Can a Neural Network Property Portfolio Selection Process Outperform the Property Market?

Executive Summary. *Evidence of the superior performance of portfolios comprised of 'value' stocks over 'growth' stocks is wide and varied. Despite this burgeoning literature, relatively little is known about the comparative performance of property sector value stocks and the performance of neural network techniques in relation to this market sector. This study addresses both of these issues by applying neural network modeling techniques to the Australian property sector stocks to construct a variety of value portfolios. Risk-adjusted performance measures show that the value portfolios outperform the market by as much as 7.14%.*

by Craig Ellis*
Patrick J. Wilson**

Introduction

Artificial Neural Networks (ANNs) have a long development history, having been studied since Rosenblatt first applied single layer perceptrons to pattern-classification learning in the late 1950s (Kantardzic, 2003). Despite this long history, it has only been since the advent of faster computers that ANNs have gained wider acceptance. In ANNs, the network is presented with data repeatedly, from which it extracts the key relationships underlying the data. A valuable aspect of neural networks is that they are well suited to deal with unstructured problems, inconsistent information, missing data, and real-time output (Hawley, Johnson and Raina, 1990). There has also been a growing interest in the use of ANNs in finance and economics because of their capacity to imitate nonlinear relationships that are otherwise difficult to identify and also because the use of ANNs require no assumptions about the distribution of the underlying data.

The aim of this research is to ascertain whether it is possible to develop a neural network model capable of building a property investment portfolio that will outperform the market. While the broad objective is to ascertain whether ANNs represent a viable practical tool for portfolio allocation, a more specific objective is to determine whether an investment platform established by a neural network model provides a suitable basis for 'what if' type simulations. While it is generally accepted that ANNs are not capable of being used for policy

*University of Western Sydney, Penrith South DC NSW 1797, Australia or c.ellis@uws.edu.au.

**University of Technology, Broadway NSW 2003, Sydney, Australia or patrick.wilson@uts.edu.au.

analysis (Brooks and Tsolacos, 2003), there is no reason to assume they cannot be used for 'what if' simulations and planning for contingent scenarios generated by changes in the input variables. However, before such a step can be taken it is necessary to answer a basic question: Can a neural network constructed portfolio outperform the market?

While much is already known about the relative performance of value stocks in general, relatively little has been written about property sector value stocks specifically. Likewise, little is known about the performance of neural network techniques in relation to this market sector. Unlike most other industry sectors, property sector stock's primary income is derived from property investments and rental returns. Property trust units are underwritten over the long-term by their corresponding underlying direct property assets, yet over the short-term property trust units often behave in a similar manner to stocks and very much unlike direct property (Webb, Seiler and Meyer, 1999). If this is the case, then the question naturally arises as to the extent to which units in property trusts have comparable concepts of value and growth to those that exist in the stock market (*i.e.*, value stocks vs. growth stocks), and the extent to which portfolios composed of value units will outperform the property market.

Following from Eakins and Stansell (2003), a select group of fundamental financial ratios will be used to determine the set of 'value' assets to enter the property portfolio. Not surprisingly other research has used one or more such fundamental variables in conventional time series models to predict stock market returns. For instance, Kothari and Shanken (1997) in a study on stock market returns over the period 1926 to 1991 found that both book-to-market and dividend yield tracked time series variations in expected real stock returns. In the current research, these variables will form the inputs to a neural network model that will have preset limits determined by the literature as indicated later to isolate 'value' from 'growth' assets. Value assets are commonly defined as those whose market value is lower than their intrinsic or liquidating value (O'Shaughnessy, 1998:2). The attraction of value assets from an investor's point of view is that the (lower) market value of the asset should rise to meet the (higher) intrinsic value. This being true, portfolios comprised of value assets only, should outperform portfolios comprised of all assets. Value stocks are typically characterized by low market price per share, low cash flow per share, or low book value per share.

Portfolio allocations to value stocks is sometimes referred to as a contrarian investment strategy and a number of studies have indicated that such a strategy has outperformed the strategy of investing in 'growth' stocks (see Fama and French, 1996, 1998; Haugen, 1999; Lakonishok, Shleifer and Vishny, 1994; Levis and Liodakis, 2001; and Yen, Sun and Yan, 2004 as indicative studies). Using stochastic dominance tests, Best, Best and Yoder (2000) confirmed the dominance of value portfolios comprised of high book-to-market stocks over low book-to-market portfolios. These findings all contradict the basic tenets of the efficient markets hypothesis (EMH). Others, however, suggest that the higher returns to value stocks are simply compensation for higher risk (Chen and Zhang, 1998; and Davis, Fama and French, 2000), or that growth stocks actually outperform value stocks in the long run (Beneda, 2002). Given this wealth of evidence, this paper seeks to answer two questions: can a neural network construct a 'value' portfolio, and will this contrarian strategy outperform the market?

The remainder of the paper is structured as follows. The following section briefly describes the basic architecture of the neural network process and the specific activation function that is used in this study. The next section outlines the relevant neural network literature and provides a context to the current study. The following sections describe the research methodology employed, the data and sample used, the empirical findings and the conclusion.

Overview of the Neural Network Process

The processing units in a neural network are called artificial neurons (or processing nodes) in reference to the processing neurons in the human brain. The basis of the ANN is to try and re-create the parallel processing power of the brain using generalized nonlinear, nonparametric models based on the brain and nervous system. In the neural network, each generic neuron receives n input signals, or variables, from the units to which it is connected. Each variable is weighted by some factor and the result is passed through to a transfer (or activation) function that produces the output from the neuron.

More formally, lets suppose there are several inputs (independent variables) x_i, $i = 1, \ldots, n$ each with a corresponding weight w_{ij} where j is the index of a given neuron. These weights simulate the synaptic strengths in a natural neuron,[1] with the initial weights usually being randomly assigned within a pre-specified range (the weights are somewhat akin to the coefficients in a regression model). Sometimes there is also an externally applied bias (equivalent to an intercept in a regression model) that has the effect of lowering or increasing the net input of the activation function. This output is sometimes called the potential or the net of the neuron and is given by Giudici (2003) as:

$$P_j = \sum_{i=1}^{n} w_{ij}x_i = net_j. \tag{1}$$

An activation function f, which exists in what is called a hidden layer, is now applied to net_j to produce the output:

$$y_j = f(P_j) = \sum_{i=1}^{n} w_{ij}x_i. \tag{2}$$

Within the hidden layer there may be numerous neurons all working on the same problem (parallel processing). If there is more than one layer, this initial output is passed to the next layer (containing fewer neurons) and, again, weights are assigned to this new input and another transfer function activates until, ultimately, a final output is produced. Exhibit 1 provides a schematic representation of a typical network architecture.

While various forms of activation function can be defined, the most commonly used is the hyperbolic tangent (tanh) given by:

$$f(P_j) = \frac{e^{P_j} - e^{-P_j}}{e^{P_j} + e^{-P_j}}. \tag{3}$$

Sigmoid functions like the hyperbolic tangent are widely used as they are non-linear, easily differentiable and not unlike the smooth transition autoregressive processes developed by Terasvirta (1994). An advantage of using a nonlinear as opposed to a linear transfer function is that since linear independence of the input patterns is not required, a wider range of problems can be tackled (Coakley and Brown, 2000).

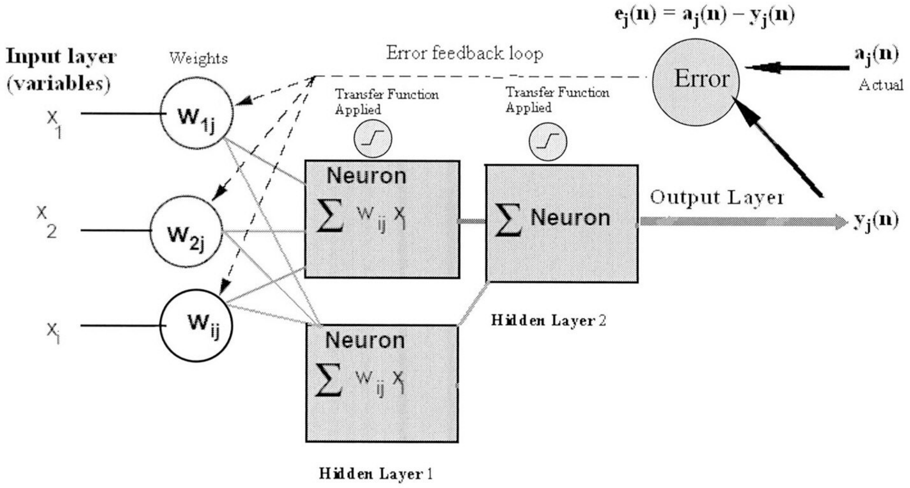
Conventionally there will be more than one layer in a network and in each layer there will be several neurons with different weights operating on a problem. The output from each layer is passed through to the next layer and subjected to further weighting and activation functions. This is an example of a feedforward multi-layered perceptron. Feedforward systems with backpropagation learning provide the basis for over 90% of commercial and industrial applications of ANNs (Kantardzic, 2003) and over 80% of all problems are trained using backpropagation with three layers: input, hidden and output (Yu, 1999).

Backpropagation is the technical term given to the error minimization process. As in a conventional statistical sense, the error is the difference between the actual (expert) response and the predicted (neural network) response viz:

$$e_j(n) = a_j(n) - y_j(n). \tag{4}$$

The j^{th} neuron produces output y_j and this is compared with the actual output, a_j, which is obtained (theoretically) from the same set of n inputs. The error functions normally employed are

Exhibit 1
Architecture of an Artificial Neural Network with Backpropagation



Based on diagrams in Tan (2001).

based on the maximum likelihood principle. If the error meets the desired goal (*e.g.*, 5% error) the final output is produced, if not then there is a closed feedback loop that sends small adjustments back to the weights and the system re-calculates. Iterations continue until either the error criteria are satisfied, or the number of iterations exceeds some pre-set limit. The procedure is similar to the weight optimization process in exponential smoothing (that for instance minimizes the mean square error), but on a much more complex scale.

As shown above, every component a_{ij} of the response vector is assumed to be the sum of a deterministic term and an error term. To extract more information, the error terms are assumed normally distributed. Giudici (2003) shows that the error function for minimization can be written as:

$$E(w) = \sum_{i=1}^{n} \sum_{j=1}^{q} (a_{ij} - y_{ij})^2, \tag{5}$$

which is minimized using a gradient descent method.

Shachmorove and Witkowska (2000) point to four distinct advantages that neural networks have over conventional statistical methods of data analysis viz: they make no assumptions about the distribution of the data under analysis; they are excellent at discovering nonlinear relationships; they perform very well with missing or incomplete data; and they can produce forecasts more quickly than, say, an econometric model. Qi (1999) notes that a clear advantage of neural networks over econometric models is that the ANNs can approximate any linear or nonlinear function to an arbitrary degree of accuracy. However, a major weakness of ANNs compared with conventional statistical methods is their lack of intuitive explanation for the models that they create.

Neural networks were developed, and have been used more extensively, in non-business environments such as medicine, engineering, geology, and physics and there is an extensive literature in these areas. The next section reviews the use of ANNs in a business environment.

Literature Review

The superior performance of value stocks versus growth stocks has been well documented (*e.g.*, Fama and French, 1996, 1998; Haugen, 1999; Lakonishok, Shleifer and Vishny, 1994; Levis and Liodakis, 2001; and Yen, Sun and Yan, 2004) and does not require repeating here. Given the focus of this paper on the ability of ANNs to identify value assets, the following review will rather concentrate on the relevant neural network literature.

Stock Selection and Portfolios

Kryzanowski, Galler and Wright (1993) used a neural network approach to pick stocks. These authors found that the neural network correctly classified 72% of the positive/negative stock returns. Wong, Wang, Goh and Quek (1992) outline an intelligent stock selection system that extends the neural network approach to handle fuzzy, probabilistic and Boolean information. Trippi and DeSieno (1992) modeled the trading of S&P 500 Index futures using a neural network and found the network outperformed a passive investment system. Shachmurove and Witkowska (2000) compared the international transmission of stock market information using a linear regression model and a neural network using a logistic function as their sigmoidal transfer function. They found that the neural network performed better at predicting daily stock returns than traditional Ordinary Least Squares (OLS) and general linear regression models. Phua, Ming and Lin (2001) used a neural network in attempting to predict the Singapore Stock Exchange Straits Times Index. These authors achieved an accuracy of 81% in predicting market direction but their network performed poorly in picking price reversals.

Property Market Applications

Applications of neural networks in the property market have largely been limited to property appraisals. Borst (1991) examined the usefulness of a feedforward, backpropagation neural network in predicting selling prices for unsold real estate. Outcomes were compared with those of a multivariate linear model used in mass appraisal. Borst found the results suggested that neural networks should be given strong consideration by the assessment community. Tay and Ho (1992) similarly used a backpropagation neural network in mass appraisal of residential apartments in Singapore and compared the outcome with appraisals based on a multiple regression. The authors found neural networks to be a good alternative to multiple regression in mass appraisal. Similarly Do and Grudnitski (1992) investigated the feasibility of using a feedforward neural network approach to residential property appraisal based on market transaction data and compared this with a multiple regression approach. The authors found the neural net approach avoided the collinearity, heteroscedasticity and other problems endemic to conventional statistical approaches to mass appraisal. They found the outcomes from neural networks (*i.e.*, property value) to be nearly twice as accurate as those of a multiple regression model.

Nguyen and Cripps (2001) used a feedforward/backpropagation neural network with one hidden layer in their comparison of predictive accuracy against a multiple regression in forecasting housing value in Rutherford County, Tennessee. They found that neural nets perform better than regression when a moderate to large data sample size is used. In a study of housing in the United Kingdom, Wilson, Paris, Ware and Jenkins (2002) use neural networks to forecast future trends within the housing market. Their architecture was a feedforward/backprogagation with two hidden layers using a hyperbolic tangent transfer function. Their analysis suggested that useful short-term (two year) forecasts (1.8% error) could be produced using the neural network. In a study on the comparative performance of statistical models and commonly used financial indicators for forecasting securitized real estate returns, Brooks and Tsolacos (2003) suggested that analysts should exploit the potential of neural networks and assess more fully their forecast performance against more traditional models.

Both Worzala, Lenk and Silva (1995) and McGreal, Adair, McBurney and Patterson (1998) adopted a more skeptical approach to the potential merits of neural networks in the valuation process. In a study based on 288 sales in Fort Collins, Colorado, Worzala et al. found that their results did not support previous research on the superiority of neural networks over regression in appraisal analysis. In a comparison using two different software packages, these authors found that the two packages alternated in their dominance over regression. Using data from the Belfast area in Northern Ireland, McGreal et. al. found the neural networks show that only 80% of properties achieve a predicted value within 15% of sale price—an outcome they claim would be unacceptable to the valuation profession. On the other hand, Connellan and James (1998) used neural networks to forecast commercial property values in the U.K. with what they called "reasonable accuracy."

Forecasting

Donaldson and Kamstra (1996) studied the outcome of combining forecasts of stock market volatility across a range of countries. These authors show that combining forecasts with nonlinear neural networks produces forecasts that routinely dominate forecasts from traditional linear techniques such as moving average variance and GARCH models.

In an interesting experiment, Hill, O'Connor and Remus (1996) compared the outcomes from the original Makridakis forecasting competition (Makridakis et al., 1982) with the outcomes using a neural network on 'level ground' (*i.e.*, the experiment was conducted as if the authors were part of the original competition). The neural network was found to perform significantly better than traditional time series methods when forecasting with monthly and quarterly data, and was comparable with annual data. The authors suggested that the crucial reason for the superior performance was the ability of neural networks to better cope with discontinuities.

In a study of the efficacy of neural networks in predicting returns on stock and bond indices, Desai and Bharati (1998) found that the neural network forecasts were conditionally efficient with respect to linear regression models for large stocks and corporate bonds, but there was no significant difference for small stocks and intermediate-term government bonds. In a study with mixed results, Narain and Narain (2002) compared stock market predictions using a multivariate statistical (MVS) model and a neural network. These authors found that the MVS model was able to predict the S&P 500 and NASDAQ indices just as well as the ANN, while the ANN model was able to predict the DJIA better than the MVS.

Walczak (2001) considered the question of the optimum amount of data to train a network model for financial forecasting. The author examined three common exchange rates: dollar/pound, dollar/mark and dollar/yen. He found that neural networks trained on a larger training set have a worse forecasting performance. Yu (1999) used a three-layer, backpropagation network in a comparison of the forecasting performance of a neural network and a conventional ARIMA model and found the neural network outperformed the ARIMA model in forecasts of the Nikkei Stock Index futures. Similarly, Kanas (2001) used a three-layer network in a forecast comparison with a conventional linear model. There were six neurons in the hidden layer of the neural network while the linear model used lagged percentage change in trading volume and lagged percentage change in dividends to predict stock returns for the Dow Jones and the FTSE. Kanas found that the neural network forecasts were preferable to the linear model in that the network could explain the forecast errors of the linear model, but not vice-versa. This indicated that the inclusion of nonlinear terms in the functional relationship between returns and explanatory factors is important in forecasting.

Research Methodology

The objectives of the research in this study are twofold. First, the performance of neural networks and their ability to identify 'value' stocks is examined from the set of all property sector stocks (individual property trusts) listed on the Australian Stock Exchange (ASX). Once identified, an evaluation of the performance of portfolios comprised of property sector value stocks versus the set of all property sector stocks is conducted. Portfolio performance relative to the market index is measured by the Sharpe ratio (Sharpe, 1966) for risk-adjusted returns, and the Sortino procedure (Sortino and Forsey, 1996; and Sortino, Miller and Messina, 1997) for adjusting returns on a downside risk basis.

In a comprehensive study of the comparative performance of listed stocks in the United States, O'Shaughnessy (1998) identified the following factors as being determinants of 'value': large stocks with low price/earnings ratios; low price/book ratios; large stocks with low price/cashflow ratios; low price/sales ratios; and large stocks with high dividend yields. 'Large' stocks are defined by O'Shaughnessy as those with a higher than average market capitalization. Given the lower volatility of large stocks relative to all stocks, value portfolios comprised of large stocks are shown by

O'Shaughnessy to typically outperform the market index by a sizeable margin in risk-adjusted terms. As per O'Shaughnessy, low (high) ratios are herein defined as those that are lower (higher) than the market average ratio (*e.g.*, low P/E = lower than market average P/E, high P/E = higher than market average P/E).

Several neural network models are tested in this study, including both single-variable and multiple-variable models. Following from O'Shaughnessy (1998) and Eakins and Stansell (2003), the set of input variables employed includes Market Capitalization (MV), Dividend Yield (DY), Price-to-Book-Value (PTBV), Price-Earnings (PE) and Price-to-Cashflow (PC). Both the single-variable models and multiple-variable models contain one output variable and are tested using two different node types: binary and linear. Using the binary node type, the output value takes on one of two predetermined values, while for the linear node type, the output value is any real number.[2] In the present analysis, the different node types are used as a cross check with the linear node type output adjusted, as described later. While the number of nodes at the input and hidden layer levels will be determined by the previous analyses, the activation function is pre-set as a hyperbolic tangent (tanh) (see Equation 3) since other research on evaluating the forecast performance of ANNs has shown that this transfer function has faster convergence than other transfer functions (Coakley and Brown, 2000).[3]

A total of 400 observations from January 1990 to February 1997 are used to train each neural network model. A further 1,548 observations (March 1997 to November 2003) are used to test the neural network output. Observations for all stocks are initially ranked by date. As not all stocks traded for the full sample period (1990–2003), this avoids any survivorship bias, which may influence the results. In order to avoid the effects of pattern bias during the neural network training phase, observations for each stock are date scrambled using a random number generator. Since each stock is tested against the value criteria on a monthly basis, value stocks in one time period are not necessarily value stocks in the next period, or in subsequent periods. To avoid look-ahead bias associated with making investment decisions based on data that is not yet known, portfolios constructed at time $t = 1$ comprise stocks that are identified by the neural network as being value stocks at time $t = 0$.

Single-Variable Neural Network Training and Testing

Single-variable neural network models in this study include a single input variable only. Individual neural networks are trained and tested for each of the five input variables listed above. Using the binary node type, the output variable for the single-variable neural network is defined as either 'VALUE' or 'NOT' according to the value criteria that was previously established for each respective input [*e.g.*, low P/E (*i.e.*, less than market average)] = VALUE, high P/E (*i.e.*, greater than or equal to the market average) = NOT). The output of the trained neural network using the binary node type is therefore also a binary of the type VALUE or NOT. Next, setting the training phase node type to linear, the output variable is now set to either 1 or 0, where, according to the relevant value criteria for each input variable, an output value of 1 corresponds to a value stock and a value of 0 to a non-value stock. Unlike the binary node type, which returns only binaries as output, the linear node type returns real values approximately $\pm$ 0.1 standard deviations above or below 0 and 1. These outputs are then rounded to the nearest whole value (0 or 1 in the case of the single-variable neural network model). Repeating the process for each of the five input variables yields a total of ten single-variable value portfolios.

Multiple-Variable Neural Network Training and Testing

As opposed to the single-variable neural network (NN) models that classify stocks as being value or non-value on the basis of a single criteria (*e.g.*, low price/book, *or* low P/E, *or* high dividend yield, etc.), the multiple-variable criteria ranks stocks in terms of the degree of value when measured against all criteria simultaneously (*e.g.*, low price/book, *and* low P/E, *and* high dividend yield, etc.). Stocks are given a score out of five in each period based on how many of the individual value criteria are satisfied. As for the single-variable NN models, the output of the multiple-variable linear node model is rounded to the nearest whole value prior to the stocks being ranked. A value score of 5 in any given period shows that the stock satisfied all of the value criteria; a value score of 0 alternatively shows that the stock satisfied none of the criteria. Using the multiple-variable value criteria, 'multi-value' stocks are then defined as those with a cumulative value score ≥ 4 since it was decided this would clearly avoid any fine line transition in stocks scoring between 2.5 and 3.5 prior to rounding. The testing of multi-value stock portfolios in addition to value stock portfolios (based on a single value criteria only) is to evaluate the potential value-added by imposing a stricter value criteria on stocks during each period.

Binary and linear neural network node types are used in the multiple-variable context in the same way as for the single-variable models. For the linear node type, stocks are scored from 0 to 5 and multi-value stocks are defined as those with a cumulative value score ≥ 4. Stocks with a cumulative value score ≥ 4 are similarly classified as 'VALUE' in the binary model. Binary node type output for the multiple-variable neural network model is unchanged from the single-variable models previously discussed. The output for the linear node type using the multiple-variable criteria is a set of real values $\{1, 2, 3, 4, 5\} \pm$ approximately 1 standard deviation. Two multi-value stock portfolios comprising stocks with a predicted cumulative value score ≥ 4 are tested.

Data and Sample

The data utilized in this study comprises nominal monthly values for stocks listed on the Australian Stock Exchange (ASX) from January 1990 to November 2003 inclusive—a total of 168 months. For each company in the sample, the following data has been collected: Closing Price (P), Dividend (D), Market Capitalization (MV), Dividend Yield (DY), Price-to-Book-Value (PTBV), Price-Earnings (PE) and Price-to-Cashflow (PC). Closing prices and dividends for each company are used to calculate the total return to the stock. The remaining above listed variables comprises the set of inputs used in training and testing the neural network. For the purposes of training and testing the neural network, the sample is divided into two parts. The training period comprises observations from January 1990 to February 1997, and the out-of-sample (test) period, from March 1997 to November 2003 inclusive.

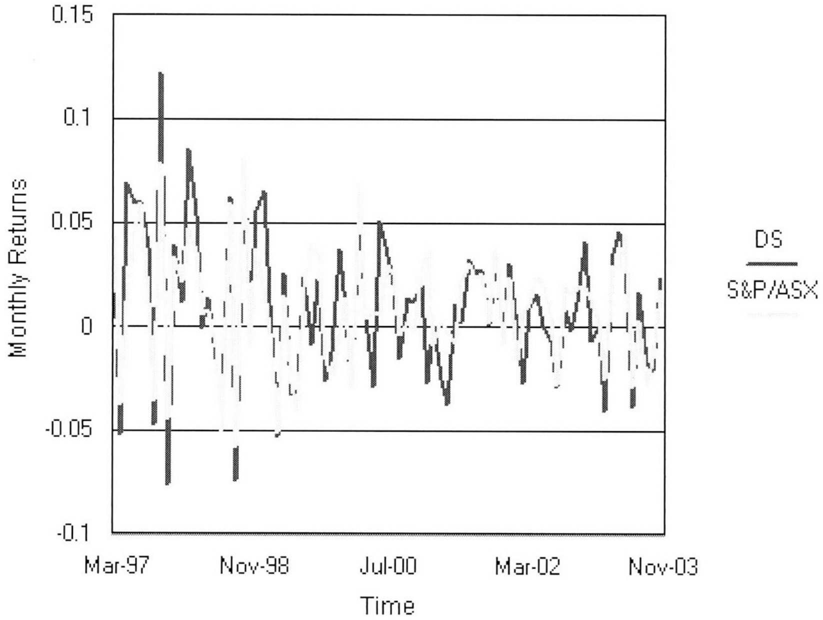
The stocks selected are those listed in the Datastream Australian Real Estate Index (DS Real Estate Index)[4] and the S&P/ASX 300 Property Trusts Index. The DS Australian Real Estate Index comprises the top 80% of property sector stocks in the Australian market. Similar to the composition of the S&P/ASX 300 Property Trusts Index, stocks included in the Index own property and derive their income from rental returns. As at November 2003, the DS Australian Real Estate Index comprised twenty-four companies and the S&P/ASX 300 Property Trusts Index twenty-eight companies. These two indices are used as the proxies for the market index, against which the performance of the neural network portfolio is compared. Market Capitalization (MV), Dividend Yield (DY), Price-to-Book-Value (PTBV), Price-Earnings (PE), Price-to-Cashflow (PC) and Total Return (TR) data is also collected for each of the two market indices. Index total return represents the cumulative points gained or lost due to changes in the share price(s) of stocks in the Index and normal dividend payments. The employment of two separate proxies for the market index in this study allows for comparisons of the findings against alternative benchmark portfolios. The choice of property sector indices as proxies for the market portfolio, as opposed to the broader market index (*e.g.*, S&P/ASX All Ordinaries Index) is consistent with the focus of this paper on the relative performance of real estate stocks. Exhibit 2 graphs the monthly returns for both indices over the out-of-sample period, demonstrating a close, although not perfect, relationship between the two series.

Results

Descriptive Statistics

Summary statistics pertaining to monthly returns for the DS Australian Real Estate Index and the

Exhibit 2
DS Real Estate Index Monthly Returns versus S&P/ASX 300 Property Index Monthly Returns



S&P/ASX 300 Property Trusts Index are provided in Exhibit 3. The DS Australian Real Estate Index started the out-of-sample period at 1815.69 and finished on November 2003 at 6727.35. The S&P/ASX 300 Property Trusts Index started at 6225.1 and finished at 20386.95. The mean return to both indices is approximately 0.95%.

To establish the degree of randomness in monthly returns for the indices, a non-parametric runs test is conducted. The runs test P-value is recorded where, for α levels $> P$-value, the data is not random. This test accepts randomness for both series. Additionally, Anderson-Darling tests for the normality of returns fail to reject the null hypothesis that data for both series follow a normal distribution. Analysis of the correlation of monthly returns finally confirms the significant relationship between the two series.

Exhibit 3
Summary Statistics

	DS Real Estate	S&P/ASX 300 Property
Value March 1993	1815.69	6225.1
Value November 2003	6727.35	20386.95
Points Gain	4911.66	14161.85
Mean Return	0.00954	0.00946
Std. Dev.	0.03685	0.02917
Skewness	0.16539	−0.05265
Kurtosis	0.31345	0.04695
Median	0.01120	0.01148
Min.	0.07630	−0.05742
Max.	0.12220	0.08061
Range	0.19850	0.13804
Sum	0.77310	0.76627
Normality Test (Anderson-Darling) (P-value)	0.941	0.505
Runs Test (P-value)	0.2135	0.5841
Correlation S&P/ASX 300 Property	0.8365	
(P-value)	0.000	

Neural Network Portfolios

The performance of each of the neural network portfolios relative to the market indices is shown in two tables. Exhibit 4 details the performance of the binary output node type neural network portfolios (called simply the binary model) and Exhibit

Exhibit 4

Neural Net Binary Model Portfolio Returns versus DS Australian Real Estate Index Return and S&P/ASX 300 Property Index Return

	Market Index	MV	DY	PE	PTBV	PC	Multi-Value
Panel A: DS Australian Real Estate Index Return							
Mean Return (%)	0.95	6.12	8.04	7.04	6.72	6.43	7.93
Std. Dev. of Return	0.0366	0.0396	0.0315	0.0303	0.0279	0.0252	0.0334
Portfolio Return-Index Return (%)		5.16	7.09	6.09	5.76	5.48	6.97
z score (difference)		8.61	13.21	11.52	11.26	11.08	12.66
P-value		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Risk-free (90-day dealer rate) (%)	5.14						
Sharpe Ratio	−1.1429	0.2467	0.9228	0.6266	0.5650	0.5111	0.8339
Portfolio Sharpe-Market Sharpe		1.3896	2.0657	1.7695	1.7079	1.6540	1.9768
z score (difference)		8.84	13.15	11.26	10.87	10.53	12.58
P-value		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Downside Deviation	0.0548	0.0220	0.0089	0.0119	0.0121	0.0119	0.0096
Sortino Ratio	−0.7644	0.4452	3.2686	1.6047	1.3027	1.0852	2.9082
Portfolio Sortino-Market Sortino		1.2096	4.0330	2.3691	2.0671	1.8496	3.6726
z score (difference)		5.66	10.07	8.06	7.75	7.48	9.31
P-value		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Panel B: S&P/ASX 300 Property							
Mean Return (%)	0.95	6.12	8.04	7.04	6.72	6.43	7.93
Std. Dev. of Return	0.0290	0.0396	0.0315	0.0303	0.0279	0.0252	0.0334
Portfolio Return-Index Return (%)		5.17	7.10	6.10	5.77	5.48	6.98
z score (difference)		9.48	14.93	13.07	12.91	12.84	14.2
P-value		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Risk-free (90-day dealer rate) (%)	5.14						
Sharpe Ratio	−1.4468	0.2467	0.9228	0.6266	0.5650	0.5111	0.8339
Portfolio Sharpe-Market Sharpe		1.6934	2.3696	2.0734	2.0117	1.9579	2.2807
z score (difference)		10.78	15.08	13.19	12.8	12.46	14.51
P-value		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Downside Deviation	0.0507	0.0220	0.0089	0.0119	0.0121	0.0119	0.0096
Sortino Ratio	−0.8266	0.4452	3.2686	1.6047	1.3027	1.0852	2.9082
Portfolio Sortino-Market Sortino		1.2718	4.0953	2.4313	2.1294	1.9118	3.7348
z score (difference)		6.05	10.27	8.34	8.07	7.83	9.51
P-value		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

5 the linear output node type neural network portfolios (called the linear model). Returns to each of the neural network portfolios are compared to both the DS Australian Real Estate Index and the S&P/ASX 300 Property Trusts Index.

Mean returns to each of the single-variable neural network models in Exhibit 4 are significantly greater than mean returns to either of the DS Real Estate or S&P/ASX 300 Property indices. The highest mean return is for the dividend yield value portfolio (8.04%) and the lowest is for the market capitalization value portfolio (6.12%). Analysis of z scores and *P*-values for the difference between mean returns to the market portfolios and single-variable neural network portfolios confirms the significance of these results. Mean returns attributable to the multi-value portfolio are also significantly greater than mean returns to both market indices. Exhibit 6a and Exhibit 6b respectively depict the difference between the multi-value (binary) portfolio returns and the DS Australian Real Estate Index and S&P/ASX 300 Property returns over the test period (excess return over market).[5]

Compared to mean returns for each of the single value portfolios, *P*-values for the difference between the multi-value portfolio mean return and single-variable mean returns confirm that the multi-value portfolio return is significantly greater

Exhibit 5
Neural Net Linear Model Portfolio Returns versus DS Real Estate Index Return and S&P/ASX 300 Property Index Return

	Market Index	MV	DY	PE	PTBV	PC	Multi-Value
Panel A: DS Australian Real Estate Index Return							
Mean Return (%)	0.95	6.18	8.03	7.10	6.72	6.45	8.08
Std. Dev. of Return	0.0366	0.0392	0.0316	0.0302	0.0279	0.0252	0.0338
Portfolio Return-Index Return (%)		5.23	7.08	6.15	5.76	5.49	7.13
z score (difference)		8.77	13.17	11.66	11.26	11.12	12.88
P-value		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Risk-free (90-day dealer rate) (%)	5.14						
Sharpe Ratio	−1.1429	0.2653	0.9150	0.6509	0.5650	0.5186	0.8711
Portfolio Sharpe-Market Sharpe		1.4082	2.0579	1.7938	1.7079	1.6615	2.0140
z score (difference)		8.96	13.1	11.42	10.87	10.57	12.82
P-value		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Downside Deviation	0.0548	0.0213	0.0090	0.0116	0.0121	0.0112	0.0090
Sortino Ratio	−0.7644	0.4872	3.2262	1.6885	1.3027	1.1632	3.2532
Portfolio Sortino-Market Sortino		1.2515	3.9906	2.4529	2.0671	1.9276	4.0176
z score (difference)		5.76	10.01	8.24	7.75	7.41	9.53
P-value		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Panel B: S&P/ASX 300 Property							
Mean Return (%)	0.95	6.18	8.03	7.10	6.72	6.45	8.08
Std. Dev. of Return	0.0290	0.0392	0.0316	0.0302	0.0279	0.0252	0.0338
Portfolio Return-Index Return (%)		5.23	7.09	6.16	5.77	5.50	7.14
z score (difference)		9.66	14.87	13.25	12.91	12.89	14.43
P-value		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Risk-free (90-day dealer rate) (%)	5.14						
Sharpe Ratio	−1.4468	0.2653	0.9150	0.6509	0.5650	0.5186	0.8711
Portfolio Sharpe-Market Sharpe		1.7121	2.3618	2.0976	2.0117	1.9654	2.3179
z score (difference)		10.9	15.03	13.35	12.80	12.51	14.75
P-value		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Downside Deviation	0.0507	0.0213	0.0090	0.0116	0.0121	0.0112	0.0090
Sortino Ratio	−0.8266	0.4872	3.2262	1.6885	1.3027	1.1632	3.2532
Portfolio Sortino-Market Sortino		1.3138	4.0529	2.5152	2.1294	1.9899	4.0798
z score (difference)		6.15	10.21	8.52	8.07	7.74	9.72
P-value		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

than all portfolios, with the exception of the dividend yield value portfolio.[6] Differences in returns between the multi-value and dividend yield value portfolios are not significant for the binary model (*P*-value = .8259).

Risk-adjusted returns in this study are calculated using the Sharpe Ratio and the Sortino Ratio. The Sharpe Ratio measures the difference between the portfolio return and the risk-free rate, and is standardized by the standard deviation of portfolio returns. The Sortino Ratio is calculated as the difference between the portfolio return (R_p) and a minimum acceptable return (MAR), divided by the downside deviation (DD) of the portfolio return versus the minimum acceptable return (DD_{MAR}). Downside deviation is similar to loss standard deviation with the exception that it (DD) only includes portfolio returns below the MAR, rather than portfolio returns below the mean. The basis of the Sortino Ratio is that investors are more concerned with the risk of loss (downside risk), than the risk of gains (upside risk). Standard deviation, as used by the Sharpe Ratio, considers both upside and downside risk. The calculation of the Sortino Ratio is given in the Appendix in Equation (A1). For consistency, the Sortino Ratio MAR is set equal to the risk-free rate of 5.14%, this being the mean of the Australian 90-day dealer rate for the test period.

Exhibit 6a
Neural Net Multi-Value Binary Model Property Portfolio Return less DS Australian Real Estate Index Return

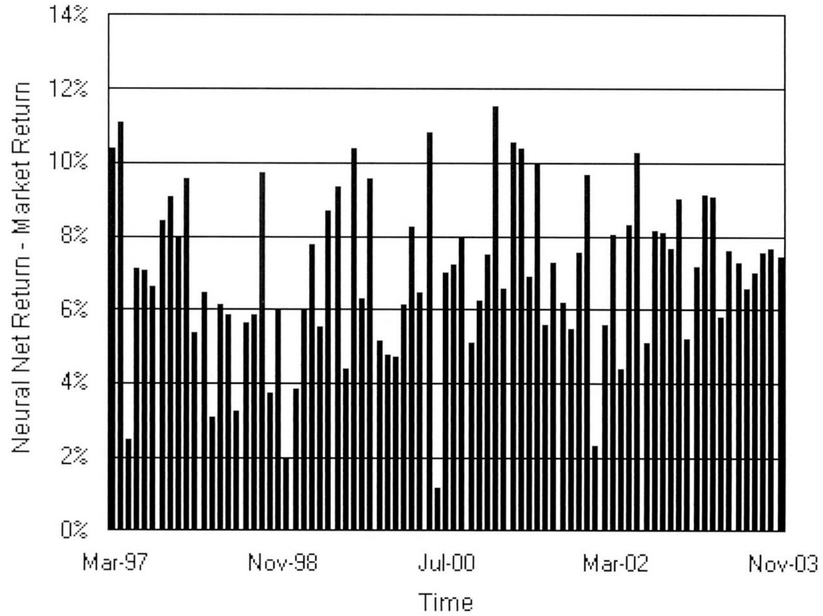


Exhibit 6b
Neural Net Multi-Value Binary Model Property Portfolio Return less S&P/ASX 300 Property Index Return

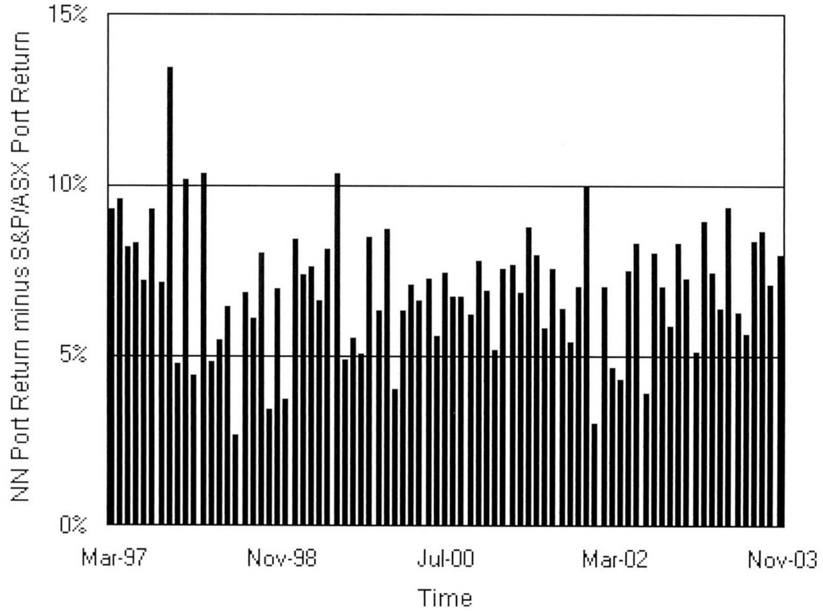


Exhibit 7a
Neural Net Multi-Value Binary Model Sharpe Ratios versus DS Australian Real Estate Index Sharpe Ratios

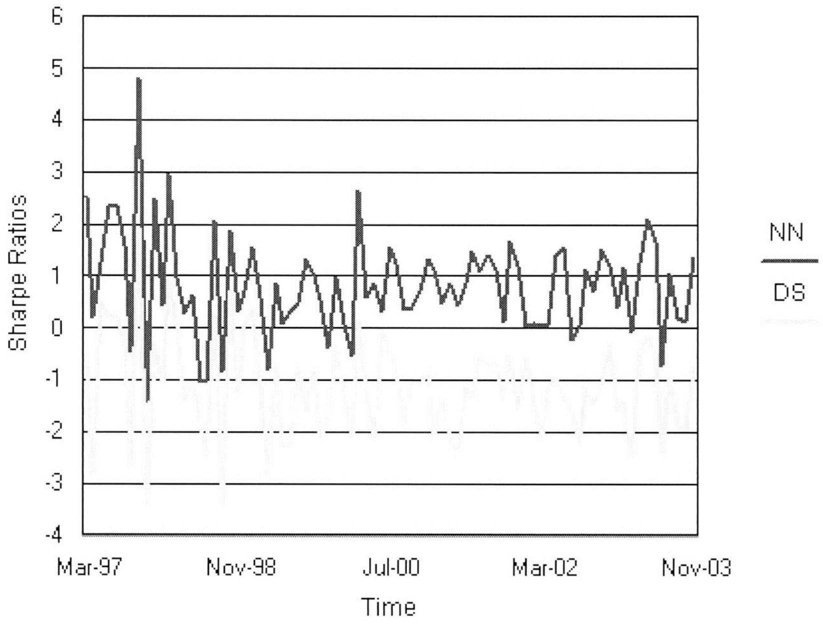


Exhibit 7b
Neural Net Multi-Value Binary Model Sharpe Ratios versus S&P/ASX 300 Property Index Sharpe Ratios

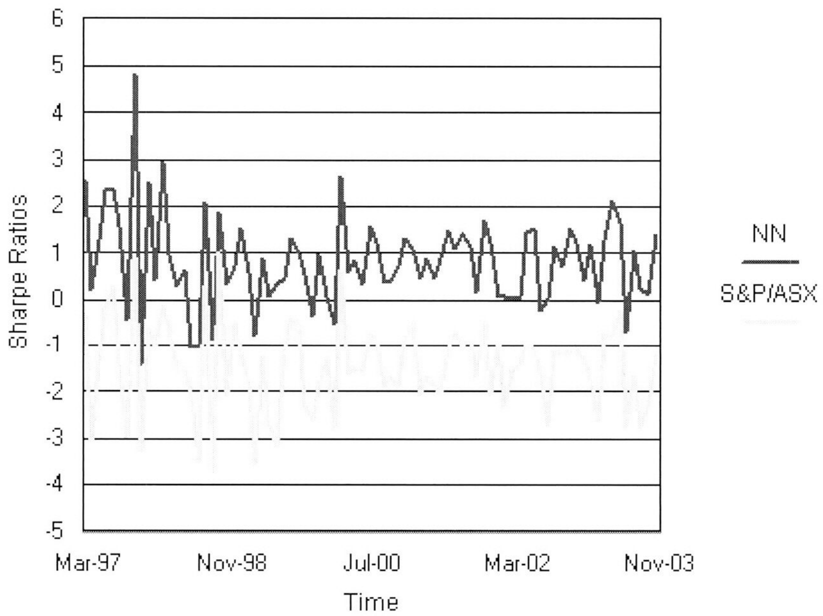


Exhibit 8a

Neural Net Multi-Value Binary Model Sortino Ratios versus DS Australian Real Estate Index Sortino Ratios

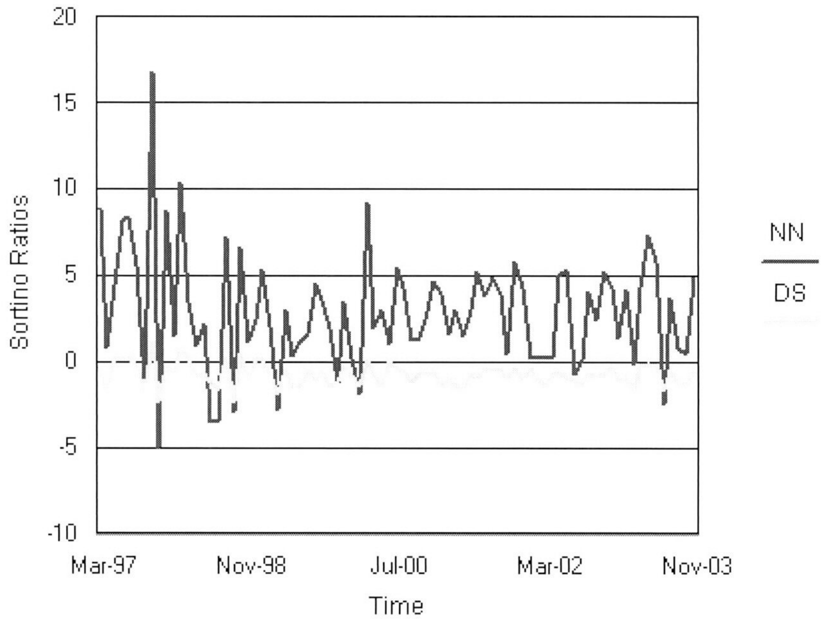
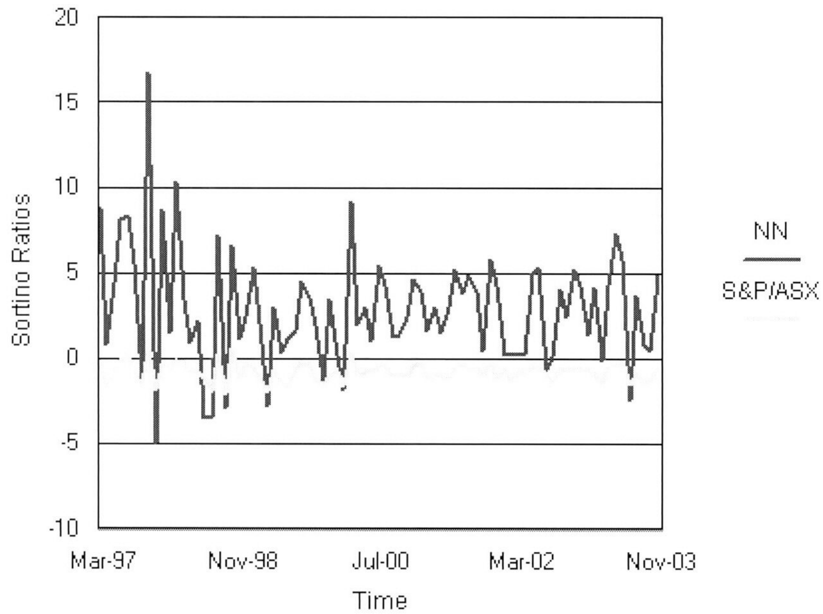


Exhibit 8b

Neural Net Multi-Value Binary Model Sortino Ratios versus S&P/ASX 300 Property Index Sortino Ratios



Analysis of z scores and P-values for the difference between the mean returns to the market portfolios and the value and multi-value portfolios in Exhibit 4 confirm the superiority of the neural network portfolios over both market indices. Analysis of the difference in value and multi-value portfolio Sharpe Ratios, however, shows no significant difference between the multi-value Sharpe Ratio and either of the dividend yield value portfolio Sharpe Ratio (*P*-value = .5687) or P/E value portfolio Sharpe Ratio (*P*-value = .1868). Sharpe Ratios for all other value portfolios are significantly lower than the multi-value Sharpe Ratio, using the binary model. These results are consistent with those attributable to the Sortino Ratio, with the exception that there is no significant difference between the multi-value Sortino Ratio and the dividend yield value portfolio Sortino Ratio only (*P*-value = .5157). Exhibit 7a and Exhibit 7b respectively depict the difference between the multi-value (binary) portfolio Sharpe Ratios and DS Australian Real Estate Index and S&P/ASX 300 Property Sharpe Ratios over the test period.

Exhibit 8a and Exhibit 8b respectively depict the difference between the multi-value (binary) portfolio Sortino Ratios and DS Australian Real Estate Index and S&P/ASX 300 Property Sortino Ratios over the test period.

Results pertaining to the neural network linear node type are presented in Exhibit 5. For the value portfolios, the highest mean return is for the dividend yield value portfolio (8.03%) and the lowest is for the market capitalization value portfolio (6.18%). The Multi-Value Linear model portfolio mean return of 8.08% is marginally higher than the binary model mean of 7.93%. Consistent with results already discussed for the binary output node type, nominal mean returns for all portfolios are significantly greater than mean returns to both market indices. While the multi-value portfolio mean return is also higher than all of the value portfolios, the difference in returns between the multi-value portfolio and dividend yield value portfolio is not significant for the linear model (*P*-value = .9283). Also consistent with results for the binary models, Sharpe and Sortino Ratios for all the linear model portfolios are significantly greater than those for both market indices. Likewise, there is no significant difference between the multi-value Sharpe Ratio and either of the dividend yield value portfolio Sharpe Ratio (*P*-value = .7795) or P/E value portfolio Sharpe Ratio (*P*-value = .1615), nor the multi-value Sortino Ratio and the dividend yield value portfolio Sortino Ratio (*P*-value = .9601) using the linear node type.

Conclusion

This paper set out to ascertain whether the non-linear, parallel processing power of a neural network is capable of producing property portfolios that would outperform the market on a regular basis. The two benchmarks were a property index comprising twenty-four property companies constructed by Datastream International and a property index comprising twenty-eight companies constructed by the Australian Stock Exchange (with a high degree, though not perfect correlation between the two markets). Based on both nominal (non-adjusted) and risk-adjusted returns, the evidence appears overwhelming that neural network constructed portfolios are quite capable of outperforming the market on a consistent basis—an outcome that is rarely achieved on anything like a regular basis by active fund managers. In terms of the number and variety of indicators of value, the dividend yield has been confirmed as the single most significant indicator investors should consider. Value portfolios containing high dividend yield stocks outperformed, or at worst marginally underperformed, relative to all other portfolios, including those constructed using multiple value criteria. The wider implication with regard to the property stock selection process is that this research adds further support to the contrarian strategy, and that there is benefit in actively managing a portfolio rather than simply selecting a portfolio to track the market. In addition, those managers who chose to actively manage a portfolio can be assisted through the use of a neural network tool.

This study, however, only dealt with the Australian property market. There are two further, and much larger, stages to this research program: (1) check for cross country consistency on these results in the U.S., U.K. and a selection of Asian and European countries, and (2) construct 'world' portfolios and benchmark against 'world' property indices.

Appendix

Calculation of the Sortino Ratio

The Sortino Ratio is calculated as the difference between the portfolio return (R_p) and the minimum acceptable return (MAR), divided by the downside deviation (DD) of the portfolio return versus the minimum acceptable return (DD_{MAR}). Downside deviation is similar to the loss standard deviation with the exception that it (DD) only includes portfolio returns below the MAR, rather than portfolio returns below the mean. The basis of the Sortino Ratio is that investors are more concerned with the risk of loss (downside risk), than the risk of gains (upside risk). Standard deviation as used by the Sharp Ratio, considers both upside and downside risk.

$$Sortino\ Ratio = \frac{R_p - R_{MAR}}{DD_{MAR}} \tag{A1}$$

$$DD_{MAR} = \sqrt{\frac{\sum_{i=1}^{N} (Li)^2}{N}}$$

$$Li = (Ri - R_{MAR}) \text{ if } (Ri - R_{MAR} < 0$$

or

$$Li = 0 \text{ if } (Ri - R_{MAR}) > 0.$$

Endnotes

1. The synapse is the locus where a nervous impulse passes from the axon of one neuron to the dendrites of another.
2. The binary node type model is the more typical output type used in ANNs.
3. The software used for the analysis is the Braincel Neural Network software version 3.62. Mr. Gideon Isaac, Technical Support Unit, Promised Land Technologies (Gideon@micronet.com), confirms that the tanh transfer function is used in both input and hidden layers.
4. A Datastream calculated Index, the 'Real Estate' series, is based on the FTSE classification and includes the following sub-sectors: Real Estate Development, Property Agencies and Real Estate Investment Trusts.
5. As is evident from Exhibit 5, the excess return is typical across all neural network portfolios, hence only a couple of graphs are presented.
6. z scores and P-values for the difference between the multi-value portfolio mean return and single-variable mean returns, not reported in this study, are available from the authors by request.

References

Beneda, N., Growth Stocks Outperform Value Stock Over the Long Term, *Journal of Asset Management*, 2002, 3:2, 112–23.

Best, R. J., R. W. Best and J. A. Yoder, Value Stocks and Market Efficiency, *Journal of Economics and Finance*, 2000, 24:1, 28–35.

Borst, R. A., Artificial Neural Networks: The Next Modeling/Calibration Technology for the Assessment Community?, *Property Tax Journal*, 1991, 10:1, 69–94.

Brooks, C. and S. Tsolacos, International Evidence on the Predictability of Returns to Securitised Real Estate Assets: Econometric Models vs. Neural Networks, *Journal of Property Research*, 2003, 20:2, 133–56.

Chen, N. and F. Zhang, Risk and Return of Value Stocks, *Journal of Business*, 1998, 71:4, 501–35.

Coakley, J. R. and C. E. Brown, Artificial Neural Networks in Accounting and Finance: Modelling Issues, *International Journal of Intelligent Systems in Accounting, Finance and Management*, 2000, 9:2, 119–44.

Connellan, O. and H. James, Estimated Realisation Price (ERP) by Neural Networks: Forecasting Commercial Property Values, *Journal of Property Valuation and Investment*, 1998, 16:1, 71–9.

Davis, J. L., E. F. Fama and K. R. French, Characteristics, Covariances, and Average Returns: 1929–1997, *Journal of Finance*, 2000, 55, 389–406.

Desai, V. S. and R. Bharati, The Efficacy of Neural Networks in Predicting Returns on Stock and Bond Indices, *Decision Sciences*, 1998, 29:2, 405–25.

Do, A. Q. and G. Grudnitski, A Neural Network Approach to Residential Property Appraisal, *The Real Estate Appraiser*, 1992, December, 38–45.

Donaldson, R. G. and M. Kamstra, Forecast Combining with Neural Networks, *Journal of Forecasting*, 1996, 15:1, 49–62.

Eakins, S. G. and S. R. Stansell, Can Value-based Stock Selection Criteria Yield Superior Risk-Adjusted Returns: An Application of Neural Networks, *International Review of Financial Analysis*, 2003, 12, 83–97.

Fama, E. F and K. R. French, Size and Book-to-Market Factors in Earnings Returns, *Journal of Finance*, 1996, 51:1, 131–55.

——., Value versus Growth: The International Evidence, *Journal of Finance*, 1998, 53:6, 1975–99.

Giudici, P., *Applied Data Mining—Statistical Methods for Business and Industry*, Chichester, UK: John Wiley and Sons, 2003.

Haugen, R. A., *The New Finance: The Case Against Efficient Markets*, Second Edition, Upper Saddle River, NJ: Prentice-Hall, 1999.

Hawley, D. D., J. D. Johnson and D. Raina, Artificial Neural Systems: A New Tool for Financial Decision-Making, *Financial Analysts Journal*, 1990, 46:6, 63–72.

Hill, T., M. O'Connor and W. Remus, 1996. Neural Network Models for Time Series Forecasts, *Management Science*, 1996, 42:7, 1082–92.

Kanas, A., Neural Network Linear Forecasts for Stock Returns, *International Journal of Finance and Economics*, 2001, 6:3, 245–54.

Kantardzic, M., Data Mining—Concepts, Models, Methods and Algorithms, New Jersey: John Wiley and Sons, 2003.

Kothari, S. P. and J. Shanken, Book-to-Market, Dividend Yield, and Expected Market Returns: A Time-Series Analysis, *Journal of Financial Economics*, 1997, 44:2, 169–203.

Kryzanowski, L., M. Galler and D. W. Wright, Using Artificial Neural Networks to Pick Stocks, *Financial Analyst Journal*, 1993, 49:4, 21–7.

Lakonishok, J., A. Shleifer and R. Vishny, Contrarian Investment, Extrapolation and Risk, *Journal of Finance*, 1994, 49:5, 1541–78.

Levis, M. and M. Liodakis, Contrarian Strategies and Investor Expectations, *Financial Analysts Journal*, 2001, 57(2), 43–56.

Makridakis, S., A. Andersen, R. Carbone, R. Fildes, M. Hibon, R. Lewandowski, J. Newton, E. Parzen and R. Winkler, The Accuracy of Extrapolation (Time Series) Methods: Results of a Forecasting Competition, *Journal of Forecasting*, 1982, 1:2, 111–54.

McGreal, S., A. Adair, D. McBurney and D. Patterson, Neural Networks: The Prediction of Residential Values, *Journal of Property Valuation and Investment*, 1998, 16:1, 57–67.

Narain, L. S. and R. L. Narain, Stock Market Prediction: A Comparative Study of Multivariate Statistical and Artificial Neural Network Models, *Journal of Accounting and Financial Research*, 2002, 10:2, 85–94.

Nguyen, N. and A. Cripps, Predicting Housing Value: A Comparison of Multiple Regression Analysis and Artificial Neural Networks, *Journal of Real Estate Research*, 2001, 22:3, 313–36.

O'Shaughnessy, J. P., What Works on Wall Street, Revised Edition, New York, McGraw-Hill, 1998.

Phua, P. K. H., D. Ming and W. Lin, Neural Network with Genetically Evolved Algorithms for Stocks Prediction, *Asia-Pacific Journal of Operational Research*, 2001, 18, 103–07.

Qi, M., Nonlinear Predictability of Stock Returns Using Financial and Economic Variables, *Journal of Business and Economic Statistics*, 1999, 17:4, 419–29.

Shachmurove, Y. and D. Witkowska, Utilizing Artificial Neural Network Model to Predict Stock Markets, CARESS Working Paper #00-11, September, 2000.

Sharpe, W. F., Mutual Fund Performance, *Journal of Business*, 1966, 39, 119–38.

Sortino, F. A. and H. J. Forsey, On the Use and Misuse of Downside Risk, *Journal of Portfolio Management*, 1996, 22:2, 35–42.

Sortino, F. A., G. A. Miller and J. M. Messina, Short-Term Risk-Adjusted Performance: A Style-Based Analysis, *Journal of Investing*, 1997, 6:2, 19–28.

Tan, C. N. W., *Artificial Neural Networks: Applications in Financial Distress Prediction & Foreign Exchange Trading,* Australia: Wilberto Publishing, 2001.

Tay, D. P. H. and D. K. K. Ho, Artificial Intelligence and the Mass Appraisal of Residential Apartment, *Journal of Property Valuation & Investment*, 1992, 10, 525–40.

Terasvirta, T., Specification, Estimation, and Evaluation of Smooth Transition Autoregressive Models, *Journal of the American Statistical Association*, 1994, 89:425, 208–18.

Trippi, R. R. and D. DeSieno, Trading Equity Index Futures With a Neural Network, *Journal of Portfolio Management*, 1992, 19:1, 27–33.

Walczak, S., An Empirical Analysis of Data Requirements for Financial Forecasting with Neural Networks, *Journal of Management Information Systems*, 2001, 17:4, 203–22.

Webb, J. R. , M. J. Seiler and F. C. Neil Myer, Are EREITs Real Estate, *Journal of Real Estate Portfolio Management*, 1999, 5:2, 171–81.

Wilson, I. D., S. D. Paris, J. A. Ware and D. H. Jenkins, Residential Property Time Series Forecasting with Neural Networks, *Journal of Knowledge-Based Systems*, 2002, 15, 335–41.

Wong, F. S., P. Z. Wang, T. H. Goh and B. K. Quek, Fuzzy Neural Systems for Stock Selection, Financial Analyst Journal, 1992, 48:1, 47–52.

Worzala, E., M. Lenk and A. Silva, An Exploration of Neural Networks and its Application to Real Estate Valuation, *Journal of Real Estate Research*, 1995, 10:2, 185–201.

Yen, J. Y., Q. Sun and Y. Yan, Value versus Growth Stocks in Singapore, *Journal of Multinational Financial Management*, 2004, 14, 19–34.

Yu, S., Forecasting and Arbitrage of the Nikkei Stock Index Futures: An Application of Backpropagation Networks, *Asian-Pacific Financial Markets*, 1999, 6:4, 341–54.