Investigating Neural Network Modeling Decisions for the Australian All-Ordinaries Index

Andrew Flitman¹, Mark Barnes², and Deniss Teng Tai Kiat¹

¹Monash University, Business Systems, Clayton, Victoria, Australia andrew.flitman@infotech.monash.edu.au
²Monash University, Computing and IT, Churchill, Victoria, Australia mbbar1@student.monash.edu.au

Abstract. Estimating stock market output depends mainly on identifying nonlinear relationships of input variables. To forecast such systems a non-linear modeling tool is required. This paper describes the experimental approaches for developing an Artificial Neural Network for the purpose of modeling the Australian All Ordinaries Index movement over a prediction horizon of 1 year. Network parameters such as network architectures, input data sizing and periodicity are considered in the development of the network. The evaluation criterion for the Neural Network output is the R Square Statistic.

1 Introduction

Previously, [3] questioned the validity of Artificial Neural Networks (ANN) as a forecasting tool. He pointed out that few comparisons with standard statistical techniques had been made. However, in recent years, Neural Networks have perhaps become the most significant forecasting tools to be applied to time series data and also to model financial markets.

The interest in the ANN capabilities to forecast has resulted in a significant amount of research in comparing how well it performs compared to previous statistical techniques. Flitman [4] modeled the Australian All Ordinaries Index using a Back Propagation Neural Network and compared those results with statistical methods. He used combinations of five explanatory variables to predict the index twelve months in advance. Often the Neural Network performed as well, if not better than traditional methods. Barnes [2] also modeled the Australian All Ordinaries Index, however, in that paper the prediction forecasts were 1, 3 and 6 months. This model included a 'sliding window', which allowed the models to be retrained with the most recently available data while maintaining the training size. It was found that the explanatory variables improved the predictive power of the Neural Network when predicting 3 and 6 months into the future, but they didn't assist in predicting one-month ahead. At this stage, not all factors that influence the forecasting performance of Neural Networks are known. As a result, issues of methodology for development of any ANN for specific purposes become increasingly important. This paper extends previous work of Flitman [4] and Barnes [2] focusing on iterating data sizing, periodicity and network architectures, for development of an ANN in trying to predict the movement of the Australian All Ordinaries Index. Development factors were evaluated by comparing the performance of different Neural Network architectures using the same input data set and the performance of a single Neural Network by varying and scaling the input data.

This paper is organized into four main sections. In section 2 a discussion of the All Ordinaries Index is presented. This is followed by a discussion on the selection of variables, section 3. Then a description of Neural Networks is section 4. In section 5 we describe the network architecture and its results. The final section 6 concludes the paper and discusses some further research.

2 All Ordinaries Index

The All Ordinaries Index is a weighted average of more than 250 share prices. It is an example of a price index, in that no account is taken of dividend payouts. As such it only measures the capital gain or loss resulting from investing in the stock market. In a wider context the All Ordinaries Index can be thought of as a reflection of the current and future health and wealth of the economy.

The All Ordinaries works by following the changes in the aggregate market values (AMVs) of companies covered by the index. For each company, the number of listed shares is multiplied by the market price. The AMV for the index portfolio is then calculated with the index calculated as:

Todays Closing Index =
$$\frac{(\text{Yesterday's closing}) \times (\text{End - of - day AMV})}{\text{Start - of - day AMV}}$$
(1)

The number of companies on the index is not fixed, and alters as company eligibility for inclusion changes. It is comparable to Standard and Poors 500, The Toronto 300 Composite and the Topix Index (Japan).

Peters [9] showed that a historical plot of the S&P 500 looked very similar irrespective of whether plotted on a monthly, weekly or daily basis. This shows that non-linear dynamic (NLD) systems demonstrate self-similarity, which means that the variability of the time series would look similar when viewed with different time scales. The AOI is assumed to belong to the same family as the S&P 500 and hence exhibit non-linear dynamic and self-similarity characteristics. Theoretical work by [9] indicated that three inputs were required to predict the pattern of the S&P index. Whilst this result may imply a relatively simple time series, the reality is that no attempt was made to identify what the variables were. Thus, given the imperfection of the input variables that are actually available to us, we will need somewhat more than the theoretical minimum. As discussed below our models utilised five input variables.

3 Selection of Variables

Flitman [4], used ten different variables in combinations of five to locate the five variables that had the most influence in improving his predictions. This was guided by texts such as [7] where variables demonstrated the statistical relevance of the data. We compared the predictive power of the models using lagged AOI values and the inclusion of the extra four explanatory variables as suggested by [4]. We also looked at all combinations of the four explanatory variables using the Neural Network and Linear Regression techniques to observe the importance, individually and collectively that they had on the models. The constant variable used in all models was the historic movements in the all ordinaries.

Historic Movements in the All Ordinaries (AOI): is included since the past history of the data often can give a guide to the future. The four explanatory variables are as follows:

Producer Prices (Manufactoring)(PP): are used in preference to Consumer Prices. Producer prices provide a leading indicator to consumer prices, and also reflect more accurately the changing prices of doing business.

Money Supply (M3): is regulated by the Government to provide sustainable economic growth. If the supply is shrinking very quickly there is a possible deflationary cycle with consequent effect on the All Ordinaries. When the rate of change is slowing down very quickly this implies a slowdown in economic activity with a consequent effect on stocks. Similarly, too rapid growth implies inflationary pressures with natural knock-on effects for stocks.

Price Earning Ratio (PE): is an excellent indicator of how cheap or expensive the stock market is. Of particular interest are real P/Es. In general the higher the market's average real P/E, the worse stocks have performed over the next twelve months [7].

Treasury 10 Year Bonds (TB): is used as an indicator of interest rates. When used independently, interest rates are only of use in predicting the stock market when expressed in real rather than nominal terms [7].

3.1 The Data Sets

Data on the above indicators were readily obtained from <u>http://www.finance.yahoo</u>, Australian Bureau of Statistics and <u>www.economagic.com</u>. Figures at the start of the month were obtained from May 1987 to May 2001. This is an extension of the data set [4] used previously and a subset of the data set [2] used. The data set consists of a total of 169 observations and divided into three categories for the Neural Network:

The data that was being used as input to develop the Neural Network model consisted of the training and test sets. Thus the network was trained using training data, with the network being saved each time an improvement was registered on the test data. Extraction of the test data from the training data was undertaken using sampling. The remainder of the data that was not used to train the network was used for validating the output (production set). The production data was not exposed to the network prior to this. The production set was simply used to compare the actual index to the results produced by the model. It is the result against this validation set which are of particular interest. The Neural Network modeling approach can be applied to fit historic data with good R-square.

4 Neural Networks

Neural Networks are built from elements that behave somewhat like individual nerve cells (or neurons). Much research has been conducted looking at the application of this technology for classifying and forecasting for various time series such as options pricing and demand forecasts and prediction of thrift failures [10]. This paper investigates the development of a Neural Network system to predict the Australian All Ordinaries Index from the explanatory variables (input drivers).

To ensure the non-linear aspect of the problem is catered for the networks have a minimum of three levels. The reader is referred to [8] for an overview of Neural Network theory.

In designing a neural network, there are a number of variable parameters that need to be selected. These include Learning Rate, Momentum, Initial Interconnection Weights (all to do with learning), architecture selection, number of internal neurons in the hidden layer, input data manipulations and data smoothing (We shall explore some of these design parameters later in the next section). The choice of these parameters can greatly influence the network's performance.

5 Network Architectures

As indicated above, there are many parameters that have to be decided upon when developing a network model to predict the All Ordinaries Index. Parameters such as interconnection rate, learning rate, momentum, neurons interconnections, number of hidden neurons, activation functions, architectures, input data and so forth have to be considered.

It is not intended here to discuss all of the above influencing factors. The reader is referred to texts such as [5] and [11] for details. Specifically, two of the most superficial but most important parameters, architecture selection (number of hidden neurons and neurons interconnection will be considered as part of the architecture) and input data will be developed as a basis for the development of the other influencing factors.

Guided by authors such as [6] and the preliminary results of Flitman [4] and Barnes [2] financial modeling applications seemed to be facilitated using tanh activation function, thus we have used that as our benchmark.

Interconnection weights, learning rate and momentum were set to defaulted values and will be kept a constant throughout the development process.

Network Parameters	Value
Interconnection Weights	0.3
Learning Rate	0.1
Momentum	0.1

Table 1.	Constant	Network	Parameters
----------	----------	---------	------------

The main architectures that were considered consisted of between 3 and 5 layers. Layer connection was either standard backpropagation or a modified version with each layer connected to each other (Jump Connection networks).

We conducted a number of structured experiments to investigate Network Architectures and Data Set Size. This can be summarized into a simple flow diagram.



Architecture Selection



Data Size and Period Selection

Fig. 1.

5.1 Determining Network Architectures

The data set used to train the network was kept constant in order to investigate network architectures as well as the number of internal neurons. Determining the size of the network (the number of neurons) has important consequences for its performance [11]. Too small a network may not reach an acceptable level of accuracy. Too many neurons may result in an inability for the network to generalise; it may be rote learning the patterns. The training set consisting of 5 inputs and a single output (109 observations) were applied to all the different network architectures. The data was normalised in the range of [-1,1]. This required knowledge of the expected maximums and minimums for the entire data set including the test and variation set. The scaling was to keep the inputs and outputs to figures relative to its maximum and minimum. This will minimise the effects of input magnitudes as well as aid network learning.

We decided to present the training sets to the network in a random order rather than rotate through the training set, so that bias could be minimized (the network could end up learning based on the position of the training data rather than the data itself).

Each time a number (200) of complete passes (epochs) of the training data passed through the network, we applied the partially trained network to the test data set and recorded the average forecast error. The network parameters that recorded the lowest error would be saved, overwriting previous saved parameters. This, however, lends itself to the risk of saving a network that has memorized the training set without the ability to generate new data.

This process will continue until a number of successive tests using the test data yielded no further improvement. We allowed 300 tests on the test data without any improvements before concluding the results.

Once the training was completed, the network was applied to the production set for validation. The production set was represented by 60 consecutive months of data. We relied entirely on the R square statistic when comparing the network quality.

We decided that the data normalization based on the minimum and maximum values of the data set was far too broad. We therefore decided to normalize based on standard deviations from the mean. Initially the maximum and minimum values for the data set were reset to 2 standard deviations from the mean and the network was retrained using the same architectures as above.

The results of the R square of all networks except for network Run 11, trained by setting the maximum and minimum of the input variables to 2 standard deviation of their mean, have recorded improved results. The best result was produced by the 3-layer jump connection network with 10 internal neurons with an R square of 44.38%. Further exploration found that increasing the number of neurons didn't improve the Neural Networks predictive power.

	Network Architecture	Internal Neurons	R Square Default Min/Max	R Square Min/Max 2 SD
1	3 layer BPN	5	0.0871	0.2051
2	3 layer BPN	10	0.089	0.4382
3	4 layer BPN	5	0.0758	0.1984
4	4 layer BPN	10	0.0332	0.2718
5	5 layer BPN	5	0.066	0.2741
6	5 layer BPN	10	0.0012	0.0744
7	3 layer jump connection	5	0.0557	0.1076
8	3 layer jump connection	10	0.1527	0.4438
9	4 layer jump connection	5	0.0223	0.2798
10	4 layer jump connection	10	0.1048	0.2864
11	5 layer jump connection	5	0.1726	0.1328
12	5 layer jump connection	10	0.0156	0.1134

Table 2. Varying architectures by adjusting the input data (Min/Max)

Results have shown that the maximum and minimum values of the input variables clearly affect the Neural Network's output. We therefore next varied the maximum and minimum values of the inputs in multiples of 2 + - 0.25 standard deviations from the mean. The 3-layer jump connection with 10 neurons (best R Square) was then trained using the varied data:

	Standard Deviations from the Mean	R Square
ир 10)	1.25	0.1794
un un	1.5	0.4256
er . ctic	1.75	0.3950
laye	2	0.4438
3 . Jon	2.25	0.2313
0	2.5	0.1753
	2.75	0.1788

Table 3. Maximum and Minimum of input variables

The best R Squared still remained with the network trained using input data with their maximum and minimum values set to 2 standard deviations from the mean.

At this stage in our experiment we narrowed down the vast parameters in the development of the neural network model for predicting the All Ordinaries Index to the following (based on the above results):

- Network Architecture: 3-layer Jump Connection with 10 Neurons
- *Input Variables Max and Min*: 2 Standard Deviations from the Mean

5.2 Determining the Input Data Set Size and Period

The input data set were figures at the start of the month obtained from May 1997 to May 2001 ([4] used data from May 1987 to October 1993). The data set consists of a total of 169 observations. The data was divided into 3 categories for the Neural Network (training, testing and production sets).

5.2.1 *Expanding the Production Set*: We trained the data set using an expanding production set. The training and test set were kept constant while the production set was being expanded from a period of 12 months (12 observations) up to 60 months in multiples of 12.

	Production Set	R Square
1	12	0.3930
2	24	0.6724
3	36	0.2423
4	48	0.2484
5	60	0.4438

Table 4. Expanding the Production Set

The best result in this exploration is produced by training the 3 layer jump connection with 10 neurons, with a training and test set of 109 observations and a production set of 24 observations. The R Square is 67.24%. The data set was from May 1987 to May 1998.

5.2.2 *Expanding Training and Test Sets*: We applied the former approach to the training and test data. The production set was kept constant at 12 observations while the training set was expanded from 109 observations up till 157 observations in multiples of 12.

	Training Set	Test Set	R Square
1	87	22	0.3930
2	96	25	0.5036
3	106	27	0.2423
4	115	30	0.0290
5	125	32	0.2685

Table 5. Expanding Training and Test Sets

The best result was again obtained using input data from May 1987 to May 1998 with an R Square of 50.36. Therefore with a constant production set size it was found that this technique did not improve the predictive power of the Neural Network. It was noted that the results were significantly worse after the Run 2. This tends to suggest that the increased training and test set size doesn't generally improve the model. To confirm this point a different architecture was tried. The best result again were obtained by training the network with data from May 1987 to May 1998 and the larger training and test sets did not improve its predictive power. This is shown below.

	Training Set	Test Set	R Square
1	87	22	0.0068
2	96	25	0.6709
3	106	27	0.6069
4	115	30	0.0414
5	125	32	0.3743

Table 6. Expanding Training and Test Sets(Cyclical)

5.2.3 *Sliding Data Windows*: Finally, we applied the 'sliding window' approach to the input data. The data set was fixed to a period of 10 years (121 observations in all). All the training set, test set and production sets were constant in size. Only the time periods were varied.

Table 7. Sliding Data Windows

	Data Range	R Square
1	May 1987 – May 1997	0.3930
2	May 1988 – May 1998	0.7569
3	May 1989 – May 1999	0.5250
4	May 1990 – May 2000	0.0038
5	May 1991 – May 2001	0.6140

It was found the sliding window approach gave the best results of the techniques tried in this paper. The best result was found to have an R Square of 75.69% for the forward 12-month forecast of the All Ordinaries Index movement produced by training input data from May 1988 to May 1997 and validating the forecast with data from June 1997 to May 1998. As with each of the previous techniques, Run 4 gave very poor results, so it can be concluded that predicting this period (i.e. the 12 months leading up to May 2000) was very volatile and difficult to make accurate predictions. The model was developed using a 3-layer jump connection with 10 internal neurons with the input data range set to 2 standard deviations from the mean.

6 Conclusions and Further Research

From the investigations conducted, it is clear that the iterative development greatly aided the production of the 'best' Neural Network for the purpose of modeling the Australian All-Ordinaries Index.

It was found that the 3 Layer Jump Connection Neural Network with 10 internal neurons produces the best result in terms of R Square. Results also show that the input

data range of 2 standard deviations (maximum and minimum) along with the sliding window technique produced the best R Square output.

With data manipulation, it was found that different periods of the same data that was used to train the network produced significantly different results. The best R Square (trained using 3 layer feed forward neural network) for the entire experiment was obtained using data from May 1988 to May 1998.

There are of course many other factors that affect the learning ability of a neural network. With the consideration of 'Network Architectures' and 'Data Sizing and Periodicity' already considered, it would be useful to broaden this research to encompass other parameters such as momentum, learning rate and interconnection rate.

Further work is also being undertaken to improve sensitivity by looking at input data of shorter time periods (weekly or daily).

References

- 1. Barnes, M.B., Rimmer, R.J., Ting, K.M. (1999) "A study of techniques for data mining from the Australian Stock Exchange" Proceedings of the Forth World Conference on Systemics, Cybernetics and Informatics, Vol. 8, Part 2, 2000, pp52–57.
- Barnes, M.B., Flitman, A., Ting, K.M., "Australian All Ordinaries Index: Re-examine the Utilities of the Explanatory Variables Using Three Different Measures", Proceedings of the Ninth International Conference on Neural Information Processing, Vol. 5, 2002, pp 2335–2339.
- 3. Chatfield, C. "Neural Networks: Forecasting Breakthrough of Passing Fad?" International Journal of Forecasting, Vol. 9, 1–3.(1993).
- Flitman, A., "A neural network forecaster of the Australian All-Ordinaries", NEORAP'95, pp 189–195.
- 5. Hertz, J., Krogh, A., Palmer, R.G. "Introduction to the Theory of Neural Computation", Redwood City CA, Addison-Wesley (1991)
- 6. Klimasauskas, C.C. "Neural Network Techniques" in Trading on the Edge, New York: John Wiley (1994)
- 7. Leeb, S. "Market Timing for the Nineties", New York: Harper Business (1994).
- 8. Lipperman, R. "An Introduction to Computing with Neural Nets", IEEE. (1987).
- 9. Peters, E. "Chaos and Order in the Capital Markets", New York: John Wiley (1991).
- 10. Salchenberger, L.M., Cinar, E.M., Lash, N.A. "Neural Networks: A New Tool for Predicting Thrift Failures", Decision Sciences Vol. 23 (1992).
- 11. Wasserman, P.D., "Advanced Methods in Neural Computing", Van Nostrand, New York, (1993).