http://doi.org/10.35784/iapgos.5729 *received: 28.12.2023 | revised: 10.05.2024 | accepted: 24.05.2024 | available online: 30.06.2024*

IMPROVING PARAMETERS OF V-SUPPORT VECTOR REGRESSION WITH FEATURE SELECTION IN PARALLEL BY USING QUASI-OPPOSITIONAL AND HARRIS HAWKS OPTIMIZATION ALGORITHM

Omar Mohammed Ismael[1], Omar Saber Qasim[2], Zakariya Yahya Algamal[3]
[1]Ministry of Education, Directorate of Education in Nineveh, Mosul, Iraq, [2]University of Mosul, Department of Mathematics, Mosul, Iraq, [3]University of Mosul, Department of Statistics and Informatics, Mosul, Iraq

Abstract. *Numerous real-world problems have been addressed using support vector regression, particularly v-support vector regression (v-SVR), but some parameters need to be manually changed. Furthermore, v-SVR does not support feature selection. Techniques inspired from nature were used to identify features and hyperparameter estimation. The quasi-oppositional Harris hawks optimization method (QOBL-HHOA) is introduced in this research to embedding the feature selection and optimize the hyper-parameter of the v-SVR at a same time. Results from experiments performed using four datasets. It has been demonstrated that, in terms of prediction, the number of features that may be chosen, and execution time, the suggested algorithm performs better than cross-validation and grid search methods. When compared to other nature-inspired algorithms, the experimental results of the QOBL-HHOA show its efficacy in improving prediction accuracy and processing time. It demonstrates QOBL-ability as well. By searching for the optimal hyper-parameter values, HHOAs can locate the features that are most helpful for prediction tasks. As a result, the QOBL-HHOA algorithm may be more appropriate than other algorithms for identifying the data link between the features of the input and the desired variable. Whereas, the numerical results showed superiority this method on these methods, for example, mean square error of QOBL-HHOA method results (2.05E-07) with influenza neuraminidase data set was the better than the others. For making predictions in other real-world situations, this is incredibly helpful.*

Keywords: v-support vector regression, Harris hawks algorithm, hyper-parameter selection, quasi-oppositional learning

POPRAWA PARAMETRÓW REGRESJI WEKTORA NOŚNEGO V Z RÓWNOLEGŁYM WYBOREM CECHY POPRZEZ WYKORZYSTANIE ALGORYTMU QUASI-OPOZYCYJNEGO I ALGORYTMU OPTYMALIZACJI HARRIS HAWKS

Streszczenie. *Liczne problemy występujące w świecie rzeczywistym rozwiązano za pomocą regresji wektora nośnego, w szczególności regresji wektora nośnego v (v-SVR), ale niektóre parametry wymagają ręcznej zmiany. Ponadto v-SVR nie obsługuje wyboru funkcji. Do identyfikacji cech i estymacji hiperparametrów wykorzystano techniki inspirowane naturą. W tym badaniu wprowadzono quasi-opozycyjną metodę optymalizacji Harris Hawks (QOBL-HHOA), aby osadzić selekcję cech i jednocześnie optymalizować hiperparametr v-SVR. Wyniki eksperymentów przeprowadzono przy użyciu czterech zbiorów danych. Wykazano, że pod względem predykcji, liczby możliwych do wybrania cech oraz czasu wykonania zaproponowany algorytm sprawdza się lepiej niż metody krzyżowej walidacji i wyszukiwania siatki. W porównaniu z innymi algorytmami inspirowanymi naturą wyniki eksperymentalne QOBL-HHOA pokazują jego skuteczność w poprawianiu dokładności przewidywań i czasu przetwarzania. Wykazuje również zdolność QOBL. Wyszukując optymalne wartości hiperparametrów, HHOA mogą zlokalizować funkcje, które są najbardziej przydatne w zadaniach predykcyjnych. W rezultacie algorytm QOBL-HHOA może być bardziej odpowiedni niż inne algorytmy do identyfikacji łącza danych pomiędzy cechami wejścia a pożądaną zmienną. Natomiast wyniki numeryczne wykazały wyższość tej metody nad wymienionymi metodami, na przykład błąd średniokwadratowy wyników metody QOBL-HHOA (2,05E-07) z zestawem danych dotyczących neuraminidazy grypy był lepszy niż w pozostałych. Jest to niezwykle pomocne przy przewidywaniu innych sytuacji w świecie rzeczywistym.*

Słowa kluczowe: regresja wektora v-nośnego, algorytm Harris hawks, wybór hiperparametrów, uczenie się quasi-opozycyjne

Introduction

Theoretical and practical benefits that explain the support vector machine's better performance in classification and regression have drawn a lot of academics, practitioners, and statisticians in recent years, according to a short description of a technique using support vector machines (SVM) [13]. SVM was initially used to address classification challenges. The Vapnik insensitive loss function (SVR) has been included, making the SVM more capable of addressing the support vector regression (SVR) problem [13, 45].

SVR offers three strengths be: (1) assured converge to the solutions of optimal due to the use of quadratic programming with linear constraints for data learning. (2) Using kernel mapping, nonlinear relationship modeling is computationally efficient. Furthermore, (3) Lower error rates on the test dataset indicate higher generalization performance [43]. Schölkopf, Smola [34] developed v-SVR as a fresh SVR category. The error in training and the amount of support vectors are adjusted in this category via the v of parameter. Numerous hyper-parameters and variables have a substantial impact on the numerical result of the v-SVR and can either both directly and indirectly affect the discovery of the best resolution. The thorough search of the grid is typically used to analyze every hyper-parameter combinations, and cross-validation is performed to test SVR's prediction performance [23]. By gradually focusing on the pertinent subset layers of information, feature selection is a technique and process intended to reduce the complexity and dimensionality of the current data set being monitored [15]. Notwithstanding SVR's amazing characteristics, there are numerous limitations, incorporating the feature selection, for example. To put it another way, SVR is not available to select feature [5]. Choosing a modest many features in regression situations minimizes complexity of computing. For compact and consistent regression models, an optimal feature selection is necessary [4, 21].

Wrapper methods and filter methods are the two primary FS strategies. The primary disadvantage of the filter techniques is that they operate independently of ML classifiers and do not use their input [31]. In several disciplines, machine learning (ML) techniques have been successfully used [35].

Algorithms inspired by nature, which they created by pulling ideas in nature, have piqued the curiosity of researchers and obtained competitive outcomes when handling optimization challenges such as feature selection and hyper-parameter tuning [12, 26, 27]. There have been several studies on tweaking the SVR's hyper-parameters utilizing methods inspired by nature, including as [9, 10, 18, 20, 24, 27, 29, 40, 42, 44]. To advance and expand the exploration and exploitation of current algorithms, scholars have recently begun working on a variety of new nature-inspired algorithms. Between these novel algorithms, the Harris hawks optimization Technique hasbecome popular among these algorithms because of its excellent efficacy [3, 17].

To achieving more solutions on the search space, Quasi-Oppositional (QO) based learning approach was presented by Rahnamayan et al. [33] with attention to the existing population and its quasi oppositional at the same time. To the greatest of knowledge, only a few efforts to concurrently do feature selection and SVR hyper-parameter tuning, particularly for v-support vector regression, however in this case, quasi opposition based learning has been added for these algorithms. The main objective of that job is to implement the feature selection using the QOBL-based Harris hawks technique and to optimize the evolution hyper-parameters of the v-support vector regression. When, the quasi-opposition method is used, it targets the search space, in other words, there is a processing of the search space, which gives an abundance of time by minimizing the search space, and two processes are integrated in data processing and analysis together (features selection, parameter optimization), and this is another addition in providing Time and performance. The remainder of the article was organized as follows: The theoretical element of the v-support vector regression is presented in Section 2. Sections 3 and 4 go over the Harris hawks algorithm and opposition based learning in detail. The proposed algorithm is provided in Section 5. Section 6 presents the experimental outcomes. Section 7 has the conclusion.

1. The v-Support Vector Regression method (v-SVR)

Several categorization problems have been effectively resolved using support vector machines (SVM). Also, the SVM is enhanced to address non-linear regression issues using Vapnik's [41] discovery of the ε-insensitive loss function, which is known as support vector regression. Given a training dataset of n observations $\{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$, the SVR can be obtained by resolving the optimization issue below, where $\mathbf{x}_i = (x_{i,1}, x_{i,2}, ..., x_{i,p}) \in \mathrm{R}^p$ is the vector's i^{th} feature, $y_i \in \mathrm{R}$ for $i = 1, ..., n$ is the variable of target, which is ε-insensitive loss function, and a quantitative variable.

$$\min_{\mathbf{w},b}\left\{\frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{n}\left(\zeta_i + \tilde{\zeta}_i\right)\right\}$$
$$\text{S.T.}\quad \begin{cases} y_i - (\mathbf{w}\varphi(\mathbf{x}_i) + b) \le \varepsilon + \tilde{\zeta}_i \\ (\mathbf{w}\varphi(\mathbf{x}_i) + b) - y_i \le \varepsilon + \zeta_i \\ \zeta_i, \tilde{\zeta}_i \ge 0, \end{cases} \tag{1}$$

Here, $\mathbf{w}$ is the weight vector, b represents bias, and $C > 0$ is a penalty parameter that regulates how training error and model complexity are balanced. $\varphi(\mathbf{x}_i)$ is a nonlinear mapping created by a kernel function ζ_i & $\tilde{\zeta}_i$ are slack variables. The Lagrangian multipliers can resolve Eq. (1) after expressing it as its dual problem.

$$\min_{\tilde{\alpha},\alpha}\frac{1}{2}\sum_{i,j=1}^{n}(\tilde{\alpha}_i - \alpha_i)(\tilde{\alpha}_j - \alpha_j)K(\mathbf{x}_i, \mathbf{x}_j) +$$
$$+\varepsilon\sum_{i=1}^{n}(\tilde{\alpha}_i - \alpha_i) - \sum_{i=1}^{n} y_i(\tilde{\alpha}_i - \alpha_i)$$
$$\text{S.T.}\quad \begin{cases} \sum_{i=1}^{n}(\alpha_i - \tilde{\alpha}_i) = 0 \\ 0 \le \alpha_i, \tilde{\alpha}_i \le C \end{cases} \tag{2}$$

Here α_i, $\tilde{\alpha}_i$ denote Lagrangian multipliers and $K(\mathbf{x}_i, \mathbf{x}_j)$ denotes kernel mapping. The underlying regression problem's regression hyperplane is thus provided by

$$y_i = f(\mathbf{x}_i) = \sum_{\mathbf{x}_i = sv}(\tilde{\alpha}_i + \alpha_i)K(\mathbf{x}_i, \mathbf{x}_j) + b \tag{3}$$

A newly proposed non-linear kernel called (v-SVR) looks across high-dimensional feature space for the optimum hyper-plane of regression that carries the least structural risk [6, 24]. It is impossible to modify the number of support vectors in the SVR with ε-insensitive loss function. [25]. Schölkopf, Smola [34] suggested an upgraded version v-SVR to improve the SVR solution time by regulating the amount of support vectors, raining mistakes, and providing a guess of the in the data. convex quadratic programming is feasible with inequality constraints by:

$$\min_{w,b}\left\{\frac{1}{2}w^T w + c[v\varepsilon + \frac{1}{\mu}\Sigma_{i=1}^{\mu}(\zeta_i + \tilde{\zeta}_i)]\right\}$$
$$S.T\begin{cases} y_i - (\varphi(x_i).w + b) \le \varepsilon + \tilde{\zeta}_i \\ (\varphi(x_i).w + b) - y_i \le \varepsilon + \zeta_i \\ \varepsilon \ge 0, \zeta_i, \tilde{\zeta}_i \ge 0, \end{cases} \tag{4}$$

Whereas $v \in [0,1]$ Schölkopf and Smola [34] demonstrated that v is a lower constraint on the fraction of support vectors and an upper bound on the fraction of margin errors.

After being transformed into its dual problem, the Lagrangian multipliers can be used to solve Equation (4) as follows:

$$\begin{aligned} L(b,\mathbf{w},\varepsilon,\zeta,\tilde{\zeta}) &= \frac{1}{2}\mathbf{w}^T\mathbf{w} \\ &+ C\,(v\varepsilon + \frac{1}{\mu}\sum_{i=1}^{\mu}(\zeta_i + \tilde{\zeta}_i)) - \sum_{i=1}^{\mu}\theta_i\zeta_i - \sum_{i=1}^{\mu}\tilde{\theta}_i\tilde{\zeta}_i - \gamma\varepsilon \\ &+ \sum_{i=1}^{\mu}\alpha_i(\mathbf{w}^T\varphi(\mathbf{x}_i) + b - y_i - \varepsilon - \zeta_i) \\ &+ \sum_{i=1}^{\mu}\tilde{\alpha}_i(\mathbf{w}^T\varphi(\mathbf{x}_i) + b - y_i - \varepsilon - \tilde{\zeta}_i), \end{aligned} \tag{5}$$

Whereas Lagrange multipliers are $\gamma, \theta_i, \tilde{\theta}_i, \alpha_i, \tilde{\alpha}_i \ge 0$. Eq. (5) can be resolved y partially differentiating with respect to $\zeta_i, \tilde{\zeta}_i, \varepsilon, b,$ and $\mathbf{w}$ as shown below.

$$\begin{cases} \dfrac{\partial L}{\partial \mathbf{w}} = \mathbf{w} + \sum_{i=1}^{n}\alpha_i x_i - \sum_{i=1}^{n}\tilde{\alpha}_i x_i = 0 \\ \dfrac{\partial L}{\partial b} = \sum_{i=1}^{n}\alpha_i - \sum_{i=1}^{n}\tilde{\alpha}_i = 0 \\ \dfrac{\partial L}{\partial \varepsilon} = \dfrac{C}{n}\sum_{i=1}^{n} v - \gamma - \sum_{i=1}^{n}(\alpha_i + \tilde{\alpha}_i) = 0 \\ \dfrac{\partial L}{\partial \zeta} = \sum_{i=1}^{n}\dfrac{C}{n} - \sum_{i=1}^{n}\alpha_i - \sum_{i=1}^{n}\theta_i = 0 \\ \dfrac{\partial L}{\partial \tilde{\zeta}} = \sum_{i=1}^{n}\dfrac{C}{n} - \sum_{i=1}^{n}\tilde{\alpha}_i - \sum_{i=1}^{n}\tilde{\theta}_i = 0 \end{cases} \Rightarrow \begin{cases} \mathbf{w} = \sum_{i=1}^{n}(\tilde{\alpha}_i - \alpha_i)x_i \\ \sum_{i=1}^{n}(\tilde{\alpha}_i - \alpha_i) = 0 \\ \sum_{i=1}^{n}(\tilde{\alpha}_i - \alpha_i) \\ = Cv - \gamma \le Cv \\ \alpha_i = \dfrac{C}{n} - \theta_i \le \dfrac{C}{n} \\ \tilde{\alpha}_i = \dfrac{C}{n} - \tilde{\theta}_i \le \dfrac{C}{n} \end{cases} \tag{6}$$

The Lagrange function can be reformulated by processing Eq. (5) through Eq. (6) as follows:

$$\begin{aligned} L &= -\frac{1}{2}\sum_{i,j=1}^{n}(\tilde{\alpha}_i - \alpha_i)(\tilde{\alpha}_j - \alpha_j)K(\mathbf{x}_i, \mathbf{x}_j) \\ &+ \sum_{i=1}^{n}(\tilde{\alpha}_i - \alpha_i)\,y_i \end{aligned} \tag{7}$$

The linked dual issue's best solution can be utilized to solve the objective function's optimization problem given by Eq. (7) under Karush-Kuhn-Tucker conditions. After that, the v-decision SVR's purpose is as follows:

$$y_i = f(\mathbf{x}_i) = \sum_{i=1}^{n}(\tilde{\alpha}_i + \alpha_i)K(\mathbf{x}_i, \mathbf{x}_j) + b \tag{8}$$

2. Optimization algorithm for Harris hawks

Heidari and Mirjalili's description of optimization algorithm for the Harris hawks (HHOA) [17], was created by simulating the actions of Harris Hawks as they hunt and catch rabbits in the wild. To discover the best solution for any given problem, the HHOA goes through three steps of optimization. These three stages are exploration, exploitation, and the change from exploration to exploitation.

2.1. Phase of exploration

The exploring phase mimics a scenario in which a Harris hawk is unable to precisely follow its prey. When that occurs, the hawks pause to watch and look for new prey. The prey is currently the best solution in the HHOA at each step, whilst the hawks are potential solutions. The hawks then use two operators, each of whom is given a chance to perch in a different area at random and wait for a meal [19]. This procedure is mathematically characterized as

$$x^{(t+1)} = \begin{cases} x^t_{rand} - r_1|x^t_{rand} - 2r_2 x^t & q \geq 0.5 \\ (x^t_{prey} - x^t_m) - r_3(L_b + r_4(U_b - L_b)) & q < 0.5 \end{cases} \quad (9)$$

where x^t_{prey} denotes the location of the desired rabbit, x^t_{rand} is the location of a hawk selected at random from the current team, and $x^{(t+1)}$ denotes the position vector of hawks in the subsequent iteration. The integers r_1, r_2, r_3 and r_4 are all random. The search space's upper and lower bounds are L_b and U_b. The following equation determines the average location of the present population of hawks as x^t_m.

$$x^t_m = \frac{1}{nh}\sum_{i=1}^{nh} x^t_i \quad (10)$$

where nh represents all of the group members while x^t_i indicates where each hawk in the group is located.

2.2. Phase of transition

The HHO algorithm switches from the exploration phase to the exploitation phase based on the energy level of the prey (escape energy), E. The following is a definition of the prey's energy loss:

$$E = 2E_0(1 - \frac{t}{t_{\max}}) \quad (11)$$

During each iteration, E_0 stands for the initially energy, which varies at random between (-1,1), and $t_{\max}$ stands for the greatest number of iterations. This value is intended to represent the prey physically identifying for a value of $E_0 \in [-1, 0)$ or strengthening for a value of $E_0 \in [0,1)$. Additionally, If $|E| \geq 1$, HHOA will begin exploring the space of search; if not, it will transition to the exploitation phase [36].

2.3. Phase of exploitation

The letter $|E|$ is taken into consideration while deciding what kind of besiege to employ to catch the target during the exploitation phase. So, when $|E| \geq 0.5$, one that is soft is picked, and when $|E| < 0.5$, one that is hard [16, 28, 32] is picked. Both the soft besiege and strong besiege tactics are effective in promoting this process.

The $r \geq 0.5$ and $|E| \geq 0.5$ conditions of the soft besiege method denote the prey's escape capability (r). As a result, the Harris hawks update their solution by picking the best candidate from the population. This indicates that the prey still has enough energy to flee. Using the following equation, this can be expressed

$$x^{(t+1)} = \Delta x^t - E\left|J\, x^t_{prey} - x^t\right| \quad (12)$$

where $\Delta x^t = x^t_{prey} - x^t$, $J = 2(1 - r_5)$, which denotes the prey's leap intensity during the escape stage, while r_5 is a random value between [0,1].

On the other side, with a rigorous besiege tactic, $r \geq 0.5$ and $|E| < 0.5$, indicate that the victim is worn out and unable to flee due to a lack of energy. The Harris's hawk's most recent location is listed as

$$x^{(t+1)} = x^t_{prey} - E\left|\Delta x^t\right| \quad (13)$$

In the event when $r < 0.5$ and $|E| \geq 0.5$, the falcon gradually chooses its best dive to take the victim in a competitive manner; this technique is referred to as soft besiege with progressive rapid dives [14, 38]. After that, the hawk's new posture is mathematically described as

$$\Upsilon = x^t_{prey} - E\left|J\, x^t_{prey} - x^t\right| \quad (14)$$

The Harris's Hawk is able to capable of diving by.

$$Z = \Upsilon + S \times \text{Levy(D)} \quad (15)$$

While $1 \times D$ is the problem's dimension and S is a random vector of size D, the levy flight function, abbreviated Levy, will be determined as:

$$\text{Levy(D)} = 0.01 \times \frac{\mu}{|\delta|^{1/\beta}} \left[\frac{\Gamma(1+\beta) \times \sin\left(\frac{\pi\beta}{2}\right)}{\Gamma(1+\beta)/2 \times \beta \times 2^{(1+\beta)/2}}\right]^{1/\beta} \quad (16)$$

Whereas β is a constant, and the value of β is 1.5 [14]. μ and δ are random numbers between (0, 1). The Harris's Hawk's location is updated during this phase as

$$x^{(t+1)} \begin{cases} \Upsilon & \text{if} \quad \text{Fitness}(\Upsilon) < \text{Fitness}(x^t) \\ Z & \text{if} \quad \text{Fitness}(Z) < \text{Fitness}(x^t) \end{cases} \quad (17)$$

$\text{Fitness}(x^t)$ is the fitness function in this case.

3. Method of quasi-opposition

Through the use of the oppositional-based learning (O-BL) method, Tizhoosh improved the evolutionary trajectory for the first time [37]. Researchers offered the following opposite-based learning strategy to deal with this issue:

Let $x \in R$ is a real number outlined on a specific interval $a \leq x \leq b$. The definition of the opposite number $\bar{x}$ is as follows:

$$\bar{x} = a + b - x \quad (18)$$

If a = 0 and b = 1, then it will be

$$\bar{x} = 1 - x \quad (19)$$

In a multidimensional example, similar definitions apply to the opposite number. When an n-dimensional system of coordinates with $x_1, x_2, \ldots\ldots\ldots, x_n \in R$ and $x_i \in [a_i, b_i]$ has a point $P(x_1, x_2, \ldots\ldots\ldots, x_n)$ in it. The $\bar{x}_1, \bar{x}_2, \ldots\ldots\ldots, \bar{x}_n$ coordinates of the opposite point $\bar{P}$ fully define it as follows:

$$\bar{x}_i = a_i + b_i - x_i \quad i = 1, \ldots\ldots\ldots, n \quad (20)$$

Last but not least, this approach assumes that g(.) is an evaluation function (such as a reward, error, and fitness function, etc.) used to determine optimality and that $f(x)$ is the fundamental function. Taking the previous into account, we have two values: x, a random initialization value in $[a, b]$, and $\bar{x}$, which is x's opposite. If it is true that $g(f(x))$ is greater than or equal $g(f(\bar{x}))$, x is chosen; otherwise, $\bar{x}$ is. For each iteration, $f(x)$ and $f(\bar{x})$ are computed, and then both are applied to the evaluation function $g(.)$. In order to more successfully attain the applicant's solution by concentrating on the existing population as well as quasi-oppositional, Rahnamayan et al. [33] established a Quasi Oppositional (QO) based learning technique for the first time. It was also shown that a quasi-opposite number generally approaches the solution more closely than an opposing number [30]. According to how the quasi-opposite number, which is a real number's opposite in D-dimensional space, $\overline{x_q}$ is displayed as follows for each real number, such as $P(x_1, x_2, ..., x_n)$, subject to $a_i \le x_i \le b_i$

$$\overline{x_{qi}} = rand\left(\frac{a_i + b_i}{2}, \bar{x}_i\right) \quad i = 1, ..., n \tag{21}$$

4. The proposed algorithm

In SVR, a number of parameters need to be fixed. The term "hyper-parameter" refers to the combination of the kernel parameter, the loss function (ε) that is ε-insensitive, and the penalized parameter (C). The exact desired values cannot be calculated using a mathematically based method, which makes these hyper-parameters were chosen particularly sensitive to SVR performance [39]. Therefore, choosing those hyper-parameters is an essential component of the SVR study. The literature has a number of attempts using various techniques to enhance SVR performance by the wise selection of these hyper-parameters [11, 12, 22, 26, 27, 39]. These different techniques, including natural-inspired algorithms, were utilized to select the SVR hyper-parameter [8-10, 18, 20, 24, 27, 29, 40, 42, 44].

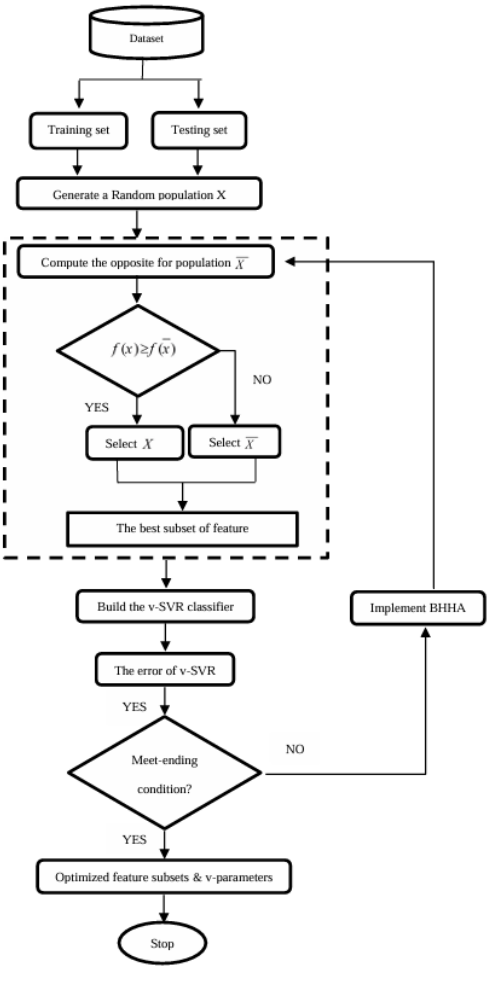

Fig. 1. The flowchart of the suggested framework

However, all of these currently used methods for selecting hyper-parameters make any attempt to select features concurrently. On the other side, little effort is made to tweak the v-SVR hyper-parameter. Using the quasi-oppositional based learning (QO-BL) to decrease search space. while, the role of Harris hawks (BHHOA) is focused to find the best solution, and this best solution is associated with hyper-parameters of v-SVR to improve this method. The type of kernel function used in the suggested technique is a kernel of Gaussian with value $\sigma > 0$. A representation of the solution is shown in figure 1. The suggested framework's flowchart is depicted in figure 2.

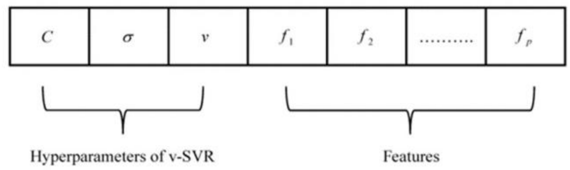

Fig. 2. Illustration of the suggested solution

In order to represent the C, ν, σ, and the features are represented by p binary values, each member of the hawk family has a location that has three quantitative values., Otherwise, 0 and 1 will be used for the relevant feature. That is to say, each hawk has three plus position (P). The parameter settings can be mentioned in the table 1.

Table 1. Description the parameter settings for the proposed algorithm

Population size	01_DOI
Maximum iteration	5, and 10
Parameters	$v \sim U(0,1)$, $\sigma \sim U(0,2)$, and $C \sim U(0,5)$
The feature-representing rest locations	$U(0,1)$

The steps of the suggested method are now being presented.

Step 1: $t_{max} = 5$ is the maximum number of iterations and $nh = 5$ is the quantity of hawks.

Step 2: The first three places, $v \sim U(0,1)$, $\sigma \sim U(0,2)$, and $C \sim U(0,5)$, indicate the hypermeters and are produced at random from a uniform distribution. The feature-representing rest locations are created as $U(0,1)$.

Step 3: Using Eq. (18), $\bar{x}$, and $\overline{x_q}$ are calculated, using (21)

Step 4: The definition of the fitness function is

$$\text{fitness} = \min\left[\frac{1}{n_{test}} \sum_{i=1}^{n_{test}} (y_{i,test} - \hat{y}_{i,test})^2\right] \tag{22}$$

where the testing dataset's fitness is determined.

Step 5: At this stage is calculate the fitness-function both all (x, $\bar{x}$, $\overline{x_q}$) These all are compared. If $f(x) \ge f(\overline{x_q})$ x is selected; otherwise, $\overline{x_q}$ is selected.

Step 6: Equations (9), (12), (13), (17), and (18) are used to update the locations of the hawks. To deal with feature selection, the binary HHOA (BHHOA-QOBL) is utilized. Here, each hawk is represented by the p-bit binary string. To update the position, the transfer function is usually used to force hawk to be in a binary space. A transfer function that restricts the new solution to to binary values can be utilized to create this binary vector [38].

$$x^{t+1} = \begin{cases} 1 & \text{if} \quad T\left(\Delta x^{t+1}\right) > rand \\ 0 & \text{otherwise} \end{cases} \tag{22}$$

where $T\left(x\right) = (1/1 + \exp(-x))$ is the sigmoid transfer function and $rand \in \left[0,1\right]$ is a random number.

Step 7: Till $t_{\max}$ is attained, steps 4, 5, and 6 are repeated.

5. Experimental results

Comprehensive comparative tests using the cross-validation approach with ten folds (CV) and the grid search strategy (GS) are used to evaluate the predictive performance of the suggested algorithm, HHOA. In this study, four separate sets of chemical datasets are employed: Data-set(A) [2], influenza neuraminidase a/PR/8/34 (H1N1) inhibitors (Data-set(B)), Data-set(C) covered the anti-cancer potential of imidazo [4,5-b] pyridine derivatives [7], and Data-set(D) covered a variety of antifungal agents [1]. Numerous thousands of descriptors are features in each of these datasets. These datasets are summarized in table 2.

A training dataset, which included 70% of all samples, and a test dataset, which included 30% of all samples, were created from each dataset. 20-times are handled with this splitting. There were two evaluation standards: The training dataset's mean-squared error ($MSE_{train} = \sum_{i=1}^{n_{train}} (y_{i,train} - \hat{y}_{i,train})^2 / n_{train}$) and the testing dataset's mean-squared error ($MSE_{test} = \sum_{i=1}^{n_{test}} (y_{i,test} - \hat{y}_{i,test})^2 / n_{test}$).

Table 3 lists the total number of features used for each approach's training set and averaged MSE-train. An essential consideration is the amount of features chosen by each strategy; solutions with a limited number of chosen features are preferable. Table 3 shows that HHOA is less feature-rich as from both the two approaches. As an illustration, Data-set(C), HHOA chose 16 characteristics as opposed to the CV's selection of 111 and the GS's selection of 138, respectively. Table 3 shows that the HHOA outperformed all other compared approaches in terms of predictive performance. As a result, the lowest MSE-train was produced by using the HHOA. Additionally, Data-set(D) shows that the HHOA's MSE-train reduction was roughly 88.86% and 90.30% lower than those of the CV and the GS, respectively. In addition, according to table 3's results, the GS technique is placed worst, while CV is second but performs worse than HHOA across the board.Once more, the test set findings in Table 4 show that the recommended strategy, HHOA, produces noticeably greater ability to predict when contrasted with GS and CV. For instance, HHOA outperformed CV and GS in terms of predictive performance in Data-set(A) with 0.0519, compared to 1.5946 and 1.8472, respectively. The GS is certainly the least accurate forecasting tool out of the ones employed.

Table 5 displays the CPU time for the suggested algorithm, CV, and GS to further emphasize the computational effectiveness. As can be observed, HHOA requires shorter time than GS and CV in terms of computing efficiency. The statistical test is required to demonstrate that the HHOA significantly outperforms the alternative approaches. It is clear that for all datasets, there's a statistical distinction with HHOA and the rest. This is expected given how much time the GA and CV require for computation.

The proposed HHOA's projected efficacy will be compared with other widely used algorithms used to address this problem in order to confirm the viability and effectiveness of the proposed HHOA in optimizing the v-SVR hyper-parameter and feature selection. These include the whale optimization algorithm, the firefly method, the bat algorithm, and the particle swarm optimization algorithm (WOA). These algorithms' parameters are set to be simple. It is anticipated that the population size and iteration counts will be the same as in HHOA. The average MSE and the computing time of the comparison approaches are displayed, respectively, in table 6 and figure 3. Table 6 amply demonstrates the HHOA's superior prediction performance by comparing its results to those of the other algorithms across all datasets. GWA is in second place, and PSO is last. In terms of global search and convergence in terms of running time, figure 3 demonstrates that HHOA performs better than other algorithms. Overall, HHOA has shown amazing outcomes when compared to WOA and superior results when compared to PSO, FF, and BA, as seen in table 6 and figure 3.

Table 2. An explanation of the used datasets

Data-sets	#Samples	#features
Data-set(A)	134	1048
Data-set(B)	479	2881
Data-set(C)	65	2540
Datas-et(D)	212	3107

Table 3. Results from experiments (on average) using training datasets

Data-sets	CV		GS		QOBL-HHOA	
	#FS	Mse_{train}	#FS	Mse_{train}	#FS	Mse_{train}
Data-set(A)	87	1.3377	104	1.5411	11	1.86×10^{-7}
Data-set(B)	101	2.1842	116	2.6691	15	0.3741
Data-set(C)	111	0.8296	138	1.0838	15	0.2525
Data-set(D)	92	1.6492	122	1.8951	10	1.9074×10^{-7}

Table 4. Results from experiments (on average) using testing datasets.

Data-sets	CV Mse_{ttest}	GS Mse_{ttest}	QOBL-HHOA Mse_{ttest}
Data-set(A)	1.5946	1.8472	2.05×10^{-7}
Data-set(B)	2.3114	2.4062	0.1888294
Data-set(C)	1.0557	1.2068	0.064013815
Data-set(D)	1.8069	1.9844	1.87946×10^{-7}

Table 5. Displays the typical time taken for computation in seconds

Data-sets	CV	GS	QOBL-HHOA
Data-set(A)	757.13	812.28	121.32
Data-set(B)	810.68	854.37	147.58
Data-set(C)	697.22	773.57	103.1
Data-set(D)	815.95	884.72	142.31

Table 6. Comparative experimental results of MSE_{test} (on average) for various algorithms using testing dataset

Algorithm	Datat-set(A)	Datat-set(B)	Datat-set(C)	Datat-set(D)
BA	0.0674	0.1108	0.0286	0.2905
FF	0.0694	0.1329	0.0309	0.3168
PSO	0.0711	0.1368	0.0326	0.3351
WOA	0.0541	0.1095	0.0246	0.2607

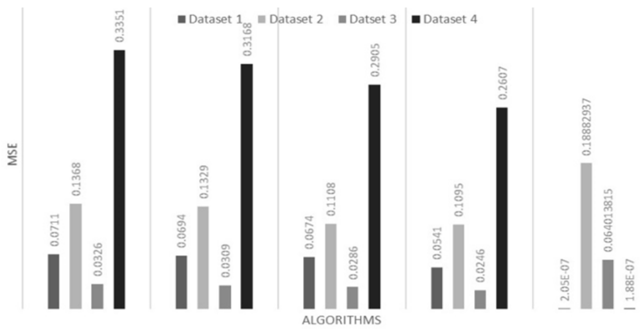

Fig. 3. displays the typical algorithm's time of running in seconds

6. Conclusion

Performing feature selection and optimizing the hyper-parameter of the v-SVR are essential steps in creating a successful research for any prediction challenge. In this study, it was suggested to use HHOA to simultaneously incorporate the feature selection and improve the hyper-parameter of the v-support vector regression. The results of the experiments and the statistical analysis of four datasets show that the proposed algorithm performs better than previous approaches and algorithms in terms of prediction, the number of features selected, and execution time. Due to this, the HHOA is a superior option than the others for describing the association between the characteristics of the input variables and the target variable. Other practical applications can use this framework for prediction, which is quite effective. This study was characterized by conducting the process of selecting features and improving the parameters at the same time, that is, the continuous and discontinuous digital system was dealt with at the same time, in addition to reducing the search area, and then it was used to make predictions after that, which opened a new way of hybridization to use other algorithms or use them in classification operations in the future.

References

[1] Al-Fakih, A. et al.: A QSAR model for predicting antidiabetic activity of dipeptidyl peptidase-IV inhibitors by enhanced binary gravitational search algorithm. SAR and QSAR in Environmental Research 30(6), 2019, 403–416.
[2] Al-Fakih A. et al.: QSAR classification model for diverse series of antifungal agents based on improved binary differential search algorithm. SAR and QSAR in Environmental Research 30(2), 2019, 131–143.
[3] Algamal Z. Y. et al.: High-dimensional QSAR prediction of anticancer potency of imidazo [4, 5-b] pyridine derivatives using adjusted adaptive LASSO. Journal of Chemometrics 29(10), 2015, 547–556.
[4] Al-Thanoon N. A., Qasim O. S., Algamal Z. Y.: A new hybrid firefly algorithm and particle swarm optimization for tuning parameter estimation in penalized support vector machine with application in chemometrics. Chemometrics and Intelligent Laboratory Systems 184, 2019, 142–152.
[5] Al-Thanoon N. A., Qasim O. S., Algamal Z. Y.: Improving nature-inspired algorithms for feature selection. Journal of Ambient Intelligence and Humanized Computing 2022, 1–11.
[6] Al-Thanoon N. A., Qasim O. S., Algamal Z. Y.: Tuning parameter estimation in SCAD-support vector machine using firefly algorithm with application in gene selection and cancer classification. Comput Biol Med 103, 2018, 262–268.
[7] Al-Thanoon N. A., Qasim O. S., Algamal Z. Y.: Variable selection in gamma regression model using binary gray wolf optimization algorithm. Journal of Physics: Conference Series. 2020.
[8] Cao G., Wu L.: Support vector regression with fruit fly optimization algorithm for seasonal electricity consumption forecasting. Energy 115, 2016, 734–745.
[9] Cheng C.-T. et al.: Optimizing Hydropower Reservoir Operation Using Hybrid Genetic Algorithm and Chaos. Water Resources Management 22(7), 2007, 895–909.
[10] Cheng J., Qian J., Guo Y.-N.: Adaptive chaotic cultural algorithm for hyperparameters selection of support vector regression. International Conference on Intelligent Computing. Springer 2009.
[11] Cherkassky V., Ma Y.: Practical selection of SVM parameters and noise estimation for SVM regression. Neural networks 17(1), 2004, 113–126.
[12] Chou J.-S., Pham A.-D.: Nature-inspired metaheuristic optimization in least squares support vector regression for obtaining bridge scour information. Information Sciences 399, 2017, 64–80.
[13] Chuang C.-C., Lee Z.-J.: Hybrid robust support vector machines for regression with outliers. Applied Soft Computing 11(1), 2011, 64–72.
[14] Fan Q., Chen Z., Xia Z.: A novel quasi-reflected Harris hawks optimization algorithm for global optimization problems. Soft Computing, 2020.
[15] Ganesh N. et al.: Efficient feature selection using weighted superposition attraction optimization algorithm. Applied Sciences 13(5), 2023, 3223.
[16] Golilarz N. A. et al.: A New Automatic Method for Control Chart Patterns Recognition Based on ConvNet and Harris Hawks Meta Heuristic Optimization Algorithm. IEEE Access 7, 2019, 149398–149405.
[17] Heidari A. A. et al.: Harris hawks optimization: Algorithm and applications. Future Generation Computer Systems 97, 2019, 849–872.
[18] Hong W.-C. et al.: SVR with hybrid chaotic genetic algorithms for tourism demand forecasting. Applied Soft Computing 11(2), 2011, 1881–1890.
[19] Houssein E. H. et al.: Optimal Sink Node Placement in Large Scale Wireless Sensor Networks Based on Harris' Hawk Optimization Algorithm. IEEE Access 8, 2020, 19381–19397.
[20] Huang C.-F.: A hybrid stock selection model using genetic algorithms and support vector regression. Applied Soft Computing 12(2), 2012, 807–818.
[21] Ismael O. M., Qasim O.S., Algamal Z.Y.: A new adaptive algorithm for v-support vector regression with feature selection using Harris hawks optimization algorithm. in Journal of Physics: Conference Series, 2021.
[22] Ito K., Nakano R..: Optimizing support vector regression hyperparameters based on cross-validation. Proceedings of the International Joint Conference on Neural Networks, 2003.
[23] Kaneko H., Funatsu K.: Fast optimization of hyperparameters for support vector regression models with highly predictive ability. Chemometrics and Intelligent Laboratory Systems 142, 2015, 64–69.
[24] Kazem A. et al.: Support vector regression with chaos-based firefly algorithm for stock market price forecasting. Applied Soft Computing 13(2), 2013, 947–958.
[25] Kong D. et al.: Tool wear monitoring based on kernel principal component analysis and v-support vector regression. The International Journal of Advanced Manufacturing Technology 89(1–4), 2016, 175–190.
[26] Laref R. et al.: On the optimization of the support vector machine regression hyperparameters setting for gas sensors array applications. Chemometrics and Intelligent Laboratory Systems 184, 2019, 22–27.
[27] Li S., Fang H., Liu X.: Parameter optimization of support vector regression based on sine cosine algorithm. Expert Systems with Applications 91, 2018, 63–77.
[28] Menesy S. A. et al.: Developing and Applying Chaotic Harris Hawks Optimization Technique for Extracting Parameters of Several Proton Exchange Membrane Fuel Cell Stacks. IEEE Access 8, 2020, 1146–1159.
[29] Nait Amar M., Zeraibi N.: Application of hybrid support vector regression artificial bee colony for prediction of MMP in CO2-EOR process. Petroleum, 2018.
[30] Naveh I. M. H. et al.: A Quasi-Oppositional Method for Output Tracking Control by Swarm-Based MPID Controller on AC/HVDC Interconnected Systems With Virtual Inertia Emulation. IEEE Access 9, 2021, 77572–77598.
[31] Priyadarshini J. et al.: Analyzing physics-inspired metaheuristic algorithms in feature selection with K-nearest-neighbor. Applied Sciences 13(2), 2023, 906.
[32] Qu C. et al.: Harris Hawks optimization with information exchange. Applied Mathematical Modelling 84, 2020, 52–75.
[33] Rahnamayan S., Tizhoosh H. R., Salama M. M.: Quasi-oppositional differential evolution. IEEE congress on evolutionary computation, 2007. IEEE. regression with outliers. Applied Soft Computing 11(1), 2011, 64–72.
[34] Schölkopf B. et al.: New support vector algorithms. Neural computation 12(5), 2000, 1207–1245.
[35] Shaik K. et al.: Big Data Analytics Framework Using Squirrel Search Optimized Gradient Boosted Decision Tree for Heart Disease Diagnosis. Applied Sciences 13(9), 2023, 5236.
[36] Shehabeldeen T. A. et al.: Modeling of friction stir welding process using adaptive neuro-fuzzy inference system integrated with harris hawks optimizer. Journal of Materials Research and Technology 8(6), 2019, 5882–5892.
[37] 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.
[38] Too A., Mohd S..: A New Quadratic Binary Harris Hawk Optimization for Feature Selection. Electronics 8(10), 2019, 1130.
[39] Tsirikoglou P. et al.: A hyperparameters selection technique for support vector regression models. Applied Soft Computing 61, 2017, 139–148.
[40] Üstün B. et al.: Determination of optimal support vector regression parameters by genetic algorithms and simplex optimization. Analytica Chimica Acta 544(1–2), 2005, 292–305.
[41] Vapnik V. N.: An overview of statistical learning theory. IEEE transactions on neural networks 10(5), 1999, 988–999.
[42] Wu C.-H., Tzeng G.-H., Lin R.-H.: A Novel hybrid genetic algorithm for kernel function and parameter optimization in support vector regression. Expert Systems with Applications 36(3), 2009, 4725–4735.
[43] Xu S. et al.: An improved variable selection method for support vector regression in NIR spectral modeling. Journal of Process Control 67, 2018, 83–93.
[44] Zhang J. et al.: Optimization enhanced genetic algorithm-support vector regression for the prediction of compound retention indices in gas chromatography. Neurocomputing 240, 2017, 183–190.
[45] Zhao Y.-P., Sun J.-G.: Robust truncated support vector regression. Expert Systems with Applications 37(7), 2010, 5126–5133.

Ph.D. Omar M. Ismael
e-mail: Omar_muhammed84@nan.epedu.gov.iq

Omar M. Ismael received the B.S. degree in Mathematics from University of Mosul, Mosul, Iraq, in 2008, the M.S. degree in Mathematics from University of Mosul, Mosul, Iraq, in 2015, the Ph.D. degree in Mathematical Science/Artificial intelligence from University of Mosul, Mosul, Iraq, in 2021. He is a teacher in Directorate of Education in Nineveh. His research interest includes the data science, intelligence technologies, deep learning, bioinformatics, and optimization algorithm.

https://orcid.org/0009-0005-6739-4790

Prof. Dr. Omar S. Qasim
e-mail: omar.saber@uomosul.edu.iq

Omar S. Qasim received his B.S. degree in Mathematics from the University of Mosul, Mosul, Iraq, in 2003, and his M.S. degree in Mathematics from the University of Mosul, Mosul, Iraq, in 2006. He obtained his Ph.D. degree in Mathematical Science from the University of Mosul in 2010. He is currently a professor of Mathematics at the University of Mosul. His research interests include applied mathematics, deep learning, bioinformatics, fuzzy logic, and optimization algorithms.

https://orcid.org/0000-0003-3301-6271

Prof. Dr. Zakariya Y. Algamal
e-mail: Zakariya.algamal@uomosul.edu.iq

Zakariya Y. Algamal received the B.S. degree in Statistics from University of Mosul, Mosul, Iraq, in 2001, the M.S. degree in Statistics from University of Mosul, Mosul, Iraq, in 2004, the Ph.D. degree in Mathematical Science/Statistics from Universiti Teknologi Malaysia (UTM), Malaysia, in 2016, and Post. Doctorate in Mathematical Science/Statistics from Universiti Teknologi Malaysia (UTM), Malaysia, in 2017. He is a professor in Statistics in University of Mosul.
His research interest includes the development of high dimensional data, generalized linear models, bioinformatics, chemoinformatics, and optimization algorithms.

https://orcid.org/0000-0002-0229-7958