Financial Forecasting using Neural Networks: A Review

Punam Varghade
M.Tech. IInd Year, Dept. of CSE, RCERT
Chandrapur, Maharashtra
India

Prof. Rahila Sheikh
Assistant Professor, Dept. of CSE, RCERT
Chandrapur, Maharashtra
India

ABSTRACT
Neural networks are good at classification, forecasting and recognition. They are also good candidates of financial forecasting tools. Forecasting is often used in the decision making process. Neural network training is an art. Trading based on neural network outputs, or trading strategy is also an art. We will discuss a seven-step neural network forecasting model building approach in this article. Pre and post data processing/analysis skills, data sampling, training criteria and model recommendation will also be covered in this article.

Keywords
Neural Networks, Finance, Time Series Analysis, Forecasting, Artificial Intelligence

1. FORECASTING WITH NEURAL NETWORKS
Forecasting is a process that produces a set of outputs by given a set of variables. The variables are normally historical data. Basically, forecasting assumes that future occurrences are based, at least in part, on presently observable or past events. It assumes that some aspects of the past patterns will continue into the future. Past relationships can then be discovered through study and observation. The basic idea of forecasting is to find an approximation of mapping between the input and output data in order to discover the implicit rules governing the observed movements. For instance, the forecasting of stock prices can be described in this way. Assume that u_i represents today's price, v_i represents the price after ten days. If the prediction of a stock price after ten days could be obtained using today's stock price, then there should be a functional mapping u_i to v_i, where $v_i = \Gamma_i(u_i)$. Using all (u_i, v_i) pairs of historical data, a general function $\Gamma()$ which consists of $\Gamma_i()$ could be obtained, that is $v = \Gamma(u)$. More generally, ur which consists of more information in today's price could be used in function $\Gamma()$. As NNs are universal approximators, we can find a NN simulating this $\Gamma()$ function. The trained network is then used to predict the movements for the future.

NN based financial forecasting has been explored for about a decade. Many research papers are published on various international journals and conferences proceedings. Some companies and institutions are also claiming or marketing the so called advanced forecasting tools or models. Some research results of financial forecasting found in references. For instance, stock trading system[4], stock forecasting [6, 22], foreign exchange rates forecasting [15, 24], option prices [25], advertising and sales volumes [13]. However, Callen et al. [3] claim that NN models are not necessarily superior to linear time series models even when the data are financial, seasonal and nonlinear.

2. TOWARDS A BETTER ROBUST FINANCIAL FORECASTING MODEL
In working towards a more robust financial forecasting model, the following issues are worth examining.

First, instead of emphasizing on the forecasting accuracy only, other financial criteria should be considered. Current researchers tend to use goodness of fit or similar criteria to judge or train their models in financial domain. In terms of mathematical calculation this approach is a correct way in theory. As we understand that a perfect forecasting is impossible in reality. No model can achieve such an ideal goal. Under this constraint, seeking a perfect forecasting is not our aim. We can only try to optimize our imperfect forecasts and use other yardsticks to give the most realistic measure.

Second, there should be adequate organization and processing of forecasting data. Preprocessing and proper sampling of input data can have impact on the forecasting performance. Choice of indicators as inputs through sensitivity analysis could help to eliminate redundant inputs. Furthermore, NN forecasting results should be used wisely and effectively. For example, as the forecast is not perfect, should we compare the NN output with the previous forecast or with the real data especially when price levels are used as the forecasting targets?

Third, a trading system should be used to decide on the best tool to use. NN is not the single tool that can be used for financial forecasting. We also cannot claim that it is the best forecasting tool. In fact, people are still not aware of which kind of time series is the most suitable for NN applications. To conduct post forecasting analysis will allow us to find out the suitability of models and series. We may then conclude that a certain kind of models should be used for a certain kind of time series. Training or building NN models is a trial and error procedure. Some researchers are not willing to test more on their data set [14]. If there is a system that can help us to formalize these tedious exploratory procedures, it will certainly be of great value to financial forecasting.

Instead of just presenting one successful experiment, possibility or confidence level can be applied to the outputs. Data are partitioned into several sets to find out the particular knowledge of this time series. As stated by David Wolpert and William Macready about their No-Free-Lunch theorems [28], averaged over all problems, all search algorithms perform equally. Just experimenting on a single data set, a NN model which outperforms other models can be found. However, for another data set one model which outperforms NN model can also be found according to No-Free-Lunch theorems. To avoid such a case of one model outperforming others, we partition the data set into several sub data sets. The recommended NN models are those that outperform other models for all sub time horizons. In other words, only those models incorporated with enough local knowledge can be used for future forecasting.

It is very important and necessary to emphasize these three issues here. Different criteria exist for the academics and the industry. In academics, sometime people seek for the accuracy towards 100%. While in industry a guaranteed 60% accuracy is typically aimed for. In addition, profit is the eventual goal of practitioners, so a profit oriented forecasting model may fit their needs.

Cohen [5] surveyed 150 papers in the proceedings of the 8th National Conference on artificial intelligence. He discovered that only 42% of the papers reported that a program had run on more than one example; just 30% demonstrated performance in some way; a mere 21% framed hypotheses or made predictions. He then concluded that the methodologies used were incomplete with respect to the goals of designing and analyzing AI system.

Tichy [20] showed that in a very large study of over 400 research articles in computer science. Over 40% of the articles are about new designs and the models completely lack experimental data. In a recent IEEE computer journal, he also points out 16 excuses to avoid experimentation for computer scientists [21]. What he is talking is true and not a joke.

Prechelt [14] showed that the situation is not better in the NN literature. Out of 190 papers published in well-known journals dedicated to NNs, 29% did not employ even a single realistic or real learning problem. Only 8% of the articles presented results for more than one problem using real world data.

To build a NN forecasting we need sufficient experiments. To test only for one market or just for one particular time period means nothing. It will not lead to a robust model based on manually, trial-and-error, or ad hoc experiments. More robust model is needed but not only in one market or for one time period. Because of the lack of industrial models and because failures in academic research are not published, a single person or even a group of researchers will not gain enough information or experiences to build up a good forecasting model. It is obvious that an automated system dealing with NN models building is necessary.

As NN training is an art, many searchers and practitioners have worked in the field to work towards successful prediction and classification. For instance, William Remus and Marcus O'connor [16] suggest some principles for NN prediction and classification are of critical importance in the chapter, “Principles of Forecasting” in “A Handbook for Researchers and Practitioners”:

- Clean the data prior to estimating the NN model.
- Scale and deseasonalize data prior to estimating the model.
- Use appropriate methods to choose the right starting point.
- Use specialized methods to avoid local optima.
- Expand the network until there is no significant improvement in fit.
- Use pruning techniques when estimating NNs and use holdout samples when evaluating NNs.
- Take care to obtain software that has in-built features to avoid NN disadvantages.
- Build plausible NNs to gain model acceptance by reducing their size.
- Use more approaches to ensure that the NN model is valid.

With the authors' experience and sharing from other researchers and practitioners, we propose a seven-step approach for NN financial forecasting model building. The seven steps are basic components of the automated system and normally involved in the manual approach. Each step deals with an important issue. They are data preprocessing, input and output selection, sensitive analysis, data organization, model construction, post analysis and model recommendation.

Step 1. Data Preprocessing

A general format of data is prepared. Depending on the requirement, longer term data, e.g. weekly, monthly data may also be calculated from more frequently sampled time series. We may think that it makes sense to use as frequent data sampling as possible for experiments. However, researchers have found that increasing observation frequency does not always help to improve the accuracy of forecasting [28].

Inspection of data to find outliers is also important as outliers make it difficult for NNs and other forecasting models to model the true underlying functional. Although NNs have been shown to be universal approximators, it had been found that NNs had difficulty modeling seasonal patterns in time series [11].When a time series contains significant seasonality, the data need to be deseasonalized.

Before the data is analyzed, basic preprocessing of data is needed. In the case of days with no trading at all exist, the missing data need to be fill up manually. Heinkel and Kraus [9] stated that there are three possible ways dealing with days with no trading:1) Ignore the days with no trading and use data for trading days. 2) Assign a zero value for the days which are no trading. 3) Build a linear model which can be used to estimate the data value for the day with no trading. In most cases, the horizontal axis is marked by the market day instead of (or in addition to) the calendar date. Suppose we are forecasting the price of next time point. If it is on a Monday, the next time point is tomorrow or Tuesday. If it is on a Friday, the next day will be next Monday in fact it is two days later.

In most of times, weekly closing price refers to each Fridays closing prices. In the event of Friday being a holiday, the most recently available closing price for the stock was used. Some researchers also pick any day as weekly prices. Normalization is also conducted in this phase. The purpose of normalization is to modify the output levels to a reasonable value. Without such transformation, the value of the output may be too large for the network to handle, especially when several layers of nodes in the NN are involved. A transformation can occur at the output of each node, or it can be performed at the final output of the network.

Original values, Y, along with the maximum and minimum values in the input file, is later entered into the equation below to scale the data to the range of [-1,+1].

$$Nm = \frac{2*Y - (Max + Min)}{Max - Min}$$

Step 2. Selection of Input & Output variables

Select inputs from available information. Inputs and targets also need to be carefully selected. Traditionally, only changes are processed to predict targets as the return or changes are the main concerns of fund managers. Three types of changes have been used in previous research: xt -xt-1 , log xt - log xt-1 and t t-1 . In addition, pure time series forecasting xt-1 techniques require a stationary time series while most raw financial time series are not stationary. Here stationarity refers to a stochastic process whose mean, variances and covariance (first and second order moments) do not change with time. xt -xt-1 is thought to be unit dependent, and hence comparisons between series are difficult and are less used in the literature. However, after NNs have been introduced, we can use the original time series as our forecasting targets. We can let the networks to determine the units or patterns from the time series. In fact, the traditional returns are not the exact returns in real life. The inflation is not taken into account at least. These returns are named as nominal returns ignoring inflation as the inflation cannot be calculated so sensibly from daily series. After the aim has been fixed. The NN model will find out the relationship between inputs and the fixed targets. The relationship is discovered from the data rather than according to the human expectation.

In addition to using pure time series, the inputs to NNs can also include some technical indicators such as moving averages, momentum, RSI, etc. These indicators are in popular use amongst chartists and floor traders. Certain indicators, such as moving averages are one of the oldest technical indicators in existence and they happen to be among the most useful indicators. In practice, a trader may only focus on one indicator and base on certain basic rules to trade. However, he needs other indicators to confirm his findings. For instance, if a short term, say 10 days, moving average crosses over a long term, say 30 days, moving average and both moving averages are in an upward direction, it is the time to go long. If the 10 day moving average crosses below the 30 day moving average and both moving averages are directed downward. most traders will consider this as a valid sell signal. With fast calculation capability, more indicators or combined indicator could be used.

Step 3. Sensitivity Analysis

Sensitivity Analysis is used to find out which indicator is more sensitive to the outputs. In other words, after a sensitivity analysis, we can easily eliminate the less sensitive variables from the input set. Usually, sensitivity analysis is used to reduce the number of fundamental factors. NN or some other forecasting models are used in forecasting as the forecast target is believed to have relationship with many other series. Sometimes, the input variables may be correlated with each other. Simply using all the available information may not always enhance the forecasting abilities. This is the same as the observation that complex models do not always out perform simple ones. Empirical studies have shown that forecasts using econometric models

are not necessarily more accurate than those employing time series methods [7]. If the ability of explaining economic or business phenomena which can increase our understanding of relationships between variables is not counted in, econometric models will be useless. Besides fundamental factors, technical indicators may also be used for sensitivity analysis. The basic idea used here is that several trainings are conducted using different variables as inputs to a NN and the performance of them are then compared. If there is no much difference on the performance with or without a variable, this variable is said to be of less significance to the target and thus can be deleted from the inputs to the network. Instead of changing the number of input variables, another approach changes the values of a particular variable. Again, several trainings are conducted using perturbed variables. Each time, a positive or negative change is introduced into the original value of a variable. If there is no much difference on the performance with or without changes of a variable, this variable is said to be of less significance.

Overfitting is another major concern in the design of a NN. When there is no enough data available to train the NNs and the structure of NNs is too complex, the NN tends to memorize the data rather than to generalize from it. Keeping the NN small is one way to avoid overfitting. One can prunes the network to a small size using the technical such as in [12].

Step 4. Data Organization

The next step is Data Organization. In data preprocessing step, we have chosen the prediction goal and the inputs that should be used. The historical data may not necessarily contribute equally to the model building. We know that for certain periods the market is more volatile than others, while some periods are more stable than others. We can emphasize a certain period of data by feeding more times to the network or eliminate some data pattern from unimportant time periods. With the assumption that volatile periods contribute more, we will sample more on volatile periods or vise versa. We can only conclude this from the experiments of particular data set.

The basic assumption for time series forecasting is that the pattern found from historical data will hold in the future. Traditional regression forecasting model building uses all the data available. However, the model obtained may not be suitable for the future. When training NNs, we can hold out a set of data, out-of-sample set apart from training. After the network is confirmed, we use the out-of-sample data to test its performance. There are tradeoffs for testing and training. One should not say it is the best model unless he has tested it, but once one has tested it one has not trained enough. In order to train NNs better, all the data available should be used. The problem is that we have no data to test the ``best'' model. In order to test the model, we partition the data into three parts. The first two parts are used to train (and validate) the NN while the third part of data is used to test the model. But the networks have not been trained enough as the third part is not used in training. The general partition rule for training, validation and testing set is 70%, 20% and 10% respectively according to the authors' experience.

Step 5. Model Construction

Model Construction step deals with NN architecture, hidden layers and activation function. A backpropagation NN is decided by many factors, number of layers, number of nodes in each layer, weights between nodes and the activation function. In our study, a hyperbolic tangent function is used as the activation function for a backpropagation network. Similar to the situation of conventional forecasting models, it is not necessarily true that a complex NN, in terms of more nodes and more hidden layers, gives a better prediction. It is important not to have too many nodes in the hidden layer because this may allow the NN to learn by example only and not to generalize [1] .When building a suitable NN for the financial application we have to balance between convergence and generalization. We use a one hidden layer network for our experiment. We adopt a simple procedure of deciding the number of hidden nodes which is also determined by the number of nodes in the input or preceding layer. For a single hidden layer NN, the number of nodes in the hidden layer being experimented are in the order of n2, n2 ±1, n2± 2 , …, where n2stands for half of the input number. The minimum number is 1 and the maximum number is the number of inputs, n, plus 1. In the case where a single hidden layer is not satisfactory, an additional hidden layer is added. Then another round of similar experiments for each of the single layer networks are conducted and now the new n2 stands for half of the number of nodes in the preceding layer. Besides the architecture itself the weight change is also quite important. The learning rate and momentum rate can lead to different models.

The crucial point is the choice of the sigmoid activation function of the processing neuron. There are several variations from the standard backpropagation algorithm which aim at speeding up its relatively slow convergence, avoiding local minima or improving its generalization ability. e.g. the use of different activation functions other than the usual sigmoid function, the addition of a small positive offset to the derivative of the sigmoid function to avoid saturation at the extremes, the use of a momentum term in the equation for the weight change. More detailed discussion can be found in [10] and [8]. Although the backpropagation algorithm does not guarantee optimal solution, Rumelhart [17] observed that solutions obtained from the algorithm come close to the optimal ones in their experiments. The accuracy of approximation for NNs depends on the selection of proper architecture and weights, however, backpropagation is only a local search algorithm and thus tends to become trapped in local optima. Random selection of initial weights is a common approach. If these initial weights are located on local grades, the algorithm will likely become trapped at a local optimum. Some researchers have tried to solve this problem by imposing constraints on the search space or by restructuring the architecture of the NNs. For example, parameters of the algorithm can be adjusted to affect the momentum of the search so that the search will break out of local optima and move toward the global solution. Another common method for finding the best (perhaps global) solution using backpropagation is to restart the training at many random points. Wang [26] proposes a “fix” for certain classification problems by constraining the NN to only approximate monotonic functions.

Another issue with the backpropagation network is the choice of the number of hidden nodes in the network. While trial-and-error is a common method to determine the number of hidden nodes in a network, genetic algorithms are also often used to find the optimum number [19]. In fact, in recent years, there has been increasing use of genetic algorithms in conjunction with NNs. The application of genetic algorithms to NNs has followed two separate but related paths. First, genetic algorithms have been used to find the optimal network architectures for specific tasks. The second direction involves optimization of the NN using genetic algorithms for search. No matter how sophisticated the NN technology, the design of a neural trading system remains an art. This art, especially in terms of training and configuring NNs for trading, and be simplified through the use of genetic algorithms.

Traditional backpropagation NNs training criterion is based on goodness-of-fit which is also the most popular criterion for forecasting. However, in the context of financial time series forecasting, we are not only concerned at how good the forecasts fit their target. In order to increase the forecastability in terms of profit earning, Yao [23] proposes a profit based adjusted weight factor for backpropagation network training. Instead of using the traditional least squares error, a factor which contains the profit, direction, and time information was added to the error function. The results show that the new approach does improve the forecastability of NN models, for the financial application domain.

Step 6. Post Analysis

In Post Analysis step, experiment results will be analysized to find out the possible relationship such as the relations between higher profit and data characters. According to the performance of each segment, we can decide how long this model can be used. In other words, how long we should retrain the NN model. The knowledge gained from experiment will be used in future practices. A major disadvantage of NNs is that their forecasts seem to come from a black box. One cannot explain why the model made good predictions by examining the model parameters and structures. This makes NN models hard to understand and difficult for some managers to accept. Some work has been done to make these models more understandable [2, 18].

Step 7. Model Recommendation

As we know, certainty can be produced through a large number of uncertainties. The behavior of an individual could not be forecast with any degree of certainty, but on the other hand, the behavior of a group of individuals could be forecast with a higher degree of certainty. With only one case of success does not mean it will be successes in the future. In our approach, we do not just train the network once using one data set. The final NN model we suggested in using of forecasting is not a single network but a group of networks. The networks are amongst the best model we have found using the same data set but different samples, segments, and architectures. The Model Recommendation could be either Best-so-far or Committee. Best-so-far is the best model for the testing data and in the hope of that it is also the best model for the future new data. As we cannot guarantee that the only one model is suitable for the future, we recommend a group of models as a committee in our final model. When forecast is made, instead of basing on one model, it can conclude from the majority of the committee. As on the average, the committee suggestion for the historical data is superior to a single model. Therefore the possibility for future correctness is greater.

3. CONCLUSION REMARKS

NNs are suitable for financial forecasting and marketing analysis. They can be used for financial time series, such as stock exchange indices, foreign exchange rates forecasting. Research experiments show that NN models can outperform conventional models in most cases. As beating the markets is still a difficult problem, if a NN cannot work as an alternative tool for traditional forecasting and analysis models, at least it can work as a complementary tool.

Some people treat NN as a panacea. However, there is no cure-all medicine in the world. When applying a NN model in a real application, attention should be taken in every single step. The usage and training of a NN is an art.

One successful experiment says nothing for real application. There is always another model that exists which is superior to the successful model for other data sets. Segmenting data into several sub sets and training with a NN with the same architecture will assure that the model will not just work for a single data set. Furthermore, building a model construction system will free human being from the tedious trial-and-error procedures.

4. REFERENCES

[1] Jingtao Yao, Palmerston North, Chew Lim TAN, “Guidelines for Financial Forecasting with Neural Networks” Proceedings of International Conference on Neural Information Processing, (2001), 14-18.

[2] E. B. Baum, D. Hassler, “What Size Net Gives Valid Generalization?”, Neural Computation, 1, (1989), 151-160.

[3] J. M. Benitez, J. L. Castro & I Requena, “Are artificial neural networks black boxes?” IEEE Transactions on Neural Networks, 8, 1997, 1156-1164.

[4] J. L. Callen, C. C.Y. Kwan, P. C.Y. Yip, Y.F. Yuan, “Neural network forecasting of quarterly accounting earnings”, International Journal of Forecasting, (12)4, 1996 pp 475-482.

[5] A. J. Chapman, “Stock Market Trading Systems Through Neural Networks: Developing a Model”, International Journal of Applied Expert Systems, Vol. 2, no. 2, 1994, pp88-100.

[6] P. R. Cohen, “A Survey of the Eighth National Conference on Artificial Intelligence: Pulling Together or Pulling Apart? ”, AI Magazine, Vol. 12, No. 1, 1992, pp17-41.

[7] R. G. Donaldson, M. Kamstra, “An artificial neural network-GARCH model for international stock return volatility”, Journal of Empirical Finance, 4(1), 1997, pp 17-46.

[8] R. Fildes, ``Quantitative forecasting --- the state of the art: Causal models'', Journal of the operational Research Society, vol. 36, 1985, pp691-710

[9] R. Hecht-Nielsen, Neurocomputing, Addison-Wesley, 1990.

[10] R. Heinkel, A. Kraus, ``Measuring Event Impacts in Thinly Traded Stocks'', Journal of Financial and Quantitative Analysis, March 1988.

[11] K. Hornik, M. Stinchcombe, White, “Multilayer feedforward networks are universal approximators”,Neural Networks, 2(5), 1989, 359-366.

[12] T. Kolarik, G. Rudorfer, “Time series forecasting using neural networks,” APL Quote Quad, 25(1), 1994, 86-94.

[13] M.C. Mozer, P. Smolensky , “Using relevance to reduce network size automatically”, Connection Science, 1(1), 1989, pp3-16.

[14] H.-L. Poh, J. T. Yao, T. Jasic, “Neural Networks for the Analysis and Forecasting of Advertising and Promotion Impact”, International Journal of Intelligent Systems in Accounting, Finance and Management, Vol. 7, No. 4, 1998, pp253-268.

[15] L. Prechelt, “A Quantitative Study of Experimental Evaluations of Neural Network Learning Algorithms: Current Research Practice”, Neural Networks, 9(3), 1996, pp457-462.

[16] A. N. Refenes, M. Azema-Barac, L. Chen and S. A. Karoussos, “Currency Exchange Rate Prediction and Neural Network Design Strategies”, Neural Computing & Applications, No. 1, 1993, pp46-58.

[17] W. Remus, M. O'connor, “Neural Networks For Time Series Forecasting”, in Principles of Forecasting: A Handbook for Researchers and Practitioners, J. Scott Armstrong, editor, Norwell, MA: Kluwer Academic Publishers, 2001.

[18] D. E. Rumelhart, D.E., G.E. Hinton, R.J. Williams, “Learning Internal Representations by Error propagation”, in: Parallel Distributed Processing, Volume 1, D.E. Rumelhart, J.L. McClelland (Eds.), MIT Press, Cambridge, MA, 1986, pp318-362.

[19] R. Setiono, W.K. Leow and J.Y-L. Thong. “Opening the neural network blackbox: An algorithm for extracting rules from function approximating neural networks”, In Proceedings of International Conference on Information Systems 2000, Brisbane, Australia, December 10 - 13, pp176-186

[20] R. S. Sexton, R. E. Dorsey, J. D. Johnson, “Toward global optimization of neural networks: A comparison of the genetic algorithm and backpropagation”, Decision Support Systems, vol. 22, no. 2, pp. 171--185, 1998.

[21] W. F. Tichy, P. Lukowicz, L. Prechelt, A. Heinz, “Experimental Evaluation in Computer Science: AQuantitative Study”, Journal of Systems and Software, 28(1):9-18. 1995.

[22] W. F. Tichy, “Should Computer Scientists Experiment More? 16 Reasons to Avoid Experimentation”, IEEE Computer, 31(5), 1998, pp32- 44.

[23] J. T. Yao, C. L. Tan and H.-L. Poh, “Neural Networks for Technical Analysis: A Study on KLCI”, International Journal of Theoretical and Applied Finance, Vol. 2, No.2, 1999, pp221-241.

[24] J. T. Yao, C. L. Tan, “Time dependent Directional Profit Model for Financial Time Series Forecasting”,Proceedings of The IEEE-INNS-ENNS International Joint Conference on Neural Networks, Como, Italy, 24- 27 July 2000, Volume V, pp291-296.

[25] J. T. Yao, C. L. Tan, “A case study on using neuralnetworks to perform technical forecasting of forex”, Neurocomputing, Vol. 34, No. 1-4, 2000, pp79-98.

[26] J. T. Yao, C. L. Tan, Y. L. Li, "Option Prices Forecasting Using Neural Networks", Omega: The International Journal of Management Science, Vol. 28, No. 4 2000, pp455-466.

[27] S. Wang, "The Unpredictability of Standard Back Propagation Neural Networks in Classification Applications," Management Science 41(3), March 1995, 555-559.

[28] D. H. Wolpert, W.G. Macready, "No Free Lunch Theorems for Search", Technical Report of The Santa Fe Institute, SFI-TR-95-02-010, 1996.

[29] B. Zhou, "Estimatinzg the Variance Parameter From Noisy High Frequency Financial Data", MIT Sloan School Working Paper, No. 3739, 1995.