



# Predict Stock Prices Using Supervised Learning Algorithms and Particle Swarm Optimization Algorithm

Mohammad Javad Bazrkar<sup>1</sup> · Soodeh Hosseini<sup>1</sup>

Accepted: 3 May 2022 / Published online: 5 June 2022  
© Springer Science+Business Media, LLC, part of Springer Nature 2022

## Abstract

Forecasting the stock market has always been one of the challenges for stock market participants to make more profit. Among the problems of stock price forecasting, we can mention its dynamic nature, complexity and its dependence on factors such as the governing system of countries, emotions, economic conditions, inflation, and so on. In recent years, many studies have been conducted to predict the capital stock market using traditional techniques, machine learning algorithms and deep learning. The lower our forecast stock error, the More we can reduce investment risk and increase profitability. In this paper, we present a machine learning (ML) approach called support vector machine (SVM) that can be taught using existing data. SVM extracts knowledge between data and ultimately uses this knowledge to predict new stock data. We have also aimed to select the best SVM method parameters using the particle swarm optimization (PSO) algorithm to prevent over-fitting and improve forecast accuracy. Finally, we compare our proposed method (SVM-PSO) with several other methods, including support vector machine, artificial neural network (ANN) and LSTM. The results show that the proposed algorithm works better than other methods and in all cases, its forecast accuracy is above 90%.

**Keywords** Stock market · Machine learning · Prediction · Support vector machine · Particle swarm optimization

---

✉ Soodeh Hosseini  
so\_hosseini@uk.ac.ir

Mohammad Javad Bazrkar  
bazrkar76@gmail.com

<sup>1</sup> Department of Computer Science, Faculty of Mathematics and Computer, Shahid Bahonar University of Kerman, Kerman, Iran

## 1 Introduction

Today's world is a world of change and knowing what awaits us in the future can greatly contribute to the success of individuals and organizations. In the past, traders took a lot of risks and always looked for a partner to share their profits and losses with, and everyone shared in the profits and losses according to the amount of their capital.

“Stock exchange” means an organized and formal capital market in which shares of companies and participation bonds are traded under certain terms and conditions. The stock exchange is a formal and reliable reference for attracting capital from individuals and the private sector to finance long-term investment projects.

With the continuous development of society's economy, the emergence of capital markets in countries has increased rapidly, so the investor needs powerful and reliable tools to predict stock prices and the ability to buy the stock they need. Stock prices are basically divided into four categories: dynamic, nonlinear, nonparametric and chaotic (Oh & Kim, 2002). Stock price forecasting has been considered by many stock market participants and traders for many years. Even today, buying and selling small stocks of listed companies is attractive to many ordinary people who do not have much capital. Stock market indices fluctuate widely and affect the amount of investment people make. Stock traders are looking for ways to increase their capital gains by predicting future stock prices. Therefore, it seems necessary that appropriate, correct and science-based methods in determining the future price of stocks be presented to investors.

In recent years, various methods for forecasting stock prices have been examined like the artificial neural network (ANN) method, which does not work properly due to many parameters in stock price forecasting and analysis (Tao et al., 2004). Support vector machine (SVM) is also mentioned as one of the efficient algorithms for stock price forecasting, which, unlike artificial neural networks, does not stick to local minimums and seeks optimal solutions globally while artificial neural networks are placed in local optimizations (Cherkassky & Ma, 2004). In support vector machine learning algorithm, regulator parameters and algorithm kernel parameters play an important role therefore, it is necessary to choose these parameters. The impact and importance of these parameters have been discussed in (Alvarez Meza et al., 2012).

In this paper, we try to predict the stock prices of Amazon and several other companies using the support vector machine, which is one of the machine learning (ML) approaches. We also use the radial base function (RBF) for the support vector machine. Finally, we try to use the particle swarm optimization (PSO) algorithm to optimize the parameters in the algorithm so that we can get the best values for the algorithm and thus get the best prediction. Although many studies have been done in the field of stock price forecasting using RBF kernel support vector machine and particle swarm optimization algorithm, in none of them the accuracy of forecasting the proposed method has reached over 90% and at best, they were about 70 percent (Karazmodeh et al., 2013). In this paper, by accurately and step-by-step optimizing the variables  $C$  and  $\delta$ , as well as by selecting the appropriate parameters of the PSO

algorithm, we were able to achieve above 90% accuracy in forecasting stock prices in all cases.

Our main motivation for doing this research is to provide a reliable way to predict stock prices so that stock market participants and the public can safely invest in the stock market and contribute to the growth and prosperity of their community economy. Our contributions in this paper are as follows. First, we improve the support vector machine algorithm for stock price forecasting using the radial basis function and particle swarm optimization algorithm and second increase the stock price forecasting accuracy and reduce investment risk.

The rest of the paper is as follows: Sect. 2 provides a brief summary of related stock price forecasting work. Section 3 describes and review the support vector machine algorithms, the radial base function, and the particle swarm optimization algorithm. Eventually, we create the proposed model in Sect. 4 and examine it. The results obtained and the evaluation of the model are done in Sect. 5, and finally in Sect. 6 we give a general conclusion of the proposed model.

## 2 Related Work

Given the significant gains and losses that may occur in the stock market, many methods for predicting stock prices have been explored to date. Hegazy et al. (2014) using the least squares support vector machine algorithm (LS-SVM) and particle swarm optimization algorithm (PSO) were able to increase the accuracy of stock price forecasting compared to methods such as artificial neural network (ANN). They also managed to avoid the over-fitting that happens in ANN'S method but the lack of forecast parameters and lack of explanation of existing parameters is one of the drawbacks of this project. Hafezi et al. (2015) using the bat-neural network multi-agent system (BNNMAS), they tried to pre-process in parallel on the data and increase the accuracy of stock price forecasting compared to ANN and genetic algorithm neural network (GANN) methods. They also proved that this method is suitable for forecasting stock prices in the long run.

Reddy (2018) using support vector machine (SVM) by radial basis function (RBF) kernel tried to develop financial data forecasting. Bruno Miranda Henrique et al. (2018) using support vector regression (SVR) method, they tried to reduce the prediction error compared to the proposed random stage model, efficient market hypothesis (EMH) and zero return random model. Unfortunately, this method does not specify items such as transaction costs or portfolio risk levels and uses a set of fixed tutorials at current prices. Rachna Sable et al. (2019) have presented an article reviewing machine learning algorithms for predicting stock prices. Hernández-Álvarez et al. (2019) used random trees (RT) and multilayer perceptron (MP) algorithms to predict stock prices and were able to increase forecast accuracy and use computers as a tool for automated data processing and data mining. Soroda Basak et al. (2019) provides a model for predicting stock movement in long-term forecasts using random forests (RF) and gradient boosted decision's trees (GBDT). Mehar Vijn et al. (2020) using artificial neural network (ANN) and random forest (RF), they have predicted the closing price of stocks and have increased the

accuracy of the forecast, but unfortunately, they do not take into account financial parameters such as final price, profit and loss, etc. JinShan Yang et al. (2020) used gradient boosting (GBDT) to predict stock prices. They first used natural language processing to deal with the text. Then, they reduce dimension by PCA, finally, with GBDT, they have forecast stock prices. Amanat Ullah et al. (2021) placed a lot of emphasis on feature selection and extracted 28 features from stock data to predict stock prices. They also combined machine learning classifiers for stock trading and were able to prove that group learning in stock trading works better than using a single algorithm. Sohrab Mokhtari et al. (2021) predicted stock prices by incorporating artificial intelligence into machine learning algorithms.

Table 1 shows some of the work done on stock price forecasting using machine learning algorithms and outlines some of the advantages and disadvantages of these methods.

### 3 Review of SVM and PSO

Considering the importance of the support vector machine algorithm and the role of the kernel in it and the importance of the particle swarm optimization algorithm in the proposed method (SVM-PSO), in this section we explain these algorithms and their application in our proposed method.

#### 3.1 Support Vector Machine (SVM)

The original SVM algorithm was invented in 1963 by Vladimir Vepnik and was generalized by Vapnik and Cortes to the nonlinear mode (1995). Support vector machine is a supervised machine learning model used to categorize data. After giving the SVM dataset to the training tags for each group, they can categorize the new data. SVM is also used in regression.

The purpose of the SVM algorithm is to find a decision boundary called a hyper-plane in a  $N$ -dimensional space ( $N$ -number of attributes) that specifically classifies data points. Maximizing the distance between the nearest points of each class and the hyper-plane leads to the separation of the hyper-plane. This distance is called the margin. The main purpose of SVM is to find the optimal hyper-plane because it not only categorizes existing data sets but also helps predict unseen data classes. The hyper-plane is the ideal aircraft with the most margins.

The general equation of a hyper-plane is as follows:

$$F(x) = W^T X + b = 0 \quad (1)$$

where  $W$  is the weight vector,  $X$  is the input vector, and  $b \in R$ . The formula for calculating the margin is as follows:

$$m = \frac{2}{\|W\|} \quad (2)$$

To increase the margin, we must minimize the denominator of the fraction ( $\|W\|$ ). For convenience and to use linear algebra, instead of minimizing  $\|W\|$ , we minimize the following function:

$$\min \frac{1}{2} \|W\|^2 \quad \left( \frac{1}{2} \text{ is convenient for taking derivative later on} \right) \quad (3)$$

There are many cases in which data cannot be fully categorized through linear segregation. In such cases, the support vector machine seeks out the extraordinary aircraft that maximizes the margin and minimizes incorrect classifications. For this, we introduce the slack variable,  $\epsilon_i$  which allows some objects to fall off the margin but it penalizes them. The slack variables  $\epsilon_i > 1$  for misclassified points, and  $0 < \epsilon_i < 1$  for points close to the decision boundary.

In this scenario, the algorithm tries to keep the slack variable zero while maximizing the margin. However, it minimizes the total distance of incorrect classification from marginal hyper-plane and the number of incorrect classifications. The optimization problem (3) is now as follows:

$$\min \frac{1}{2} \|W\|^2 + C \sum_{i=1}^N \epsilon_i \quad (4)$$

Here, the parameter  $C$  is the regularization parameter that controls the trade-off between the slack variable penalty (misclassifications) and width of the margin and  $N$  is the number of slack variables. Selecting the  $C$ -parameter is very important because: Small  $C$  makes it easy to ignore constraints, which leads to large margins and contains a lot of unwanted data. Also in this case the training error and the test error are both large and we get under fitting. Large  $C$  allows the constraints hard to be ignored which leads to a small margin. In this case, our training error is small while the test error is high. In other words, in this case we suffer from over fitting. There are no rules or solutions for choosing the value of  $C$ . In this paper, we use the particle swarm optimization (PSO) algorithm to select the optimal value of parameter  $C$  optimally.

### 3.1.1 Kernel Functions

For SVM, the easiest way to separate two data classes is one line for two-dimensional data and one plane for three-dimensional data. But lines or planes cannot always be used, and we sometimes need a nonlinear region to separate these classes. Support vector machine manage such situations using a kernel function that maps data to another space and can be used as a linear hyper-plane to separate classes. This kernel trick is known to turn the data kernel performance into a higher next feature space to allow linear separation.

There are several functions for the kernel  $K$ , including:

- Linear kernel  $K(X, X_i) = X_i^T X$  (5)
- Polynomial kernel of degree  $d$   $K(X, X_i) = (1 + X^T X_i)^d$  for any  $d > 0$  (6)

**Table 1** A summary of related works is considered

Authors	Method	Advantage	Challenges
Osman Hegazy et al. (2014)	LS-SVM with PSO algorithm	<ol style="list-style-type: none"> <li>1. Increases the accuracy and convergence of the model compared to the ANN method</li> <li>2. Reduces forecast error</li> <li>3. Prevents over fitting</li> <li>4. Easy selection and best parameters using PSO algorithm</li> </ol>	Lack of parameters and lack of proper explanation for existing parameters
Reza Hafezi et al. (2015)	Bat-neural network multi-agent system (BNNMAS)	<ol style="list-style-type: none"> <li>1. Performs data processing in parallel</li> <li>2. Improving results and predictive accuracy compared to neural network methods, genetic algorithm (GANN) and some other models such as generalized regression neural network (GRNN)</li> <li>3. A convenient tool for forecasting stock prices in the long run</li> </ol>	<ol style="list-style-type: none"> <li>1. Failure to implement step-by-step scenario planning to complete the entire proposed model</li> <li>2. Inadequate comparison with other models</li> <li>3. Uncertainty of complexity in the inherent data set</li> </ol>
Reddy (2018)	Support vector machine (SVM) by RBF kernel	<ol style="list-style-type: none"> <li>1. Development of financial data forecasts</li> <li>2. Prevent over fitting in the support vector machine (SVM)</li> <li>3. Based on the proposed algorithm, it creates practical models with high performance</li> </ol>	Insufficient comparison with other models
Bruno Miranda Henrique et al. (2018)	Support vector regression (SVR)	<ol style="list-style-type: none"> <li>1. Periodic model updates</li> <li>2. Reduction of forecast error compared to the proposed random stage model, efficient market hypothesis (EMH) and zero return random model</li> <li>3. Increased forecast accuracy during lower swing periods</li> </ol>	<ol style="list-style-type: none"> <li>1. Uses a set of fixed tutorials at current prices</li> <li>2. It does not specify items such as transaction costs or portfolio risk level</li> <li>3. It is possible to create risk</li> <li>4. Some of the results presented suffer from poor data input</li> </ol>
Rachna Sable et al. (2019)	An overview of machine learning algorithms (SVM / ANN / PSO-LSM and et al.) for stock price forecasting	<ol style="list-style-type: none"> <li>1. Examines different types of machine learning algorithms to predict stock prices</li> <li>2. Focuses on different data</li> <li>3. Uses different evaluation criteria</li> </ol>	Lack of explanation of compared algorithms and evaluation criteria

**Table 1** continued

Authors	Method	Advantage	Challenges
Hernández-Álvarez et al. (2019)	Random trees and multilayer perceptron algorithms	<ol style="list-style-type: none"> <li>1. Using computers as a tool for automated data processing and data mining</li> <li>2. Appropriate and useful use of weka tools</li> <li>3. Prediction accuracy increases</li> </ol>	<ol style="list-style-type: none"> <li>1. No tested with various datasets</li> <li>2. Failure to parse the quantity of emotions expressed to increase the accuracy of the prediction</li> </ol>
Suryoday Basak et al. (2019)	Random forests, and gradient boosted decision 'n trees	<ol style="list-style-type: none"> <li>1. Forecast for stock movement and high accuracy in long-term forecasts</li> <li>2. Increased prediction accuracy compared to SVM and ANN methods</li> <li>3. Understands the scope of big data using proposed and developed models</li> <li>4. No severe bias and no data bias effect</li> </ol>	Reinforced tree models have not been used to predict short-term trends
Mehar Vijh et al. (2020)	Artificial neural network (ANN) and random forest (RF)	<ol style="list-style-type: none"> <li>1. The efficiency of the proposed model in predicting the closing price of stocks</li> <li>2. Finds hidden features through a self-learning process</li> <li>3. Increases the accuracy of the proposed model and reduces the error</li> </ol>	<ol style="list-style-type: none"> <li>1. Lack of comparison with different algorithms</li> <li>2. It does not take into account financial parameters such as final price, profit and loss, etc</li> </ol>
JinShan Yang et al. (2020)	Gradient Boosting (GBDT)	<ol style="list-style-type: none"> <li>1. Reliable public news processing using natural language processing</li> <li>2. Increase forecast accuracy to 65%</li> </ol>	<ol style="list-style-type: none"> <li>1. Lack of proper reduction of the scope of calculating the vote model</li> <li>2. For unlabeled data, semi-supervised learning is not performed</li> </ol>
Amanat Ullah et al. (2021)	Combining four Bayesian naive Gaussian methods, decision tree, logistic regression by setting L1 and random gradient descent	<ol style="list-style-type: none"> <li>1. Extract appropriate features using modern methods to predict stock prices</li> <li>2. Increase the accuracy of prediction compared to individual methods</li> </ol>	<ol style="list-style-type: none"> <li>1. Resource constraints</li> <li>2. Inability to predict high frequency stock trading</li> <li>3. Inability to implement neural network</li> </ol>

**Table 1** continued

Authors	Method	Advantage	Challenges
Sohrab Mokhtari et al. (2021)	ANN, DT, and SVM models, logistic regression (LR), Gaussian naive Bayes (GNB), Bernoulli Naive Bayes (BNB), random forest (RF), k-nearest neighbor (KNN), and XGboost (XGB). moreover, linear regression and long short-term memory (LSTM) are used in regression problem	1. Stock market analysis from a technical and fundamental point of view 2. Comparison of the performance of machine learning algorithms considering artificial intelligence	1. Inadequacy of the model for long-term forecasting 2. Lack of proper development of computational power of artificial intelligence

- Multi-layer perceptron (MLP) kernel  $K(X, X_i) = \tanh(KX_i^T X + \theta)$  (7)
- Radial basis function (RBF) kernel  $K(X, X_i) = \frac{\exp(-X - X_i^2)}{2\delta^2}$  for any  $\delta > 0$  (8)

In this paper, we use the kernel of the radial basis function for the support vector machine. The choice of parameter  $\delta$  in RBF is of great importance. If the  $\delta$  parameter is selected correctly, the accuracy of the algorithm will increase. So far, unfortunately, there are no exact rules or solutions for choosing parameter  $\delta$ . In this paper we use the particle swarm optimization (PSO) algorithm to select the optimal value of parameter  $\delta$ . The proposed algorithm is implemented in Google Colab using python language. Algorithm1 shows the pseudo-code of the SVM algorithm with the RBF kernel and the selection of parameters by PSO.

**Algorithm1:** Pseudo-code of SVM algorithm with RBF kernel

1. Start
2. Load data ()
3. For 1 through Stop condition
4. Specify the C range and  $\delta$  with PSO
5. SVM (kernel = RBF, C = ' \ ' ,  $\delta$  = ' \ ' )
6. Error calculation
7. Print output
8. End

### 3.2 Particle Swarm Optimization (PSO) Algorithm

Particle swarm optimization algorithm is one of the most important intelligent optimization algorithms in the field of collective intelligence. The PSO implements a model for us that is based on sharing information and experiences. This algorithm was introduced by James Kennedy and Eberhart (1995). The PSO algorithm is inspired by the social behaviour of animals such as fish and birds. These animals live together in small and large groups. In the PSO algorithm, the members of the population of answers are directly related to each other and solve the problem by



exchanging information with each other, recalling good memories of the past. Like other collective intelligence algorithms, the PSO algorithm uses a set of possible answers. The algorithm continues until an optimal answer is found or the conditions for the end of the algorithm are met. In this method, each  $x$  response is represented as a particle, and a group of particles is a set of responses. In PSO, the velocity equation guarantees the movement of particles towards the optimal region. This equation is usually based on three main elements: The speed of stillness, cognitive component (Pbest), collective component (Gbest). In algorithm simulation, the behavior of each particle can be influenced by the best local particle within a given neighborhood or the best general particle.

In general, if  $\vec{X}_i(t)$  represents the position of the  $P_i$  particle in the search space at moment  $t$ , position  $P_i$  changes by adding speed  $\vec{V}_i(t)$  to the current position as follows:

$$\vec{X}_i(t) = \vec{X}_i(t - 1) + \vec{V}_i(t) \tag{9}$$

$$\vec{V}_i(t) = \vec{V}_i(t - 1) + c_1r_1(\vec{P}_b - \vec{X}_i(t - 1)) + c_2r_2(\vec{G}_b - \vec{X}_i(t - 1)) \tag{10}$$

where  $\vec{V}_i(t)$  is the velocity vector in step  $t$ ,  $c_1$  and  $c_2$  are positive fixed values and  $r_1$  and  $r_2$  are random numbers that are normally generated in the interval  $[0, 1]$ . Parameters  $\vec{P}_b$  and  $\vec{G}_b$  also represent the best position for personal and collective experience, respectively. In order to create a better search capability, a parameter called inertial weight is added to the algorithm as a factor in the speed parameter as shown below:

$$\vec{V}_i(t) = W\vec{V}_i(t - 1) + c_1r_1(\vec{P}_b - \vec{X}_i(t - 1)) + c_2r_2(\vec{G}_b - \vec{X}_i(t - 1)) \tag{11}$$

The inertial weight determines the effect of the velocity of the particles in the previous step on the velocity now. In this way, with large amounts of inertia weight, the general search ability of the algorithm is improved, and more space is explored, however, with small amounts of inertial weight, the space under study is limited and the search is performed in this limited space. Therefore, the algorithm typically starts moving with a large amount of inertial weight, which causes a large space search at the beginning of the run. Over time, the weight of inertia decreases, which concentrates the search for small space in the final steps. Algorithm2 shows the pseudo-code of the PSO algorithm. The PSO algorithm is suitable for a variety of continuous and discrete problems and provides substantial answers to various optimization problems.

**Algorithm2: PSO pseudo-algorithm code**


---

```

1. For each particle
{
2. Randomly initialized position and velocity of particles:
    $X_i(0)$  and  $V_i(0)$ 
}
3. Do perform maximum or minimum error repeat criteria
{
4. For each particle
{
5. Calculate data fitness value
6. If the amount of fitness value is better than Pbest
{
7. Set Pbest = the amount current fitness value
}
8. If Pbest is better than Gbest
{
9. Set Gbest = Pbest
}
}
10. For each particle
{
11. Calculate the velocity of the particles
12. Use Gbest and velocity to update particle data
}
}

```

---

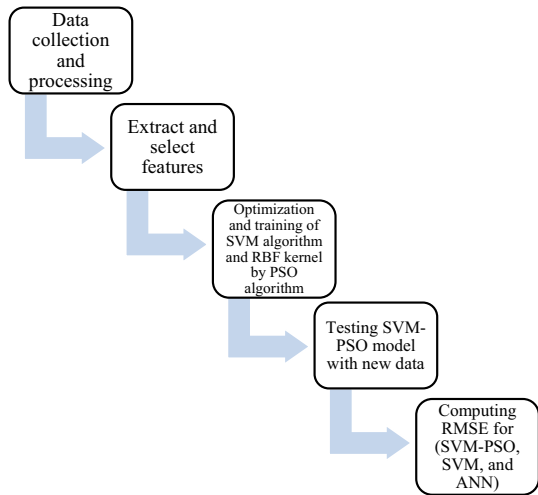
## 4 Proposed Model

In this paper, we try to predict the closing price of the stock day by using the support vector machine with RBF kernel. We also use the PSO algorithm to optimize variables  $C$  and  $\delta$  to increase the model prediction accuracy. We used hybrid methods in this paper because hybrid learning techniques are an ideal way to achieve the highest accuracy (Abraham et al., 2009). Our proposed model is based on a study of data from the last 5 years (2016–2020) of Amazon (is an e-commerce company), Oracle (is a software super company) and IBM (is an international commercial machinery company). We use the root mean square error (RMSE) as a criterion for comparing the proposed model with the artificial neural network (ANN) and SVM method without optimizing the mentioned parameters ( $C, \delta$ ). The proposed model structure uses six features to predict the closing price of the stock and the output of the model is the same as the closing price of the stock the next day. Figure 1 shows the structure of the proposed model. In the following, we will explain the structure steps of the proposed model.

### 4.1 Data Collection and Processing

Collecting stock market data inside the country has many obstacles because many companies on the stock exchange, in order to compete and protect their privacy, refuse to disclose their data to the public. In order to avoid such problems as well as other problems such as the system of countries, emotions, economic conditions and inflation, in this paper, the stock market data of three famous companies Amazon, Oracle and IBM have been used. All datasets used are available on the Yahoo

**Fig. 1** Structure of the proposed model



Finance website (Yahoo Finance website), which is a media asset that is part of the Yahoo network. This website includes financial reports, news, data and commentary on the stock prices of various companies, press releases and more. Yahoo Finance also has some online financial management tools that it offers to its users.

Data preprocessing plays a key role in the data processing process and its results. There are several steps and tools for data preprocessing (Alasadi & Bhaya, 2017). Fortunately, the data used did not contain any missing values, but in some cases included outliers. The stock price on some days has not followed its usual pattern and has increased or decreased significantly, and this has led to the inclusion of outliers in the data set used. In this paper, to solve the problem of outliers data, we replace their values with the average of the previous normal data. We get the average value for non-outliers data and replace it with outliers data. This way all the data is within a reasonable range.

Given that the values of the data properties used in stock price forecasting are significantly different from each other, data normalization plays an important role in our work. To achieve more accurate results, it is necessary that the intervals related to the different features be somehow identical or close to each other. There are many ways to normalize data (Yu et al., 2009). In fact, normalizing the data increases the accuracy of the forecast (Panigrahi & Behera, 2013). In this paper, we use the Min–Max normalization method. In this method, each set of data is mapped to arbitrary intervals, the minimum and maximum values of which are already known. In this method, any desired interval can be mapped to new intervals with just a simple conversion. Suppose attribute A is to be mapped from the data set between  $Min_A$  to  $Max_A$  to the new range  $new_{Min}$  to  $new_{Max}$ . For this purpose, any initial value such as  $t$  in the initial interval will be converted to the new value  $t'$  in the new interval according to the following equation (Yu et al., 2009):

$$t' = \frac{(new_{Max} - new_{Min})}{Max_A - Min_A} (t - Min_A) + new_{Min} \tag{12}$$

## 4.2 Extract and Select Features

Many features can be used to predict stock closing price. In this paper, we use six features to predict stock closing prices. According to studies, these six characteristics play an essential role in predicting stock prices (Adebiyi et al., 2014; Göçken et al., 2016; Li et al., 2014; Patel et al., 2015). In the following, we will briefly explain these extracted features.

*Date of exchanges:* In the proposed model, we have the date of the exchanged shares in each time period and we use this feature to calculate the closing price of the shares in the next date and days. *Open stock price:* This feature represents the price for the first stock trading in a period of time (usually one day). The open stock price starts the first trading price of a day or a certain period.

*High stock prices:* This feature actually represents the price at which the stock reaches its highest level during the trading period. This is the point where there are more sellers than stock buyers because stock sellers always tend to sell their stocks at high prices so supply volumes and the number of suppliers in this price increase. Also, the high price of stocks indicates that this is the highest price that buyers have been willing to buy stocks during that period. In other words, the high stock price shows that shareholders have been willing to buy stocks up to this price during this period.

*Low stock prices:* The low stock price actually represents that the stock has reached its lowest price level during the trading period. In this price there are more buyers than stock sellers. Obviously, shareholders always tend to buy stocks at low prices and based on this incentive, at low prices, the amount and number of demand increases. Also, the low price indicates that this is the lowest price at which Stock sellers were willing to sell their stock during that time.

*Volume Trades:* This feature shows the number of sales and buy transactions that take place over a period of time. The trading volume confirms the stock trading trend. Heavier trading volume indicates the buyers' desire for a share or the selling pressure on a share, which is in line with the moving trend of stocks. If the price trend is an uptrend, the volume needs to be heavier on positive days to continue this uptrend, conversely, if the downtrend dominates the share, the downtrend will continue as long as the trading volume is heavier on negative days.

*Closing price of shares:* This feature represents the last price that the stock has been traded during the trading period. The closing price of stocks is the weighted average price of stocks during the trading period, taking into account the base volume. The base volume is the minimum number of shares of a company that must be traded during the day so that the price of that share can increase or decrease based on the range of the fluctuation range. To calculate the base volume, it is enough to mint the number 0.0004 (it varies in different countries) in the number of shares of the desired company to calculate the base volume of that company. The effect of the base volume on the closing price of the stock is that if the volume of transactions is more than the base volume, the calculated average is recorded as the closing price of the share in the trading board but if the volume of transactions is not

more than the base volume, the final price will be adjusted as much as the difference between the two volumes. In general, the closing price of the stock is calculated according to the following formula:

$$P_2 = P_1 + \left[ (VWAP - P_1) \frac{N}{M} \right] \tag{13}$$

where  $P_1$  is the closing price of the stock previous day, VWAP is the volume weighted average price of the current day,  $N$  is the trading volume of that share in the current trading session,  $M$  is the base volume and  $P_2$  is the closing price of the stock today. How to calculate VWAP is as follows:

$$VWAP = \frac{\text{Price} \times N}{\sum_{i=1}^n N} \tag{14}$$

where the price is the price of each transaction and  $n$  is the number of transactions per share. As mentioned, if the volume of transactions is more than or equal to the base volume the same VWAP price will be the basis for calculating the closing price of stocks in the stock index:

$$P_2 = VWAP \tag{15}$$

Also, the closing price of stocks on each day is the basis of price fluctuations the next trading day. For example, if the closing price of today's stock is 1,000 \$ per share, the allowable range of price fluctuations on tomorrow's trading day is 5% positive (1,050 \$) and 5% negative (950 \$) of that share. (This range varies from country to country). The closing price of stocks is often the most used in stock market analysis. In this paper, we use this feature as the target feature and try to predict the closing price of the stock using the proposed model.

### 4.3 Optimization and Training of SVM Algorithm and RBF Kernel by PSO Algorithm

To optimize parameter  $C$  in the SVM algorithm as well as parameter  $\delta$  in the RBF kernel, no specific rules have been defined to date. Since the values selected for these two parameters have a great impact on the prediction accuracy of the model and also affect the error rate, in this paper, we try to obtain the best values for these two parameters by using the particle swarm optimization algorithm. In the third part, Algorithm1 showed the selection of  $C$  and  $\delta$  parameters by PSO algorithm.

After optimizing the  $C$  and  $\delta$  parameters by the PSO algorithm, we train our proposed model with two support vector machine models without optimizing the mentioned parameters and the artificial neural network. We used data from January 2016 to December 2019 Amazon, Oracle and IBM as training data.

**Table 2** Parameters of algorithms used in this work

Parameter	Value
Iteration PSO	20
Inertia-weight PSO	1.1
$c_1$ and $c_2$ PSO	2
$r_1$ and $r_2$ PSO	Random numbers
SVM-PSO kernel	RBF
C-min $_{SVM-PSO}$	1
C-max $_{SVM-PSO}$	200
$\delta$ -min $_{SVM-PSO}$	0.01
$\delta$ -max $_{SVM-PSO}$	2
Validation type	K-fold
n-validations	10
Validation-size	0.1
Build ANN network	By tensor flow
Number of ANN layers	3
Number of first layer neurons	512
Number of second layer neurons	256
Number of third layer neurons	128
ANN activation function	Relu function
Sigma $_{ANN}$	1
Initial weights and bias in ANN	Random numbers
Epochs $_{ANN}$	10
Batch size $_{ANN}$	50
SVM kernel	RBF
$C$ $_{SVM}$	2000
$\delta$ $_{SVM}$	0.01
LSTM <sup>a</sup> network	Tensor flow
Number of LSTM layers	3
Number of first layer neurons	50
Number of second layer neurons	50
Number of third layer neurons	50
LSTM activation function	Relu function
LSTM time step	60

<sup>a</sup>Long short term memory

#### 4.4 Testing SVM-PSO Model with New Data

To test the proposed SVM-PSO model, we use the 2020 datasets of three companies, Amazon, Oracle and IBM, to determine the accuracy of the proposed model. We also calculate the accuracy of the two models SVM and ANN and compare it with the proposed model. The results obtained in the fifth section show

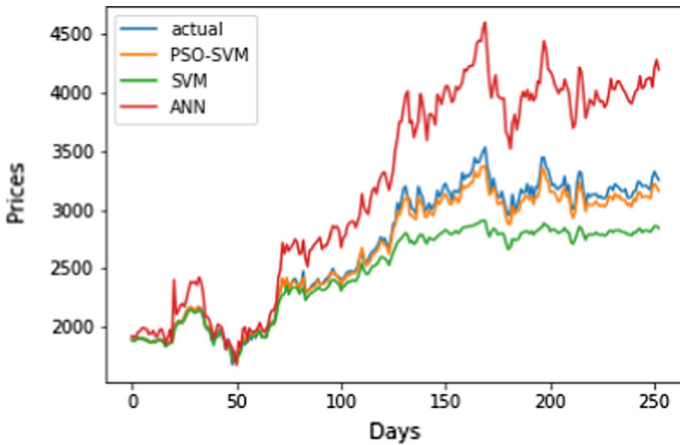


Fig. 2 Results for Amazon company

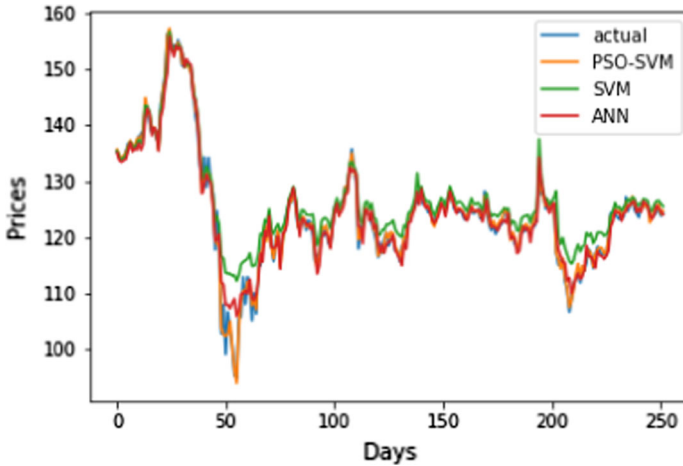


Fig. 3 Results for IBM company

that the accuracy of the proposed model in all cases is higher than the SVM and ANN models.

#### 4.5 Computing RMSE for SVM-PSO, SVM, and ANN

After calculating the accuracy of the proposed model and comparing it with the other two models, we will then compare the three models using the root mean square error evaluation criterion. The results obtained in the fifth section show that the proposed SVM-PSO model has better performance compared to the other two models and has a lower RMSE error rate.

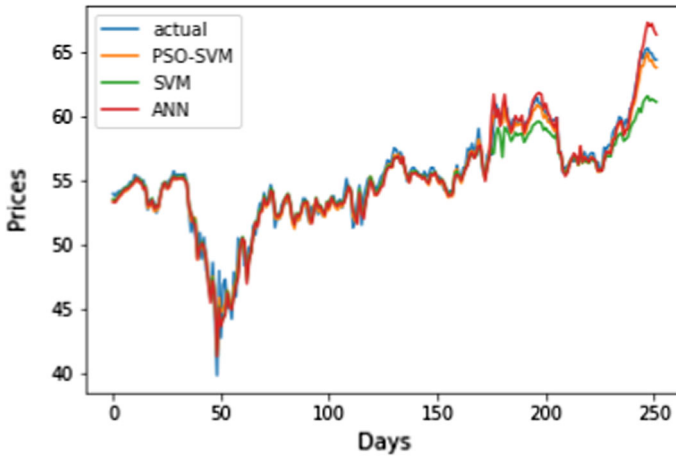


Fig. 4 Results for Oracle company

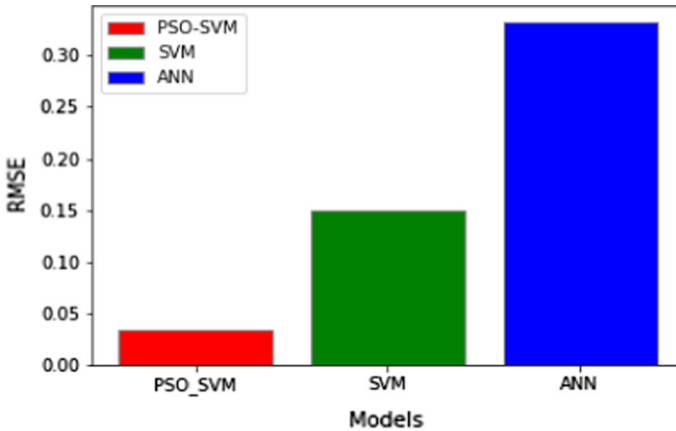


Fig. 5 RMSE for Amazon company

The RMSE compares the values predicted by the model with the actual measured values (Dennison et al., 2004). The following is how to calculate RMSE (Dennison & Roberts, 2003):

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \hat{x}_i)^2} \quad (16)$$

where  $N$  is the number of data,  $x_i$  is the actual measured value of the data, and  $\hat{x}_i$  the amount of data predicted by the learning model. In this paper  $x_i$  is the closing price of stocks in one day and  $\hat{x}_i$  is the closing price of stocks on the same day as predicted by the model.



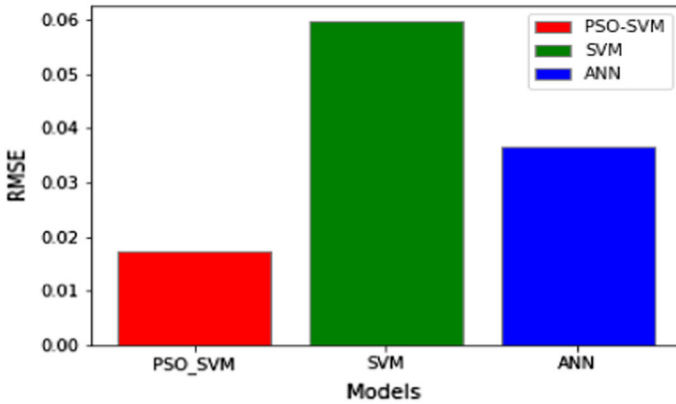


Fig. 6 RMSE for IBM company

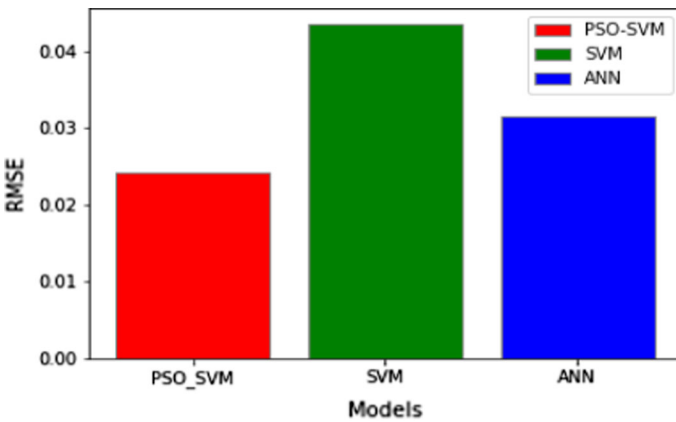


Fig. 7 RMSE for Oracle company

**Table 3** Root Mean Square Error (RMSE) for proposed algorithm

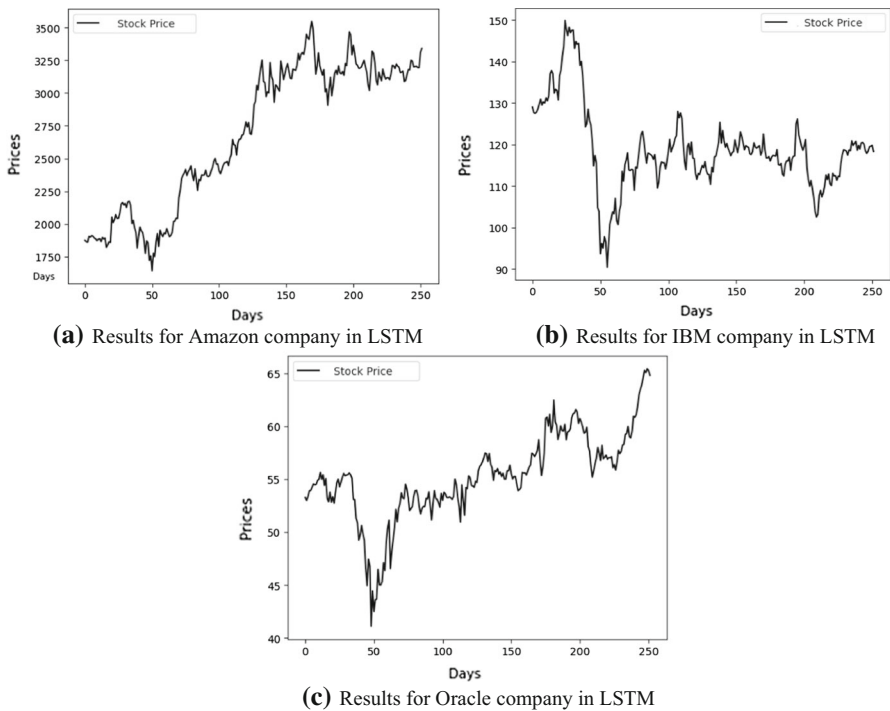
Company	Algorithm			
	SVM-PSO	SVM	ANN	LSTM
Amazon	<b>0.0333</b>	0.1490	0.3328	0.0424
IBM	<b>0.0173</b>	0.0597	0.0365	0.0207
Oracle	<b>0.0241</b>	0.0435	0.0314	0.0261

### 5 Results and Discussion

SVM-PSO, SVM and ANN algorithms were trained and tested with datasets form January 2016 to December 2020. All datasets used are available in (Yahoo Finance website). All data sets are divided into two parts: education from 2016 to 2019 (80%

**Table 4** Stock forecast accuracy for the proposed algorithm

Company	Algorithm			
	SVM-PSO	SVM	ANN	LSTM
Amazon	<b>0.9871</b>	0.7423	0.2047	0.9385
IBM	<b>0.9895</b>	0.8764	0.9361	0.9774
Oracle	<b>0.9770</b>	0.9250	0.9387	0.9701

**Fig. 8** Stock prices in Amazon, IBM and Oracle companies

of data) and the test part, which is 2020 data (20% of data). Table 2 describes the parameters used by SVM, ANN and SVM-PSO.

Figures 2, 3 and 4 shows the application of the proposed SVM-PSO model in comparison with SVM and ANN (multilayer perceptron) algorithms in different data sets. In Fig. 2, which are the results of Amazon, the results show that the proposed SVM-PSO algorithm has the highest accuracy with the optimizations performed, followed by the SVM algorithm without optimization and finally the ANN algorithm with the highest accuracy. Figure 3 shows the results from IBM. As can be seen, the proposed SVM-PSO algorithm has the highest accuracy, followed by the ANN algorithm, and finally the SVM algorithm has the highest accuracy. Figure 4 shows the results from Oracle. It can be stated that the predicted curve

using the proposed SVM-PSO algorithm is closer to the real curve in terms of accuracy, and followed by ANN, while SVM is the worst one.

Figures 5, 6 and 7 shows a comparison between SVM-PSO, SVM and ANN algorithms with respect to root mean square error function (RMSE) for the three companies surveyed. As can be seen, the error of the proposed algorithm in all three companies is less compared to other algorithms. Table 3 also shows the performance of the RMSE function for the proposed algorithm. It can be seen that the proposed SVM-PSO algorithm has always performed better than the SVM, ANN and LSTM algorithms, especially in cases where there are fluctuations in time series performance. Also, Table 4 shows the accuracy of stock forecasting in the three companies for the proposed method and other methods. The proposed algorithm can predict the stock price of each company in accordance with the mentioned features and show the high accuracy. As shown in Tables 3 and 4, we consider LSTM algorithm to compare the proposed method with this method. The results show that the error of the proposed method in three companies is less compared to other algorithms and the accuracy of the proposed method is higher than other methods. The results of stock prices for Amazon, IBM and Oracle companies by considering LSTM algorithm is presented in Fig. 8. This algorithm is to directly use yesterday's closing price as the predicted value of tomorrow's closing price.

## 6 Conclusions

In this paper, we used a machine learning method called support vector machine (SVM) to predict stock closing price. We also optimized the parameter  $C$  in SVM and  $\delta$  in the RBF kernel using the particle swarm optimization (PSO) algorithm and were able to obtain the best parameters for our proposed model. The results show that the proposed SVM-PSO algorithm in all cases has more accuracy and less error than SVM ANN and LSTM algorithms.

We can also predict the closing price of each company's stock according to the characteristics of open stock price, high stock price, low stock price and trading volume. Rather, the shares of each company can be predicted according to the considered indicators, which are also among the global indicators.

Among the challenges and limitations in this subject can be mentioned the following:

- Fragmented and non-standard data.
- Long data training time.
- Lack of comprehensive cooperation of stock exchange organizations to provide the required information.

## 6.1 Practical Implications

The findings of this study also provide important suggestions for stock market researchers and investors. Stock and capital markets are one of the key facts about the development and progress of the economy of societies. In order to have an active and dynamic stock market, it is necessary to create conditions for the general public to be able to have an active presence in this market.

Prerequisite for the growth and development of the stock market is appropriate and exciting advertising by government agencies. Also, the existence of reliable ways to predict stock prices that increase profitability and reduce investor losses is one of the important factors in encouraging investors to invest in stocks.

So far, many evolutionary algorithms have been used to optimize machine learning parameters and deep learning algorithms in stock price prediction forecasting (Hegazy et al., 2015), but none of them have been more than 90% accurate. Compared to previous work done in the field of stock price forecasting, the highest accuracy of which was 70%, our proposed algorithm was able to achieve above 90% accuracy in all cases by accurately optimizing the parameters. Achieving this level of accuracy can create a dramatic start and change in the market economy of communities and cause more people to enter the stock market due to the accurate forecast of this market.

## 6.2 Suggestions for Future Research and Further Scope of Research

As future works, we can use other evolutionary algorithms such as firefly algorithm (FA) to adjust  $C$  and  $\delta$  parameters and see what changes are achieved in the results obtained. In general, evolutionary algorithms play an important and effective role in parameter optimization (Vala et al., 2021).

We can also, as future works, increase the accuracy of stock price forecasting by considering the precise and periodic feelings of people at any given time and building an algorithm for it.

## Declarations

**Ethical Approval** This article does not contain any studies with human participants or animals performed by any of the authors.

**Conflict of interest** The authors declare that they have no conflict of interest.

**Informed Consent** The participant has consented to the submission of the case report to the journal.

## References

- Abraham, A., Corchado, E., & Corchado, J. M. (2009). Hybrid learning machines. *Neurocomputing: an International Journal*, 72, 2729–2730.
- Adebisi, A. A., Adewumi, A. O., & Ayo, C. K. (2014). Comparison of ARIMA and artificial neural networks models for stock price prediction. *Journal of Applied Mathematics*, 2014, 1–7.
- Alasadi, S. A., & Bhaya, W. S. (2017). Review of data preprocessing techniques in data mining. *Journal of Engineering and Applied Sciences*, 12(16), 4102–4107.

- Alvarez Meza, A. M., Daza Santacoloma, G., Acosta Medina, C. D., & Castellanos Dominguez, G. (2012). Parameter selection in least squares-support vector machines regression oriented, using generalized cross-validation. *Dyna*, 79(171), 23–30.
- Basak, S., Kar, S., Saha, S., Khaidem, L., & Dey, S. R. (2019). Predicting the direction of stock market prices using tree-based classifiers. *The North American Journal of Economic and Finance*, 47, 552–567.
- Cherkassky, V., & Ma, Y. (2004). Practical selection of SVM parameters and noise estimation for SVM regression. *Neural Networks*, 17(1), 113–126.
- Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3), 273–297.
- Dennison, P. E., Halligan, K. Q., & Roberts, D. A. (2004). A comparison of error metrics and constraints for multiple endmember spectral mixture analysis and spectral angle mapper. *Remote Sensing of Environment*, 93(3), 359–367.
- Dennison, P. E., & Roberts, D. A. (2003). Endmember selection for multiple endmember spectral mixture analysis using endmember average RMSE. *Remote Sensing of Environment*, 87(2–3), 123–135.
- Göçken, M., Özçalıcı, M., Boru, A., & Dosdoğru, A. T. (2016). Integrating metaheuristics and artificial neural networks for improved stock price prediction. *Expert Systems with Applications*, 44, 320–331.
- Hafezi, R., Shahrabi, J., & Hadavandi, E. (2015). A bat-neural network multi-agent system (BNNMAS) for stock price prediction: Case study of DAX stock price. *Applied Soft Computing*, 29, 196–210.
- Hegazy, O., Soliman, O. S., & Salam, M. A. (2014). A machine learning model for stock mark prediction. arXiv preprint [arXiv:1402.7351](https://arxiv.org/abs/1402.7351).
- Hegazy, O., Soliman, O. S., & Salam, M. A. (2015). Comparative study between FPA, BA, MCS, ABC, and PSO algorithms in training and optimizing of LS-SVM for stock market prediction. *International Journal of Advanced Computer Research*, 5, 2277–7970.
- Henrique, B. M., Sobreiro, V. A., & Kimura, H. (2018). Stock price prediction using support vector regression on daily and up to the minute prices. *The Journal of Finance and Data Science*, 4(3), 183–201.
- Hernández-Álvarez, M., Hernández, E. A. T., & Yoo, S. G. (2019). Stock market data prediction using machine learning techniques. In: *International conference on information technology & systems*, 2019. Springer, pp. 539–547.
- Karazmodeh, M., Nasiri, S., & Hashemi, S. M. (2013). Stock price forecasting using support vector machines and improved particle swarm optimization. *Journal of Automation and Control Engineering*, 1(2), 173–176.
- Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. In: *Proceedings of ICNN'95-international conference on neural networks*, Vol. 4: IEEE, pp. 1942–1948.
- Li, X., Xie, H., Chen, L., Wang, J., & Deng, X. (2014). News impact on stock price return via sentiment analysis. *Knowledge-Based Systems*, 69, 14–23.
- Mokhtari, S., Yen, K. K., & Liu, J. (2021). Effectiveness of artificial intelligence in stock market prediction based on machine learning. arXiv preprint [arXiv:2107.01031](https://arxiv.org/abs/2107.01031).
- Oh, K. J., & Kim, K.-J. (2002). Analyzing stock market tick data using piecewise nonlinear model. *Expert Systems with Applications*, 22(3), 249–255.
- Panigrahi, S., & Behera, H. (2013). Effect of normalization techniques on univariate time series forecasting using evolutionary higher order neural network. *International Journal of Engineering and Advanced Technology*, 3(2), 280–285.
- Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2015). Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques. *Expert Systems with Applications*, 42(1), 259–268.
- Reddy, V. K. S. (2018). Stock market prediction using machine learning. *International Research Journal of Engineering and Technology (IRJET)*, 5(10), 1033–1035.
- Sable, R., Goel, S., & Chatterjee, P. (2019). Empirical study on stock market prediction machine learning. In: *2019 International conference on advances in computing, communication and control (ICAC3)*, 2019: IEEE, pp. 1–5.
- Tao, X., Renmu, H., Peng, W., & Dongjie, X. (2004). Input dimension reduction for load forecasting based on support vector machines. In: *Proceedings of 2004 IEEE international conference on electric utility deregulation, restructuring and power technologies*, Vol. 2. IEEE, pp. 510–514.
- Ullah, A., Imtiaz, F., Ihsan, M. U. M., Alam, M., Rabiul, G., & Majumdar, M. (2021). Combining machine learning classifiers for stock trading with effective feature extraction. arXiv preprint [arXiv:2107.13148](https://arxiv.org/abs/2107.13148).

- Vala, T. M., Rajput, V. N., Geem, Z. W., Pandya, K. S., & Vora, S. C. (2021). Revisiting the performance of evolutionary algorithms. *Expert Systems with Applications*, *175*, 114819.
- Vijh, M., Chandola, D., Tikkiwal, V. A., & Kumar, A. (2020). Stock closing price prediction using machine learning techniques. *Procedia Computer Science*, *167*, 599–606.
- Yahoo Finance Website: <http://in.finance.yahoo.com>
- Yang, J., Zhao, C., Yu, H., & Chen, H. (2020). Use GBDT to predict the stock market. *Procedia Computer Science*, *174*, 161–171.
- Yu, L., Pan, Y., & Wu, Y. (2009). Research on data normalization methods in multi-attribute evaluation. In: *2009 International conference on computational intelligence and software engineering*, pp. 1–5.

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.