A Comparison between Regression, Artificial Neural Networks and Support Vector Machines for Predicting Stock Market Index

Alaa F. Sheta  
Computers and Systems Department  
Electronics Research Institute Giza, Egypt

Sara Elsir M. Ahmed  
Computer Science Department  
University of Science and Technology Khartoum, Sudan

Hossam Faris  
Business Information Tech. Dept.  
The University of Jordan Amman, Jordan

Abstract—Obtaining accurate prediction of stock index significantly helps decision maker to take correct actions to develop a better economy. The inability to predict fluctuation of the stock market might cause serious profit loss. The challenge is that we always deal with dynamic market which is influenced by many factors. They include political, financial and reserve occasions. Thus, stable, robust and adaptive approaches which can provide models have the capability to accurately predict stock index are urgently needed. In this paper; we explore the use of Artificial Neural Networks (ANNs) and Support Vector Machines (SVM) to build prediction models for the S&P 500 stock index. We will also show how traditional models such as multiple linear regression (MLR) behave in this case. The developed models will be evaluated and compared based on a number of evaluation criteria.

Index Terms—Stock Market Prediction; S&P 500; Regression; Artificial Neural Networks; Support Vector Machines.

I. INTRODUCTION

Understanding the nature of the relationships between financial markets and the country economy is one of the major components for any financial decision making system [1]–[3]. In the past few decades, stock market prediction became one of the major fields of research due to its wide domain of financial applications. Stock market research field was developed to be dynamic, non-linear, complicated, non-parametric, and chaotic in nature [4]. Much research focuses on improving the quality of index prediction using many traditional and innovative techniques. It was found that significant profit can be achieved even with slight improvement in the prediction since the volume of trading in stock markets is always huge. Thus, financial time series forecasting was explored heavenly in the past. They have shown many characteristics which made them hard to forecast due to the need for traditional statistical method to solve the parameter estimation problems. According to the research developed in this field, we can classify the techniques used to solve the stock market prediction problems to two folds:

- **Econometric Models:** These are statistical based approaches such as linear regression, Auto-regression and Auto-regression Moving Average (ARMA) [5], [6]. There are number of assumptions need to be considered while using these models such as linearity and stationary of the the financial time-series data. Such non-realistic assumptions can degrade the quality of prediction results [7], [8].

- **Soft Computing based Models:** Soft computing is a term that covers artificial intelligence which mimic biological processes. These techniques includes Artificial Neural Networks (ANN) [9], [10], Fuzzy logic (FL) [11], Support Vector Machines (SVM) [12], particle swarm optimization (PSO) [13] and many others.

ANNs known to be one of the successfully developed methods which was widely used in solving many prediction problem in diversity of applications [14]–[18]. ANNs was used to solve variety of problems in financial time series forecasting. For example, prediction of stock price movement was explored in [19]. Authors provided two models for the daily Istanbul Stock Exchange (ISE) National 100 Index using ANN and SVM. Another type of ANN, the radial basis function (RBF) neural network was used to forecast the stock index of the Shanghai Stock Exchange [20]. In [21], ANNs were trained with stock data from NASDAQ, DJIA and STI index. The reported results indicated that augmented ANN models with trading volumes can improve forecasting performance in both medium-and long-term horizons. A comparison between SVM and Backpropagation (BP) ANN in forecasting six major Asian stock markets was reported in [22]. Other soft computing techniques such as Fuzzy Logic (FL) have been used to solve many stock market forecasting problems [23], [24].

Evolutionary computation was also explored to solve the prediction problem for the S&P 500 stock index. Genetic Algorithms (GAs) was used to simultaneously optimize all of a Radial Basis Function (RBF) network parameters such that an efficient time-series is designed and used for business forecasting applications [25]. In [26], author provided a new prediction model for the S&P 500 using Multigene Symbolic Regression Genetic Programming (GP). Multigene GP shows more robust results especially in the validation/testing case than ANN.
In this paper, we present a comparison between traditional regression model, the ANN model and the SVM model for predicting the S&P 500 stock index. This paper is structured as follows. Section II gives a brief idea about the S&P 500 Stock Index in the USA. In Section III, we provide an introduction to linear regression models. A short introduction to ANN and SVM is provided in Section IV and Section V, respectively. The adopted evaluation methods are presented in Section VI. In Section VII, we describe the characteristics of the data set used in this study. We also provide the experimental setup and results produced in this research.

II. S&P 500 Stock Index

The S&P 500, or the Standard & Poor’s 500, is an American stock market index. The S&P 500 presented its first stock index in the year 1923. The S&P 500 index with its current form became active on March 4, 1957. The index can be estimated in real time. It is mainly used to measure the stock prices levels. It is computed according to the market capitalization of 500 large companies. These companies are having stock in the The New York Stock Exchange (NYSE) or NASDAQ. The S&P 500 index is computed by S&P Dow Jones Indices. In the past, there were a growing interest on measuring, analyzing and predicting the behavior of the S&P 500 stock index [27]–[29]. John Bogle, Vanguard’s founder and former CEO, who started the first S&P index fund in 1975 stated that:

“The rise in the S&P 500 is a virtual twin to the rise in the total U.S. stock market, so of course investors, and especially index fund investors, who received their fair share of those returns, feel wealthier.”

In order to compute the price of the S&P 500 Index, we have to compute the sum of market capitalization of all the 500 stocks and divide it by a factor, which is defined as the Divisor (D). The formula to calculate the S&P 500 Index value is given as:

\[
\text{Index Level} = \frac{\sum P_i \times S_i}{D}
\]

\(P\) is the price of each stock in the index and \(S\) is the number of shares publicly available for each stock.

III. Regression Analysis

Regression analysis have been used effectively to answer many question in the way we handle system modeling and advance associations between problem variables. It is important to develop such a relationships between variables in many cases such as predicting stock market [13], [14], [30], [31]. It is important to understand how stock index move over time.

A. Single Linear Regression

In order to understand how linear regression works, assume we have \(n\) pairs of observations data set \(\{x_i, y_j\}_{i=1, \ldots, n}\) as given in Figure 1. Our objective is to develop a simple relationship between the two variables \(x\) (i.e. input variable) and \(y\) (i.e output variable) so that we can develop a line equation (see Equation 1).

\[
y = ax + b
\]  

where \(a\) is a constant (i.e. bias) and \(b\) is the slope of the line. It is more likely that the straight line will not pass by all the points in the graph. Thus, Equation 1 shall be re-written as follows:

\[
y = a + bx + \epsilon
\]  

where \(\epsilon\) represents the error difference between the values of \(x_i\) and \(y_i\) at any sample \(i\). Thus, to find the best line that produce the most accurate relationship between \(x\) and \(y\). We have to formulate the problem as an optimization problem such that we can search and find the best values of the parameters (i.e. \(\hat{a}\) and \(\hat{b}\)). In this case, we need to solve an error minimization problem. To minimize the sum of the error over the whole data set. We need to minimize the function \(L\) given in Equation 3.

\[
L = \sum_{i=1}^{n} \epsilon_i^2 = \sum_{i=1}^{n} (y_i - a - bx_i)^2
\]  

To find the optimal values for \(\hat{a}\) and \(\hat{b}\) we have to differentiate \(L\) with respect to \(a\) and \(b\).

\[
\frac{\partial L}{\partial a} = -2 \sum_{i=1}^{n} (y_i - \hat{a} - \hat{b}x_i) = 0
\]

\[
\frac{\partial L}{\partial b} = -2 \sum_{i=1}^{n} (y_i - \hat{a} - \hat{b}x_i)x_i = 0
\]

By simplification of Equations 4, we get to the following two equations:

\[
n \hat{a} + \sum_{i=1}^{n} x_i \hat{b} = \sum_{i=1}^{n} y_i
\]

\[
\sum_{i=1}^{n} x_i \hat{a} + \sum_{i=1}^{n} x_i^2 \hat{b} = \sum_{i=1}^{n} x_i y_i
\]
Equations 5 is called least square (LS) normal equations. The solution of these normal equations produce the least square estimate for $a$ and $b$.

B. Multiple Linear Regression

The simple linear model Equation 2 can be expanded to a multivariate system of equations as follows:

$$y = a_1x_1 + a_2x_2 + \cdots + a_nx_n$$  \hfill (6)

where $x_j$ is the $j^{th}$ independent variable. In this case, we need to use LS estimation to compute the optimal values for the parameters $a_1, \ldots, a_n$. Thus, we have to minimize the optimization function $L$, which in this case can be presented as:

$$L = \sum_{i=1}^{n} \epsilon_i^2 = \sum_{i=1}^{n} (y_i - \hat{a}_1x_1 - \hat{a}_2x_2 - \cdots - \hat{a}_nx_n)^2$$  \hfill (7)

To get the optimal values of the parameters $\hat{a}_1, \ldots, \hat{a}_n$, we have to compute the differentiation for the functions:

$$\frac{\partial L}{\partial a_1} = \frac{\partial L}{\partial a_2} = \cdots = \frac{\partial L}{\partial a_n} = 0$$  \hfill (8)

Solving the set of Equations 8, we can produce the optimal values of the model parameters and solve the multiple regression problem. This solution is more likely to be biased by the available measurements. If you have large number of observations the computed estimate of the parameters shall be more robust. This technique provide poor results when the observations are small in number.

IV. ARTIFICIAL NEURAL NETWORKS

ANNs are mathematical models which were inspired from the understanding of some ideas and aspects of the biological neural systems such as the human brain. ANN may be considered as a data processing technique that maps, or relates, some type of input stream of information to an output stream of processing. Variations of ANNs can be used to perform classification, pattern recognition and predictive tasks [15], [19], [20], [22], [30].

Neural network have become very important method for stock market prediction because of their ability to deal with uncertainty and insufficient data sets which change rapidly in very short period of time. In Feedforward (FF) Multilayer Perceptron (MLP), which is one of the most common ANN systems, neurons are organized in layers. Each layer consists of a number of processing elements called neurons; each of which contains a summation function and an activation function. The summation function is given by Equation 9 and an activation function can be a type of sigmoid function as given in Equation 10.

Training examples are used as input the network via the input layer, which is connected to one, or more hidden layers. Information processing takes place in the hidden layer via the connection weights. The hidden layers are connected to an output layer with neurons most likely have linear sigmoid function. A learning algorithms such as the BP one might be used to adjust the ANN weights such that it minimize the error difference between the actual (i.e. desired) output and the ANN output [32]–[34].

$$S = \sum_{i=0}^{n} w_i x_i$$  \hfill (9)

$$\phi(S) = \frac{1}{1 + e^{-S}}$$  \hfill (10)

There are number of tuning parameters should be designated before we can use ANN to learn a problem. They include: the number of layers in the hidden layer, the type of sigmoid function for the neurons and the adopted learning algorithm.

V. SUPPORT VECTOR MACHINES

Support vector machine is a powerful supervised learning model for prediction and classification. SVM was first introduced by Vladimir Vapnik and his co-workers at AT&T Bell Laboratories [35]. The basic idea of SVM is to map the training data into higher dimensional space using a nonlinear mapping function and then perform linear regression in higher dimensional space in order to separate the data [36]. Data mapping is performed using a predetermined kernel function. Data separation is done by finding the optimal hyperplane (called the Support Vector with the maximum margin from the separated classes. Figure 2 illustrates the idea of the optimal hyperplane in SVM that separates two classes. In the left part of the figure, lines separated data but with small margins while on the right an optimal line separates the data with the maximum margins.

A. Learning Process in SVM

Training SVM can be described as follows; suppose we have a data set $\{x_i, y_i\}_{i=1,...,n}$ where the input vector $x_i \in \mathbb{R}^d$ and the actual $y_i \in \mathbb{R}$. The modeling objective of SVM is to find the linear decision function represented in the following equation:

$$f(x) \leq w, \quad \phi_i(x) > +b$$  \hfill (11)

where $w$ and $b$ are the weight vector and a constant respectively, which have to be estimated from the data set. $\phi$ is a nonlinear mapping function. This regression problem can be formulated as to minimize the following regularized risk function:

$$R(C) = \frac{C}{n} \sum_{i=1}^{n} L_C(f(x_i), y_i) + \frac{1}{2} \| w \|^2$$  \hfill (12)

where $L_C(f(x_i), y_i)$ is known as $\varepsilon$–intensive loss function and given by the following equation:

$$L_C(f(x), y) = \begin{cases} |f(x) - y| - \varepsilon & |f(x) - y| \geq \varepsilon \\ 0 & \text{otherwise} \end{cases}$$  \hfill (13)

To measure the degree of miss classification to achieve an acceptable degree of error, we use slack variables $\xi_i$ and
\[\xi_i^*\] as shown in Figure 3. This addition makes the problem presented as a constrained minimum optimization problem (See Equation 14).

\[
\begin{align*}
\text{Min. } R(w, \xi_i^*) &= \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{n} (\xi_i + \xi_i^*) \\
\text{Subject to: } &
\begin{cases}
y_i - \langle w, x_i \rangle - b \leq \epsilon + \xi_i \\
\langle w, x_i \rangle + b - y_i \leq \epsilon + \xi_i^* \\
\xi_i, \xi_i^* \geq 0
\end{cases}
\end{align*}
\]

(14)

where \(C\) is a regularized constant greater than zero. Thus it performs a balance between the training error and model flatness. \(C\) represents a penalty for prediction error that is greater than \(\epsilon\). \(\xi_i\) and \(\xi_i^*\) are slack variables that form the distance from actual values to the corresponding boundary values of \(\epsilon\). The objective of SVM is to minimize \(\xi_i, \xi_i^*\) and \(w^2\).

The above optimization with constraint can be converted by means of Lagrangian multipliers to a quadratic programming problem. Therefore, the form of the solution can be given by the following equation:

\[
f(x) = \sum_{i=1}^{n} (\alpha_i - \alpha_i^*) K(x_i, x) + b
\]

(16)

\(K(\cdot)\) is the kernel function and its values is an inner product of two vectors \(x_i\) and \(x_j\) in the feature space \(\phi(x_i)\) and \(\phi(x_j)\) and satisfies the Mercer’s condition. Therefore,

\[
K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j)
\]

(18)

Some of the most common kernel functions used in the literature are shown in Table I. In general, SVMs have many advantages over classical classification approaches like artificial neural networks, decision trees and others. These advantages include: good performance in high dimensional spaces; and the support vectors rely on a small subset of the training data which gives SVM a great computational advantage.

<table>
<thead>
<tr>
<th>Polynomial Kernel</th>
<th>(K(x_i, x_j) = (x_i \cdot x_j + 1)^d)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hyperbolic Tangent Kernel</td>
<td>(K(x_i, x_j) = \tanh(c_1 (x_i \cdot x_j) + c_2))</td>
</tr>
<tr>
<td>Radial Basis Kernel</td>
<td>(p: K(x_i, x_j) = \exp(x_i \cdot x_j/2p^2))</td>
</tr>
</tbody>
</table>

VI. EVALUATION CRITERION

In order to assess the performance of the developed stock market prediction models, a number of evaluation criteria will be used to evaluate these models. These criteria are applied to measure how close the real values to the values predicted using the developed models. They include Mean Absolute Error (MAE), Root Mean Square Error (RMSE) and correlation coefficient \(R\). They are given in Equations 19, 20 and 21, respectively.
\[ MAE = \frac{1}{n} \sum_{i=1}^{n} |(y_i - \hat{y}_i)| \]  
(19)

\[ RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2} \]  
(20)

\[ R = \frac{\sum_{i=1}^{n} (y_i - \bar{y})(\hat{y}_i - \bar{y})}{\sqrt{\sum_{i=1}^{n} (y_i - \bar{y})^2 \sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2}} \]  
(21)

where \( y \) is actual stock index values, \( \hat{y} \) is the estimated values using the proposed technique, \( n \) is the total number of measurements.

VII. EXPERIMENTAL RESULTS

A. S&P 500 Data Set

In this work, we use 27 potential financial and economic variables that impact the stock movement. The main consideration for selecting the potential variables is whether they have significant influence on the direction of (S&P 500) index in the next week. While some of these features were used in previous studies [30]. The list, the description, and the sources of the potential features are given in Table III show the 27 features of data set.

The categories of these features include: S&P 500 index return in three previous days SPY(t-1), SPY(t-2), SPY(t-3); Financial and economic indicators (Oil, Gold, CTB3M, AAA); The return of the five biggest companies in S&P 500 (XOM, GE, MSFT, PG, JNJ); Exchange rate between USD and three other currencies (USD-Y, USD-GBP, USD-CAD); The return of the four world major indices (HIS, FCHI, FTSE, GDAXI); and S&P 500 trading volume (V).

S&P 500 stock market data set used in our case consists of 27 features and 1192 days of data, which cover five-year period starting 7 December 2009 to 2 September 2014. We sampled the data on a weekly basis such that only 143 samples were used in our experiments. The S&P 500 data were split into 100 samples as training set and data for 43 samples as testing set.

B. Multiple Regression Model

The regression model shall have the following equation system.

\[ y = a_0 + \sum_{i=1}^{27} a_i x_i \]  
(22)

The values of the parameters \( a \)'s shall be estimated using LS estimation to produce the optimal values of the parameters \( \hat{a} \)'s. The produced linear regression model can be presented as given in Table II. The actual and Estimated S&P 500 index values based the MLR in both training and testing cases are shown in Figure 4 and Figure 5. The scattered plot of the actual and predicted responses is shown in Figure 6.
\[ \hat{y} = -0.0234 \times x_1 + 0.13 \times x_2 + 0.021 \times x_3 + 0.021 \times x_4 - 0.021 \times x_5 \\
- 10.303 \times x_6 + 6.0301 \times x_7 + 0.7738 \times x_8 + 0.2779 \times x_9 - 0.4391 \times x_{10} \\
- 0.27754 \times x_{11} + 0.12733 \times x_{12} - 0.058638 \times x_{13} + 13.646 \times x_{14} + 9.5224 \times x_{15} \\
- 0.0003 \times x_{16} + 0.24856 \times x_{17} - 0.0016 \times x_{18} + 0.0 \times x_{19} - 2.334 \times 10^{-9} \times x_{20} \\
+ 0.16257 \times x_{21} + 0.63767 \times x_{22} - 0.14301 \times x_{23} + 0.08 \times x_{24} + 0.074 \times x_{25} \\
- 0.0002 \times x_{26} + 0.026301 \times x_{27} + 6.9312 \]

(23)

**C. Developed ANN Model**

The proposed architecture of the MLP Network consists of three layers with single hidden layer. Thus input layer of our neural network model has 27 input nodes while the output layer consists of only one node that gives the predicted next week value. Empirically, we found that 20 neurons in the hidden layer achieved the best performance. The BP algorithm is used to train the MLP and update its weight. Table IV shows the settings used for MLP. Figure 7 and Figure 8 show the actual and predicted stock prices for training and testing cases of the developed ANN. The scattered plot for the developed ANN model is shown in Figure 9.

**D. Developed SVM Model**

SVM with an RBF kernel is used to develop the S&P 500 index model. The RBF kernel has many advantages such as the ability to map non-linearly the training data and the ease of implementation [37]–[39]. The values of the parameters \( C \) and \( \sigma \) have high influence on the accuracy of the SVM model. Therefore, we used grid search to obtain these values. It was found that the best performance can be obtained with \( C = 100 \) and \( \sigma = 0.01 \). Figure 10 and Figure 11 show the actual and predicted stock prices for training and testing cases of the developed SVM model. The scattered plot for the developed SVM model is shown in Figure 12.
Fig. 7. ANN: Actual and Estimated S&P 500 Index values in Training Case

Fig. 10. SVM: Actual and Estimated S&P 500 Index values in Training Case

Fig. 8. ANN: Actual and Estimated S&P 500 Index values in Testing Case

Fig. 11. SVM: Actual and Estimated S&P 500 Index values in Testing Case

Fig. 9. ANN Scattered Plot

Fig. 12. SVM Scattered Plot
E. Comments on the Results

The calculated evaluation criterion of the regression, MLP and SVM models for training and testing cases are shown in Table V. Based on these results it can be noticed that SVM outperformed the MLP and MLR models in both training and testing cases. SVMs have many advantages such as using various kernels which allows the algorithm to suits many classification problems. SVM are more likely to avoid the problem of falling into local minimum.

VIII. Conclusions and Future Work

In this paper, we explored the use MLP and SVM to develop models for prediction the S&P 500 stock market index. A 27 potential financial and economic variables which impact the stock movement were adopted to build a relationship between the stock index and these variables. The basis for choosing these variables was based on their substantial impact on the course of S&P 500 index. The data set was sampled on a weekly bases. The developed SVM model with RBF kernel model provided good prediction capabilities with respect to the regression and ANN models. The results were validated using number of evaluation criteria. Future research shall focus on exploring other soft computing techniques to solve the stock market prediction problems.

REFERENCES


### TABLE V
**EVALUATION CRITERIA FOR THE DEVELOPED MODELS**

<table>
<thead>
<tr>
<th></th>
<th>Regression</th>
<th>ANN</th>
<th>SVM-RBF</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Training</td>
<td>Testing</td>
<td>Training</td>
</tr>
<tr>
<td>Correlation coefficient</td>
<td>0.998</td>
<td>0.995</td>
<td>0.999</td>
</tr>
<tr>
<td>Mean absolute error</td>
<td>0.373</td>
<td>4.961</td>
<td>0.433</td>
</tr>
<tr>
<td>Root mean squared error</td>
<td>0.482</td>
<td>5.749</td>
<td>0.529</td>
</tr>
<tr>
<td>Relative absolute error</td>
<td>4.429%</td>
<td>11.838%</td>
<td>4.683%</td>
</tr>
<tr>
<td>Root relative squared error</td>
<td>4.982%</td>
<td>12.313%</td>
<td>5.040%</td>
</tr>
</tbody>
</table>


