Embedding Dimension as Input Dimension of Artificial Neural Network: A Study on Stock Prices Time Series

Amir Hosein Ghaderi¹, *, Bahar Barani², Hoda Jalalkamali¹

¹Neuroscience Lab., University of Tabriz, Tabriz, Iran
²Medical School, University of Kansas, Kansas City, USA

Abstract

Recently artificial neural networks (ANNs) have become crucial for the analysis of several phenomena in the world. Stock market prices is a nonlinear phenomenon that several studies are accomplished in this context using ANN. Determine the input dimension of network is a basic problem in application of ANN to prediction of stock prices. On the other hand, stock market behaves as a chaotic system with nonlinear and deterministic manner. Therefore chaos theory can be helpful to determine the input dimension of network. In this study a multilayer perceptron with Backpropagation learning algorithm is used for forecasting of stock prices in Tehran Stock Exchange (TSE). Prediction accuracy of ANN with various input dimensions is compared and the best result is achieved when the input dimension is equal to Takens embedding dimension. It is concluded that selection of Takens embedding as ANN input dimension can be led to very accurate predictions and best results in TSE.

Keywords

Neural Network, Chaos Theory, Multilayer Perceptron, Embedding Dimension, Stock Market, Forecasting

1. Introduction

Overall, there are several nonlinear phenomena across the world. The change in climate, the activity of the mammalian brain, the social behavior of humans, the probability of an earthquake occurring, and the fluctuating prices in the stock market are just a few examples of nonlinearity in the world. Recently, several methods have been suggested for the prediction of these nonlinear phenomena and therefore, the analysis and prediction of these has become a vital tool to the advancement of science. Chaotic and fractal analysis, time series analysis, the genetic algorithm, fuzzy logic, and neural networks are continually being extended for a nonlinear world analysis.

Upon a first glance at the fluctuations of the prices of stock, it seems that there is no pattern or order to the fluctuations. However, in numerous studies, it has been shown that these fluctuations do in fact follow a specific order. Some of the studies show that this behavior is chaotic and deterministic, therefore making the prediction of these fluctuations possible [1–3]. Recently, nonlinear analysis has become crucial for the analysis of price fluctuations in the stock market. SVR [4–6], fuzzy logic, genetic algorithm [8, 9] and neuro fuzzy [10, 11] are some of these nonlinear approaches and have shown that fluctuations of the prices of stock can be predicted.

One of the well-known approaches to the prediction of time series changes is the artificial neural network method [12–14]. Artificial neural network (ANN) is a reliable model for classification and prediction in many areas such as signal processing, aerology, motor control and thermodynamics and so on. Many studies suggest this model for stock analysis. In decade of 1990 ANN was used for price forecasting and showed its superiority over the linear models such as ARIMA.
price increments with a success rate slightly above 50% but performance than other methods in learning ability and pattern dimension. In most studies, there is no specific rule in perceptrons (MLPs) and back-propagation learning algorithm has been used repeatedly as network learning. L. Wu proposed a bacterial chemotaxis optimization (IBCO), which is then integrated into the BP algorithm for stock price forecasting in emerging markets, such as China. Other studies use recurrent networks rather than feedforward networks [25–27]. In recent years, several studies have been accomplished to improve network learning. The genetic algorithm has been used repeatedly as network learning algorithm in stock price applications [28–30]. Y. Zhang and L. Wu proposed a bacterial chemotaxis optimization (IBCO), which is then integrated into the BP algorithm for stock market prediction. They found that, this method show better performance than other methods in learning ability and generalization. W. Shen et al. used a radial basis function neural network (RBFNN) to train data and forecast the stock indices of the Shanghai Stock Exchange. They compared forecasting result of radial basis function optimized by several methods. Finally they found that RBF optimized by AFSA is an easy-to-use algorithm with considerable accuracy. A recurrent neural network (RNN) based on Artificial Bee Colony (ABC) algorithm was used by T.J. Hsieh et al. for stock markets forecasting. They used Artificial Bee Colony algorithm (ABC) to optimize the RNN weights. C.J. Lua and J.Y. Wu employed a cerebellar model articulation controller neural network (CAMC NN) for stock index forecasting to improve the forecasting performance. The forecasting results were compared with a support vector regression (SVR) and a back-propagation neural network (BPNN) and experimental results show that performance of the proposed CMAC NN scheme was superior to the SVR and BPNN models. An integrated approach based on genetic fuzzy systems (GFS) and ANN for constructing a stock price forecasting expert system is presented by E. Hadavandi et al. and they show that the proposed method is a suitable approach for stock price forecasting.

But, one of the problems that arise with the use of the supervisor learning ANN in time series analysis, such as the fluctuations in stock prices, is the specification of the input pattern dimension. In most studies, there is no specific rule in place for the specification of the input pattern dimension and the selection of this dimension has been accomplished experimentally [23, 24, 26, 28]. F. Castiglione stated that there is no way to determine input and hidden units and N. Khoa et al. represented that network inputs, were decided based on experimental results.

On the other hand, in chaotic analysis the correlation dimension and the embedding dimension of a time series signal can be determined. If it is assumed that the price fluctuations in the stock market are a chaotic time series signal [1–3], then the chaotic analysis can be used to determine the correlation dimension and the embedding dimension of stock market price in time series. Z. Shang et al. have demonstrated that the employment of the embedding dimension as the input pattern dimension can lead to an appropriate result in a feedforward neural network method. They have shown that, in the chaotic maps, the input pattern dimension in the neural network must be equal to the embedding dimension. However, the embedding dimension can be found through the use of two different methods: the Grassberger method and the Takens method [37, 38] and these two embedding dimensions have very different values in the chaotic analysis of the fluctuating prices of the stock market. O.J Kyong and K. Kyoung-jae show that embedding dimension can be used for determination of time lag size in input variable of ANN and they use Grassberger embedding dimension in stock price time series as input dimension for ANN learning.

In this work the first goal is to obtain a highly accurate method for the forecasting of the price of Iranian stock within short interval of time by a multi-layer perceptron ANN. The second objective mainly focuses on the application of chaotic analysis in order to determine the ANN input dimension. And, in more detail, the second goal is to utilize a new approach in order to determine whether Grassberger embedding or Takens embedding is most useful in the determination of ANN input dimension in a given real-life application, such as stock market price fluctuations.

This paper is organized as follows; the chaotic analysis on time series and embedding dimension is described in section 2, the multilayer perceptrons and back-propagation learning algorithm is displayed in section 3, section 4 is devoted to the application of the embedding dimension for the determination of the neural network input dimension, and finally sections 5 and 6 are dedicated to the discussion and conclusion.

2. The Chaotic Dimensions

2.1. Correlation Dimension

Dynamical chaotic systems have a strange attractor in phase space. Strange attractor characterized by a correlation
3. Multilayer Perceptron and Backpropagation Algorithm

In a feedforward ANN, the signals, fellow directly from input neurons to output layer. A multilayer feedforward neural network is constructed when some hidden layers are located between input and output layers. Layered architectures are those in which the set of computing units (neurons) in input layer only connected with neurons in first hidden layer and so on. Multilayer perceptrons or MLPs is termed to feedforward multilayer networks with sigmoid nonlinearities. In the MLP, the output of each neuron is:

\[ y_i = \text{sig} \left( \sum_{j=0}^{n} \omega_{ij} x_j \right), \]

where \( x_i \) is input signal, \( \omega \) is synaptic weight of neuron \( k \) and \( \text{sig} \) is a sigmoid function.

4. Application of Chaos Theory in ANN Input Dimension

The analysis is accomplished on the stocks of two corporations in Tehran Stock Exchange (TSE), Pars oil and Iran Transfo. All data are obtained from TSE at http://www.irbourse.com/en/stats.aspx. The data is related to daily exchange stock price of these corporations from March 1995 to May 2013 and used for training (90%) and testing (10%). For price forecasting, the data is divided into time lag windows and each window can be used for forecasting the next day price. This is illustrated schematically in Fig. 1.

A multilayer perceptron with two hidden layers is used for forecasting. There are 20 neurons in each hidden layer and one neuron in output layer. The number of input neurons is related to time lag window. The architecture of the ANN is shown in Fig. 2. The trainrp algorithm is used for network training in MATLAB. trainrp is a network training function that updates weight and bias values according to the resilient back-propagation algorithm. Additional data about ANN is represented in Tab. 1.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0.05</td>
<td>tansig</td>
<td>logsig</td>
<td>trainrp</td>
<td>50000</td>
</tr>
</tbody>
</table>

The correlation dimension and embedding dimension are calculated for each time series. These values are shown in Tab. 2. In this table embedding dimension is calculated in two ways. Takens embedding dimension is achieved by the first integer above \( 2D_{cor}+1 \) where \( D_{cor} \) is correlation dimension and Grassberger embedding dimension is obtained according to Fig. 3, 4. The performance of each input dimension is shown in Tab. 4, 3. The best performances of ANN forecasting with three neurons in input layer are shown in Fig. 5, 7 for two time series and the regressions of predicted prices are shown in Fig. 6, 8 for best and worst results.
5. Discussion

The neural network input data is a critical matter in ANN training. The results show that various input size of ANN can be led to different accuracy of prediction. The importance of patterns input dimension in accuracy of prediction is demonstrated in Tab. 4, 3 and Fig. 6, 8. Fig. 8 indicates that there is a very little difference between input sizes of training patterns for best and worst predictions and Fig. 6 shows that selection of wrong input dimension can be led to false predictions. Therefore it is essential to determine input dimension of ANN, correctly.

The correlation dimension and the embedding dimension determine the properties of the phase space and attractor dimensions of the time series. Embedding dimension is equal to the number of independent variables of the system. On the other hand, the number of inputs of ANN and independent variables are identical, it seems that embedding dimension could be used for determining the input dimension of ANN.

In Oh and Kim study the Grassberger embedding dimension is used to determine the input dimension of ANN for stocks prices forecasting but they did not test performance of other input dimensions. Therefore in this study all input dimensions are tested and the results show that best prediction is achieved when the Takens embedding is selected as input dimension. Table 2 indicates that there is a large difference between Grassberger embedding and Takens embedding. In the Iranian stock market, Grassberger embedding is approximately 20, while Takens embedding is approximately 3. Choosing Takens’ embedding dimension leads to reliable results which are shown in Fig. 5, 7. These results suggest a new way to determine of the input size in ANN training.

Table 3. The performance results of Iran Transfo for various input dimensions.

<table>
<thead>
<tr>
<th>Input size</th>
<th>No. training patterns</th>
<th>No. testing patterns</th>
<th>R value</th>
<th>MSE</th>
<th>MAE</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>793</td>
<td>88</td>
<td>0.75994</td>
<td>0.0000695</td>
<td>0.0066</td>
</tr>
<tr>
<td>3</td>
<td>528</td>
<td>58</td>
<td>0.97310</td>
<td>0.0000028</td>
<td>0.0014</td>
</tr>
<tr>
<td>4</td>
<td>396</td>
<td>44</td>
<td>0.97196</td>
<td>0.0000078</td>
<td>0.0024</td>
</tr>
<tr>
<td>5</td>
<td>316</td>
<td>35</td>
<td>0.94881</td>
<td>0.0000430</td>
<td>0.0060</td>
</tr>
<tr>
<td>6</td>
<td>263</td>
<td>29</td>
<td>0.97018</td>
<td>0.0000776</td>
<td>0.0084</td>
</tr>
<tr>
<td>7</td>
<td>225</td>
<td>25</td>
<td>0.94958</td>
<td>0.0001269</td>
<td>0.0108</td>
</tr>
<tr>
<td>8</td>
<td>197</td>
<td>21</td>
<td>0.93333</td>
<td>0.0000087</td>
<td>0.0024</td>
</tr>
<tr>
<td>9</td>
<td>175</td>
<td>19</td>
<td>0.95837</td>
<td>0.0002840</td>
<td>0.0164</td>
</tr>
</tbody>
</table>
### Table 4. The performance results of Pars Oil for various input dimensions.

<table>
<thead>
<tr>
<th>Input size</th>
<th>No. training patterns</th>
<th>No. testing patterns</th>
<th>R value</th>
<th>MSE</th>
<th>MAE</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>1354</td>
<td>151</td>
<td>0.95822</td>
<td>0.0001177</td>
<td>0.0027</td>
</tr>
<tr>
<td>3</td>
<td>906</td>
<td>101</td>
<td>0.99654</td>
<td>0.0000063</td>
<td>0.0018</td>
</tr>
<tr>
<td>4</td>
<td>679</td>
<td>76</td>
<td>0.84597</td>
<td>0.0002768</td>
<td>0.0040</td>
</tr>
<tr>
<td>5</td>
<td>543</td>
<td>60</td>
<td>0.99491</td>
<td>0.000082</td>
<td>0.0021</td>
</tr>
<tr>
<td>6</td>
<td>452</td>
<td>50</td>
<td>0.99552</td>
<td>0.000089</td>
<td>0.0025</td>
</tr>
<tr>
<td>7</td>
<td>387</td>
<td>43</td>
<td>0.98076</td>
<td>0.0000352</td>
<td>0.0043</td>
</tr>
<tr>
<td>8</td>
<td>339</td>
<td>38</td>
<td>0.99612</td>
<td>0.0000066</td>
<td>0.0018</td>
</tr>
<tr>
<td>9</td>
<td>301</td>
<td>34</td>
<td>0.99351</td>
<td>0.0000105</td>
<td>0.0024</td>
</tr>
<tr>
<td>10</td>
<td>271</td>
<td>30</td>
<td>0.99484</td>
<td>0.000082</td>
<td>0.0023</td>
</tr>
<tr>
<td>11</td>
<td>246</td>
<td>27</td>
<td>0.97164</td>
<td>0.0000452</td>
<td>0.0046</td>
</tr>
<tr>
<td>12</td>
<td>225</td>
<td>25</td>
<td>0.99447</td>
<td>0.0000109</td>
<td>0.0028</td>
</tr>
<tr>
<td>13</td>
<td>208</td>
<td>23</td>
<td>0.98709</td>
<td>0.0000212</td>
<td>0.0034</td>
</tr>
<tr>
<td>14</td>
<td>193</td>
<td>22</td>
<td>0.99361</td>
<td>0.0000138</td>
<td>0.0030</td>
</tr>
<tr>
<td>15</td>
<td>180</td>
<td>20</td>
<td>0.99414</td>
<td>0.0000077</td>
<td>0.0022</td>
</tr>
<tr>
<td>16</td>
<td>169</td>
<td>19</td>
<td>0.98822</td>
<td>0.0000216</td>
<td>0.0037</td>
</tr>
<tr>
<td>17</td>
<td>159</td>
<td>17</td>
<td>0.99374</td>
<td>0.0000175</td>
<td>0.0031</td>
</tr>
<tr>
<td>18</td>
<td>150</td>
<td>17</td>
<td>0.99438</td>
<td>0.0000165</td>
<td>0.0033</td>
</tr>
<tr>
<td>19</td>
<td>142</td>
<td>16</td>
<td>0.99563</td>
<td>0.0000069</td>
<td>0.0020</td>
</tr>
<tr>
<td>20</td>
<td>135</td>
<td>15</td>
<td>0.99140</td>
<td>0.0000164</td>
<td>0.0032</td>
</tr>
<tr>
<td>21</td>
<td>128</td>
<td>14</td>
<td>0.98468</td>
<td>0.0000281</td>
<td>0.0043</td>
</tr>
</tbody>
</table>

**Figure 3.** Grassberger embedding for Iran Transfo time series.
Figure 4. Grassberger embedding for Pars Oil time series.

Figure 5. Comparison between predicted and actual data for Iran Transfo with Takens embedding input dimension.

Figure 6. The best and worst regressions of ANN for Iran Transfo predicted price. a) Best result: 3 neurons in input layer. 
b) Worst result: 15 neurons in input layer.
6. Conclusion

The results show that best performance of stock price forecasting is achieved when ANN input dimension is selected equal to Takens’ embedding dimension. This result is satisfied for both Iran Transfo (MSE=0.0000028) and Pars Oil (MSE=0.0000063) corporations as indicated in Tab 4, 3. Finally we found that good predicts could be performed in TSE using an ANN when input dimension is chosen based on the Takens’ embedding dimension. The reliability of these predictions is shown in Fig. 5, 7. Therefore, we suggest that Takens’ embedding dimension could be used in other time series predications with ANN.

References


