. Mohamad, "Prediction of flyrock distance induced by mine blasting using a novel Harris Hawks optimization-based multi-layer perceptron neural network," *Journal of Rock Mechanics and Geotechnical Engineering*, vol. 13, no. 6, pp. 1413–1427, 2021, https://doi.org/10.1016/j.jrmge.2021.08.005.

[10] S. M. Cherian and K. Rajitha, "Random forest and support vector machine classifiers for coastal wetland characterization using the combination of features derived from optical data and synthetic aperture radar dataset," *Journal of Water and Climate Change*, vol. 15, no. 1, pp. 29–49, 2024, https://doi.org/10.2166/wcc.2023.238.

[11] E. Elgeldawi, A. Sayed, A. R. Galal, and A. M. Zaki, "Hyperparameter tuning for machine learning algorithms used for arabic sentiment analysis," *Informatics*, vol. 8, no. 4, pp. 1–21, 2021, https://doi.org/10.3390/informatics8040079.

[12] E. S. El-Kenawy and M. Eid, "Hybrid gray wolf and particle swarm optimization for feature selection," *International Journal of Innovative Computing, Information and Control*, vol. 16, no. 3, pp. 831–844, 2020, https://doi.org/10.24507/ijicic.16.03.831.

[13] M. Weiel, M. Götz, A. Klein, D. Coquelin, R. Floca, and A. Schug, "biomolecular simulation parameters with flexible objective functions," *Nature Machine Intelligence*, vol. 3, no. August, pp. 727–734, 2021, https://doi.org/10.1038/s42256-021-00366-3.

[14] E. M. Senan, M. H. Al-Adhaileh, F. W. Alsaade, T. H. Aldhyani, A. A. Alqarni, N. Alsharif, M. I. Uddin, A. H. Alahmadi, M. E. Jadhav, and M. Y. Alzahrani, "Diagnosis of Chronic Kidney Disease Using Effective Classification Algorithms and Recursive Feature Elimination Techniques," *Journal of Healthcare Engineering*, vol. 2021, 2021, https://doi.org/10.1155/2021/1004767.

[15] J. Xia, L. Sun, S. Xu, Q. Xiang, J. Zhao, W. Xiong, Y. Xu, and S. Chu, "A model using support vector machines recursive feature elimination (Svm-rfe) algorithm to classify whether copd patients have been continuously managed according to gold guidelines," *International Journal of COPD*, vol. 15, pp. 2779–2786, 2020, https://doi.org/10.2147/COPD.S271237.

[16] S. Zhao and Z. Zhao, "A Comparative Study of Landslide Susceptibility Mapping Using SVM and PSO-SVM Models Based on Grid and Slope Units," *Mathematical Problems in Engineering*, vol. 2021, 2021, https://doi.org/10.1155/2021/8854606.

[17] M. F. Akbar, M. I. Mazdadi, H. Saragih, and F. Abadi, "Implementation of Information Gain Ratio and Particle Swarm Optimization in the Sentiment Analysis Classification of Covid-19 Vaccine Using Support Vector Machine," vol. 5, no. 4, pp. 261–270, 2023.

[18] J. Wang, X. Wang, X. Li, and J. Yi, "A Hybrid Particle Swarm Optimization Algorithm with Dynamic Adjustment of Inertia Weight Based on a New Feature Selection Method to Optimize SVM Parameters," *Entropy*, vol. 25, no. 3, 2023, https://doi.org/10.3390/e25030531.

[19] C. Aroniadi and G. N. Beligiannis, "Applying Particle Swarm Optimization Variations to Solve the Transportation Problem Effectively," *Algorithms*, vol. 16, no. 8, 2023, https://doi.org/10.3390/a16080372.

[20] A. Performance, E. Alshdaifat, D. Alshdaifat, A. Alsarhan, F. Hussein, and S. Moh, "The Effect of Preprocessing Techniques , Applied to Numeric," *Data*, vol. 6, no. 11, 2021.

[21] M. Ramkumar Prabhu, R. Sivaraman, N. Nagabhooshanam, R. Sampath Kumar, and S. Sathish Salunkhe, "Empowering artificial intelligence-based multi-biometric image sensor for human identification," *Measurement: Sensors*, vol. 33, no. April, p. 101082, 2024, https://doi.org/10.1016/j.measen.2024.101082.

[22] U. M. Khaire and R. Dhanalakshmi, "Stability of feature selection algorithm: A review," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 4, pp. 1060–1073, 2022, https://doi.org/10.1016/j.jksuci.2019.06.012.

[23] K. Shah, H. Patel, D. Sanghvi, and M. Shah, "A Comparative Analysis of Logistic Regression , Random Forest and KNN Models for the Text Classification," *Augmented Human Research*, vol. 0, 2020, https://doi.org/10.1007/s41133-020-00032-0.

[24] J. Alcaraz, M. Labbé, and M. Landete, "Support Vector Machine with feature selection : A multiobjective approach," vol. 204, no. September 2021, 2022.

[25] M. Pan, C. Li, R. Gao, Y. Huang, H. You, T. Gu, and F. Qin, "Photovoltaic power forecasting based on a support vector machine with improved ant colony optimization," *Journal of Cleaner Production*, vol. 277, 2020, https://doi.org/10.1016/j.jclepro.2020.123948.

[26] M. Jain, V. Saihjpal, N. Singh, and S. B. Singh, "An Overview of Variants and Advancements of PSO Algorithm," *Applied Sciences (Switzerland)*, vol. 12, no. 17, pp. 1–21, 2022, https://doi.org/10.3390/app12178392.

[27] H. Reza, R. Zaman, and F. Soleimanian, *An improved particle swarm optimization with backtracking search optimization algorithm for solving continuous optimization problems*. Springer London, 2021, no. 0123456789, https://doi.org/10.1007/s00366-021-01431-6.

[28] N. Van Tinh, "Enhanced Forecasting Accuracy of Fuzzy Time Series Model Based on Combined Fuzzy C-Mean Clustering with Particle Swam Optimization," *International Journal of Computational Intelligence and Applications*, vol. 19, no. 2, 2020, https://doi.org/10.1142/S1469026820500170.

[29] S. Surono, K. W. Goh, C. W. Onn, A. Nurraihan, N. S. Siregar, A. B. Saeid, and T. T. Wijaya, "Optimization of Markov Weighted Fuzzy Time Series Forecasting Using Genetic Algorithm (GA) and Particle Swarm Optimization (PSO)," *Emerging Science Journal*, vol. 6, no. 6, pp. 1375–1393, 2022, https://doi.org/10.28991/ESJ-2022-06-06-010.

[This page intentionally left blank.]