**Adv. Math. Fin. App.,** 2022, **7**(2), P. 477-487

**Advances in Mathematical Finance & Applications**
**www.amfa.iau-arak.ac.ir**
**Print ISSN**: 2538-5569
**Online ISSN**: 2645-4610
**Doi:** 10.22034/AMFA.2021.1936352.1623

Applied-Research Paper

# Support Vector Regression Parameters Optimization using Golden Sine Algorithm and its Application in Stock Market

Mohammadreza Ghanbari [a] , Mahdi Goldani [b]

[a] *Department of Mathematical Sciences, Sharif University of Technology, Tehran 1458889694, Iran*
[b] *Department of Economics, Hakim Sabzevari University, Sabzevar 9617976487, Iran*

ARTICLE INFO

*Article history:*
Received 2021-08-03
Accepted 2021-11-12

Keywords:
Golden Sine Algorithm
Meta-heuristics Optimization Algorithms
Parameter Tuning
Support Vector Regression
Time Series Prediction

ABSTRACT

Stock price prediction is one of the most important concerns of stockholders. This prediction, independent of the method which is used or the assumptions which are applied, is welcomed and trusted if it can guarantee a high fitting. So due to the high performance prediction, using some complicated models as Machine Learning family such as Support Vector Regression (SVR) was recommended instead of older and lower performance approaches such as multiple discriminant technique. SVR model have achieved high performance on forecasting problems, however, its performance is highly dependent on the appropriate selection of SVR parameters. In this study, a novel GSA-SVR model based on Golden Sine Algorithm is presented. The performance of the proposed model is compared with eleven other meta-heuristic algorithms on some stocks from NASDAQ. The results indicate that the given model here is capable of optimizing the SVR parameters very well and indeed is one of the best models judged by both prediction performance accuracy and time consumption.

## 1 Introduction

Forecasting is a major target in finance or economic future studies that utilizes many methods in mathematics, especially in optimization fields. Among these different type of models, machine learning algorithms becomes more considerable and reliable since they have shown better results than other techniques. [27] By expanding these algorithms hyper-parameters becomes an important tasks in learning algorithms configuration since it is a core component of the model's architecture and must be set before using the model for the problems. Automated hyper-parameters optimization is the task of tuning automatically in contrast to the trial and error method as the traditional procedure. It reduces the human efforts needed for using machine learning algorithms and leads to improvement of learning processes. Meta-heuristics algorithms are one of the set of the suitable methods for parameter tuning. In this article the hyper-parameters of support vector regression are optimized using these algorithms. For validation, eleven other meta-heuristic algorithms, namely Whale Optimization Algorithm (WOA), Salp Swarm Algorithm (SSA), Neural Network Algorithm (NNA), Firefly Algorithm (FA), Multi-Verse Optimizer (MVO), Moth-Flame Optimization (MFO), Harris Hawks Optimization (HHO), Grey Wolf Optimization (GWO), Butterfly Optimization Algorithm (BOA), Biogeography-Based Optimization (BBO) and Artificial Bee Colony Optimization (ABC) are also used to optimize the parameters, and their results are compared with the proposed algorithm based on the mean square error, mean absolute percentage error and consumption time.( Some Cases for Meta Heuristic algorithm in portfolio optimization can be studied in [17]) The remainder of our work is as follows: In Section 2, a

* Corresponding author. Tel.: +985144247333
E-mail address: *mahdi.goldani@gmail.com*

literature of the work is reviewed. In section 3 we discuss the presented GSA-SVR model, the support vector regression and GSA. In Section 4, the model is tested on some datasets and compared with other models and the obtained experimental results are discussed. Conclusions and future research directions are provided in Section 5.

## 2 Literature Review

Support Vector Machine (SVM) was originally introduced by Vapnik [5] for classification problems. It can potentially solve small-sample, non-linear and high dimensions' problems by using structural risk minimization principle instead of the empirical risk principle. Essentially, the SVM is a convex quadratic programming method by which it is possible to find the global, rather than local optimum. After on, SVM was developed and extended to Support Vector Regression (SVR) to solve regression problems [5]. SVR is arguably one of the best techniques and experimental results show considerable performance compared to other nonlinear methods [23]. However, setting of the parameters for the SVR plays a significant role and the performance accuracy changes considerably upon a bad choice of parameters [24]. Concerning this issue, a main approach to select the SVR parameters optimally is to make use of an optimization technique for finding optimal values.
Generally, three common techniques to optimize the SVR parameters are grid search [20], which in practical applications, it is usually vulnerable to get to a local optimum, gradient descent [11] and meta-heuristics algorithms [16]. Meta-heuristics algorithms have been introduced that may provide a sufficiently good solution to an optimization problem especially on presence of incomplete information or limited computing capacity. They have shown superior results in the case of solving optimization problems for parameter tuning of complex models [9]. In the past, many meta-heuristics algorithms have been proposed for the selection of optimal SVR parameters with different application area including stock market, load capacity, traffic management and weather. Example are, Genetic Algorithm (GA) [8, 15, 25], Grey Wolf Optimizer (GWO) [14], Particle Swarm Optimization (PSO) [10, 30], Butterfly Optimization Algorithm (BOA) [7], Henry gas solubility optimization algorithm (HGS) [3], Harris Hawks Optimization (HHO) [18], whale optimization algorithm (WOA) [18], Sine Cosine Algorithm (SCA) [13], Firefly Algorithm (FA) [27] and Bat Algorithm (BA) [24, 29]. Recently, a novel math-based meta-heuristic optimization algorithm inspired by sine function, named as Golden Sine Algorithm (GSA), was designed by [22]. The GSA algorithm searches to approach a better solution in each iteration by trying to bring the current point closer to the target value and the solution space gets to be narrowed by the golden section algorithm so that the areas with supposedly good results instead of the whole solution space are examined. Here, we propose a novel GSA based SVR model where GSA is used to set the parameters of SVR.

## 3 Materials and Methods

In this section, we briefly discuss about Support Vector Regression (SVR) and the Golden Sine Algorithm (GSA).

### 3.1 Support Vector

Support vector Machine (SVM) is a machine learning algorithm introduced by Vapnik in 1995 for classification problems. It has been one of the more widely used methods in recent years as a powerful method. It was first used to address a binary pattern classification problem. Then, it was promoted to support vector regression (SVR) for regression problems by using $\epsilon$-insensitive loss function to penalize data when they are greater than $\epsilon$ [5]. SVR aims to provide a nonlinear mapping function to map the training dataset to a high dimensional feature space [26]. The given training dataset is$\{(x_i, y_i)\}_{i=1}^{n}$, where $x_i \in R^d$ is input data, $y_i \in R$ is the output value of the i-th data point in the dataset, d is the dimension of samples and n is the number of samples. The nonlinear function between the input and the output is formulated as:

$$y = f(x) = w^T \varphi(x) + b \tag{1}$$

Where $\Phi: R^d \to F$ is a nonlinear mapping to the feature space, $w \in F$ is a vector of weight coefficients and b is a bias constant. The w and F are estimated by minimizing the following optimization problem:

$$MIN \ \frac{1}{2}\|w\|^2, \quad \text{s.t.} \begin{cases} y_i - w^T\varphi(x) - b \le \epsilon \\ y_i - w^T\varphi(x) - b \ge -\epsilon \end{cases} \tag{2}$$

The slack variables $\varepsilon$ and $\varepsilon^*$ are used to penalize points from $\epsilon$ -insensitive band:

$$MIN \ \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}(\varepsilon + \varepsilon^*), \quad \text{s.t.} \begin{cases} y_i - w^T\varphi(x) - b \le \epsilon + \varepsilon \\ y_i - w^T\varphi(x) - b \ge -\epsilon - \varepsilon^* \\ \varepsilon . \varepsilon^* \ge 0 , i = 1, \dots , n \end{cases} \tag{3}$$

Where C is a constant known as the penalty parameter to specify the trade-off between the empirical risk and regularization terms, $\epsilon$ is the insensitive loss function and the slack variables $\varepsilon$ and $\varepsilon^*$, correspond to upper and lower deviations, respectively, and n is the number of training patterns. Using the Lagrangian and corresponding optimality conditions, the obtained generic equation is written as [5]:

$$f(x) = \sum_{i=1}^{n}(\beta_i - \beta_i^*)K(x_i . x) + b \tag{4}$$

Where $\beta_i$ and $\beta_i^*$ are nonzero Lagrange multipliers and $K(x_i . x)$ is the kernel function. In our work, Radial Basis Function (RBF) has been used as kernel function:

$$K(x_i . x) = exp\ (-\gamma\|x_i - x_j\|^2) \tag{5}$$

Where $\gamma$ is the RBF width parameter.

### 3.2 Golden Sine Algorithm

Golden Sine Algorithm (GSA) is a novel math-based meta-heuristic optimization algorithm inspired by sine function for solving optimization problems [22]. Sine, a trigonometric function, is the coordinate relative to the y-axis of a point on a 1-unit radius circle that is the central origin. An orthogonal triangle with an angle made by the $y$-axis of a straight line drawn from the origin or with the same angle is calculated with the hypotenuse section of the edge opposite this angle. The defining range of the function is $[-1, 1]$. The scan of the unit circle of all values of the sine function is similar to the search of the search space in optimization problems. This similarity has inspired the development of GSA. The operator used in the algorithm is shown by

$$V_{ij} = V_{ij}|sin(r_1)| - r_2 sin(r_1)|x_1 D_j - x_2 V_{ij}| \tag{6}$$

Where $V_{ij}$ is the value of current solution in the i-th dimension, $D$ is the determined target value, $r_1$ is a random number in the range $[0, 2\pi]$ and $r_2$ is a random number in the range $[0, \pi]$, and $x_1$ and $x_2$ are the coefficients obtained by the golden section method. These coefficients limit the search space and also allow the current value to approach the target value. The wide range of the search space is a major issue for solving problems. The effect of limiting the search space in solving problems is significantly affecting the results. GSA uses the golden section method to make this process the best possible way. Golden section search is an optimization technique that can be used to find the maximum or minimum value of a single unimodal function. The name is from the golden ratio. Two numbers, $p$ and $q$, are in a golden ratio if

$$\frac{p+q}{p} = \frac{p}{q} = \tau \tag{7}$$

Or equivalently,

$$1 + \frac{p}{q} = \tau \tag{8}$$

Or

$$1 + \frac{1}{\tau} = \tau \tag{9}$$

Solving Eq. 9, we get the positive root as

$$\tau = \frac{1-\sqrt{5}}{2} \approx 0/618033 \tag{10}$$

Here $\tau$ is called the golden number. In GSA, initial default values for $a$ and $b$ are considered to be $-\pi$ and $\pi$, respectively. These two coefficients are applied to the current and target values in the first iteration. Then, the coefficients $x_1$ and $x_2$ are updated as the target value changes. To avoid the situation of equality for $x_1$ and $x_2$ , an equality check is performed. It means that if the two values are equal, then the random numbers $rand_1$ and $rand_2$ are generated in the range, respectively, $[0, \pi]$, $[0, -\pi]$, $x_1$ and $x_2$ are recalculated.

## 3.3 GSA for Parameter Optimization of SVR

Before presentation the algorithm, we first discuss the data pre-processing of time series. Phase Space Reconstruction [21] is a method in which uncover the hidden information embedded in the time series dynamics. This method provides a simplified multidimensional representation of data. Let $\{x\}_{i=1}^{n}$ represent an n point time series. Then, the reconstructed phase space can be expressed as a matrix as follows:

$$X = \begin{bmatrix} x_1 & x_{1+\tau} & \cdots & x_{1+(m-1)\tau} \\ x_2 & x_{2+\tau} & \cdots & x_{2+(m-1)\tau} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n-1-(m-1)\tau} & x_{n-1-(m-2)\tau} & \cdots & x_{n-1} \end{bmatrix} \tag{11}$$

Where $\tau$ the time delay constant and m is called the embedding dimension of the reconstructed phase space. [12] proposed an efficient method of finding the minimal sufficient embedding dimension, named as false nearest neighbors (FNN) procedure, in which the nearest neighbors of every point in a given dimension are found, and then checks are made to see if these points are still close neighbors in one higher dimension. To estimate the delay parameter, here we use the first minimum of the Mutual Information (MI) function [1]. After finding the optimal $m$ and $\tau$, the input data and the output vector were designed by Eq. (11) and Eq. (12).

$$Y = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix} = \begin{bmatrix} x_{2+(m-1)\tau} \\ x_{2+(m-1)\tau} \\ \vdots \\ x_{n-1} \end{bmatrix} \tag{12}$$

After construction of the input and output matrix, we normalize the data by using min-max formula

$$x_{new} = \frac{x_{old} - x_{min}}{x_{max} - x_{min}} \tag{13}$$

To scale to the range $[0,1]$. Finally, the data is divided into the training set and the testing set. In the first step of our proposed algorithm, the GSA parameters including the number of agents and maximum number of iterations are set. Then, GSA-SVR starts with a set of candidate solutions generated randomly within predetermined lower and upper bounds. In this case, each solution is a three-dimensional vector represented by $(C, \gamma. \epsilon)$, where $C, \gamma$ and $\epsilon$ are the SVR parameters to be optimized. The objective function is equal to the Mean Square Error (MSE), Eq. 15, of the tested SVR model. A predetermined maximum number of iterations is used as a criteria to stop the algorithm. Figure 1 shows flowchart of the complete the complete procedure. A step-wise procedure of the proposed algorithm is described next:

Step 1: Assign the parameters including the number of search agents and the maximum number of iterations. Set the iteration number, $t$, equal to zero.

Step 2: Initialize the random solutions of search agents with

$$s_i = Lb_i + (Ub_i - Lb_i).u \quad (14)$$

And evaluate fitness function using Eq. (15) and Eq. (16) on the test data. Here, Lb (Ub) is lower (upper) bound and $i \in \{C, \gamma. \epsilon\}$, and u is a uniform random number in the interval $(0, 1)$.

Step 3: Update the position of search agents for every dimension based on Eq. (6) and set $t = t + 1$.

Step 4: If the maximum number of iterations is reached then the optimized parameters of SVR are selected, thus go to step 5; otherwise go back to step 3.

Step 5: Use the SVR model with the optimal parameters $(C, \gamma. \epsilon)$ for prediction.

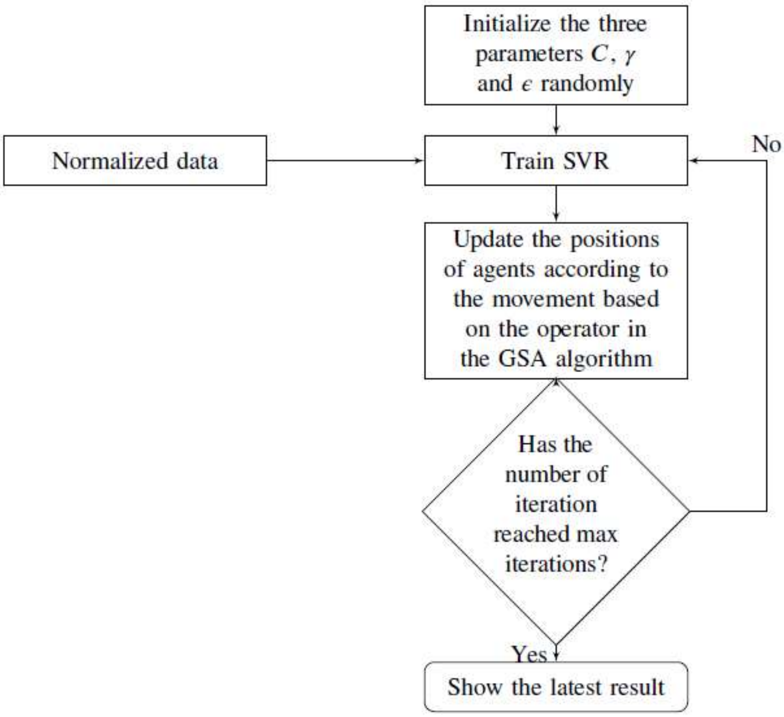



**Fig. 1:** GSA-SVR procedure

## 4 Experimental Results

In this section, a number of stocks are chosen to test the performance of our proposed GSA-SVR model. The proposed algorithm is compared to the other meta-heuristic algorithms, being used for parameter optimization of SVR, including Whale Optimization Algorithm based SVR (WOA-SVR), Salp Swarm Algorithm based SVR (SSA-SVR), Neural Network Algorithm based SVR (NNA-SVR), Firefly Algorithm based SVR (FA-SVR), Multi-Verse Optimizer based SVR (MVO-SVR), Moth-Flame Optimizer based SVR (MFO-SVR), Harris Hawks Optimization Algorithm based SVR (HHO-SVR), Grey Wolf Optimization Algorithm based SVR (GWO-SVR), Butterfly Optimization Algorithm based SVR (BOA-SVR), Biogeography-Based Optimization Algorithm based SVR (BBO-SVR) and

Artificial Bee Colony Optimization Algorithm based SVR (ABC-SVR). Stock market price prediction is regarded as one of the most challenging tasks of financial time series prediction. The difficulty of forecasting arises from the inherent non-linearity and nonstationary of the stock market and financial time series. Thus, daily closing stock market prices of three companies, namely Alibaba Group Holding Limited (BABA) (from 03/10/2016 to 01/10/2019), Tesla, Inc. (TSLA) (from 03/10/2016 to 01/10/2019) and Taiwan Semiconductor Manufacturing (TSM) Company Limited (from 03/10/2016 to 01/10/2019), were extracted from Yahoo Finance historical quotes. After finding the time delay, $\tau$, the embedding dimension, m and reconstructing the phase space, 80% of the data were used as the training set and the remaining were used as the testing set. All the predictions were based on one-step ahead prediction results and the computations were carried out in MATLAB R2019a environment using the LIBSVM Toolbox [2] on a laptop with an Intel(R) Core(TM) i3-3110M CPU @ 2.40GHz and 4 Gbytes memory. In our work, Mean Squared Error (MSE) and Mean Absolute Percent Error (MAPE) were used in order to calculate the accuracy

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - f_i). \tag{15}$$

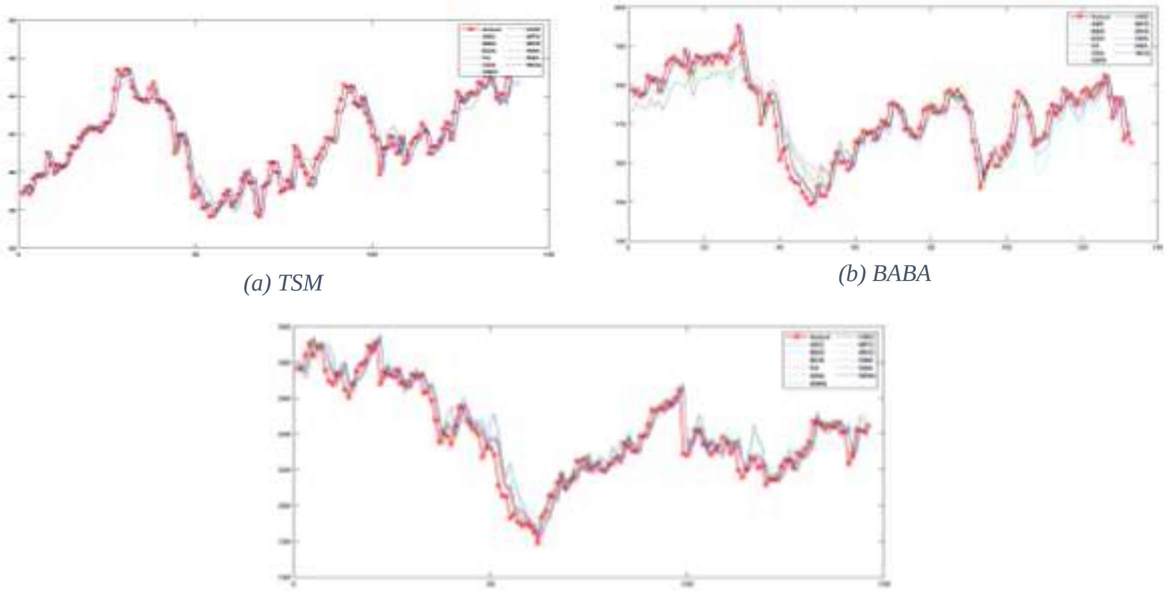
$$MAPE = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{y_i - f_i}{y_i}\right|. \tag{16}$$

Where $y_i$and $f_i$ denote the actual and predicted values for the i-th data point, respectively and $N$ is the number of forecasting days. Since meta-heuristics algorithms use initial random population, we ran each algorithm several times to get the optimal answer. However, to increase the probability of finding the global optimum we used diversity in population and a sufficiently large number of iterations. In this study, the number of populations and the maximum number of iterations are selected to be 20 and 50 respectively. Also, the search space for both parameters $C$ and $\gamma$ were $[4^{-7}, 4^{4}]$and the range for parameter $\epsilon$ was$[4^{-7}, 0/25]$. All details of datasets including name, embedding dimension m and time delay $\tau$ are shown in Table 1. The size of training datasets is equal to 530 and the size of testing datasets is equal to 133 for all of three datasets. For phase space reconstruction we used the recurrence plot and recurrence quantification analysis of MATLAB toolbox [4].

**Table 1:** Estimation of $m$ and $\tau$ for phase space reconstruction.

| Parameters | BABA | TSLA | TSM |
|---|---|---|---|
| $\boldsymbol{m}$ | 10 | 12 | 10 |
| $\boldsymbol{\tau}$ | 10 | 2 | 5 |

Actual and predicted values obtained by our model compared to eleven other methods for the three datasets TSM (Fig. (2a)), BABA (Fig. (2b)) and TSLA (Fig. (2c)) are illustrated in Fig. 2 after de-normalization. Also, Table 3 presents the optimal values for the three parameters $C, \gamma$ and $\epsilon$, as well as MSE, MAPE and the computing time for BABA testing dataset for all of SVR-based methods. The same results for TSLA and TSM testing datasets are shown in Tables 4 and 5, respectively. We now discuss results of our proposed algorithm in comparison with others. As shown in Fig. 3a, the MAPE of GSA-SVR ranked 6th with a slight difference of 0.001 with the first rank, MFO-SVR. Even though MAPE accuracy of GSA-SVR is slightly below the other five methods MFO-SVR, MVO-SVR, GWO-SVR, ABC-SVR and FA-SVR, its computing time is significantly better than these methods except for GWO-SVR. Also, Fig. 3b illustrates that GSA-SVR, FA-SVR, ABC-SVR, MFO-SVR, GWO-SVR and SSA-SVR achieved the best MSE accuracy below 0.0012 in comparison with all the other methods. Thus it shows that the proposed algorithm is one of the best based on MSE error. Finally, Fig. 4a, Fig. 4b and Fig. 4c respectively depict the bar plots of MAPE, MSE and cost time. Based on Fig. 3c, the average time consumption of the GSA-SVR model ranked seventh but competitively close to the other methods. Also it is important to mention that although BBO-SVR, BOA-SVR, HHOSVR and NNA-SVR algorithms are computationally less expensive, but their MSE and MAPE accuracy are very bad in comparison with GSA-SVR algorithm as it is shown in Fig. 3a and Fig. 3b. Also to compare the predictive accuracy results of our method with others here we used Diebold-Mariano test [6]. Base on the test, the null hypothesis of equality of any two given methods at the 5% confidence level is rejected if $|DM| > 1.96$, where DM is the test statistic of the Diebold-Mariano test calculated based on the corresponding squared-error residuals.



*(a) TSM* *(b) BABA*

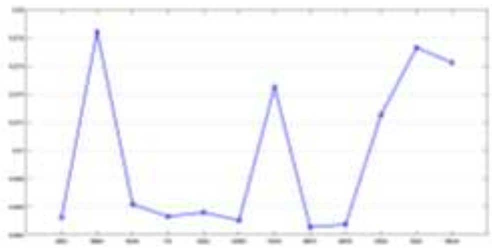
*(c) TSLA*

**Fig. 2**: Prediction comparison for datasets a) TSM b) BABA and c) TSLA show that among the twelve algorithms, GSA-SVR performs as one of the best based on accuracy.

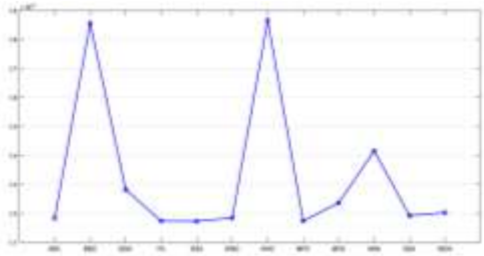
**Table 2.** Diebold Mariano Test

| Methods | BABA | TSLA | TSM |
|---|---|---|---|
| **ABC-SVR** | -0.2507 | 1.2437 | -0.5011 |
| **BBO-SVR** | 5.8814 | 5.0089 | 0.2784 |
| **BOA-SVR** | 1.8033 | 2.8169 | 0.2922 |
| **FA-SVR** | -0.7461 | 1.2767 | -0.6720 |
| **GWO-SVR** | -0.5394 | 1.2555 | -0.6085 |
| **HHO-SVR** | 7.3394 | 3.9379 | 2.5401 |
| **MFO-SVR** | -0.5634 | 1.0731 | -0.5184 |
| **MVO-SVR** | 3.9999 | 0.2153 | 0.1945 |
| **NNA-SVR** | 3.8801 | 2.5598 | 0.1225 |
| **WOA-SVR** | -0.3112 | 1.5994 | 2.1655 |
| **SSA-SVR** | 1.3241 | 1.4862 | -0.0073 |

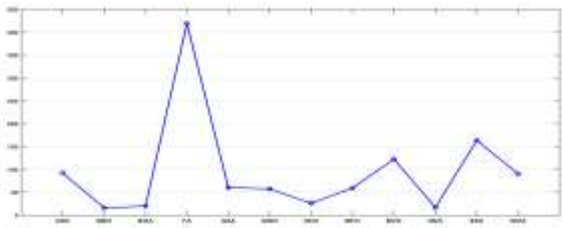
DM-values obtained by the Diebold-Mariano test on our three stocks is presented in Table 2. As shown in the table, there is a significant difference between our algorithm and BBO-SVR, HHO-SVR, MVO-SVR and NNA-SVR and in fact GSA-SVR is better than these algorithms and also there is no significance difference between the proposed algorithm and the rest of methods for BABA data. For TSLA, the null hypothesis of equality is rejected for GSA-SVR and BBO-SVR, BOA-SVR, HHO-SVR and NNA-SVR and indeed, GSA-SVR based on forecasting accuracy performed better. For TSM, HHO-SVR, WOA-SVR have DM-test absolute value greater than 1.96, therefore there is only significant difference between these models and GSA-SVR. To summarize, based on time efficiency, MSE and MAPE measures, we conclude that our GSA-SVR algorithm is capable to find the optimal values of the SVR parameters and can yield promising results and also is one of the best models among the same other meta-heuristics based SVR methods studied.



*(a) MAPEs*

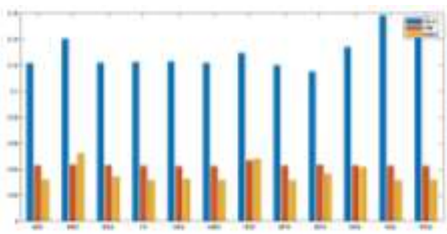


*(b) MSEs*

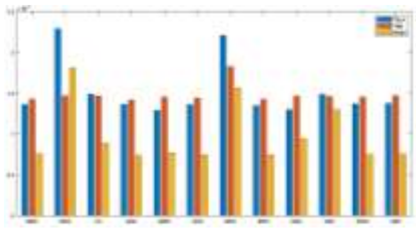


*(c) Time Consumption*

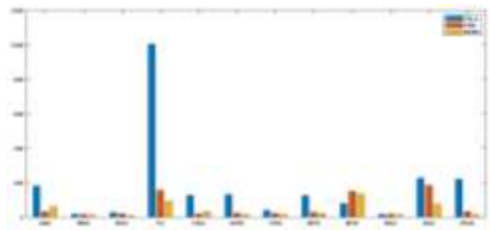
**Fig. 3:** Fig. 3a and Fig. 3b compare the MAPE and the MSE of the twelve methods. Also Fig. 3c compares the cost time.



*(a) MAPEs*



*(b) MSEs*



*(c) Time Consumption*

**Fig.4:** Fig. 4a and Fig. 4b compare the MAPE and the MSE bar time plots of the twelve methods. Also Fig. 4c compare the cost time.

## 5 Conclusion and Future Research

In support vector regression, parameters namely, penalty factor, $C$, RBF kernel function width parameter, $\gamma$ and radius of the epsilon tube, $\epsilon$ can change the performance of the algorithm considerably. Therefore, there is a need to optimize the parameters in an appropriate way. In this study, a novel hybrid method based on support vector regression and Golden Sine algorithm for selecting these parameters is presented. The proposed method were tested on three financial time series of technology based companies, Alibaba Group Holding Limited, Tesla, Inc. and Taiwan Semiconductor Manufacturing Company Limited, using their daily closing stock market prices. For invalidation, the results are compared with eleven other meta-heuristics algorithms based SVR. According to the experimental results, GSA-SVR is capable of tuning the parameters efficiently in terms of computational time, MSE and MAPE errors.

## Appendix

**Table 3:** Optimized parameters for BABA data.

| ***Models*** | $C$ | $\gamma$ | $\epsilon$ | **MSE** | **MAPE** | **Cost time** |
|---|---|---|---|---|---|---|
| ***ABC-SVR*** | 113.2752 | 0.002117915 | 0.01608924 | 0.00076193 | 0.03159439 | 61.5194 |
| ***BBO-SVR*** | 0.257373 | 3.12473 | 6.10E-05 | 0.00180953 | 0.0518886 | 13.957 |
| ***BOA-SVR*** | 41.3041 | 0.00109727 | 0.0124801 | 0.00089146 | 0.0339914 | 10.3919 |
| ***FA-SVR*** | 254.4579 | 0.001383811 | 0.008534498 | 0.0007394 | 0.0312368 | 94.71895 |
| ***GSA-SVR*** | 1.25782 | 0.220345 | 0.000172123 | 0.00077329 | 0.0321525 | 33.0719 |
| ***GWO-SVR*** | 254.6702 | 0.00127615 | 0.01754627 | 0.00074881 | 0.03133175 | 15.51003 |
| ***HHO-SVR*** | 246.7128 | 0.9452953 | 0.06805752 | 0.00156628 | 0.04784143 | 15.86751 |
| ***MFO-SVR*** | 251.2928 | 0.001859763 | 0.01025836 | 0.00074726 | 0.03121457 | 22.78994 |
| ***MVO-SVR*** | 251.3184 | 0.05405353 | 0.009142361 | 0.00094357 | 0.036104 | 135.5701 |
| ***NNA-SVR*** | 256 | 6.10E-05 | 0.006907721 | 0.00130049 | 0.04111579 | 16.66189 |
| ***SSA-SVR*** | 254.9338 | 0.0041949 | 0.0123411 | 0.00075183 | 0.03122928 | 77.0712 |
| ***WOA-SVR*** | 246.8061 | 0.001580238 | 0.01645499 | 0.00075927 | 0.0315578 | 15.11345 |

**Table 4:** Optimized parameters for TSLA data.

| **Models** | $C$ | $\gamma$ | $\epsilon$ | **MSE** | **MAPE** | **Cost time** |
|---|---|---|---|---|---|---|
| **ABC-SVR** | 256 | 0.0931836 | 0.01365015 | 0.00136099 | 0.1213265 | 182.4114 |
| **BBO-SVR** | 7.85094 | 0.9454 | 6.10E-05 | 0.00228979 | 0.140091 | 18.7108 |
| **BOA-SVR** | 0.807825 | 0.751053 | 0.00616621 | 0.00148971 | 0.121654 | 27.5524 |
| **FA-SVR** | 254.41284 | 0.098945875 | 0.01357314 | 0.00136376 | 0.12223358 | 1004.9159 |
| **GSA-SVR** | 26.58903 | 0.1041562 | 0.00074545 | 0.00129006 | 0.1225588 | 127.6254 |
| **GWO-SVR** | 255.0317 | 0.0955308 | 0.01363054 | 0.00136184 | 0.1215525 | 132.128 |
| **HHO-SVR** | 197.209 | 0.4205269 | 0.01706072 | 0.00220655 | 0.1291157 | 40.27025 |
| **MFO-SVR** | 194.3398 | 0.08970023 | 0.01441153 | 0.00134894 | 0.1197894 | 126.3 |
| **MVO-SVR** | 181.4413 | 0.05951673 | 0.01497712 | 0.00129902 | 0.1149064 | 79.70025 |
| **NNA-SVR** | 253.9341 | 0.1465462 | 0.00812801 | 0.00148744 | 0.1337156 | 17.06885 |
| **SSA-SVR** | 255.3342 | 0.1333912 | 0.00467436 | 0.00136979 | 0.1581744 | 227.6579 |
| **WOA-SVR** | 256 | 0.1364358 | 0.00424138 | 0.00137741 | 0.1548629 | 221.5676 |

**Table 5:** Optimized parameters for TSM data.

| **Models** | $C$ | $\gamma$ | $\epsilon$ | **MSE** | **MAPE** | **Cost time** |
|---|---|---|---|---|---|---|
| **ABC-SVR** | 256 | 0.001393767 | 0.0320872 | 0.001427 | 0.042772 | 32.55109 |
| **BBO-SVR** | 13.4353 | 0.14468 | 6.10E-05 | 0.00147 | 0.043286 | 15.6106 |
| **BOA-SVR** | 51.0281 | 0.0233701 | 0.0106542 | 0.001466 | 0.042897 | 21.3761 |
| **FA-SVR** | 145.3738 | 0.004001169 | 0.03353944 | 0.001417 | 0.042468 | 157.6236 |
| **GSA-SVR** | 1.4889 | 0.0970193 | 0.00213057 | 0.001454 | 0.042101 | 20.1198 |
| **GWO-SVR** | 6.35558 | 0.0320962 | 0.00284335 | 0.001442 | 0.042212 | 24.3695 |
| **HHO-SVR** | 171.9925 | 0.9798621 | 0.1031784 | 0.001826 | 0.04655 | 22.02656 |
| **MFO-SVR** | 230.3153 | 0.003261379 | 0.04130702 | 0.001425 | 0.042671 | 28.87215 |
| **MVO-SVR** | 252.9431 | 0.03697545 | 6.10E-05 | 0.001465 | 0.043254 | 151.9899 |
| **NNA-SVR** | 256 | 0.01576342 | 0.0005394 | 0.00146 | 0.042764 | 16.40153 |
| **SSA-SVR** | 255.9636 | 0.004876394 | 0.00013112 | 0.001454 | 0.042515 | 185.3799 |
| **WOA-SVR** | 1.76772 | 0.0728386 | 6.10E-05 | 0.001471 | 0.04229 | 33.9208 |

## References

[1] Abarbanel, H., *Analysis of observed chaotic data*, Springer Science and Business Media, 2012.

[2] Chang, C.C., Lin, C., *Libsvm* J.,, *A library for support vector machines*, ACM transactions on intelligent systems and technology (TIST), 2011, **2**, 27. Doi: 10.1145/1961189.1961199

[3] Cao, W., Liu, X., and Ni, J., *Parameter optimization of support vector regression using Henry gas solubility optimization algorithm*, IEEE Access,2020, **8**, P.88633-88642. Doi**:** 10.1109/ACCESS.2020.2993267

[4] Chen, Y., Yang, H., *Multiscale recurrence analysis of long-term nonlinear and nonstationary time series, Chaos*, Solitons and Fractals, 2012, **45**, P.978–987. Doi: 10.1016/j.chaos.2012.03.013

[5] Cortes, C., Vapnik, V., *Support-vector networks*, Machine learning, 1995, **20**, P.273–297. Doi: 10.1007/BF00994018

[6] Diebold, F.X., Mariano, R.S.: *Comparing predictive accuracy, Journal of Business and economic statistics*, 2002, **20**, P.134–144. Doi: 10.1198/073500102753410444

[7] Ghanbari, M., Arian, H., *Forecasting stock market with support vector regression and butterfly optimization algorithm*, arXiv preprint arXiv, 2019,1905.11462.

[8] Huang, C.F., *A hybrid stock selection model using genetic algorithms and support vector regression*, Applied Soft Computing, 2012, **12**, P.807–818. Doi:10.1016/j.asoc.2011.10.009

[9] Huang, C., Li, Y., and Yao, X., *A survey of automatic parameter tuning methods for metaheuristics*, IEEE transactions on evolutionary computation, 2019, **24**, 2, P.201-216. Doi: 10.1109/TEVC.2019.2921598

[10] Kazemi, M.S., Banihabib, M.E., and Soltani, J., *A hybrid SVR-PSO model to predict concentration of sediment in typical and debris floods*, Earth Science Informatics, 2021, **14**, 1, P.365-376. Doi: 10.1007/s12145-021-00570-0

[11] Keerthi, S.S., Sindhwani, V., Chapelle, O., *An efficient method for gradient-based adaptation of hyperparameters in svm models*, Advances in neural information processing systems, 2007, P.673–680.

[12] Kennel, M.B., Brown, R., Abarbanel, H.D., *Determining embedding dimension for phase-space reconstruction using a geometrical construction*, Physical review A, 1992, **45**, 3403. Doi: 10.1103/PhysRevA.45.3403

[13] Li, S., Fang, H., Liu, X., *Parameter optimization of support vector regression based on sine cosine algorithm, Expert systems with Applications*, 2018, **91**, P.63–77. Doi: 10.1016/j.eswa.2017.08.038

[14] Liu, M., Luo, K., Zhang, J., and Chen, S., *A stock selection algorithm hybridizing grey wolf optimizer and support vector regression*, Expert Systems with Applications, 2021, **179**. Doi: 10.1016/j.eswa.2021.115078

[15] Luo, Z., Hasanipanah, M., Bakhshandeh Amnieh, H., Brindhadevi, K., and Tahir, M.M., *GA-SVR: a novel hybrid data-driven model to simulate vertical load capacity of driven piles*, Engineering with Computers, 2021, **37**(2), P.823-831. Doi: 10.1007/s00366-019-00858-2

[16] Malik, H., Iqbal, A., Joshi, P., Agrawal, S., and Bakhsh, F.I., eds., *Metaheuristic and Evolutionary Computation: Algorithms and Applications*, Springer Nature Singapore, 2021

[17] Rahmani, M., Khalili Eraq, M., Nikoomaram H., *Portfolio Optimization by Means of Meta Heuristic Algorithms*, Advances in mathematical finance applications, 2019, **4**, P.83-97. Doi: 10.22034/amfa.2019.579510.1144

[18] Sai, L., and Huajing, F., *WOA-based algorithm for parameter optimization of support vector regression and its application to condition prognostics*, In 2017 36th Chinese Control Conference (CCC), 2017, P.7345-7350, IEEE, Doi: 10.23919/ChiCC.2017.8028516

[19] Setiawan, I.N., Kurniawan, R., Yuniarto, B., Caraka, R.E., and Pardamean, B., *Parameter Optimization of Support Vector Regression Using Harris Hawks Optimization*, Procedia Computer Science, 2021,**179**, P.17-24. Doi: 10.1016/j.procs.2021.12.003

[20] Sun, Y., Ding, S., Zhang, Z. and Jia, W., *An improved grid search algorithm to optimize SVR for prediction, Soft Computing*, 2021, **25**(7), P. 5633-5644. Doi: 10.1109/ICCT.2012.6511415

[21] Takens, F., *Detecting strange attractors in turbulence, in: Dynamical systems and turbulence*, Warwick 1980, Springer, P.366–381. Doi: 10.1007/BFb0091924

[22] Tanyildizi, E., Demir, G., *Golden sine algorithm: a novel math-inspired algorithm*, Advances in Electrical and Computer Engineering, 2017, **17**, P. 71–79. Doi: 10.4316/AECE.2017.02010

[23] Vapnik, V., *The nature of statistical learning theory*, Springer science and business media, 2013

[24] Wang, D., Lu, Y., Chen, B., and Zhao, Y., *Wind weather prediction based on multi-output least squares support vector regression optimised by bat algorithm*, International Journal of Embedded Systems, 2020, **12**(2), P.137-145. Doi: 10.1504/IJES.2020.105936

[25] 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, 2009, **36**, P.4725–4735. Doi: 10.1016/j.eswa.2008.06.046

[26] Yeh, C.Y., Huang, C.W., Lee, S.J., *A multiple-kernel support vector regression approach for stock market price forecasting*, Expert Systems with Applications, 2011, **38**, P.2177–2186. Doi: 10.1016/j.eswa.2010.08.004

[27] Zanjirdar, M., *Overview of Portfolio Optimization Models*, Advances in mathematical finance and applications, 2020, **5**(4), P.419-435, Doi: 10.22034/amfa.2020.1897346.1407

[28] Zhang, J., Teng, Y.F., and Chen, W., *Support vector regression with modified firefly algorithm for stock price forecasting*, Applied Intelligence, 2019, **49(**5), P.1658-1674. Doi: 10.1007/s10489-018-1351-7

[29] Zheng, J., Wang, Y., Li, S., and Chen, H., *The Stock Index Prediction Based on SVR Model with Bat Optimization Algorithm*, Algorithms, 2021, **14**(10). Doi: 10.3390/a14100299

[30] Zhang, D., Xiang, W., Cao, Q., Chen, S., *Application of incremental support vector regression based on optimal training subset and improved particle swarm optimization algorithm in real-time sensor fault diagnosis*, Applied Intelligence, 2021, **51**(6), P.3323-3338. Doi: 10.1007/s10489-020-01916-9