Philippine Stock Prediction using   
Artificial NeuralNetworks

Pre-final Paper for the Project

Antonio, Daniel

Electronics and Communications Engineering Department

De La Salle University - Manila

Manila, Philippines

Eco, Lance Randall Lim

Electronics and Communications Engineering Department

De La Salle University – Manila

Manila, Philippines

lance\_eco@dlsu.edu.ph

Tan, Sidney Sheldon Urlanda

Electronics and Communications Engineering Department

De La Salle University - Manila

Manila, Philippines

sidney\_tan@dlsu.edu.ph

Zarzoso, Aaron John Pascua

Electronics and Communications Engineering Department

De La Salle University - Manila

Manila, Philippines

aaron\_zarzoso@dlsu.edu.ph

*Abstract*—This paperrepresent the students’ preliminary report on the project for the course LBYCP29. It contains the procedures done in the implementation of stock price prediction in the Philippines through the use ofneural network. There is also the use of the backpropagation algorithm which is like the gradient descent.There is also the use of preprocessing data and the definition of the error function which is to be minimized.

In this work we present an Artificial Neural Network approach to predict stock market indices. We outline the design of the Neural Network model with its features parameters. A number of activation functions are implemented along with options for cross validation sets. We test our algorithm on PLDT stock index dataset.

Keywords—matlab; octave; Artificial neural networks; Multi-layer neural network; Prediction methods; Stock markets; logistic regression; training; testing; optimization; accuracy;

# Introduction

Neural network is an information processing paradigm that is influenced by the path organic nervous systems process data such as the mind of humans. It generally shown as systems of interconnected neurons which can exchange messages to each other. In addition, there connections have individual numeric weights that can be tuned based on experience which makes the neural nets adaptive to any inputs and have the capacity to learn.

Stock market remains one of the primary means of investment for investors for at least several decades. An attempt is made to predict the closing prices of stock market using neural networks. Many people tried on stock market to extract some useful patterns in predicting their movements. The Artificial Neural Networks have the capacity to determine the nonlinear relationship in the input data set without a previous assumption of knowledge of relation between the input and the output which is compatible in other models in predicting stock prices.

# Instructions/procedure

## Artificial Neural Networks

In this section we describe the structure of Artificial Neurons and how they are connected to construct Artificial Neural Network.

1. Artificial Neurons

Biological neuronal structure is the inspiration for artificial neurons. Specific transmitter substances are released from the sending side of the junction

Artificial neurons are inspired from biological neuronal structure. The transmission of a signal starting with one neuron then onto the next through neurotransmitters is an intricate synthetic procedure in which particular transmitter substances are discharged from the sending side of the intersection. The ending result is to raise or lower the electrical potential inside the body of the cell that will receive it. On the off chance that this potential achieves a limit, the neuron fires. It is this trademark that the artificial neuron model try to imitate. The neuron model as can be seen in Figure 1 is the one that is generally utilized in artificial neural systems with some minor adjustments on it



Fig. 1. Artificial Neuron Structure

The artificial neuron given in this figure has N number of inputs, which are represented as *u1, u2, ...uN*. Every line associating these inputs to the neuron has a corresponding weight, which are represented as *w1, w2,.., wN,* accordingly. The synaptic networks in biological neuronsare similar and comparable to the weights in the artificial model. The limit in the artificial neuron is generally denoted by θ and the activation that matches to the graded potential is given by the equation:

The real values are the inputs and weights. Weights with a negative value shows inhibitory in the connection while a positive valuemeans that the connection is excitatory. In spite of the fact that in biological neurons, has a negative value, it might be allocated a positive value in artificial neuron models. At certain times, the threshold is added into the summation part to make it simple by expecting an imaginary input data *u0 =+1* and a connectionweight *w0 = θ*. Thus the activation equation gets to be:

The resulting and yielded value of the neuron is a component of its activationwhich is similarly comparable to the firing frequency of the biological neurons:

There are various functions that are utilized. Some incorporate binary threshold, linear threshold, sigmoid, hyperbolic tan and Gaussian.

1. Artificial Neural Networks

While a solitary artificial neuron is not ready to actualize some Boolean functions, the issue is overcome by combining the outputs of neurons as input to the others, so constructing a neural system. Assume that we have connected numerous artificial neurons to structure a neural system. In this case, there are a lot of neurons in that are considered, so we correspond a number of index to the neurons to separate them. At that point, to denote the activation *ith* neuron, the equations are altered in this manner:

where 𝑥j could either be the output of another neuron or an input coming from outside the network.

There are various architectures being used for ANNs. The neurons are arranged as layers in feedforward neural systems. The neurons in a layer get input from the preceding layer and provide to their output to the following layer. The same or preceding layers are not allowedin this type of networksconnectionsto the neurons. The outputisthe last layer of neurons and the hidden layers are thelayers between the information and output layers. The input layer is comprised of unique input neurons, which transmit the input coming externally to their output. Systems are called single layer network if in a system there is just the layer of inputnodes and a solitary layer of neurons making up the output layer. Systems are called multilayer systems if there are one or more hidden layers. The structures are called repetitive networks, ifconnections with the neurons of the same layer or to the preceding layers are permitted.



Fig. 2. Artificial Neural Network

## Back PropagationBackpropagation Learning Algorithm

Gradient descent algorithms, which expect to identity the minima/maxima of a given function in the process of iteratively moving toward the negative of the slope of the function to be minimized/maximized, is where the backpropagation algorithm categorizes. The primary target is to minimize the error function. The averageerrorfunction to be minimized is given by:

The weights are updated on apattern bypattern basis until a complete epoch has been managed during this process. The modifications to the weights are made as per the individualerrors processed for every example exhibited to the network. The arithmetic average of these individual weights over the whole training set is an approximation of the true change that would come about because of the adjustment of the weights due to the error function.

A gradient descent technique is utilized to minimize the error. The chain rule for differentiation turn out to be:

After further simplifying:

The final rule for updating weights becomes

where,

for the last layer and

for the intermediate hidden layers.

We utilize the batch learning plan for updating the weight– all the training samples are taken as inputs into the network and the adjustment in all weights is processed from every inputsample. At that point toward the end we update the weights as indicated by the summation of all updates. One is equivalent to one iteration of inputting all the training samples.

For practicality, ANNs executing the backpropagation algorithm don't have an excess of layers, because ofthe time for training the networksincreases in an exponential manner. Additionally, there are modifications to the backpropagation algorithm which permit a better learning.

Consequently, the algorithm above can be utilized to train an Artificial Neural Network (ANN) given the training information/input and the learning rate. The above network can usually have a subjective number of hidden layers and a discretionary number of hidden neurons in every layer. The number of input layer neurons is dependent on the quantity of input components in every example, and the quantity of output layer neurons is dependent on the quantity of output elements in the target values.

There are a couple of consattributed to backpropagation learning also:

• The convergence acquired from backpropagation learning is slow.

• The convergence in backpropagation learning is not ensured.

• The outcome might merge to any local minimum on the error surface because stochastic gradient descent exists on a surface which is not flat.

• Backpropagation learning requires data scaling or normalization.

• Backpropagation requires the activation function utilized by the neurons to be differentiable.

## Ann Model Features

1. Activation Function

The user likewise has an alternative of three activationfunctions for the neurons:

* + Unipolar sigmoid:
  + Bipolar sigmoid:
  + Tan hyperbolic:
  + Radial basis function

The activation function utilized is common to every one of the neurons in the ANN.

1. Hidden Layers and Nodes

The artificial neural system we train for the prediction of picture and stock input has a discretionary number of hidden layers, and discretionary number of hiddennodes in every layer, both of which the user chooses amid run-time.

1. Data Normalization

The input is normalized before being fed to the ANN. The input vectors of the training information are normalized such that every one of the components are zero-mean and unit variance. The values that are needed are normalized such that if the activationfunction is Unipolar sigmoid, they are normalized to a value somewhere around 0 and 1 (since these are the base and most extreme estimations of the activationfunction, and so the output of the ANN), and if the activationfunction is Bipolar sigmoid or Tan hyperbolic, they are normalized to a value between - 1 and 1 and 0 and when the activation function is RBF.

The test input vector is scaled again by the same variables with which the trainingdata was normalized. The output value from the ANN for this test vector is additionally scaled back with the same component as the needed values for the training data.

1. Stopping Criterion

The learning rate 𝜂can control the rate of convergence for the backpropagation algorithm. A bigger value of 𝜂 would guarantee speedier convergence, but as consequence it may make the calculation swing around the minima, while a smallervalue of 𝜂 would make the convergence slow.

We need some stoppingcriterion for the calculation also, to guarantee that it doesn't run all the time and will eventually stop. For our tests, we utilize a three-fold stopping criterion. The backpropagation algorithm stops if any of these conditions are satisfied:

• The differnce in error starting with one iteration then onto the next falls beneath a limit that the user can set.

• The errorvalue starts to increment. There is an unwinding component here too that permitsnegligible increment as it is noted that the error tends to increment a small amount and afterward decrease again.

• If the quantity of iterations (epochs) goes past a threshold. For our situation the threshold is 200.

1. Error Calculation

The rms error between the target values and the actual outputs is the calculation of the error for convergence. We use the same error to report the performance of the algorithm on the test set.

1. Cross-validation set

We give an alternative of utilizing a cross-validation set to gauge the error of the backpropagation algorithm after each iteration in our algorithm. The cross-validation set is autonomous of the training set and aides in a more broad measure of the error and gives better results.

## Data Pre-Processing

The Nifty Sensex information included the date, time of day, opening price, closingprice, high, low and fragmentary change in cost from the preceding time step. Among these, just four factors were considered, the opening price, closingprice, high cost and the low cost. The output comprised of one factor, the closing value. The information was further divided into 60% for training and 40 % for testing the data.

In the 60 % for training, 40 % was utilized for simply training the model and the rest 20 % for cross validation of the model, wherein whilst the model was being constructed, the error was additionally being processed at the same time.

Mix of a couple samples of the data into one single vector is to be fed to the ANN. This acquaints some latencyto the framework. A moving window is utilized to add the data points as follows:

Here k is the latency or the quantity of past observations used to predict the following value. Target values are essentially the data vector values at the next time step:

Before being fed into the ANN, the inputs were normalized by "zscore" function characterized in MATLAB, wherein the mean was subtracted and the valuedivided by the difference of the data. The needed outputs were likewise normalized by target functions, dividing by their greatest values, noting the upper and lower threshold points for the separate activationfunctions ((0,1) for unipolar sigmoid, (- 1, 1) for the bipolar sigmoid and the tan hyperbolic functions).

# Data and results

## User input data

Parameters:

• Inputs – the data of the specific company’s stock index which is composed of the following: open, low, high, and close value.

• Output – the predicted close value.

• Percentage of training data (40%)

• Percentage of testing data (40%)

• Percentage of validation data (20%)

# analysis and conclusion

In our study of the artificial neural network, a highly flexible non-linear modeling technique has been implemented to predict the stock prices of selected sectors. The input used are opening price, high, low, closing price as well as stock volume. The predicted outcome demonstrate that artificial neural network has been able to predict stock prices with better accuracy if we increase the number of input data. However, there is a considerable scope definitely to build on to attempt all possible ways to predict the stock prices with a higher accuracy rate.

##### References

1. MathWorks. (N.D.). *MATLAB -The Language of Technical Computing*. [Online]. Available: http://www.mathworks.com/  
   products/matlab/
2. Eaton, J. (N.D.)*. About GNU Octave.* [Online]. Available: https://www.gnu.org/software/octave/about.html
3. https://www.python.org/doc/essays/blurb/
4. http://spin.atomicobject.com/2014/06/24/gradient-descent-linear-regression/
5. http://www.seattlerobotics.org/encoder/nov98/neural.html
6. http://cs231n.github.io/neural-networks-3/#gradcheck
7. Python.org, 'Welcome to Python.org', 2015. [Online]. Available: https://www.python.org/doc/essays/blurb/. [Accessed: 03- Nov- 2015].
8. Doc.ic.ac.uk, 'Neural Networks', 2015. [Online]. Available: http://www.doc.ic.ac.uk/~nd/surprise\_96/journal/vol4/cs11/report.html#What is a Neural Network. [Accessed: 27- Oct- 2015].
9. Atomic Spin, 'An Introduction to Gradient Descent and Linear Regression', 2014. [Online]. Available: http://spin.atomicobject.com/2014/06/24/gradient-descent-linear-regression/. [Accessed: 03- Nov- 2015].
10. Seattlerobotics.org, 'An Introduction to Back-Propagation Neural Networks', 2015. [Online]. Available: http://www.seattlerobotics.org/encoder/nov98/neural.html. [Accessed: 03- Nov- 2015].
11. J.E. Dayhoff, J.M. DeLeo, “Artificial neural networks: Opening the black box”. Cancer, 91 (8 Suppl.) (2001), pp. 1615–1635.
12. McCulloch, W. and Pitts, W. (1943). “A logical calculus of the ideas immanent in nervous activity”. Bulletin of Mathematical Biophysics, 7:115 - 133.
13. Herculano-Houzel, Suzana. ”The human brain in numbers: a linearly scaled-up primate brain”. Frontiers In Human Neuroscience.doi:10.3389/neuro.09.031.2009.
14. T. Kimoto, K. Asakawa,M. Yoda,M. Takeoka. “Stockmarket prediction systemwithmodular neural network”. Proceedings of the International Joint Conference on Neural Networks, 1990, pages 1–6
15. H. Mizuno, M. Kosaka, H. Yajima, N. Komoda. “Application of Neural Network to Technical Analysis of Stock Market Prediction”. Studies in Informatic and Control, Vol.7, No.3, 1998, pages.111–120.
16. Beale, R. and Jackson, T.Neural Computing: An Introduction. Hilger, Philadelphia, PA, 1991.
17. B.G. Malkiel. “A Random Walk Down Wall Street” (6th ed.). W.W. Norton & Company, Inc., 1973