STATISTICS, OPTIMIZATION AND INFORMATION COMPUTING
Stat., Optim. Inf. Comput., Vol. **14**, October 2025, pp 0–17.
Published online in International Academic Press (www.IAPress.org)

Kernel ridge regression improving based on golden eagle optimization algorithm for multi-class classification

Shaimaa Waleed Mahmood, Zakariya Yahya Algamal*

Department of Statistics and Informatics, University of Mosul, Iraq

Abstract Kernel Ridge Regression combines the principles of machine learning supervision and ridge regression by employing the kernel trick. This approach is particularly effective for regression problems with non-linear relationships between inputs and outputs. The kernel trick allows Kernel Ridge Regression (KRR) to perform ridge regression by learning non-linear functions in a high-dimensional space using ridge regression's regularization techniques. The success of KRR depends on the hyper-parameter settings, which determine the type of kernel used. Current methods for determining hyper-parameter values encounter three primary challenges: high computational costs, large memory demands, and low accuracy. This research introduces a significant improvement to the golden eagle optimization framework by incorporating elite opposite-based learning (EOBL) to enhance population diversity in the search space. We apply this strategy to efficiently select optimal hyper-parameters. Combining EOBL with KRR can lead to improved predictive accuracy. By selecting elite solutions and incorporating opposition-based methodologies, the model can circumvent local optima and broaden the range of potential solutions, leading to improved results, especially in complex datasets. The proposed enhancement to Kernel Ridge Regression was evaluated on ten publicly available multi-class datasets to demonstrate its effectiveness. The results from various evaluation criteria showed that the proposed enhancement achieved superior classification performance compared to all baseline techniques.

Keywords regularization, multi-class, golden eagle, polynomial, optimizer

AMS 2010 subject classifications 62Exx, 03E72

DOI: 10.19139/soic-2310-5070-2569

1. Introduction

Researchers have shown significant interest in kernel ridge regression-based binary or multi-class classifiers in recent years because of their non-iterative learning approach [1]. Kernel ridge regression is a method that combines conventional least squares regression and ridge regression[2]. (KRR) has gained popularity as a powerful nonlinear forecasting technique applicable in various scenarios. The main concept behind (KRR) is to utilize a versatile collection of nonlinear prediction functions while also mitigating over fitting through penalization, hence restricting computational complexity. This is accomplished by transforming the collection of predictors into a high-dimensional (or infinite dimensional in some cases) space consisting of the nonlinear functions of the predictors [3].

Kernel ridge regression offers a significant benefit in that the kernel function does not need to adhere to Mercer's theorem, and there is no necessity for randomness in allocating connection weights between input and hidden layers [4]. Kernel approaches are a powerful framework for improving the modeling capacity by transferring the data from the original space to a higher-dimensional feature space known as the reproducing kernel Hilbert space (RKHS)

*Correspondence to: Zakariya Yahya Algamal (zakariya.algamal@uomosul.edu.iq). Department of Statistics and Informatics, University of Mosul, Iraq

ISSN 2310-5070 (online) ISSN 2311-004X (print)

[45, 46, 5, 6]. Kernel estimation is a preliminary nonparametric technique used to estimate statistical functions in the presence of nonlinearity in the data [7, 32, 33, 34]. The kernel trick is a technique employed to implicitly project the input characteristics into a higher-dimensional space without explicitly calculating the transformation. This is accomplished by utilizing a kernel function that computes the dot product of the modified feature vectors [8].

Nonlinearity and multimodality hinder the effectiveness of standard approaches like the hill-climbing method, trapping in incorrect solutions. Another increasingly difficult issue comes when the quantity of decision factors expands or when n is exceedingly large. Furthermore, the situation is exacerbated by the presence of nonlinearity combined with the intricate nature of the big scale. Met-heuristic and heuristic algorithms are computational techniques that draw inspiration from natural systems to solve complex problems. These algorithms employ a combination of trial-and-error, learning, and adaptation to consistently discover satisfactory or optimal solutions, making them very reliable for a diverse set of challenging optimization issues [9, 35, 36].

In recent years, many meta-heuristic optimization algorithms have been employed in different research, the binary (PSO) - (WOA) for feature selection and predictive modeling. Researchers at Grand Canyon University did a linear regression analysis involving 250 students to examine their interactions with AI in their academic pursuits [10, 11]. introduced a novel nature-inspired optimization technique known as the Greylag Goose Optimization (GGO) algorithm.

The Greylag Goose serves as the inspiration for the suggested algorithm (GGO), a swarm-based algorithm. We initially evaluate the GGO technique by applying it to nineteen datasets from the UCI Machine Learning Repository. Each dataset has a diverse array of attributes, instances, and classes utilized for feature selection. We are contrasting the (GGO) with various other algorithms.

The (PO) is introduced as a novel optimization algorithm inspired by the intelligence and behavior of pumas. This method incorporates distinctive and robust techniques in each phase of exploration and exploitation, enhancing its performance across various optimization challenges [12].

In the following sections of the paper, Partition 2 will provide a detailed explanation of the kernel ridge regression. Partition 3 presents the steps of the golden eagle optimizer algorithm. Partition 4 discusses the proposed methodology. Partition 5 provides an exposition of the experiment's outcomes, specifically focusing on the multi-class data. Partition 6 contains the paper's conclusion.

2. Kernel ridge regression (KRR)

Ridge regression (RR) is a classical technique for modeling data that addresses the issue of multicollinearity among factors in a sample. Multicollinearity occurs when there is a significant correlation between numerous predictor variables in a multiple regression model [5, 37, 38]. Provided training data,

$$H = \{x_t, y_t\}_{t=1}^{m} \quad , x \in X \subset R^h,\ y_t \in Y \subset R$$

The (RR) approach calculates the parameter vector, $u \in R^h$, of a linear model $f(x)$ by minimizing the objective function [8, 13, 39, 40].

$$G(u) = \frac{1}{2}\|u\|^2 + \frac{\lambda}{m}\sum_{t=1}^{m}(y_t - f(x))^2 \tag{1}$$

The objective function employed in (RR) Eq. (1) incorporates Tikhonov regularization to minimize the sum of squared errors (sse) metric. The tuning parameter $\lambda$is used to regulate the trade-off between bias and variance [14, 15]. This refers to the penalized maximum likelihood estimation (MLE) of the variable u, under the assumption that the observed targets have been affected by a random sample from a Gaussian noise process. The noise process is independent and identically distributed (i.i.d.), with a mean of zero and a variance ofσ^2 [8].

$$y_t = f(x) + \varepsilon_i \quad , \varepsilon \sim N(0, \sigma_2)$$

KRR, a variant of ridge regression, can be derived using the "kernel trick." This involves constructing a linear ridge regression model in a high-dimensional feature space,$F(\phi : X \to F)$, created by a non-linear kernel function

that defines the inner product $K(x_t, x_j) = \phi(x_t).\phi(x_j)$. The kernel function,$K(x_t, x_j)$, can be any positive, definite "Mercer" kernel. The goal function that is minimized when developing a kernel ridge regression model is expressed as

$$G(u) = \frac{1}{2}\|u\|^2 + \frac{\lambda}{m}\sum_{t=1}^{m}(y_t - f(\phi(x)))^2 \tag{2}$$

The represented theorem implies that the solution to an optimization problem of this kind may be expressed as a linear combination of the training patterns, denoted as $u = \sum_{t=1}^{m}\alpha_t\phi(x_t)$and $\alpha = (K + \lambda I)^{-1} y$, with the identity matrix I[13].

3. Golden Eagle Optimizer (GEO)

The golden eagle, a member of the Accipitridae family, is a professional hunter with exceptional vision, high speed, and powerful talons. With a flight speed of 190 km/h, it can catch a wide range of prey. It also uses a spiral trajectory to hunt, monitoring nearby boulders and prey for a proper angle of attack while also scouting for better food sources[16, 17].

Golden eagles effectively manage their inclination to attack and glide in order to capture the most optimal prey within an acceptable timeframe and with a little expenditure of energy. Their hunting expedition commences with soaring in expansive loops within their domain, diligently scouring for potential prey. Upon detection, they proceed along the circumference of an imaginary circle with the prey in its center. The eagle commits the prey's location to memory and proceeds to orbit it, gradually decreasing its circumference. In addition, they conduct surveys of adjacent areas in search of superior alternatives, occasionally collaborating with other eagles by divulging the optimal hunting grounds. Eagles execute the ultimate assaults along a linear trajectory [18]. GEO algorithm was proposed by Mohammadi-Balani, Nayeri [18]. The implementation of the GEO algorithm involves following a series of stages.

1- GEO is derived from the spiral motion shown by golden eagles. As previously said, every golden eagle commits to memory the optimal spot it has previously visited. The eagle exhibits a dual inclination, both towards hunting and towards soaring in quest of more favorable sustenance. During each iteration, every individual golden eagle, denoted as$i$, randomly chooses the prey targeted by another golden eagle, denoted as f. Subsequently, $i$circles about the most optimal spot that $f$has visited thus far. The golden eagle i has the ability to navigate its own memories, thereby resulting in the equation $f \in \{1, 2, ..., PopSize\}$.

2- During each iteration, a golden eagle is required to select a victim from the flock's memory in order to execute cruise and assault maneuvers. Each golden eagle retains the best solution in its memory, representing the most optimal solutions discovered thus far. With respect to the chosen prey, the golden eagles compute the attack and cruise vectors. The memory is revised if the new location is superior to the prior one. The implementation of this stochastic objective mapping approach enables golden eagles to effectively navigate the terrain.

3- The golden eagle represents the attack with a vector that originates at its current position and terminates at the location of the prey stored in its memory. Equation (3) computes the attack vector for golden eagle$i$.

$$\vec{A}_i = \vec{X}_f^* - \vec{X}_i \tag{3}$$

The attack vector of eagle$i$ is represented by$\vec{A}_i$, the best location visited thus far by eagle $f$is denoted as$\vec{X}_f^*$, and the current position of eagle $i$is given by$\vec{X}_i$. The attack vector directs the population of golden eagles to the most frequently frequented places, emphasizing the exploiting stage in GEO.

4- The cruise vector is determined from the attack vector. Cruise vectors are perpendicular to assault vectors and tangent to circles. The golden eagle's cruise is its linear speed relative to prey. To calculate the cruise vector in n-dimensions, we must first find the equation of the tangent hyperplane to the circle to find the equation of a hyperplane in n-dimensions, use an arbitrary point and a perpendicular vector, known as the normal vector.

Equation (4) shows the scalar hyperplane equation in n-dimensional space.

$$h_1x_1 + h_2x_2 + ... + h_nx_n = d \Rightarrow \sum_{j=1}^{n} h_jx_j = d \tag{4}$$

The normal vector, denoted as $\vec{H} = [h_1, h_2, ..., h_n]$, represents the variables vector$X = [x_1, x_2, ..., x_n]$, and the arbitrary point on the hyper plane is $\vec{P} = [p_1, p_2, ..., p_n]$. The dot product of $\vec{H}$ and $\vec{P}$, defined as d, can be expressed as$\sum_{j=1}^{n} h_jx_j$. If we designate $\vec{X}_i$ as the position of the eagle i on the hyperplane and $\vec{A}_i$as the attack vector, which is perpendicular to the hyper plane, it is possible to demonstrate that $\vec{C}_i^t$, the cruise vector for the golden eagle i in iteration t, belongs to a specific hyper plane as defined by Equation Eq.(5).

$$\sum_{j=1}^{n} a_jx_j = \sum_{j=1}^{n} a_j^t x_j^* \tag{5}$$

The attack vector is represented by$\overrightarrow{A}_i = [a_1, a_2, ..., a_n]$, the decision-variables vector is represented by$X = [x_1, x_2, ..., x_n]$, and the position of the selected prey is represented by$X* = [x_1^*, x_2^*, ..., x_n^*]$.

Once the cruise hyper plane for eagle $i$in iteration $t$is determined, the cruise vector for this golden eagle must be determined. Golden eagles can choose any cruise hyperplane destination. Finding a random vector on the cruise hyperplane requires finding a new destination point C, other than the current location of the golden eagle i. Note that the cruise vector begins at the present location of the golden eagle i.

The generation of random points is impossible because hyperplanes exist with one fewer dimension than their surrounding space. The cruise hyperplane does not contain every possible random point in n-dimensional space. Equation (4) defines the hyperplane equation as the method to determine the last dimension of new points on the n-dimensional cruise hyperplane that contains degrees of freedom. The last dimension selection to fulfill the hyperplane equation creates 1 free variables and one fixed variable. The following method helps us generate a random n-dimensional destination point Cthat lies on the cruise hyperplane for golden eagle i.

a) Select one variable from a set of $n$variables at random to be designated as the fixed variable. The index of the selected variable is denoted as$k$. It is important to understand that the fixed variable cannot be selected from the variables whose corresponding element in the attack vector $\vec{A}_i$ is 0. The hyper plane in Equation Eq. (4) is parallel to the axis of a variable when its coefficient is 0. In this case, the variable can have any value while the other variables are randomly combined.

b) Assign arbitrary values to all variables except the k-th variable, as the k-th variable remains constant.

c) Determine the value of the constant variable by utilizing Equation Eq. (6).

$$C_k = \frac{d - \sum_{j,j\neq k} a_j}{a_k} \tag{6}$$

The destination point C has elementsC_k, where C_k represents the k-th element. The attack vector $\vec{A}_i$ has elementsa_j, where a_j represents the j-th element. The right-hand side of Equation (4) is denoted as d. The attack vector $\vec{A}_i$ also has elements a_k^t, where a_k^t represents the k-th element. The index of the fixed variable is represented by k. The random destination point on the cruise hyper plane has been located. Equation (7) presents the overall depiction of the target location on the cruise hyper plane.

$$\vec{C}_i = \left(c_1 = random, c_2 = random, ..., C_k = \frac{d - \sum_{j,j\neq k} a_j}{a_k}, ..., c_n = random \right) \tag{7}$$

Once the destination point has been established, the cruise vector for the golden eagle $i$in iteration t can now be calculated. The components of the resulting destination point consist of random numbers ranging from 0 to 1. The cruise vector of the golden eagle population is known to draw them to regions that are different from those they remember, thereby highlighting the exploration stage of GEO.

5- The golden eagles execute their movements through attack and vector operations. The step vector for golden eagle during iteration defines by Equation (8).

$$\Delta x_i = \vec{r}_1 p_a \frac{\vec{A}_i}{\left\|\vec{A}_i\right\|} + \vec{r}_2 p_c \frac{\vec{C}_i}{\left\|\vec{C}_i\right\|} \tag{8}$$

In each iteration t, p_a^t represents the attack coefficient and p_c^t represents the cruise coefficient. These coefficients determine the impact of attack and cruise on golden eagles. The vectors $\vec{r}_1$and $\vec{r}_2$are randomly generated with elements that range from 0 to 1. The topics of pa and pc will be addressed at a later point. Equation (9) determines the Euclidean norm of the attack vector,$\left\|\vec{A}_i\right\|$, and the Euclidean norm of the cruise vector, $\left\|\vec{C}_i\right\|$.

$$\left\|\vec{A}_i\right\| = \sqrt{\sum_{j=1}^{n} a_j^2}, \quad \left\|\vec{C}_i\right\| = \sqrt{\sum_{j=1}^{n} c_j^2} \tag{9}$$

The addition of the step vector from iteration $t$to the positions in iteration t determines the position of the golden eagles in iteration $t+1$.

$$x^{t+1} = x^t + \Delta x_i^t \tag{10}$$

The golden eagle updates its memory position with a new location when the fitness value of this new position exceeds its current stored position. The memory stays unchanged as the eagle chooses to settle at the new position. Each golden eagle in the latest version randomly picks another eagle from the population to orbit its most visited location. The algorithm calculates attack vector and cruise vector and the step vector to establish the new position for the next iteration. The loop runs until any termination condition becomes valid.

6- GEO employs the utilization of $p_a$and $p_c$to transition from the phase of exploration to the phase of exploitation. The procedure commences by initializing the values of p_a as low and p_c as high. Through the iterations, the value of p_a is incrementally raised while the value reached p_c is gradually lowered. Users specify the starting and ending values for each parameter. The linear transition described in Equation (11)enables the calculation of intermediate values.

$$\begin{cases} p_a = p_a^0 + \frac{t}{T}\left|p_a^T - p_a^0\right| \\ p_c = p_c^0 + \frac{t}{T}\left|p_c^T - p_c^0\right| \end{cases} \tag{11}$$

Let $t$represent the current iteration, Trepresent the maximum iterations, $p_a^0$and p_a^Trepresent the initial and final values for propensity to attackp_a, respectively, $p_c^0$and p_c^Trepresent the initial and final values for propensity to cruisep_c, respectively. Equation (11) induces linear variations in the parameters. Nevertheless, they can be altered exponentially or through any alternative mathematical function.

4. The Proposed Method

The hyper-parameter variables that determine the kernel type have a significant impact on the optimization solution for the objective function (Eq (2)). Currently, there is no dependable mathematical approach for accurately finding the exact values of these hyper-parameters. [19, 20]. Although these choices can have a significant impact on the kernel function's performance. Therefore, the examination of (KRR) models is contingent upon the selection of these hyper-parameters. The predominant techniques in the literature for determining hyper-parameters are grid search (GS), generalized cross validation (GCV), and cross validation (CV). However, studies have demonstrated that this procedure is not consistently fast [21].

The natural world has inspired the development of various meta-heuristic algorithms. Swarm intelligence is an indispensable tool in research for effectively addressing a variety of complex challenges. [22, 23, 40, 41, 42]. Researchers have thoroughly researched and effectively applied swarm intelligence techniques to tackle a wide range of complex optimization issues.

The Elite Opposition-Based Learning (EOBL) approach is an enhanced iteration of the Opposition-Based Learning (OBL) strategy, initially proposed by Tizhoosh [24]. The OBL technique serves as a machine intelligence method which seeks to boost optimization algorithm efficiency. The optimization algorithm starts randomly by evaluating both solutions for opposite outcomes and better solutions from current options. The evaluation function helps us determine which solution is better before selecting it for the next iteration.

To develop a mathematical model of OBL, follow these steps: Within the existing population, let $t = (t_1, t_2, ..., t_B)$ denote a specific location. B represents the problem of dimensional space and $t \in [w_i, e_i]$, where $i = 1, 2, ..., B$. The opposition point $\breve{t} = (\breve{t}_1, \breve{t}_2, ..., \breve{t}_B)$ may therefore be described by the equation that follows:

$$\breve{t}_i = w_i + e_i - t_i \tag{12}$$

EOBL hires an exceptional individual to guide the public to the most effective solution on a global scale. The privileged few are more likely to possess useful knowledge. EOBL generates equal and opposite particles within the search dimension by utilizing the most optimal member of the existing population. Therefore, the particles will be directed by a select few until they reach a potential location where the most favorable solution could be found. Consequently, employing the EOBL technique will enhance population diversity and promote exploration in the GEO algorithm. Several optimization strategies in the literature have utilized EOBL. This work utilizes the EOBL technique to augment the exploring capabilities of GEO algorithms. The opposition position might be defined as follows: in favor of the individual $T_i = (t_{i,1}, t_{i,2}, ..., t_{i,B})$ within the present population $T_m = (t_{m,1}, t_{m,2}, ..., t_{m,B})$; thus, The concept of the elite opposite position can be $\breve{T}_i = (\breve{t}_{i,1}, \breve{t}_{i,2}, ..., \breve{t}_{i,B})$ defined as

$$\breve{t}_{i,j} = S \times (dw_j + de_j) - t_{i,j} \tag{13}$$

$S \in [0, 1]$ and S represent factors of generality. The variables dw_j and de_j represent dynamic boundaries that are defined as

$$dw_j = \min(t_{i,j}), \quad de_j = \max(t_{i,j}) \tag{14}$$

However, the outcome may surpass the search bounds$[w_i, e_i]$. To resolve this problem, a random value is assigned to the transferred individual within the specified range of $[w_i, e_i]$, which is indicated as

$$\breve{t}_{i,j} = rand(w_j, e_j), \quad if \;\; \breve{t}_{i,j} < w_j || \breve{t}_{i,j} > e_j \tag{15}$$

EOBL segregates itself from rival solutions to promote diversity within the population. Given this, the GEO now possesses enhanced exploratory capacity. Kernel hyper-parameter population initialization is an essential operation that provides valuable insights into the results of subsequent phases. It is essential to alter the initial population's quality depending on the issue's nature. The GEO exhibits seamless convergence, a comprehensive scope, a time-consuming process for final resolution, and a reliable solution. The distribution of the initial population derived from the adjacency matrix needs evaluation to determine its ability to spread across the search space while avoiding local or suboptimal solution areas. Different genetic optimization algorithms commonly use this practice to establish their initial conditions before running the original algorithm [43, 44].

Simplifying the issue domain involves highlighting the representation of opposites in a binary manner. Afterward, we implement these fields of study in other locations to generate more comprehensive benefits for the spectrum of potential solutions. EOBL improves the GEO method by expanding the search space, allowing for optimal exploration of a wider variety of superior potential solutions. Therefore, by considering this inverse in the search space, we enhance the search's capacity to differentiate between different applications and aid in avoiding convergence to a local optimum.

This section provides a comprehensive explanation of the GEO approach's structure to enhance the recommended model's performance. We alter the GEO's elemental composition by including the EOBL technique to address and improve its limitations, such as its inability to reach the optimal value or explore the search domain quickly. The recommended approach, KRR-EOGEO, improves upon the GEO algorithm through a two-step process: Initially, we employ EOBL to initialize the population of the kernel hyper-parameters, aiming to accelerate convergence. Additionally, it prevents the GEO method from becoming trapped in a local optimum

by doing a comprehensive search across the whole search space domain for potential solutions. Furthermore, it employs the widely recognized conventional GEO technique to update the pelican's position. The EOBL is also utilized to ascertain if the fitness function value of the opposing solution is better than the new solution.

The initial stage randomly generates the first population. The EOBL (Eq. 13) determines the formation of the opposing population by using elite individuals from opposite locations. The second stage utilizes EOBL to generate elite opponent positions while updating pelican positions through the original GEO algorithm steps. The KRR-EOGEO selects its optimal population and opposite population through fitness function value assessment to create the new population. KRR-EOGEO operates with a time complexity of $O(n \times \log n) + O(t \times n \times d)$. The process can be summarized through Figure 1.

5. Experimental results

The experimental analysis of ten real multi-class data sets evaluates the effectiveness of KRR-EOGEO against KRR-CV, KRR-GCV, KRR-GS and KRR-GEO.

5.1. Multi-class Data Description

In this experiment, we employed ten multi-class real-world datasets. We sourced all downloadable data sets from the UCI repository [25, 26]. Table 1 presents a comprehensive overview of the features, including the number of samples and the number of multi-class instances for each data set. We normalized all data set as $x_{new} = \left(\frac{x - x_{\min}}{x_{\max} - x_{\min}}\right)$ to reduce the discrepancy between the variable values.

Table 1. Multi-class Data Description

Data	Samples	Features	Class
Thyroid	215	5	3
Wine	178	13	3
Seeds	210	7	3
Balance	625	4	3
Soybean2	136	35	4
Glass	214	9	4
Dermatology	366	34	6
Yeast	1484	8	10
Vowel	528	10	11
Soybean1	266	35	15

5.2. validation criteria

The evaluation of our proposed KRR-EOGEO method used both classification accuracy and F-measure criteria against KRR-CV, KRR-GCV, and KRR-GS approaches [4, 27, 28]. We can represent these criteria mathematically as follows:

$$AC = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \tag{16}$$

$$F - measure = \frac{2 \times TP}{2 \times TP + FP + FN} \tag{17}$$

TP represents the number of true positives, FP represents the number of false positives, TN represents the number of true negatives, and FN represents the number of false negatives.

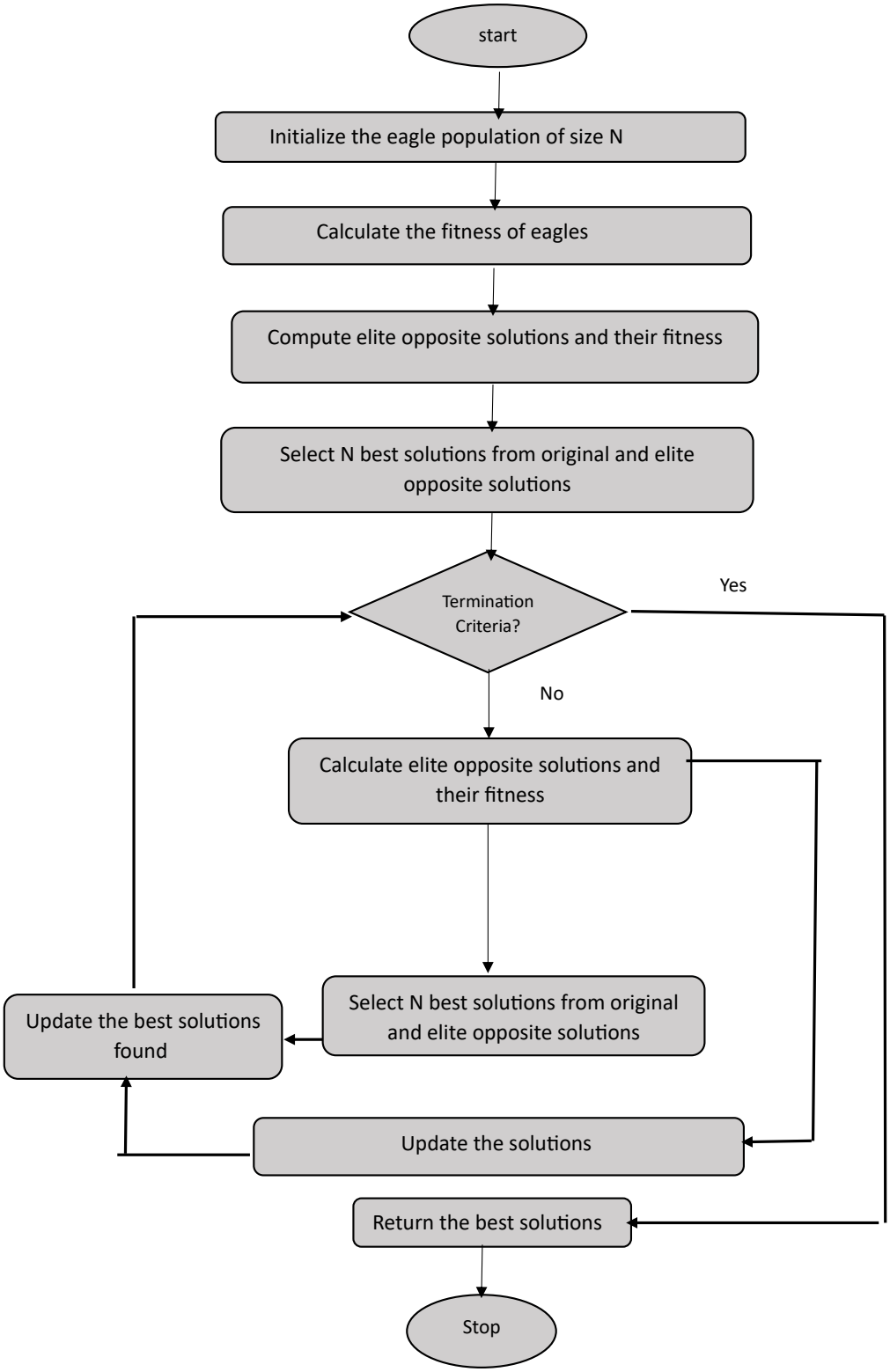

Figure 1. Flowchart of the KRR-EOGEO algorithm.

5.3. performance comparisons

We separated the data into train and test sets where the train set contained 70% of the data and the test set contained 30% of the data for evaluating the proposed approach against alternative methods. This division was done randomly. This section employs both the radial basis function kernel $K(x_t, x_j) = e^{-\theta\|x_t - x_j\|^2}$ and the polynomial kernel $K(x_t, x_j) = ((x_t, x_j) + u)^d$ [29]. For the value of θ and d, they were as $\theta > 0$ and $d = 2, 3, 4$.

In Tables 2-5, we can see the results for each technique and dataset with two kernel functions in terms of classification accuracy, F-measure and standard deviation.

Our proposed KRR-EOGEO method demonstrates the highest AC and F-measure values across all datasets. The suggested KRR-EOGEO method demonstrates the highest mean AC value (89.946) (89.898) and the highest mean F-measure value (0.9078) (0.8988) compared to other similar methods. According to AC, the mean value of KRR-EOGEO improves by (2.526%) (1.451%), (4.596%) (4.872%), (2.797%) (3.252%) and (6.544%) (6.380%) compared to KRR-GEO, KRR-CV, KRR-GCV, and KRR-GS, respectively. In addition, the F-measure shows that the mean value of KRR-EOGEO is higher than KRR-GEO, KRR-CV, KRR-GCV, and KRR-GS by (0.0142%) (0.0123%), (0.0443%) (0.0406%), (0.0326%) (0.0277%), and (0.0677%) (0.0568%) respectively. Further, according to the standard deviation values in Tables 2-5, it is noted that the proposed scheme (KRR-EOGEO) obtained the lowest values for all classifiers. This shows very clearly that the proposed KRR-EOGEO is more reliable than the other approaches. It also turns out that KRR-EOGEO regulates the amount of population diversity, too. The deceasing nature of a rule for opposite solutions also assists the developed technique in avoiding high diversity or perturbation of search agents during the searching phases. This strategy can help make exploration in a project plan equal, that is, balance with the last phase, which is exploitation.

Hence, we can employ the EOGEO technique to enhance the optimization of the KRR model, leading to improved classification performance and enhanced generalization ability. Therefore, the speed of computation is acceptable. The KRR-EOGEO offers several benefits for addressing real-life problems, particularly in small industries because of its ability to model complex, non-linear relationships through the use of kernel functions and the epsilon-insensitive loss function used in KRR allows the model to ignore small errors, making it robust to noise in the data. The bar graph in Figures 2-5 illustrates how long it takes to compute each strategy and data set for two function kernels.

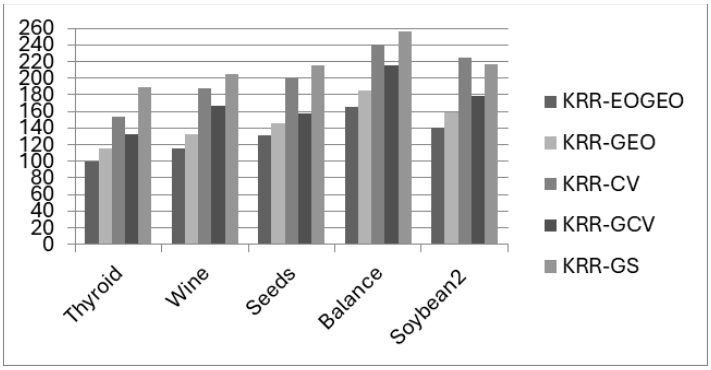

Figure 2. Time Calculate for the first group based on radial basis kernel

Analysis presented through Figures 2-5 demonstrates that KRR-EOGEO achieves quick training duration. KRR-EOGEO achieves accurate global optimum determination without requiring exhaustive examination of all parameter combinations in the grid. KRR-EOGEO achieves faster computation times than KRR-GEO and KRR-CV as well as both KRR-GCV and KRR-GS at a comparable or higher predictive ability level. Training durations between the other approaches show minimal differences.

A statistical evaluation of KRR-EOGEO's effectiveness takes place to determine its performance. The objects are present [30, 31].

(1) The Friedman test functions as a statistical method to analyze and evaluate multiple classifier performance.

Table 2. Classification Accuracies for training and testing set based on radial basis kernel

Data	Methods	Train Set	Test Set
Thyroid	KRR-EOGEO	**97.59 ± 2.09**	**96.59 ± 1.99**
	KRR-GEO	96.85 ± 1.89	95.41 ± 2.41
	KRR-CV	89.89 ± 3.19	88.95 ± 3.01
	KRR-GCV	94.87 ± 0.44	92.11 ± 1.79
	KRR-GS	88.50 ± 2.89	86.72 ± 2.67
Wine	KRR-EOGEO	**90.23 ± 3.04**	**88.65 ± 3.11**
	KRR-GEO	88.11 ± 2.78	87.59 ± 2.45
	KRR-CV	86.33 ± 4.17	85.81 ± 4.01
	KRR-GCV	87.16 ± 3.21	86.15 ± 2.51
	KRR-GS	84.04 ± 2.91	82.58 ± 2.30
Seeds	KRR-EOGEO	**95.76 ± 1.25**	**94.05 ± 1.69**
	KRR-GEO	94.54 ± 2.61	93.99 ± 2.44
	KRR-CV	92.88 ± 3.16	90.97 ± 3.01
	KRR-GCV	93.85 ± 2.32	91.91 ± 2.30
	KRR-GS	89.58 ± 2.81	88.37 ± 2.11
Balance	KRR-EOGEO	**94.79 ± 1.03**	**92.66 ± 1.22**
	KRR-GEO	93.52 ± 2.38	91.18 ± 2.19
	KRR-CV	90.76 ± 3.28	89.08 ± 3.20
	KRR-GCV	92.77 ± 1.69	90.35 ± 1.29
	KRR-GS	89.03 ± 1.10	88.00 ± 1.02
Soybean2	KRR-EOGEO	**90.95 ± 3.69**	**88.88 ± 3.35**
	KRR-GEO	88.89 ± 2.87	87.90 ± 2.55
	KRR-CV	85.49 ± 1.87	84.62 ± 1.66
	KRR-GCV	87.44 ± 5.04	85.78 ± 4.89
	KRR-GS	83.59 ± 2.09	81.96 ± 2.45
Glass	KRR-EOGEO	**75.79 ± 5.34**	**74.66 ± 5.05**
	KRR-GEO	74.70 ± 4.79	73.15 ± 4.33
	KRR-CV	71.53 ± 3.89	69.58 ± 3.51
	KRR-GCV	72.34 ± 6.42	70.81 ± 6.11
	KRR-GS	70.33 ± 4.11	68.58 ± 3.96
Dermatology	KRR-EOGEO	**96.93 ± 2.10**	**94.89 ± 2.22**
	KRR-GEO	95.58 ± 3.01	94.50 ± 3.13
	KRR-CV	92.78 ± 1.92	91.68 ± 1.62
	KRR-GCV	94.62 ± 1.13	92.89 ± 1.49
	KRR-GS	90.18 ± 1.52	88.39 ± 1.84
Yeast	KRR-EOGEO	**64.92 ± 1.87**	**62.99 ± 1.71**
	KRR-GEO	63.56 ± 1.24	62.31 ± 1.28
	KRR-CV	57.98 ± 2.06	56.79 ±2.19
	KRR-GCV	59.89 ± 1.65	58.59 ± 1.77
	KRR-GS	56.54 ± 2.31	55.43 ± 1.98
Vowel	KRR-EOGEO	**99.61 ± 0.71**	**97.89 ± 1.04**
	KRR-GEO	99.29 ± 0.89	97.51 ± 0.99
	KRR-CV	98.01 ± 1.24	96.58 ± 1.07
	KRR-GCV	99.10 ± 0.64	97.24 ± 0.93
	KRR-GS	96.54 ± 1.67	95.21 ±1.33
Soybean1	KRR-EOGEO	**92.89** ± 2.79	**91.00 ± 2.42**
	KRR-GEO	91.03 ± 1.72	90.36 ± 1.82
	KRR-CV	87.85 ± 4.05	86.39 ± 4.12
	KRR-GCV	89.45 ± 3.54	87.74 ± 3.20
	KRR-GS	85.69 ± 3.66	84.89 ± 3.58

Table 3. Classification Accuracies for training and testing set based on polynomial kernel

Data	Methods	Train Set	Test Set
Thyroid	KRR-EOGEO	**98.29 ± 2.19**	**97.10 ± 1.88**
	KRR-GEO	97.06 ± 1.98	96.11 ± 1.55
	KRR-CV	93.89 ± 2.89	92.31 ± 2.66
	KRR-GCV	95.44 ± 0.58	93.16 ± 0.89
	KRR-GS	90.89 ± 1.89	88.89 ± 1.77
Wine	KRR-EOGEO	**89.32 ± 4.03**	**88.24** ± 3.69
	KRR-GEO	87.18 ± 2.37	86.37 ± 2.56
	KRR-CV	85.01 ± 3.26	84.32 ± 3.36
	KRR-GCV	86.09 ± 2.13	84.89 ± 2.15
	KRR-GS	83.06 ± 1.99	81.85 ± 2.03
Seeds	KRR-EOGEO	**94.67 ± 2.51**	**92.79 ± 1.92**
	KRR-GEO	93.17 ± 1.69	91.88 ± 2.03
	KRR-CV	91.38 ± 2.57	90.84 ± 2.71
	KRR-GCV	92.83 ± 3.22	91.13 ± 3.87
	KRR-GS	88.89 ± 2.18	87.73 ± 2.33
Balance	KRR-EOGEO	**95.59 ± 3.01**	**93.76 ± 2.21**
	KRR-GEO	94.66 ± 2.83	93.08 ± 2.24
	KRR-CV	91.79 ± 2.77	90.27 ± 2.56
	KRR-GCV	92.97 ± 1.74	91.33 ± 1.25
	KRR-GS	90.32 ± 1.44	89.14 ± 1.86
Soybean2	KRR-EOGEO	**91.59 ± 3.24**	**89.89 ± 3.53**
	KRR-GEO	89.97 ± 2.78	88.11 ± 2.35
	KRR-CV	86.02 ± 3.81	85.40 ± 2.99
	KRR-GCV	87.84 ± 4.39	86.79 ± 4.11
	KRR-GS	84.93 ± 1.11	83.28 ± 0.91
Glass	KRR-EOGEO	**75.43 ± 4.45**	**74.19 ± 4.00**
	KRR-GEO	73.89 ± 4.11	72.77 ± 4.35
	KRR-CV	70.35 ± 2.65	69.31 ± 2.51
	KRR-GCV	71.54 ± 3.71	70.61 ± 3.39
	KRR-GS	69.08 ± 1.65	68.10 ± 1.88
Dermatology	KRR-EOGEO	**97.39 ± 0.91**	**95.98 ± 0.86**
	KRR-GEO	96.22 ± 1.72	95.89 ± 1.39
	KRR-CV	93.07 ± 1.44	92.55 ± 1.78
	KRR-GCV	94.97 ± 2.13	93.41 ± 1.95
	KRR-GS	91.81 ± 1.25	88.87 ± 1.48
Yeast	KRR-EOGEO	**66.96 ± 1.81**	**64.92 ± 1.77**
	KRR-GEO	64.68 ± 1.38	63.05 ± 1.82
	KRR-CV	58.84 ± 2.60	57.58 ± 2.91
	KRR-GCV	61.81 ± 0.99	59.95 ± 1.33
	KRR-GS	57.77 ± 3.44	56.97 ± 3.34
Vowel	KRR-EOGEO	**97.84 ± 1.79**	**96.81 ± 1.61**
	KRR-GEO	97.01 ± 0.98	96.18 ± 1.10
	KRR-CV	95.02 ± 1.39	94.85 ± 1.81
	KRR-GCV	96.59 ± 2.11	95.73 ± 1.93
	KRR-GS	94.49 ± 1.88	93.29 ± 1.32
Soybean1	KRR-EOGEO	**91.90 ± 1.72**	**90.41 ± 1.83**
	KRR-GEO	90.63 ± 1.59	89.88 ± 1.67
	KRR-CV	84.89 ± 3.84	83.73 ± 3.66
	KRR-GCV	86.38 ± 3.51	85.83 ± 3.29
	KRR-GS	83.94 ± 2.79	82.99 ± 2.94

Table 4. F-measure values for training and testing set based on radial basis kernel

Data	Methods	Train Set	Test Set
Thyroid	KRR-EOGEO	**0.961 ± 1.99**	**0.952 ± 1.87**
	KRR-GEO	0.954 ± 1.81	0.943 ± 1.65
	KRR-CV	0.909 ± 2.19	0.893 ± 2.71
	KRR-GCV	0.938 ± 0.49	0.921 ± 1.97
	KRR-GS	0.894 ± 1.89	0.879 ± 1.43
Wine	KRR-EOGEO	**0.933 ± 2.04**	**0.921 ± 2.51**
	KRR-GEO	0.929 ± 2.81	0.914 ± 2.31
	KRR-CV	0.881 ± 0.78	0.870 ± 0.91
	KRR-GCV	0.908 ± 1.91	0.893 ± 2.11
	KRR-GS	0.871 ± 1.41	0.864 ± 1.53
Seeds	KRR-EOGEO	**0.956 ± 2.52**	**0.942 ± 2.31**
	KRR-GEO	0.944 ± 2.21	0.935 ± 2.14
	KRR-CV	0.927 ± 3.05	0.917 ± 3.04
	KRR-GCV	0.935 ± 2.66	0.926 ± 2.03
	KRR-GS	0.909 ± 1.71	0.882 ± 1.91
Balance	KRR-EOGEO	**0.959 ± 3.11**	**0.939 ± 2.94**
	KRR-GEO	0.938 ± 2.68	0.921 ± 2.44
	KRR-CV	0.892 ± 3.66	0.873 ± 3.31
	KRR-GCV	0.919 ± 1.72	0.901 ± 1.84
	KRR-GS	0.871 ± 2.81	0.862 ± 2.59
Soybean2	KRR-EOGEO	**0.914 ± 2.62**	**0.899 ± 2.51**
	KRR-GEO	0.888 ± 1.91	0.870 ± 1.84
	KRR-CV	0.865 ± 1.12	0.850 ± 1.48
	KRR-GCV	0.879 ± 2.94	0.854 ± 2.55
	KRR-GS	0.849 ± 1.73	0.830 ± 1.87
Glass	KRR-EOGEO	**0.779 ± 3.45**	**0.761** ± 3.83
	KRR-GEO	0.756 ± 3.69	0.749 ± 3.21
	KRR-CV	0.729 ± 2.51	0.713 ± 2.68
	KRR-GCV	0.741 ± 2.91	0.734 ± 2.78
	KRR-GS	0.701 ± 1.88	0.691 ± 1.79
Dermatology	KRR-EOGEO	**0.972 ± 2.11**	**0.959 ± 2.33**
	KRR-GEO	0.965 ± 3.11	0.945 ± 3.29
	KRR-CV	0.924 ± 1.77	0.910 ± 1.20
	KRR-GCV	0.949 ± 2.39	0.931 ± 2.59
	KRR-GS	0.904 ± 1.89	0.887 ± 1.56
Yeast	KRR-EOGEO	**0.684 ± 1.78**	**0.666 ± 1.88**
	KRR-GEO	0.669 ± 1.04	0.652 ± 1.30
	KRR-CV	0.630 ± 1.91	0.619 ± 1.83
	KRR-GCV	0.645 ± 2.15	0.633 ± 2.11
	KRR-GS	0.618 ± 1.84	0.605 ± 1.65
Vowel	KRR-EOGEO	**0.986 ± 0.94**	**0.973 ± 0.67**
	KRR-GEO	0.974 ± 1.61	0.964 ± 1.51
	KRR-CV	0.942 ± 1.69	0.928 ± 1.77
	KRR-GCV	0.966 ± 2.18	0.954 ± 1.97
	KRR-GS	0.925 ± 1.43	0.911 ± 1.58
Soybean1	KRR-EOGEO	**0.934** ± 2.61	**0.921 ± 2.89**
	KRR-GEO	0.919 ± 1.95	0.904 ± 1.28
	KRR-CV	0.879 ± 1.59	0.854 ± 1.79
	KRR-GCV	0.897 ± 3.11	0.876 ± 3.43
	KRR-GS	0.859 ± 2.31	0.839 ± 2.12

Table 5. F-measure values for training and testing set based on polynomial kernel

Data	Methods	Train Set	Test Set
Thyroid	KRR-EOGEO	**0.979 ± 1.25**	**0.951 ± 1.67**
	KRR-GEO	0.968 ± 2.94	0.945 ± 2.32
	KRR-CV	0.939 ± 1.97	0.927 ± 2.11
	KRR-GCV	0.947 ± 3.14	0.931 ± 3.02
	KRR-GS	0.911 ± 1.98	0.891 ± 2.05
Wine	KRR-EOGEO	**0.891 ± 5.03**	**0.872 ± 4.89**
	KRR-GEO	0.878 ± 4.16	0.852 ± 3.81
	KRR-CV	0.854 ± 2.62	0.830 ± 2.53
	KRR-GCV	0.869 ± 2.77	0.840 ± 2.90
	KRR-GS	0.832 ± 0.96	0.818 ± 1.09
Seeds	KRR-EOGEO	**0.947 ± 0.89**	**0.921 ± 0.58**
	KRR-GEO	0.934 ± 1.78	0.917 ± 1.59
	KRR-CV	0.912 ± 1.68	0.901 ± 1.74
	KRR-GCV	0.920 ± 3.08	0.910 ± 3.19
	KRR-GS	0.898 ± 2.43	0.879 ± 2.58
Balance	KRR-EOGEO	**0.951 ± 2.11**	**0.939 ± 2.84**
	KRR-GEO	0.945 ± 1.79	0.933 ± 1.89
	KRR-CV	0.920 ± 3.24	0.909 ± 3.80
	KRR-GCV	0.935 ± 1.28	0.914 ± 1.48
	KRR-GS	0.902 ± 0.78	0.888 ± 0.55
Soybean2	KRR-EOGEO	**0.908 ± 2.99**	**0.882 ± 2.17**
	KRR-GEO	0.894 ± 2.87	0.875 ± 2.59
	KRR-CV	0.861 ± 1.81	0.853 ± 1.99
	KRR-GCV	0.879 ± 3.93	0.869 ± 3.81
	KRR-GS	0.849 ± 1.64	0.838 ± 1.58
Glass	KRR-EOGEO	**0.765 ± 3.71**	**0.749 ± 3.87**
	KRR-GEO	0.754 ± 3.43	0.731 ± 3.25
	KRR-CV	0.727 ± 2.79	0.711 ± 2.81
	KRR-GCV	0.736 ± 0.97	0.720 ± 0.85
	KRR-GS	0.719 ± 2.11	0.700 ± 2.09
Dermatology	KRR-EOGEO	**0.979 ± 0.88**	**0.960 ± 0.76**
	KRR-GEO	0.961 ± 1.86	0.952 ± 1.84
	KRR-CV	0.931 ± 2.04	0.923 ± 1.95
	KRR-GCV	0.948 ± 1.58	0.936 ± 1.77
	KRR-GS	0.911 ± 1.29	0.890 ± 1.41
Yeast	KRR-EOGEO	**0.676 ± 0.89**	**0.659 ± 0.90**
	KRR-GEO	0.658 ± 1.84	0.635 ± 1.61
	KRR-CV	0.624 ± 2.30	0.590 ± 2.09
	KRR-GCV	0.639 ± 1.21	0.619 ± 1.50
	KRR-GS	0.615 ± 2.78	0.583 ± 2.61
Vowel	KRR-EOGEO	**0.974** ± 0.97	**0.966 ± 0.62**
	KRR-GEO	0.965 ± 1.05	0.956 ± 120
	KRR-CV	0.945 ± 1.58	0.932 ± 1.87
	KRR-GCV	0.957 ± 2.13	0.947 ± 2.44
	KRR-GS	0.935 ± 2.01	0.923 ± 2.29
Soybean1	KRR-EOGEO	**0.918 ±1.87**	**0.906 ± 1.72**
	KRR-GEO	0.908 ± 1.66	0.895 ± 1.86
	KRR-CV	0.869 ± 2.87	0.853 ± 2.90
	KRR-GCV	0.881 ± 3.00	0.870 ± 3.14
	KRR-GS	0.848 ± 2.41	0.829 ± 2.55

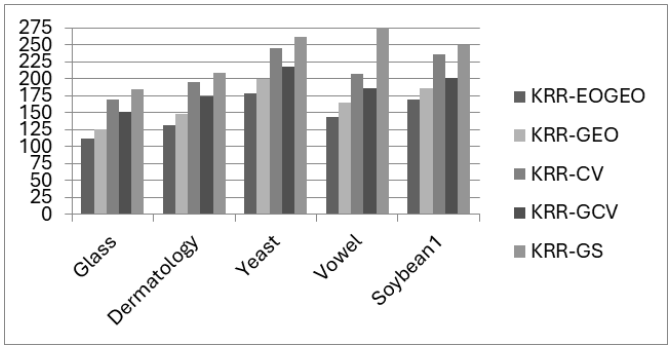

Figure 3. Time Calculate for the second group based on radial basis kernel

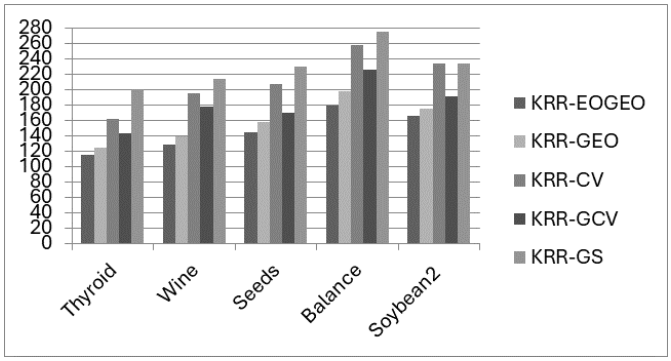

Figure 4. Time Calculate for the first group based on polynomial kernel.

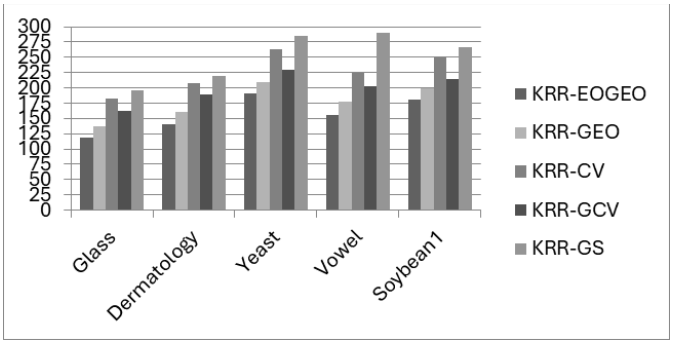

Figure 5. Time Calculate for the second group based on polynomial kernel.

(2) The Wilcoxon signed-rank test is a statistical test that does not rely on specific assumptions about the distribution of the data. We employ the Wilcoxon signed-rank test (WST) to identify significant differences between KRR-EOGEO and the other discussed approaches.

The results of the WST are displayed in Tables 6 and 7, with a significant level of$\alpha = 0.05$. Tables 6 and 7 indicate the proposed approach, which is based on AUC, has more statistical significance compared to other approaches. The p-values from all approaches remain below 0.05.

The KRR-EOGEO algorithm demonstrates superior performance compared to other algorithms based on Table 8 statistical results because its WST test values remain below 0.05 for most datasets. The statistical data shows KRR-EOGEO achieves superior performance compared to the 4 comparison algorithms.

Table 6. ρ-values for the Wilcoxon signed-rank test of the suggested approach results when compared to four competitor approaches based on radial basis kernel.

Pairwise comparison	ρ-Value
KRR-EOGEO vs KRR-GEO	0.0065
KRR-EOGEO vs KRR-CV	0.0212
KRR-EOGEO vs KRR-GCV	0.0448
KRR-EOGEO vs KRR-GS	0.0314

Table 7. ρ-values for the Wilcoxon signed-rank test of the suggested approach results when compared to four competitor approaches based on polynomial kernel

Pairwise comparison	ρ-Value
KRR-EOGEO vs KRR-GEO	0.0134
KRR-EOGEO vs KRR-CV	0.0245
KRR-EOGEO vs KRR-GCV	0.0324
KRR-EOGEO vs KRR-GS	0.0355

Table 8. ρ-values for the Wilcoxon signed-rank test of comparisons among methods based on computational time.

Pairwise comparison	ρ-Value for radial basis kernel	ρ-Value for polynomial kernel
KRR-EOGEO vs KRR-GEO	0.0094	0.00118
KRR-EOGEO vs KRR-CV	0.00314	0.00378
KRR-EOGEO vs KRR-GCV	0.00403	0.00393
KRR-EOGEO vs KRR-GS	0.00264	0.00297

Table 9 shows the classification accuracy performance by 4 popular metaheuristic algorithms, namely WaOA, GA, SHO, and GSA. Experimental results show that KRR-EOGEO is significantly superior to other algorithms.

6. Conclusion

KRR is an adaptation of ridge regression and classification incorporating the kernel trick. Consequently, it acquires knowledge of the linear function within the feature space determined by the kernel above and the data. The remaining candidate non-linear kernels express this as a non-linear function in the original data domain. However, the KRR model's success relies heavily on the hyper-parameters determining the kernel type, making them a critical factor. This study presents an enhanced golden eagle optimization technique incorporating elite opposite-based learning to identify the optimal values for these hyper-parameters. We evaluated the efficacy of the suggested method, KRR-EOGEO, on ten publicly available datasets with multi-class. We compared the KRR-EOGEO algorithm with other established approaches, such as KRR-GEO, KRR-CV, KRR-GCV, and KRR-GS, regarding CA and F-measure. The comparison studies and evaluations demonstrated that the KRR-EOGEO outperformed other methods. The results also showed that using both EOBL and GEO together improved the GEO algorithm's performance and solution quality, which led to fast convergence on the ideal solution. Although of the best performance of KRR-EOGEO, but KRR-EOGEO is subject to several limitations. The KRR model requires users to adjust both the regularization parameter and the kernel parameters during its implementation. KRR training

Table 9. Comparisons with other metaheuristic algorithm based on classification accuracy

Data	Methods	Train Set	Test Set
Thyroid	KRR-EOGEO	**97.59 ± 2.09**	**96.59 ± 1.99**
	WaOA	96.97 ± 1.80	95.85 ± 2.40
	GA	95.89 ± 2.15	94.75 ± 2.05
	SHO	96.50 ± 0.84	95.09 ± 1.49
	GSA	94.12 ± 2.66	93.00 ± 1.37
Wine	KRR-EOGEO	**90.23 ± 3.04**	**88.65 ± 3.11**
	WaOA	88.19 ± 2.88	87.00 ± 2.55
	GA	86.64 ± 4.02	85.79 ± 3.88
	SHO	87.58 ± 2.31	86.23 ± 2.81
	GSA	88.95 ± 2.11	87.48 ± 1.30
Seeds	KRR-EOGEO	**95.76 ± 1.25**	**94.05 ± 1.69**
	WaOA	94.22 ± 2.31	93.54 ± 2.40
	GA	94.77 ± 3.14	93.87 ± 3.11
	SHO	93.81 ± 1.38	92.98 ± 2.20
	GSA	91.98 ± 2.72	90.07 ± 2.09
Balance	KRR-EOGEO	**94.79 ± 1.03**	**92.66 ± 1.22**
	WaOA	93.36 ± 2.25	92.14 ± 1.19
	GA	91.11 ± 2.28	89.99 ± 2.31
	SHO	93.68 ± 1.83	92.55 ± 1.18
	GSA	92.53 ± 1.56	91.00 ± 1.42
Soybean2	KRR-EOGEO	**90.95 ± 3.69**	**88.88 ± 3.35**
	WaOA	88.18 ± 2.41	87.66 ± 2.50
	GA	89.41 ± 1.37	88.31 ± 1.66
	SHO	87.21 ± 4.04	85.98 ± 3.81
	GSA	87.49 ± 2.12	86.78 ± 2.31
Glass	KRR-EOGEO	**75.79 ± 5.34**	**74.66 ± 5.05**
	WaOA	73.93 ± 3.71	72.16 ± 3.43
	GA	74.84 ± 3.12	73.58 ± 3.92
	SHO	72.84 ± 4.12	71.85 ± 4.01
	GSA	70.93 ± 4.03	68.78 ± 3.95
Dermatology	KRR-EOGEO	**96.93 ± 2.10**	**94.89 ± 2.22**
	WaOA	94.00 ± 2.61	93.02 ± 2.43
	GA	92.98 ± 1.22	91.60 ± 1.58
	SHO	91.45 ± 1.41	90.85 ± 1.72
	GSA	95.22 ± 1.32	94.39 ± 1.71
Yeast	KRR-EOGEO	**64.92 ± 1.87**	**62.99 ± 1.71**
	WaOA	63.43 ± 1.89	62.38 ± 1.67
	GA	61.88 ± 2.36	60.58 ±2.29
	SHO	59.96 ± 1.10	58.58 ± 1.89
	GSA	57.97 ± 2.51	56.00 ± 2.61
Vowel	KRR-EOGEO	**99.61 ± 0.71**	**97.89 ± 1.04**
	WaOA	97.49 ± 1.89	96.09 ± 1.58
	GA	97.15 ± 1.42	95.89 ± 1.61
	SHO	99.17 ± 1.68	97.58 ± 1.92
	GSA	98.24 ± 1.69	97.21 ±1.45
Soybean1	KRR-EOGEO	**92.89** ± 2.79	**91.00 ± 2.42**
	WaOA	91.53 ± 1.78	90.46 ± 1.88
	GA	90.85 ± 4.00	88.99 ± 4.08
	SHO	89.40 ± 3.58	87.93 ± 3.25
	GSA	88.00 ± 2.69	86.89 ± 2.78

using elite opposite-based learning approaches requires large amounts of computational resources when it handles complex or extensive datasets. The future work will integrate filter-based feature selection with KRR-EOGEO. A multi-stage multi-objective GEO based feature selection approach will be integrated with KRR.

REFERENCES

1. Gautam, C., A. Tiwari, and M. Tanveer, *KOC,:Kernel ridge regression based one-class classification using privileged information.*, Information Sciences, 2019. 504: p. 324-333.
2. Exterkate, P., et al., *Nonlinear forecasting with many predictors using kernel ridge regression* , International Journal of Forecasting, 2016. 32(3): p. 736-753.
3. Exterkate, P., *Modelling issues in kernel ridge regression.* , 2011.
4. Mohapatra, P., S. Chakravarty, and P. Dash, *Microarray medical data classification using kernel ridge regression and modified cat swarm optimization based gene selection system*, Swarm and Evolutionary Computation, 2016. 28: p. 144-160.
5. He, J., et al., *Kernel ridge regression classification. in 2014 International Joint Conference on Neural Networks (IJCNN)*, 2014. IEEE.
6. Shawe-Taylor, J. and N. Cristianini, *Kernel methods for pattern analysis.* , 2004: Cambridge university press.
7. Muhammad, L.A. and S.H. Kazem, *Estimate Kernel Ridge Regression Function in Multiple Regression*, Journal of Economics and Administrative Sciences, 2018. 24(103): p. 411-411.
8. Cawley, G.C., et al., *Heteroscedastic kernel ridge regression*, Neurocomputing, 2004. 57: p. 105-124.
9. Yang, X.-S., *Engineering optimization: an introduction with metaheuristic applications*, 2010: John Wiley Sons.
10. Towfek, S., et al., *AI in Higher Education: Insights from Student Surveys and Predictive Analytics using PSO-Guided WOA and Linear Regression*, Journal of Artificial Intelligence in Engineering Practice, 2024. 1(1): p. 1-17.
11. El-Kenawy, E.-S.M., et al., *Greylag goose optimization: nature-inspired optimization algorithm*, Systems with Applications, 2024. 238: p. 122147.
12. Abdollahzadeh, B., et al., *Puma optimizer (PO): A novel metaheuristic optimization algorithm and its application in machine learning*, Cluster Computing, 2024: p. 1-49.
13. Cawley, G.C. and N.L. Talbot, *Reduced rank kernel ridge regression*, Neural Processing Letters, 2002. 16: p. 293-302.
14. Geman, S., E. Bienenstock, and R. Doursat, *Neural networks and the bias/variance dilemma*, Neural computation, 1992. 4(1): p. 1-58.
15. Tikhonov, A.N. and V. Arsenin, *Solutions of ill-posed problems*, New York: John Wiley, 1977.
16. Tack, J., et al., *Ecosystem processes, land cover, climate, and human settlement shape dynamic distributions for golden eagle across the western US*, Animal conservation, 2020. 23(1): p. 72-82.
17. Tikkanen, H., et al., *Modelling golden eagle habitat selection and flight activity in their home ranges for safer wind farm planning*, Environmental Impact Assessment Review, 2018. 71: p. 120-131.
18. Mohammadi-Balani, A., et al., *Golden eagle optimizer: A nature-inspired metaheuristic algorithm*, Computers Industrial Engineering, 2021. 152: p. 107050.
19. Algamal, Z.Y., et al., *Improving grasshopper optimization algorithm for hyperparameters estimation and feature selection in support vector regression*, Chemometrics and Intelligent Laboratory Systems, 2021. 208.
20. Ismael, O.M., O.S. Qasim, and Z.Y. Algamal, *Improving Harris hawks optimization algorithm for hyperparameters estimation and feature selection in v-support vector regression based on opposition-based learning*, Journal of Chemometrics, 2020. 34(11).
21. Al-Thanoon, N.A., O.S. Qasim, and Z.Y. Algamal, *Tuning parameter estimation in SCAD-support vector machine using firefly algorithm with application in gene selection and cancer classification*, Computers in Biology and Medicine, 2018. 103: p. 262-268.
22. Algamal, Z.Y., *A new method for choosing the biasing parameter in ridge estimator for generalized linear model*, Chemometrics and Intelligent Laboratory Systems, 2018. 183: p. 96-101.
23. Kahya, M.A., S.A. Altamir, and Z.Y. Algamal, *Improving firefly algorithm-based logistic regression for feature selection*, Journal of Interdisciplinary Mathematics, 2020. 22(8): p. 1577-1581.
24. Tizhoosh, *H.R. Opposition-based learning: a new scheme for machine intelligence*, in International conference on computational intelligence for modelling, control and automation and international conference on intelligent agents, web technologies and internet commerce (CIMCA-IAWTIC'06). 2005. IEEE.
25. Dua, D. and C. Graff, *UCI machine learning repository [http://archive. ics. uci. edu/ml]*, Irvine, CA: University of California, School of Information and Computer Science. IEEE transactions on pattern analysis and machine intelligence, 2019. 1(1): p. 1-29.
26. Frank, A., *Uci machine learning repository. irvine, ca: University of california, school of information and computer science*, http://archive. ics. uci. edu/ml, 2010.
27. Hazarika, B.B. and D. Gupta, *Affinity based fuzzy kernel ridge regression classifier for binary class imbalance learning*, Engineering Applications of Artificial Intelligence, 2023. 117: p. 105544.
28. Subasi, A., *Automatic recognition of alertness level from EEG by using neural network and wavelet coefficients*, Expert systems with applications, 2005. 28(4): p. 701-711.
29. Maalouf, M. and D. Homouz, *Kernel ridge regression using truncated newton method*, Knowledge-Based Systems, 2014. 71: p. 339-344.
30. Demšar, J., *Statistical comparisons of classifiers over multiple data sets*, The Journal of Machine learning research, 2006. 7: p. 1-30.
31. Woolson, R.F., *Wilcoxon signed-rank test*, Wiley encyclopedia of clinical trials, 2007: p. 1-3.
32. Algamal, Z. Y., Asar, Y. *Liu-type estimator for the gamma regression model* , Communications in Statistics-Simulation and Computation, vol. 49, no . 8, p. 2035-2048, 2020.

33. Algamal, Z. Y., Lee, M. H. *A new adaptive L1-norm for optimal descriptor selection of high-dimensional QSAR classification model for anti-hepatitis C virus activity of thiourea derivatives* , SAR and QSAR in Environmental Research, vol. 28, no. 1 , p. 75-90, 2017.
34. Kahya, M. A., Altamir, S. A., Algamal, Z. Y. *Improving whale optimization algorithm for feature selection with a time-varying transfer function* , Numerical Algebra, Control and Optimization, vol. 11, no. 1 , p. 87-98 , 2020.
35. Algamal, Z. Y., Qasim, M. K., Ali, H. T. M. *A QSAR classification model for neuraminidase inhibitors of influenza A viruses (H1N1) based on weighted penalized support vector machine* , SAR and QSAR in Environmental Research, vol. 28 , no. 5, p. 415-426 , 2017.
36. Algamal, Z. Y., Lee, M. H., Al-Fakih, A. M. *High-dimensional quantitative structure–activity relationship modeling of influenza neuraminidase a/PR/8/34 (H1N1) inhibitors based on a two-stage adaptive penalized rank regression* , Journal of Chemometrics, vol. 30 , no.2 m p. 50-57 , 2016.
37. Algamal, Z. Y., Lee, M. H., Al-Fakih, A. M., Aziz, M. *High-dimensional QSAR classification model for anti-hepatitis C virus activity of thiourea derivatives based on the sparse logistic regression model with a bridge penalty* , Journal of Chemometrics, vol. 31 , p.6, p. e2889 , 2017.
38. Algamal, Z. Y., Qasim, M. K., Lee, M. H., Ali, H. T. M. *High-dimensional QSAR/QSPR classification modeling based on improving pigeon optimization algorithm* , Chemometrics and Intelligent Laboratory Systems, vol. 206, p. 104170 , 2020.
39. Ismael, O. M., Qasim, O. S., Algamal, Z. Y. *Improving Harris hawks optimization algorithm for hyperparameters estimation and feature selection in v-support vector regression based on opposition-based learning* , Journal of Chemometrics, vol 34 , no. 11, e3311, 2020.
40. Abonazel, M. R., Algamal, Z. Y., Awwad, F. A., Taha, I. M. *A new two-parameter estimator for beta regression model: method, simulation, and application* , Frontiers in Applied Mathematics and Statistics, vol. 7,p. 780322 ,2022.
41. Algamal, Z. Y., Abonazel, M. R. *Developing a Liu-type estimator in beta regression model* , Concurrency and Computation: Practice and Experience, vol.34 ,no. 5 , p. e6685.
42. Algamal, Z., Ali, H. M. *An efficient gene selection method for high-dimensional microarray data based on sparse logistic regression* , Electronic Journal of Applied Statistical Analysis, vol . 10 , no . 1 , 242-256 , 2017.
43. Salih, A. M., Algamal, Z., Khaleel, M. A. *A new ridge-type estimator for the gamma regression model* , Iraqi Journal for Computer Science and Mathematics, vol . 5 , no .1 , p. 85-98.
44. Alkhateeb, A., Algamal, Z. *). Jackknifed Liu-type estimator in Poisson regression model* , Journal of the Iranian Statistical Society,Vol .11 ,no . 1, p. 21-37, 2022.
45. Mahmood, S. W., Basheer, G. T., Algamal, Z. Y. *Quantitative Structure–Activity Relationship Modeling Based on Improving Kernel Ridge Regression* , Journal of Chemometrics, vol. 39, no. 5, p. e70027, 2025.
46. Mahmood, S. W., Basheer, G. T., Algamal, Z. Y. *Improving kernel ridge regression for medical data classification based on meta-heuristic algorithms* , Kuwait Journal of Science, vol. 52, no. 3 , p. 100408 , 2025