IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 4 Issue 8, August 2017
ISSN (Online) 2348 – 7968 | Impact Factor (2016) – 5.264
www.ijiset.com

Neural Networks for Financial Time Series

Odeta Shkreli

Department of Informatics, Faculty of Natural Sciences, University of Tirana, Tirana, Albania

Abstract
This research presents the analysis of neural network systems and how it is efficiently utilized for financial analysis and in particular for cash predictions and forecasts. It attempts to analyze the usefulness of artificial neural networks systems in financial analysis, focusing on its architecture and the way it can be implemented for analyzing financial time series. In this research we will elaborate its components and how they can be practically configured. The recommendations on improvements for better forecasts are also done on various segments as needed. In the document it is also highlighted the procedure of the approach.

Keywords: *Cash forecasting, data collection, implementation, neural network model, time series.*

1. Introduction

Analyzation of financial data is a critical and indispensable process for the evaluation of the business environment, prediction of the future financial situation and make accurate strategic decision. Implementation of information technology, especially of business intelligence have shown to have significantly improved the way the financial data are analyzed and increased the accuracy and the timing of the predictions as well as of the decisions. A specific area of financial analysis is cash forecasting. Cash forecasting is predicting future cash demand in terms of currency within a realistic accuracy and precision.

Business Intelligence (BI) is a wide term including a number of various techniques that can be applied for financial analysis and specifically for cash forecasting, such as OLAP tools and data mining techniques. In [1] and [2] the authors identify that a good BI system should provide these tools: end-user reporting, query and reporting, OLAP, dashboard/screen tools, data mining tools, and planning and modeling tools.

The financial data that needs to be analyzed consist mainly on numerical values of revenues and expenditures (actual, planned and forecasted), recorded at equal time intervals and are categorized as time series data. Time series databases are widely used in many applications such as market stock analysis, economic forecasts, budget analyzes, and so on.

The development of statistical methods has produced a set of data analysis techniques that are useful to confirm predefined hypothesis. Such techniques are, however, inadequate in the process of discovering new correlations and dependences between data, which on the other side, grow in quantity, dimensions and complexity. Through the use of data mining, large quantities of data can be explored and analyzed in order to discover valid, novel, potentially useful, and ultimately understandable patterns in data, through different methods and techniques. Data mining includes many tools and techniques that can be used for cash forecasting and other financial predictions. These tools can be used individually or simultaneously depending on the concrete need and analysis that needs to be performed. In terms of financial prediction, Neural Networks (NN) can be considered one of the most powerful approaches despite their complexity.

Forecasting the future situations in a business environment determines many critical business decisions like procurement of supplies and equipment, inventory control, product demand estimation, cash requirement prediction for expenses, labor cost estimation among others. These predictions are usually very difficult and some may even prove impossible [3]. To avoid difficulties and uncertainties system identification models that use historic data are used. Input data and output data pertinent to the given and specific application is collected over specific periods and then evaluated by the programmed processes. Prediction processes in these systems are capturing then processing past historic data, which provides a vision into the future. Predicted results can easily and efficiently be used to decide future actions and make strategic decisions. It is therefore vital to get predictions and forecasts on cash and liquidity accurately through the analysis of public finance. With the rapid development of computing and storing abilities of computers, many enhancements on predicting tasks in different kind of fields has been achieved one of them being neural networks. Neural networks is one of the main and effective methods which has been and can be used in this finance analysis to predict cash flows.

Cash forecasting ensures efficient and effective use of cash in a business and improves the cash management process. As discussed above, NN is one of the efficient forecasting technique used among other methods like time series analysis forecast, factor technique and expert approach. According to [4], the most frequent areas of NN

IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 4 Issue 8, August 2017
ISSN (Online) 2348 – 7968 | Impact Factor (2016) – 5.264
www.ijiset.com

applications in past 10 years are production/operations (53.5%) and finance (25.4%).

2. Neural Networks

A neural network is a massively parallel distributed processor made up of simple processing units that has a natural propensity for storing experiential knowledge and making it available for use. It resembles the brain in two respects:

1. Knowledge is acquired by the network from its environment through a learning process.
2. Interneuron connection strengths, known as synaptic weights, are used to store the acquired knowledge [5].

Neural networks are designed or structured to avail ways to solve problems with ease and with limited or no input of an expert or the need to conduct a programming procedure.

2.1 Artificial Neural Networks (ANN)

ANN arise from the will to artificially simulate the organization and the physiological functioning of human brain structures. ANN are information processing systems that try to internally simulate the biological nervous system, that are made up of a large number of neuron cells connected to each other in a complex network. The intelligence behavior emerges from the large number of interactions between the connected units called neurons. ANN consists of a number of neurons, which are distributed in hierarchical layers. The Multilayer Perceptron (MLP) model is one of the most widely used NN architecture. This network has a multilayered, feed forward, hierarchical structure that consists of input and output layers as well as one or more hidden layers. The total number of neurons, number of neurons on each layer, as well as number of layers determine the accuracy of the network model. The input layer neurons receive information, called stimuli, from the environment, the output layer neuron emit responses to the environment and the hidden layer neurons communicate only internally in the network. The connections between the neurons are called synapses. The stimuli can either excite the neuron or inhibit it. If the receiving neuron is excited by the information it receives, it will pass the information on to other neighboring neurons. If the neuron is inhibited, the input of the information is dampened and not passed on [6]. Figure 1 shows a graphical presentation of a neural network.

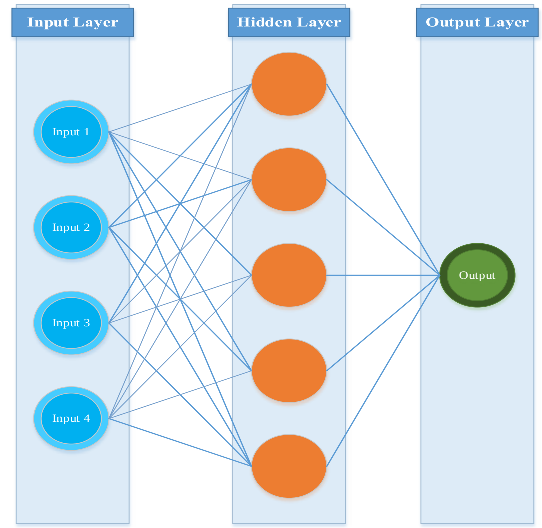

Fig. 1. Graphical presentation of the Neural Network.

Each connection between two nodes has a weight, which encapsulate the “knowledge” of the system. By processing existing cases with inputs and expected outputs, these weights would be adjusted based on differences between actual and expected outputs.

The relationship between inputs and outputs, so the function of the network, is not programmed, but is obtained from a learning process based on empirical data and can be:

Supervised learning: when we have a set of training data including typical input examples with the corresponding outputs. This way the network can learn to infer the relationship that binds them. Subsequently, the network is trained via an appropriate algorithm (usually backpropagation that is a supervised learning algorithm) that uses these data to modify the weights and other network parameters in such a way to minimize the prediction error and in case of success the network learns to recognize the relationships between input and output variables and is able to make predictions. The objective of the supervised learning is the prediction of the output values for each valid input, based only on a limited number of relationship examples. Supervised learning can help solve regression or classification problems.

Unsupervised learning: it is based on training algorithms that modify the weights of the network referring only to a set of input data. These kind of algorithms try to group the input data and identify appropriate cluster representations of the data, using mainly topologic or probabilistic methods. Unsupervised learning can be used to develop data compression techniques.

Reinforcement learning: where an appropriate algorithm is used to identify a modus operandi, from the observation of the external environment: each activity has a specific impact in the environment and the environment produces a particular reaction that guides the algorithm in the learning process. The algorithm focuses on on-line performance, which implies a balancing between exploring unknown situations and exploiting current knowledge.

3 Artificial Neural Networks Approach

The network relays on computing structures that learn by experience and predict patterns in a given data set. To implement neural networks a set of conditions should be fulfilled for a successful prediction. This includes [3]:

- Identification of the prediction objectives;
- The sample data should entail necessary information that defines the issues to be addressed;
- Proper awareness of the problem to address ensures that precise decision on developing the network can be made. It also helps to understand the underlying patterns;
- Sufficiently sized and relevant data set for preparation and testing of the network;
- The generalization of the output for the prediction.

3.1 Predicting Financial Time Series with Artificial Neural Networks

In order to use ANN for financial time series predictions, a number of steps should be followed and decisions to be made. As specified previously, we first need to identify the prediction objectives and after profound understanding we can identify the necessary data for analysis, the frequency of the data that will be used for obtaining the output and the time period for the prediction.

3.1.1 Data Archive

The next action is the creation of the data archive that includes:

Data gathering: when we have a full knowledge of the problem, we can proceed with the identification of the necessary data to be collected and the data sources. In our specific case, the information we need to collect for cash forecasting relates to data about revenues and receivables. These data are can be collected from the accounting system of the main unit in charge of financial accounting, i.e. Ministry of Finance or other agencies that deal with financial data;

Analysis and transformation: this is an important step for the preparation of both input and output data. Usually, financial time series contain noises that needs to be removed using transformation techniques that do not change the dynamics of the phenomenon. Some financial time series do not keep a certain ascending or descending trend for a long time period. In many cases, time series contain also outliers, which are exceptions from the normal behavior of the trend. Depending on the specific task, like in case we want to instruct the network to recognize the patterns in normal phases, then we need to make adequate transformations to the data.

Selection of the input and output variables: in this phase we use the first data set for the first network learning. We then can evaluate the information supplied by the individual variables and analyze the correlation. We can proceed with eliminating the least significant variables and perform the next network learning with the reduced data set.

3.1.2. Neural Network Architecture

The selection of the most appropriate NN architecture and the connection mechanism between neurons, is an important and decisive element. The architectural parameters that need to be defined are:

Time division of the database: time series need to be divided in sub periods that determine the scope of learning (training set) and evaluation. The network performs the learning process from the test set, extracted from the training set and then it is applied on the other set of data, called generalization.

Determination of the hidden layers and the quantity of neurons for each layer: Despite many researches, there is no fixed rule to determine the correct number of hidden layers. Although, it seems that two hidden layers are a good approach in many prediction problems.

Regarding the number of hidden neurons, in literature there are various formulas, but they do not prove satisfactory results in all prediction problems. In practice, the number of hidden neurons can be defined though a trial and error process. If the forward selection is chosen, we can start by selecting a small number of hidden neurons and observe the forecast accuracy. Next we increase the number of hidden neurons and perform the test and observation again, until the error is acceptably small. If the backward selection is chosen, then we start with a large number of hidden neurons and then decrease it gradually until we are satisfied with the forecast accuracy.

The number of input neurons depend on the study case as well. In cash forecasting it will depend on (1) number of cash withdrawal inputs included in the model and (2) the way this inputs are encoded. Calendar effects can be included as parameters affecting cash withdrawal: i.e. working day, weekday, holiday effect, salary day effect, and in this example the total number of input neurons would be four, each representing the values of an individual variable at a particular instant of time.

For the same case study, number of output neurons would be one, which will indicate the value of forecasted cash.

Connection mechanism between layers: There are various connection methods that can be used such as:

Standard connections: providing direct connections between input and output that pass through one or more hidden layers, without returning to themselves.

Jumper connections: providing the network to assign connection weights even between neurons of non-adjacent layers.

Repetitive connections: providing the possibility that neurons assigned to hidden layers can return to input variables with iterative processes to quantify precisely the connective weight. This type of connection is widely used for financial time series.

Activation function: There are various activation functions that can be used such as: linear, logistic (sigmoid), symmetric logistics, hyperbolic tangent, correct tangent, sinusoids, Gaussian, invers Gaussian etc. The activation functions used for different layers can be different, or the same for all layers. In case of financial time series, the logistic function is widely used in the hidden layers. In cases where financial time series present dynamic features, the symmetric function is more appropriate especially in the input and hidden layers.

Calculating the errors: When a training set of data is processed through the network, we need to calculate the difference between the actual output and the expected output. This difference represents the error of the network. A low difference (a small error) indicates that the prediction is close to reality, otherwise appropriate changes in the schema or parameters of the NN need to be done. There are no general rules for calculating the error. In case of cash forecasting, as indicated in [3] the forecast error for each pair of actual and forecasted cash withdrawal can be given by the following formula:

$$Error_i = \frac{(withdrawal_{\ actual})_i - (withdrawal_{\ predicted})_i}{(withdrawal_{\ actual})_i} x\ 100\%$$

for i= 1, 2,., N, where N is the number of testing data points. After calculating the forecast error, forecasting accuracy can be calculated as:

Forecasting Accuracy = 100 - % of error in forecasting

Learning modality of the network: As specified earlier the network can perform supervised or unsupervised learning. In financial time series data, the supervised learning is commonly used. There are several algorithms, each of them with feature peculiarities that make them more effective for solving certain problems. The most commonly used algorithm in financial time series prediction is the back propagation algorithm.

3.1.3 Back propagation algorithm

The back propagation algorithm allows the evaluation of the gradient of the error function for a feed forward network characterized by an arbitrary architecture, with different activation and error functions. It learns during a training epoch, and will probably require several epochs before the network has sufficiently learnt to handle all the provided data and the end result is satisfactory. The steps of the training epoch are [7]:

For each input entry in the training data set:

- feed input entry data into the network (feed forward).
- Initialize weights.
- check output against desired value and feedback error (back propagate)

Back propagation consists of:

- calculate error gradients
- update weights

4. Implementation

Based on the description above, the neural network will be composed of one input layer, two hidden layers and the output layer. The network will use the financial time series to predict their future values. Let us consider a series t_i $(i \leq h)$, where h is the prediction origin. From the t_i series, it can be extracted a training set.

The followed approach consists in the division of t_i, in sequential groups with a fixed number of components as shown below:

$$Groups\ of\ 3 \rightarrow \begin{Bmatrix} t_1 \\ t_2 \\ t_3 \end{Bmatrix} \begin{Bmatrix} t_2 \\ t_3 \\ t_4 \end{Bmatrix} \begin{Bmatrix} t_3 \\ t_4 \\ t_5 \end{Bmatrix} \begin{Bmatrix} 4 \\ t_5 \\ t_6 \end{Bmatrix}$$

$$Groups\ of\ 4 \rightarrow \begin{Bmatrix} t_1 \\ t_2 \\ t_3 \\ t_4 \end{Bmatrix} \begin{Bmatrix} t_2 \\ t_3 \\ t_4 \\ t_5 \end{Bmatrix} \begin{Bmatrix} t_3 \\ t_4 \\ t_5 \\ t_6 \end{Bmatrix}$$

Each of these elements is an input to the network, to which it is produced the corresponding output. The output layer contains a single neuron, meaning that the correct value is the first term of the series not contained in the input data as shown below:

$$Input : \begin{Bmatrix} t_1 \\ t_2 \\ t_3 \\ t_4 \end{Bmatrix} \gg output : t_5;$$

$$Input : \begin{Bmatrix} t_2 \\ t_3 \\ t_4 \\ t_5 \end{Bmatrix} \gg output : t_6;$$

The window size *(ws)* parameter is defined as the number of the elements of each subset plus 1 (output value). This way the training set will be composed of *h – (ws-1)* copies: an input vector to which a correct output vector is associated. So, it builds a matrix containing input values *(InM)* and a vector *(Ou)* containing the correct output values, as illustrated below:

$$InM = \begin{Bmatrix} t_1 & t_2 \\ t_2 & t_3 \\ t_3 & t_4 \\ t_4 & t_5 \end{Bmatrix}; \quad Ou = \begin{Bmatrix} t_5 \\ t_6 \end{Bmatrix}$$

The network is configured to have *ws-1* neurons in the first layer and one neuron in the last layer.

When all the patterns included in the training set have been presented, the weights are being adjusted and one iteration has been completed. The iteration numbers performed before terminating the learning (defined as *MxIt*) is an arbitrary parameter.

Once the training process of the network is completed, it can be used to make predictions.

3.3 Arbitrary Parameters

This network contains as free parameters only the weights, but the implementation introduces other arbitrary parameters as follows:

1. Hidden neurons *(hN)*. As explained above, there is no specific rule to define the number of hidden neurons, but it can be used a trial and error process. A low number of *hN* can lead to non-accurate predictions, whereas a large number can lead to overfitting. In general, we can start by specifying the same *hN* number as the input neurons.
2. Window size *(ws)*. Determines the number of neurons of the input layer *(ws-1)* and as researches have shown, it impacts significantly the network answer.
3. Maximal number of iterations *(MxIt).* If presenting the patterns more often, the error in the training set decreases, but a large number of iterations can cause overfitting.
4. Weight choice *(wc)*. Represents the number of times the weights have been evaluated within each simulation. The configuration that presents the lower error is then selected.
5. Number of simulation *(simul)*. Different simulations are performed with the same input data in order to have a “distribution” of the obtained results. This way, it is possible to associate a possible uncertainty to the performed simulations.

The random nature of the results obtained depends on the initial choice of synaptic weights. There are in total *(ws x hN) + (hN + 1)* weights, and their initial values are randomly selected. This is necessary because the error function can present some local minimums. Performing more simulations with a different initial weight configuration, does not avoid these minimums, but can mediate them.

4. Conclusions

Artificial neural network forecasts systems outcomes are not usually error free, but they are better and reliable techniques compared to the others.
During this research we tried to present the structure of a neural network in general and the theoretical steps that need to be followed for the implementation. We focused on cash forecasting as a specific area of financial analysis, where prediction of the future values has a significant importance for proper cash management. We built a NN model for predicting financial time series data and we also tried to present the necessary parameters and the way they can be defined during this process.
While predicting cash forecast this technique should be adopted by organizations although the cost of its operation may be costly in terms of hardware and software procurement and also adequate training of the implementers of the system. Despite that, it is worth is in the long run due to the more accurate predictions.

References

[1] S. Rouhani, S. Asgari, and S. V. Mirhosseini, Review Study: Business Intelligence Concepts and Approaches, American Journal of Scientific Research. 50. pp. 62-75.
[2] W. W. Eckerson, What Are Performance Dashboards, Available at http://bpmpartners.com/documents/Chapter1Excerpt.pdf.
[3] PCh. Kumar and E. Walia. "Cash Forecasting: An Application of Artificial Neural Networks in Finance." IJCSA 3, no. 1 (2006).

[4] B.K. Wong., T.A. Bodnovich, Y. Selvi, 1995. A bibliography of neural networks business application research: 1988-Sep-tember 1994. Expert Systems 12 (3).
[5] S. Haykin, Neural Networks and Learning Machines, Third Edition, 2009.
[6] M. Anandarajan, P. Lee, A. Anandarajan. Bankruptcy Prediction Using Neural Networks. In Business Intelligence Techniques. A Perspective from Accounting and Finance.
[7] K.G. Akintola, B. K. Alese & A.F. Thompson. Time series forecasting with neural network: A case study of stock prices of intercontinental bank Nigeria. International Journal of Recent Research and Applied Studies, 9 (3), December 2011.