INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact factor: 4.295
(Volume 4, Issue 1)
Available online at www.ijariit.com

Stock Price Prediction using Neural Networks

Shallu Bashambu
shallubashambu@rediffmail.com
Maharaja Agrasen Institute of Technology, Rohini, Delhi

Aman Sikka
amansikka220@gmail.com
Maharaja Agrasen Institute of Technology, Rohini, Delhi

Pallav Negi
negi.pallav@gmail.com
Maharaja Agrasen Institute of Technology, Rohini, Delhi

ABSTRACT

Trading stocks on the stock market is one of the major investment activities. In the past, investors developed a number of stock analysis method that could help them predict the direction of stock price movement. Modelling and predicting of equity future price, based on the current financial information and news, is of enormous use to the investors. Introduction of machine learning caused that new models can be developed based on the past data. Our main hypothesis was that by applying machine learning and training it on the past data, it is possible to predict the movement of the stock price. Particularly, we want to determine the percentage of growth or fall in a stock price for the next day which can be variable. Usually, investors want to select the stocks which will grow in price substantially.

Keywords: Machine Learning, Deep Learning, Neural Networks.

1. INTRODUCTION

In the past few decades, forecasting of stock market is gaining more attention as the profitability of investors in the stock market mainly depends on the predictability. If the direction of the market is successfully predicted the investors can yield enough profits out of market using prediction.

Stock price prediction s rather a hazardous operation. A good analyst is therefore not the one who is always right, but someone who is at average, someone who has higher efficiency than his colleagues. In the last few years, it has become clear that neural networks have become part of this class of analysts. Neural networks are programs that are based on the geometry of the human brain. The theory was developed in 1943, when the first computers were not even produced. The domain of neural networks has become one of the fastest growing sub-areas in computer science in the last ten years.

2. NEURAL NETWORKS

An Artificial Neural Network (ANN) is an information processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. The key element of this paradigm is the novel structure of the information processing system. It is composed of a large number of highly interconnected processing elements (neurons) working in unison to solve specific problems. ANNs, like people, learn by example. An ANN is configured for a specific application, such as pattern recognition or data classification, through a learning process. Learning in biological systems involves adjustments to the synaptic connections that exist between the neurons. This is true of ANNs as well.

One property of neural networks is their ability to learn. Most newly programmed neural networks are not able to perform their task with the desired accuracy at once. Usually a network's behavior is adapted in a learning or training process. During this process the network is iteratively provided with a set of input patterns together with the corresponding output patterns until it produces the desired output. This set of input patterns & corresponding output patterns is called a training set. While training, the network may change the values of its parameters according to the applied learning rule.

The purpose of training a neural network on a certain task depends on an important assumption. After the training phase the neural network is assumed to perform its task satisfactory on previously un-encountered input patterns: the training is useful only if the knowledge gained from the training pattern generalizes to other input patterns. Therefore it is important for the training set to be representative of all input patterns on which the network will perform its task.

2.1 Stock Price Prediction Using Neural Networks

Complex relationships between inputs and outputs may not always allow us to find patterns. Neural Networks is gaining much attention these days because of its capability of solving such problems. It has robust ability to discover relationship in the input data set without a priori assumption of the knowledge of relation between the input and the output data. It can be used to build model that identify unknown hidden patterns in data which can be further used for prediction purposes. Neural Network has already been successfully applied in the fields related to finance, econometrics, medicine and engineering.

A lot of data is produced in the stock market in a very short period of time. Therefore we need a fast response in this area. Most of the research is done in the area of analysis of the time series (prediction of future values based on stock history). However, the history is not the only factor of the stock to be predicted: the stock price of a company say Google for example will be highly dependent on the new developments of the software industry & the new products released by Google. Time series analysis is used in many different areas. Our objective in this particular case is to predict the next value in a time series: the next stock price on a particular date.

In a chaotic system like stock market, there may be a very large number of unknown factors which may affect the stock price, therefore no significant mathematical relation between the factors & the price can be found. We can't use mathematical formulas to predict those factors. No law exists which governs the stock prices using these underlying factors. Taking this into consideration, application of neural networks would be very beneficial in predicting the stock price.

In this paper, feed forward propagation neural network is used for prediction. Feed forward neural networks is unidirectional connection between the neurons that means that the information can flow only in one direction which is the forward direction. Input has to be fed into the first layer & then the hidden layers come into play, the hidden layers are connected to the last layer which produces the output. There is no connection among neurons in the same layer. Since the information is fed from one layer to another, it is called feed forward neural network.

The back propagation algorithm falls into the general category of gradient descent algorithms. Purpose of gradient descent algorithm is to find the minima/maxima of a function by iteratively moving in the direction of negative slope of the function that we want to minimize/maximize. In our problem objective is to minimize the error function. In back-propagation algorithm the network is trained by repeatedly processing the training data set and comparing the network output with the actual output to reduce the mean square error. Weights of the connections between various neurons has been modified and this process has been continued till the error comes within threshold value. It is back-propagation algorithm as errors from output is back-propagating towards the input layer during training sessions with the objective of minimizing the mean square error. As errors back-propagated model can be continuously updated to minimize error. Along with this, few disadvantages are also associated with back propagation algorithm-

- It is not guaranteed that result always converges to global minima on the error function.
- Even it is not guaranteed that result always converge.
- If result converges, rate of convergence is very slow.
- Typically each backpropagation training session starts with different initial weights and biases, and different divisions of data into training, validation, and test sets. These different conditions can lead to different solutions for the same problem.

3. METHODOLOGY

The work flow for the general neural network design process has five primary steps:

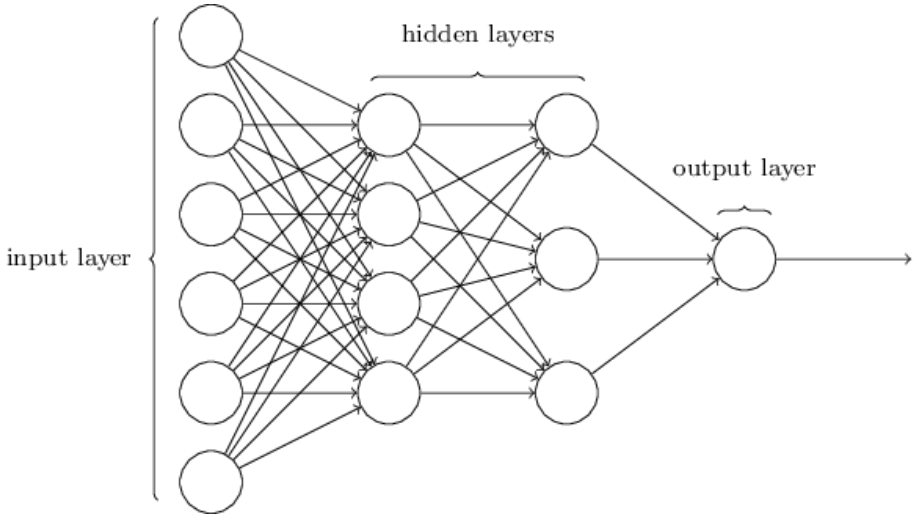
- Data Collection and Preparation
- Network Creation
- Training the network
- Validating the network
- Using the network

a) Data Collection and Preparation

Data collection is the primary step and it is necessary in order to train, validate and test the neural network. General practice is to divide data into three subsets, namely, training set, validation set and testing set. Training set is foremost as it is used to compute the gradient and updating the network weights and biases. Validation set is used as during training phase error on validation is monitored and when the network begins to over fit the data, the error on validation phase starts rising. Then, the network weights and biases corresponding to minimum validation error can be used for the prediction purpose.

b) Network Creation

It is crucial step for prediction, in this step network has been created with the decision about the inputs, number of layers, number of neurons in each layer, transfer function to be used, outputs etc. We have used date of prediction and opening price as an input. Three hidden layers and one output layer has been used, while number of neurons in each layer and transfer function has been decided by MATLAB itself to predict adjusted closing price of a stock as an output.



Structure of a Neural Network

c) Training the Network

The process of training a neural network involves tuning the values of the weights and biases of the network to optimize network performance (by minimizing the Mean Square Error). Out of two years historical data of stock, we can take 80-90% of the data to train the model while remaining is used to test the model.

d) Validating the Network
Once the network is trained the network is validated using validation data to enhance the performance of network.

e) Using the Network
Once the network was optimized, it has been tested using the test data. In our case collected data of google (registered on NASDAQ as GOOG)

4. STOCK PRICE PREDICTION USING PYTHON

4.1 Python

Python is a high level, interpreted programming language, created by Guido van Rossum. The language is very popular for its code readability and compact line of codes. It uses white space inundation to delimit blocks.

Python provides a large standard library which can be used for various applications for example natural language processing, machine learning, data analysis etc.

It is favored for complex projects, because of its simplicity, diverse range of features and its dynamic nature.

4.2 NumPy

NumPy is the fundamental package for scientific computing with Python. It provides a high-performance multidimensional array object, and tools for working with these arrays. It contains among other things:

- A Powerful N-dimensional array object
- Sophisticated (broadcasting) functions
- Tools for integrating C/C++ and Fortran code
- Useful linear algebra, Fourier transform, and random number capabilities.

4.3 Tensorflow

TensorFlow is an open source software library for numerical computation using data flow graphs. Nodes in the graph represent mathematical operations, while the graph edges represent the multidimensional data arrays (tensors) communicated between them. The flexible architecture allows us to deploy computation to one or more CPUs or GPUs in a desktop, server, or mobile device with a single API.

4.4 Theano

Theano is a Python library that allows us to define, optimize, and evaluate mathematical expressions involving multi-dimensional arrays efficiently. Theano features:

- Tight integration with NumPy – Use numpy.ndarray in Theano-compiled functions.
- Transparent use of a GPU – Perform data-intensive computations much faster than on a CPU.
- Efficient symbolic differentiation – Theano does your derivatives for functions with one or many inputs.
- Speed and stability optimizations – Get the right answer for log (1+x) even when x is really tiny.
- Dynamic C code generation – Evaluate expressions faster.
- Extensive unit-testing and self-verification – Detect and diagnose many types of errors.

4.5 Keras

Keras is a high-level neural networks API, written in Python and capable of running on top of TensorFlow, CNTK, or Theano. It was developed with a focus on enabling fast experimentation. Being able to go from idea to result with the least possible delay is key to doing good research.

We use Keras if we need a deep learning library that:

- Allows for easy and fast prototyping (through user friendliness, modularity, and extensibility).
- Supports both convolutional networks and recurrent networks, as well as combinations of the two.
- Runs seamlessly on CPU and GPU.

5. FUTURE SCOPE

Using neural networks to forecast stock market prices will be a continuing area of research as researchers and investors strive to outperform the market, with the ultimate goal of bettering their returns. It is unlikely that new theoretical ideas will come out of this applied work. However, interesting results and validation of theories will occur as neural networks are applied to more complicated problems.

Financial neural networks must be trained to learn the data and generalize, while being prevented from overtraining and memorizing the data. Also, due to their large number of inputs, network pruning is important to remove redundant input nodes and speed-up training and recall. The major research thrust in this area should be determining better network architectures. The commonly used backpropagation network offers good performance, but this performance could be improved by using recurrence or reusing past inputs and outputs. The architecture combining neural networks and expert systems shows potential. Currently, implemented neural networks have shown that the Efficient Market Hypothesis does not hold in practice, and that stock markets are probably chaotic systems. Until we more fully understand the dynamics behind such chaotic systems, the best we can hope for is to model them as accurately as possible.

6. CONCLUSIONS

Predicting the direction of movements of the stock market index is important for the development of effective market trading strategies. It usually affects a financial trader's decision to buy or sell an instrument. Successful prediction of stock prices may promise attractive benefits for investors. These tasks are highly complicated and very difficult. We can use the principle of artificial neural networks to correctly predict the future value of a particular stock. This is done in steps, first we collect the past data of a stock. Then we create a neural network & train the network on the basis of the past data. Generally 80-90% of the data is taken to train the model. Then we use our model to predict the value of the stock & compare the value with the pre obtained data set. Although neural networks are not perfect in their prediction, they outperform all other methods and provide hope that one day we can more fully understand dynamic, chaotic systems such as the stock market.

7. REFERENCES

[1] J.E. Dayhoff, J.M. DeLeo, "Artificial neural networks: Opening the black box". Cancer, 91 (8 Suppl.) (2001), pp. 1615–1635
[2] J.hertz, A.Krogh & R.G. Palmer. Introduction to the Theory of Neural Computation. Addison-Wesley, 1991
[3] Abu-Mostafa, Y. S., & Atiya, A. F. (1996). Introduction to financial forecasting. Applied Intelligence, 6(3), 205–213.
[4] H. Mizuno, M. Kosaka, H. Yajima, N. Komoda. "Application of Neural Network to Technical Analysis of Stock Market Prediction". Studies in Informatics and Control, Vol.7, No.3, 1998, pages.111–120
[5] Z.Tang and P.A.Fishwick, "Backpropagation neural nets as models for time series Forecasting," ORSA journal on computing, vol.5, No.4, pp 374-384, 1993.

