Courant Research Centre
'Poverty, Equity and Growth in Developing and Transition Countries: Statistical Methods and Empirical Analysis'

Georg-August-Universität Göttingen
(founded in 1737)

Discussion Papers

No. 202

Forecasting the Market Risk Premium with Artificial Neural Networks

Leoni Eleni Oikonomikou

April 2016

Wilhelm-Weber-Str. 2 · 37073 Goettingen · Germany
Phone: +49-(0)551-3914066 · Fax: +49-(0)551-3914059

Email: crc-peg@uni-goettingen.de Web: http://www.uni-goettingen.de/crc-peg

Forecasting the Market Risk Premium with Artificial Neural Networks

by Leoni Eleni OIKONOMIKOU[1]

First version: March 2015

Revised version: November 2015

[1] Research Associate and PhD Candidate at Georg-August University of Goettingen, Germany.
Contact address: Platz der Göttinger Sieben 5, 37073, Göttingen, Germany
E-Mail: leoni-eleni.oikonomikou@wiwi.uni-goettingen.de , le.oikonom@gmail.com
I would like to thank the DFG, German Research Foundation, RTG 1723, fort he provided financial support along the preparation of this research paper.

Forecasting the Market Risk Premium with Artificial Neural Networks

Abstract:

This paper aims to forecast the Market Risk premium (MRP) in the US stock market by applying machine learning techniques, namely the Multilayer Perceptron Network (MLP), the Elman Network (EN) and the Higher Order Neural Network (HONN). Furthermore, Univariate ARMA and Exponential Smoothing models are also tested.

The Market Risk Premium is defined as the historical differential between the return of the benchmark stock index over a short-term interest rate. Data are taken in daily frequency from January 2007 through December 2014. All these models outperform a Naive benchmark model. The Elman network outperforms all the other models during the insample period, whereas the MLP network provides superior results in the out-of-sample period.

The contribution of this paper to the existing literature is twofold. First, it is the first study that attempts to forecast the Market Risk Premium in a daily basis using Artificial Neural Networks (ANNs). Second, it is not based on a theoretical model but is mainly data driven. The chosen calculation approach fits quite well with the characteristics of ANNs. The forecasting model is tested with data from the US stock market.

The proposed model-based forecasting method aims to capture patterns in the data that improve the forecasting accuracy of the Market Risk Premium in the tested market and indicates potential key metrics for investment and trading purposes for short time horizons.

Keywords: nonlinear models, forecasting performance metrics, market risk premium,
US equity market.

JEL classification: C45, C52, G15, G17

1. Introduction

The market risk premium is often used by academics and financial market practitioners for asset valuation purposes. The motivation of this study is to assess whether there exists any temporal patterns that better forecast the market risk premium in short time horizons. This will potentially be of fundamental importance for asset managers who invest in equity market indices and desire to make timely investment decisions about asset allocation, rebalancing of portfolios and formulating appropriate investment strategies. Moreover, it is important for risk managers who seek an indication of the premium for bearing non-diversifiable risk, in order to avoid particular scenarios and limit the damage from others. It is also useful for pension funds on exploring whether it is advantageous to invest in a particular stock or bond market.

Market risk premium refers to the excess return over the risk free rate that an investor is requesting on a market investment as a compensation for the risk he or she is undertaking. Investing in equities is riskier for the following reasons: For one, historically stocks have been more volatile relative to other asset classes and an investor can lose a lot more money in equities in the short run. Secondly, equity holders get paid out when all the other claims on a company's cash flows and the employee's get paid first, thus, indicating that the cash flows to equity holders contain a higher probability of default.

In this study, the market risk premium time series is calculated via the historical premium approach which is a purely backward looking approach. More precisely, it is calculated as the historical differential of the total return on the benchmark stock index for a particular market and a short-term interest rate (as a proxy for the riskless rate). This historical risk premium approach also assumes that the premium is equal for all investors. For an extensive discussion on appropriate methods of calculating the market risk premium, as well as estimation and forecasting methods, refer to A. Damodaran (2013); Goetzmann and Ibbotson (2006) and E. Dimson et al. (2002).

The historical premium calculation method was selected because of its convenience to be implemented in an Artificial Neural Network modeling context. The time period chosen is such that it includes pre and post crisis data so that the adaptability of the models can be tested across different market conditions.

The remainder of the paper is organized as follows. Section 2 summarizes relevant literature. Section 3 provides an extensive description of the data and Section 4 introduces the methods used in this study. Finally, Section 5 presents the empirical results and Section 6 concludes.

2. Literature Review

Forecasting the market risk premium has been the subject of a long debate among finance academics. A couple of empirical studies have pointed out the weak "out-of-sample" performance of widely used regression-based predictors of risk premium (Welch and Goyal, 2006), and unstable forecasting relationships (Lettau and Nieuwerburgh, 2008). Pastor and Stambaugh (2001) propose a framework in order to model and estimate the market risk premium. This approach encloses the information for the entire historical dataset on returns, with economically motivated prior beliefs about the risk premium. They also account for the existence or not of a mean-variance link mechanism for the risk premium. In his survey paper, M. Spiegel (2008) describes the controversy of the previously unstable forecasting relationships and the regression-based predictors of risk premium.

Therefore, my paper aims to propose a different methodological alternative that is based on the inherent nonlinearities of financial time series, in order to more accurately forecast the market risk premium on a daily basis.

Faria *et al.*(2009) present a methodological alternative to traditional Exponential Smoothing forecasting modeling in which the parameter a is updated along the prediction. In the same vein, Lai *et al.* (2007) propose a hybrid synergy model that integrates an exponentially smoothing (ES) model and a Backpropagation Neural Network (BPNN) and test it with two exchange rates, EUR/USD and JPY/USD. An individual exponential smoothing model and an individual BPNN model are used as benchmark models, for comparison.

Perez-Rodriguez *et al.* (2005) apply STAR and ANN in the task of forecasting the Spanish Ibex-35 index using one-step and multi-step ahead forecasting techniques. They assess their results based on statistical and economic metrics. Olson and Mossman (2003) compare the forecasting performance of neural networks, logistic regression and ordinary least squares methods to the task of forecasting one-year ahead Canadian stock returns. Their findings support the superiority of neural networks over the other tested techniques. They confirm their results by applying trading strategies.

Previous work has also investigated the modeling and trading of the gasoline crack spread where the spread is considered as the profit margin gained by cracking crude oil. (Dunis *et al.*, 2006). Roh (2007) proposes hybrid models by combining artificial neural networks and time-series models in the task of forecasting the deviation and direction of the volatility of the stock price index. Krollner (2011) develops an ANN based market timing model in order to predict the Australian stock index futures market one-month ahead. This model is then used for portfolio risk management and setting up appropriate hedging strategies.

Qi (1999) examines the nonlinear predictive ability of excess returns in US stock market with a recursive neural network and economic and financial variables as inputs to the network. Empirical results demonstrate the superiority of neural networks over its linear counterparts. Thawornwong and Enke (2004) aim to uncover the structural relationship of various financial and economic variables by applying data mining analysis and neural networks to the S&P 500 stock portfolio.

3. Data and Methodology

This section aims to provide a detailed description of the data used in this study as well as the transformations that have been made. More precisely, the market risk premium for the US financial market is defined via the historical premium approach, as following:

$$MRP = R_m - R_f \qquad (1)$$

where:

- MRP is the Market risk premium

- R_m is the market rate of return
- R_f is the risk-free rate return

The choice of a short-term Treasury Bill yield as a proxy for the Risk-free Rate return has been selected because the total dataset comprises of eight years and the focus of the study is on short-term horizons.

All data has been sourced from DataStream for the period from January 2007 through to December 2014. More precisely daily closing prices (adjusted for dividends and splits) of the S&P 500 composite Index are used as a proxy for the market rate of return for the tested US market. In the same vein, the chosen proxy for the Risk-free Rate of Return for the US market is: US T-Bill Secondary Market 3 Months, Middle Rate.

The figure below, shows the calculated daily historical Market Risk Premium time series for the US financial market:

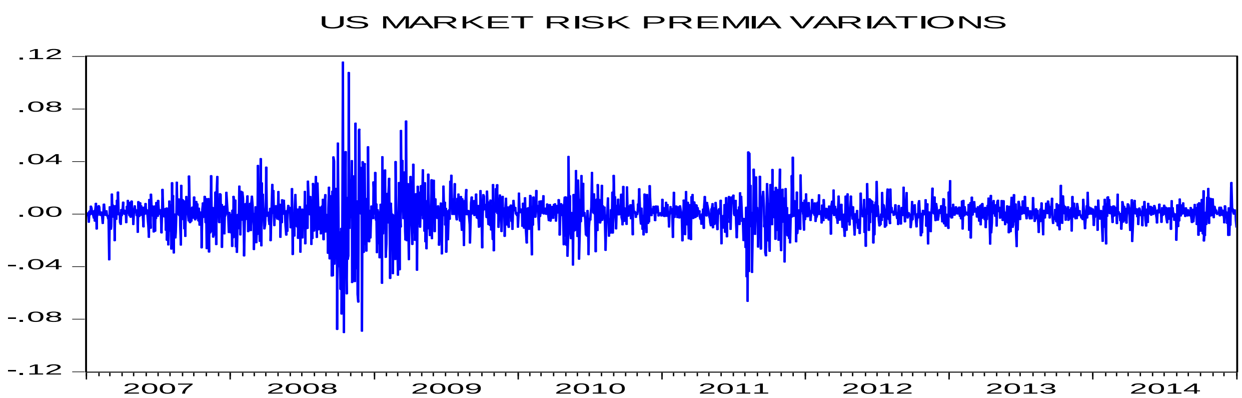

Figure 1: US Market risk premium during the entire sample period.

This time-series contains slight skewness, high kurtosis and is non-normal. The Jarque-Bera test confirms it at 99 per cent confidence interval, as presented by the summary statistics in the Appendix A.1. In addition, the daily closing prices are non stationary and are transformed to daily return series in order to assure the stationary property of this time series.

Taking into consideration the Price level $P_1, P_2, \dots, P_t$, the return at time t is calculated as:

$$R_t = \left(\frac{P_t}{P_{t-1}}\right) - 1 \qquad (2)$$

The selection of inputs (explanatory variables) inserted to the network has been done via variable/feature selection. This method is used in order to apply some measures that can be used to quantify the relevance of variables hidden in a large data set with respect to a given class or concept description. Such measures include information gain, the Gini index, uncertainty and correlation coefficients (Thawornwong and Enke, 2004). In this paper correlation coefficients are used in order to determine the necessary inputs that were included in the network. Therefore, the resulting variables having high correlation with the target/output time-series are chosen as the relevant input variables provided to the neural network models.

The number of lags has been selected after testing Autoregressive Models with different lags in an appropriate statistical software and selecting the best performing model with the lowest information criteria. The 10 main sector indices as well as the 3-month Treasury Bill are chosen as inputs to the neural networks since they have a clear explanatory power and correlation with the Market Risk Premium. The sector indices include the companies listed in the benchmark stock index. Therefore, sector indices are deemed as the most appropriate for our universal set of inputs.

In total, 11 inputs were inserted to the network. Table 1 below shows all the inputs to neural networks as well as the lags with which they were inserted. The final inputs are also normalized. This is intended for faster approaching to global minima at error surface and can also assure faster training.

Number	Explanatory Variables	Lag
1	S&P 500 Consumer Discretionary TR	2
2	S&P 500 Consumer Staples TR	2
3	S&P 500 Energy TR	2
4	S&P 500 Financials TR	3
5	S&P 500 Healthcare TR	3
6	S&P 500 Industrials TR	1
7	S&P 500 Information Technology TR	2
8	S&P 500 Materials TR	1
9	S&P 500 Telecommunication Services TR	2
10	S&P 500 Utilities TR	2
11	US T-Bill Secondary Market 3 months	4

Table 2 : Explanatory Variables for US Stock Exchange.

In a previous version of this paper, volatility index and yield curves have also been included in the initial universe of inputs. However, they were finally dropped out due to poor forecasting accuracy results. This leads to the conclusion of no predictability link between market risk premium and volatility. In addition, the proposed model has been applied to Polish, Peruvian and Philippines datasets which were not included as well in this paper, due to poor results and lack of comparability of risk-free rates. In the case of Peru, there was a serious liquidity problem in the chosen time period that deterred an unbiased economic comparison with the market risk premia from the other countries.

4. Forecasting Models

4.1. Naïve Model

The benchmark model that is usually used by the forecasting literature to predict the future is the Naïve one. This model assumes that all forecasts for the future are equal to the last observed value of the series at time . Hence, the model takes the following form:

$$Y_t = Y_{t+1} \qquad (3)$$

where: Y_t is the current rate of return at time t and Y_{t+1} is the forecast rate of return at time $t+1$.

4.2. Exponential Smoothing

4.2.1. Single Exponential Smoothing

Exponential Smoothing assigns *exponentially decreasing weights* as the observations get older. Hence, *recent observations are given relatively more weight in forecasting than the older observations*. In simple exponential smoothing, forecasts are calculated using weighted averages where the weights decrease exponentially as observations come from the distant past. The smallest weights are given to the oldest observations. The exponential smoothing model is defined as:

$$Y_{t+1} = aY_t + a(1-a)Y_{t-1} + a(1-a)^2Y_{t-2} + \cdots + a(1-a)^nY_{t-n} \qquad (4)$$

where:

- Y_{t+1} is the forecast for the next period $t+1$

- Y_t is the rate of return in the present period t
- $Y_{t-1}, Y_{t-2}, \dots, Y_{t-n}$ are the lagged values of the return

at periods $t-1,\ t-2, \dots,\ t-n$

- a is the smoothing parameter, $0 \leq a \leq 1$

4.3. Autoregressive Moving Average Model (ARMA)

In Autoregressive Moving Average models (ARMA), the current value of a time series is assumed to depend on its previous values and on previous residual values. Therefore, the model includes autoregressive and moving average components and takes the following form:

$$Y_t = \varphi_0 + \varphi_1 Y_{t-1} + \varphi_2 Y_{t-2} + \cdots + \varphi_p Y_{t-p} + \varepsilon_t - w_1 \varepsilon_{t-1} - w_2 \varepsilon_{t-2} - \cdots - w_q \varepsilon_{t-q} \quad (5)$$

where:

- Y_t is the dependent variable at time t
- $Y_{t-1},\ Y_{t-2}, \dots, Y_{t-p}$ are the lagged dependent variables
- $\varphi_0, \varphi_1, \dots, \varphi_p$ are the regression coefficients
- ε_t is the residual term
- $\varepsilon_{t-1}, \varepsilon_{t-2}, \dots, \varepsilon_{t-q}$ are the previous values of the residual terms
- $w_1, w_2, \dots, w_q$ are the residual weights

Taking into account the insample correlogram (training and test subsets) an ARMA (2,3) was chosen for the US market. All the coefficients (except for the constant) are statistically significant at 99 per cent confidence interval. This ARMA model is specified below and the output results are cited in the Appendix B.1:

$$Y_t = 0.014097 - 1.725901 Y_{t-1} - 0.856905 Y_{t-2} - 1.604583 \varepsilon_{t-1} - 0.594618 \varepsilon_{t-2} + 0.16293 \varepsilon_{t-3} \qquad (6)$$

4.4. Artificial Neural Networks

Artificial Neural Networks (ANNs) gained immense popularity over the recent past and are widely applied in a variety of financial time-series forecasting problems and investment decision-making. They were initially developed as a model that mimics the intelligence of a human brain to a machine (Trippi and Turban, 1992). ANNs became quite popular due to their particular and attractive features which are enumerated below.

First, they are data-driven, self-adaptive models. This ensures no need to make any a priori assumption about the statistical distribution of the data or to specify any particular model form in the model building process; instead the network model will be adaptively formed based on the features presented from the data. Second, they are nonlinear models which make them more statistically accurate in modeling complex data patterns, compared to their traditional linear counterparts, such as ARIMA techniques (Adhikari and Agrawal, 2013; G. Rozenberg *et al.*, 2012). Finally, they are universal functional approximators and can approximate a large class of functions with a high degree of accuracy (Khashei and Bijari, 2010).

However, ANNs exhibit some drawbacks, from a statistical point of view. An often cited one, is the fact that model parameters are difficult, if not impossible, to interpret, thus they are considered as "black box" models and built principally for pattern recognition and forecasting. Another one concerns the risk of overfitting or under fitting the data. Overfitting occurs when the constructed model is fairly complex and may fit irregular or unpredicted noise in the data. In this case, the model will be less robust for out-of-sample forecasting. Under fitting occurs when a model is excessively simple to capture the underlying trend of the data and does not fit them well enough.

There are numerous types of Artificial Neural Network models in the literature. In this paper, three Artificial neural network models are applied, namely, the Multilayer Perceptron network (MLP), the Elman Network (EN) and the Higher Order Neural Network (HONN) in order to forecast the daily Market Risk Premium.

4.4.1. Multilayer Perceptron Network

A standard neural network has at least three layers. Similar to a multivariate regression, the number of nodes in the first layer, also called as input layer, corresponds to the number of explanatory variables. The number of nodes in the last layer, corresponds to the number of dependent variables. The number of nodes in the intermediate layer, or alternatively hidden layer, defines the amount of complexity that the network model is able to fit. Further, the input and hidden layers contain an additional node, which is the bias node and has a fixed value of one. This bias node is similar to the intercept in a standard regression model. Each node in one layer is connected to the next layer. These connections, called network weights, are the model parameters in a standard regression.

The information to the network is processed as follows:

- the explanatory variables are inserted to the network via its input nodes (in this paper, that includes lagged values of total returns on sector indices and 3-month T-Bill).
- Then, information is processed to each node in the hidden layer via the network weights, as the weighted sum of its inputs.
- Finally, each node in the hidden layer processes the information, via a nonlinear activation function (transfer sigmoid function) on the output layer, where the information is processed via a linear transfer function on to the final output of the model (the desired market risk premium).

A simplified, single output MLP model is shown in the Figure 2 below:

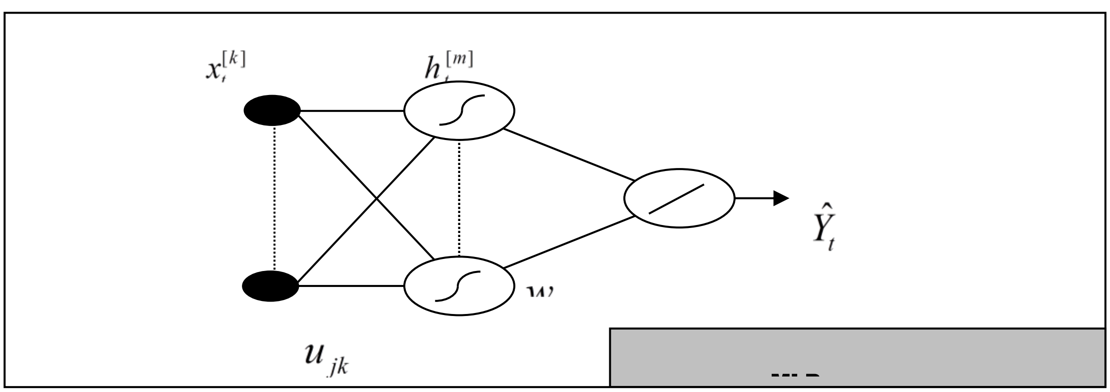

Figure 2: A single output, fully connected MLP (source: Dunis *et al.*(2014)).

- $x_t^{[n]}$ $\left(n = 1,2,\cdots,k+1\right)$ are the model inputs (including the input bias node) at time t (these are the explanatory variables described above)
- $h_t^{[m]}$ $\left(m = 1,2,...,m+1\right)$ are the hidden nodes outputs (including the hidden bias node) at time t
- $\hat{Y}_t$ is the MLP output (Market Risk Premium one-day ahead)
- u_{jk} , w_j are the network weights
- is the transfer sigmoid function: $S\left(x\right) = \frac{1}{1+e^{-x}}$ (7)
- is a linear function: $F\left(x\right) = \sum_i x_i$ (8)

The error function to be minimized is:

$$E(u_{jk}, w_j) = \frac{1}{T}\sum(\mathrm{Y}_t - \hat{Y}_t(u_{jk}, w_j))^2 \qquad (9)$$

where Y_t is the actual value and $\hat{Y}_t$ is the target value.

4.4.2. Elman Networks (EN)

Elman networks were initially developed in 1990. ENs with one or more hidden layers are feedforward networks which have the ability to learn any dynamic relationship between inputs and output arbitrarily well, given enough neurons in the hidden layer.

A simple Elman Network has activation feedback that encloses short-term memory. These additional memory units enable the network to yield better results in comparison to simple MLPs. However, as Tenti (1996) highlights, their main disadvantage is that they require substantially more connections and more memory in simulation, than standard backpropagation networks. This results in a significant increase in computational time, during the training process.

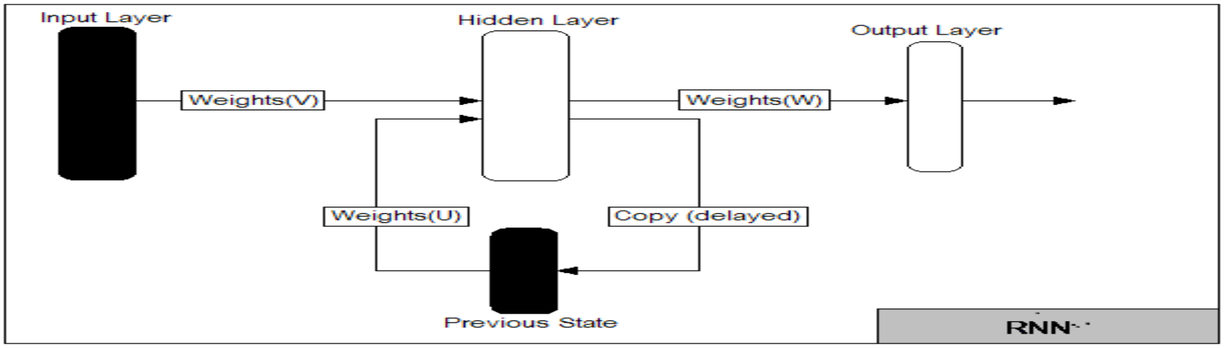

Figure 3: Architecture of Elman or Recurrent Neural Network (source: Dunis *et al.*, 2006).

The state/hidden layer shown in Figure 3 is updated with external inputs, similar to simple MLP, but also with activation from previous forward propagation, shown as "Previous State" in the Figure. In essence, the Elman architecture can provide more accurate outputs because the inputs are potentially taken from all previous values.

Similar to MLP network architecture, the Elman network uses the transfer sigmoid function, error function and linear function. This has been done intentionally in order to have the chance of comparing the architectures of all the models together.

4.4.3. Higher Order Neural Network

Higher Order Neural Networks were initially presented by Giles and Maxwell (1987). This type of neural network (NN) architecture has been widely used in pattern recognition, nonlinear simulation, classification and prediction in computer science and engineering. However, their financial applications remain quite limited.

The structure of a three input second-order HONN is displayed in Figure 4 below:

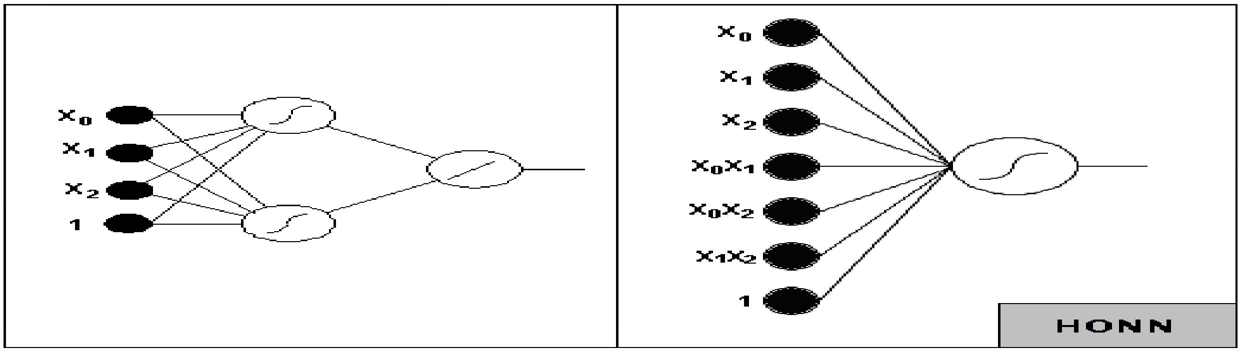

Figure 4: Left, MLP with three inputs and two hidden nodes; right, second-order HONN with three inputs (source: Dunis *et al.*, 2006).

Higher Order NNs use joint activation functions which reduce the need to establish relationships between inputs when training. This diminishes the number of free weights, resulting in potential faster training than even MLPs. Moreover the reduction of free weights means that the problems of overfitting and local optima affecting the results can be largely avoided (Kamruzzaman, Begg and Sarker, 2006). These type of models are also referred to as "open-box" models, where each neuron maps variables to a function through weights/coefficients without the use of hidden layers (Shawash, 2012). The omission of hidden layers in this type of NN architecture (as shown at the right part of Figure 4) permits the use of easier training methods, leads to faster convergence, reduced network size and more accurate curve fitting, compared to other kind of NN architectures (Zhang, Xu and Fulcher, 2002). However, since the number of inputs can become quite large for higher order architectures, orders of four and above are not frequently applied.

4.5. <u>Training the Neural Network</u>

The network training (which is the adjustment of its weights in such a way that the network maps the input value of the training dataset to the output/predicted value) begins with randomly selected weights and continues by applying the backpropagation of errors learning algorithm. This learning algorithm searches those weights that minimize an Error function.

Depending on the amount of nodes in the hidden layer, it is probable that the network learns the training data exactly (alternatively referred as overfitting). Therefore, the network training has to stop early (usually referred as 'early stopping'). This is done by dividing the dataset into three subsets (as shown in the Table 2 below).

Dataset	Trading Dates	From	To
Total set	2004	02/01/2007	31/12/2014
Training set (insample)	1196	02/01/2007	27/10/2011
Test set (insample)	398	28/10/2011	31/05/2013
Validation set (out-of-sample)	399	03/06/2013	31/12/2014

Table 2: Neural networks training dataset for US stock exchange

First, the training set is used to optimize the model and the backpropagation learning algorithm is used to determine optimal weights from the initial random weights. Then, the test set is used to stop the training set from potential overfitting problems. More precisely, the training set optimization stops when the test set reaches at the maximum positive return. Both training and test sets refer to the insample subset and are split in such a way in order to avoid overfitting and make sure that any patterns in the data will be captured by the network model. At last, the out-of-sample subset is used to simulate future values of the time-series under study.

All NNs are trained 80 times. The best ten performing networks are selected (these demonstrate the lowest error within the insample set) for the out-of-sample forecasting task. In addition, forecasts of each network differ depending on the different architectures being tested and their initial random set of weights. Therefore, a simple average of the committee of these 10 NNs is presented as a way to eliminate possible outlier network and avoid the problem of local optima that might have arisen during the training process.

5. Empirical Results

The forecasting performance of the tested models is evaluated, based on a few commonly used statistical metrics namely, Root Mean Square Error (RMSE), Normalized Mean Square Error (NMSE), Directional Symmetry (DS), Correct Up Trend (CU), Correct Down Trend (CD).

RMSE and NMSE measure the deviation between actual and forecast values. A small value of these measures indicates higher accuracy in forecasting. DS measures the correct direction of change predictions. CU measures the correct positive direction of change predictions and CD the correct negative direction of change predictions. The last three statistical metrics are expressed in percentages and all the formulas are presented in the Appendix C.1.

All neural network architectures were trained with 11 inputs each, one hidden layer (applies only to MLP and EN architectures) and one output in order to predict the Market risk premium. The number of hidden nodes varied from 3~8 along the training of the networks and the HONN architecture is also tested for 2~4 orders. As previously mentioned, the NNs were trained 80 times and 10 networks were finally chosen based on their insample performance. The number of hidden nodes of the 10 selected networks varied from 3~5. The choice of a short time period (2007-2014) has been done intentionally in order to avoid potential structural breaks.

The tables 3 and 4 below show the insample and out-of-sample performance of the tested models in comparison with the benchmark naive model. The presented performance metrics for MLP, EN and HONN models represent a combined forecast (simple average) of the 10 best performing networks out of the total 80 trained networks.

	NAIVE	SES*	ARMA(2,3)	MLP	EN	HONN
RMSE	2.276	1.644	2.247	1.527	1.084	1.485
NMSE	2.195	1.155	2.136	0.988	0.497	0.933
DS	50	50.69	49.373	52.635	60.163	53.639
CU	47.429	49.371	47.886	49.943	57.829	51.657
CD	53.203	52.368	51.253	55.989	63.092	56.128

*SES is Single Exponential Smoothing with decay parameter $a = 0,2$

Table 3: Insample statistical performance for the US financial market.

	NAIVE	SES	ARMA(2,3)	MLP	EN	HONN
RMSE	0.997	0.747	0.710	0.709	0.719	0.729
NMSE	1.987	1.116	1.008	1.005	1.032	1.061
DS	48.872	48.628	48.120	62.657	58.897	54.637
CU	44.589	47.619	48.485	59.74	60.173	53.68
CD	54.767	50	47.619	66.667	57.143	55.952

Table 4: Out-of-Sample statistical performance for the US financial market.

The insample performance metrics clearly propose the Elman network as the best performer among the rest of the tested models. The Multilayer Perceptron Network outperforms all the other models in out-of-sample and achieves to predict the downside moves better than the rest of the tested models. The Elman Network performs slightly better in predicting the upside moves. More precisely, the proposed forecast model achieves in predicting around 62.66 percent of correct direction of change moves for the Market risk premium. In particular, it succeeds in predicting 60.17 percent of the upside moves (EN network) and 66.67 percent of the downside moves (MLP network) in the US stock market. All these models outperform the benchmark naive model both insample and out-of-sample.

This leads us to the conclusion that the proposed forecast model can be used in order to indicate possible downside or upside moves in the market, form appropriate market timing strategies and decide on whether it is profitable to invest in the stock market compared to deposit your money in a riskless bank account.

In the same vein, insample performance metrics go in line with the out-of-sample ones, leading us to the conclusion that the model has good generalization capabilities. This means that since the results are similar for both insample and out-of-sample periods, it is highly likely that they will continue to be similar in the future. This also confirms the adaptive properties that neural networks have as they are learning the data which are given to them.

Therefore, both insample and out-of-sample results demonstrate the superiority of Artificial Neural Networks over simple univariate forecasting models for the tested market.

6. Concluding Remarks

This paper has developed, applied and compared three forecast models based on Artificial neural network architectures, in order to predict the one-day ahead Market risk premium for the US equity market, using top level sector indices and middle rates on 3-month T-

Bill. Univariate Single Exponential Smoothing and ARMA models are also tested against the benchmark naive model.

The empirical results demonstrate the superiority of ANNs over simpler univariate forecast models as well as over the benchmark naive model. In particular, MLP performed better out-of-sample, whereas EN in the insample period. Therefore, the proposed forecast model achieves to capture patterns in the data that better forecast the Market risk premium. Directional Accuracy measures can further be used as an indicator for trading and investment purposes.

The limitations of this methodological approach stem from the method itself. On the one hand, it takes time to train the networks. On the other hand, the results of the forecast model are sensitive to the inputs selected and the time period chosen.

Some proposals for future research may include the application of this framework to a developing/emerging market with the same inputs in order to favor from comparing the results between the Market risk premium from a developed and a developing market. How do NNs perform in that case? Another interesting extension for future work is to include some trading strategies and check the performance of the proposed forecast model in a simulated trading experiment. This might provide good grounds for forming appropriate market timing and investment strategies for portfolio managers on whether to invest in the stock or the bond market.

References

1. R. Adhikari and R. K. Agrawal (2013), *An Introductory Study on Time Series Modeling and Forecasting,* LAP LAMBERT Academic Publishing.
2. A. K. Alexandridis. and A. D. Zapranis (2014), *Wavelet Neural Networks: With Applications in Financial Engineering, Chaos and Classification,* 1st ed., Wiley.
3. G. S. Atsalakis and K. P. Valavanis (2009), 'Surveying Stock Market Forecasting Techniques — Part II: Soft Computing Methods', *Expert Systems with Applications*, 36, 5932 — 41.
4. A. Damodaran. (2013), *Equity Risk Premiums (ERP): Determinants, Estimation and Implications – The 2013 Edition* Stern School of Business.
5. H. Demuth *et al.* (2013), *Neural Network Toolbox 8 User's Guide*, The MathWorks, Inc., Natick.
6. E. Dimson *et al.* (2002), *The Triumph of Optimists: 101 Years of Global Investment Returns, 1st ed.,* Princeton University Press, New Jersey, US.
7. C. Dunis *et al.* (2006), 'Modeling and Trading the gasoline crack spread: A nonlinear story', *Derivatives Use, Trading & Regulation*, 12(1/2), 126 --145.
8. J. L. Elman (1990), "Finding Structure in Time", *Cognitive Science*, 14, 179-211.
9. E. L. de Faria *et al.* (2009), 'Predicting the Brazilian Stock Market through Neural Networks and Adaptive Exponential Smoothing Methods', *Expert Systems with Applications*, 12506 — 09.
10. C. I. Giles and T. Maxwell (1987), 'Learning, Invariance and Generalization in High-Order Neural Networks', *Applied Optics*, 26(23), 4972 – 4978.
11. W. N. Goetzmann and R. G. Ibbotson (2006), *The Equity Risk Premium: Essays and Explorations*, Oxford University Press, New York, US.
12. J. Kamruzzaman *et al.* (2006), *Artificial Neural Networks in Finance and Manufacturing*, Idea Group Inc, US.
13. M. Khashei. and M. Bijari (2010), 'An artificial neural network (p,d,q) model for timeseries forecasting', *Expert Systems with Applications*, 37, 470-489.

14. B. Krollner (2011), Risk Management in the Australian Stock market using Artificial Neural Networks, PhD thesis, School of Information Technology, Bond University.

15. S. Lai *et al.* (2007), *Foreign-Exchange-Rate Forecasting with Artificial Neural Networks,* (Springer US), pp 121-131.

16. M. Lettau and S. Van Nieuwerburgh (2008), 'Reconciling the Return Predictability Evidence'*, Review of Financial Studies*, 21(4), 1607 – 1652.

17. B. Ning *et al.* (2009), 'Using chaotic Neural Network to forecast stock index', *Advances in Neural Networks* 5551, 870 — 876.

18. D. Olson and C. Mossman (2003), 'Neural Network forecasts of Canadian stock returns using accounting ratios'*, International Journal of Forecasting* 19(3), 453 — 465.

19. L. Pastor and R. F. Stambaugh (2001), 'The Equity Premium and Structural Breaks', *Journal of Finan*ce 56, 1207 -- 39.

20. J. V. Perez-Rodriguez *et al.* (2005), 'STAR and ANN models: forecasting performance on the Spanish ibex-35 stock index', *Journal of Empirical Finance* 12(3), 490— 509.

21. M. Qi (1999), 'Nonlinear Predictability of Stock Returns Using Financial and Economic Variables', *Journal of Business & Economic Statistics*, 17 (4), 419-429.

22. T. H. Roh (2007), 'Forecasting the volatility of stock price index', *Expert Systems with Applications* 33(4), 916 — 922.

23. G. Rozenberg *et al.* (2012), *Handbook of Natural Computing*, Springer Berlin Heidelberg, 461-477.

24. G. Sermpinis *et al.* (2012), 'Trading the EUR/USD exchange rate with stochastic Neural Network combination and time-varying leverage', *Decision Support Systems*, 54(1), 316 – 329.

25. J. Shawash (2012), Generalized correlation higher order neural networks, neural network operation and Levenberg-Marquardt training on field programmable gate arrays, Doctoral thesis, UCL (University College London).

26. M. Spiegel(2008), 'Forecasting the Equity Premium: Where we stand today', *Review of Financial Studies,* 21(4), 1453 -- 59.

27. S. Thawongnwong and D. Enke, 'The adaptive selection of financial and economic variables for use with artificial neural networks', *Neurocomputing*, 56, 205 - 232.

28. R. R. Trippi and E. Turban (1992), *Neural Networks in Finance and Investing: Using Artificial Intelligence to Improve Real World Performance*, McGraw-Hill Inc., NY, US.

29. I. Welch and A. Goyal (2006), 'A Comprehensive Look at the Empirical Performance of Equity Premium Prediction', *Review of Financial Studies* 21(4), 1510 – 31.

30. M. Zhang *et al.* (2002), 'Neuron-adaptive higher order neural-network models for automated financial data modeling', *IEEE Transactions on Neural Networks*, 13(1): 188 – 204.

31. G. Zhang *et al.* (1998) 'Forecasting with artificial neural networks: the state of the art', *International Journal of Forecasting*, 14, 35 — 62.

7. Appendix

A.1 Summary Statistics

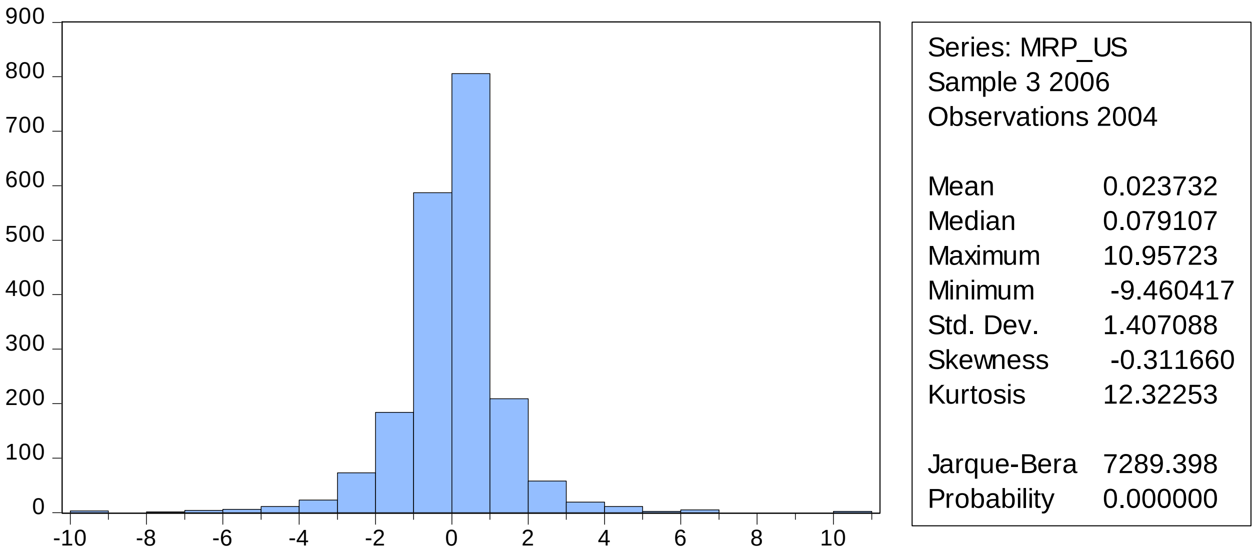

Figure 5: Summary Statistics for the US risk premium time series.

B.1 ARMA OUTPUT RESULTS

Dependent Variable: MRP_US
Method: ARMA Maximum Likelihood (OPG - BHHH)
Sample: 1 1604
Included observations: 1604
Convergence achieved after 182 iterations
Coefficient covariance computed using outer product of gradients

Variable	Coefficient	Std. Error	t-Statistic	Prob.
C	0.014097	0.034367	0.410193	0.6817
AR(1)	-1.725901	0.038901	-44.36661	0.0000
AR(2)	-0.856905	0.037688	-22.73669	0.0000
MA(1)	1.604583	0.041475	38.68809	0.0000
MA(2)	0.594618	0.047271	12.57902	0.0000
MA(3)	-0.162930	0.015061	-10.81831	0.0000
SIGMASQ	2.285780	0.041485	55.09844	0.0000

R-squared	0.026254	Mean dependent var	0.014010
Adjusted R-squared	0.022593	S.D. dependent var	1.532604
S.E. of regression	1.515192	Akaike info criterion	3.673411
Sum squared resid	3664.106	Schwarz criterion	3.696903
Log likelihood	-2937.239	Hannan-Quinn criter.	3.682134
F-statistic	7.171785	Durbin-Watson stat	1.994448
Prob(F-statistic)	0.000000		

Inverted AR Roots	-.86-.33i	-.86+.33i	
Inverted MA Roots	.18	-.89-.34i	-.89+.34i

Figure 6: ARMA(2,3) estimation output for the US Market risk premium time series.

C.1 Statistical Performance Metrics

$$RMSE = \frac{1}{n}\sqrt{\sum_{i=1}^{n}(X_i - \hat{X}_i)^2}$$

$$NMSE = \frac{1}{n}\frac{\sum_{i=1}^{n}(X_i - \hat{X}_i)^2}{\sum_{i=1}^{n}(X_i - \bar{X})^2}$$

$$DS = \frac{100}{n}\sum_{i=1}^{n} d_i, \qquad d_i = \begin{cases} 1, & if \quad (X_i - X_{i-1})(\hat{X}_i - \hat{X}_{i-1}) \geq 0 \\ 0, & otherwise \end{cases}$$

$$CU = 100\frac{\sum_{i=1}^{n} d_i}{\sum_{i=1}^{n} t_i} \qquad d_i = \begin{cases} 1, & if \;\; (X_i - \hat{X}_{i-1}) > 0, \quad (X_i - X_{i-1})(\hat{X}_i - \hat{X}_{i-1}) \geq 0 \\ 0, & otherwise \end{cases}$$

$$t_i = \begin{cases} 1, & if \;\; (X_i - X_{i-1}) > 0 \\ 0, & otherwise \end{cases}$$

$$CD = 100\frac{\sum_{i=1}^{n} d_i}{\sum_{i=1}^{n} t_i} \qquad d_i = \begin{cases} 1, & if \;\; (X_i - \hat{X}_{i-1}) < 0, \quad (X_i - X_{i-1})(\hat{X}_i - \hat{X}_{i-1}) \geq 0 \\ 0, & otherwise \end{cases}$$

$$t_i = \begin{cases} 1, & if \;\; (X_i - X_{i-1}) < 0 \\ 0, & otherwise \end{cases}$$

Table 5: Statistical Performance Metrics that evaluate the forecasting accuracy of the tested Models.