UNIVERSITY OF GOTHENBURG
SCHOOL OF BUSINESS, ECONOMICS AND LAW

Master's Thesis in Finance

Comparing forecasts of ARMAs and ANNs on OMXS30

Examining from a economic and statistical point of view

Lars Pilerot & David Waldenbäck Hellman

900328 , 890809

Abstract

Forecasting is of great importance within economics and a vast number of papers have been published on financial forecasting. One of the most used forecasting models in economics is the autoregressive moving average (ARMA). This study compares the ARMA models to artificial neural networks (ANNs). ANNs have proven to be successful in other fields and have increased in popularity with the increase of computing power in recent times. The study includes six different versions of the ARMA model and three different ANNs. These models are examined using statistical and economical measures in order to determine their forecasting performance. The study shows a discrepancy between these two types of performance measures and also shows the difficulty of evaluating a forecast from a financial perspective. The results are inconclusive and dependent on the purpose of the evaluation.

Keywords: *forecasting, artificial neural networks, auto regressive moving average, forecast evaluation*

Supervisor: *Marcin Zamojski*
Graduate School *Master degree project no. 2017:XX*

2017 - 06 - 02

Acknowledgements

We would like to thank our supervisor Marcin Zamojski for his advice and feedback. Marcin's supervision has been invaluable during the process of writing this paper.

Contents

List of Figures

List of Tables

1 Introduction

Forecasting is of great interest in economics and finance. Both governments and firms use forecasting models to guide their decision making processes (Giacomini & White 2006). The importance of forecasting within economics and finance is also reflected in the vast number of publications on the subject — searching for 'forecasting financial time series' on Google Scholar produces approximately 1.2m hits. However, financial time series are among the most difficult to forecast due to their inherent noisy, non-stationary, and chaotic behavior (Tay & Cao 2001). Also, forecasting financial time series is difficult as many factors may play a role, e.g., political events, economic news, traders' expectations etc.; these may causes prices to move (Huang et al. 2005). Further, Van Gestel et al. (2001) state that the signal to noise ratio is low in financial time series, which makes prediction of the next points in the time series challenging.

In this paper, we investigate the performance of two return forecasting models: autoregressive moving average (ARMA) model and artificial neural network (ANN). We use the OMXS30, the index of Swedish blue chip firms, to evaluate the models. The ARMA model is used due to its popularity within time series analysis (cf. De Gooijer & Hyndman 2006) while the ANN is used due its successful application in other complex areas (cf. Kuan & White 1994). Another reason for comparing these models is that they differ in structure, the ARMA belongs to a family of parametric or semi-parametric models and the ANN is a non-parametric model.

ARMA models have been widely used in forecasting time series (cf. De Gooijer & Hyndman 2006, Du Preez & Witt 2003, Ediger & Akar 2007, French et al. 1987, Merh et al. 2011, Dhrymes & Peristiani 1988). The ARMA is regarded as the most efficient forecasting technique in social science and is used extensively (Adebiyi et al. 2014). Its origin is related to Yule (1927) work on stochastic processes and, later, to the introduction of Autoregressive (AR) and moving average (MA) models (cf. De Gooijer & Hyndman 2006). The approach was further popularized by Box et al. (1974).

Artificial neural networks (ANN) have existed for a long time (cf. McCulloch & Pitts 1943) but their popularity increased in the 1990s (Zhang & Hu 1998) as a result of an increase in computing power, which meant that larger models could be used. ANNs are universal and highly flexible function approximators (Kaastra & Boyd 1996) and are classified in a branch of artificial intelligence called 'machine learning' (Gardner & Dorling 1998). ANNs are designed to mimic the human brain's ability to learn and recognize patterns (Vaisla & Bhatt 2010). Zhang & Hu (1998) argue that ANNs are well suited for complex problems where there is a large amount of data available but the solutions require knowledge that may be difficult to specify. They further state that ANNs have the ability to generalize when the data is noisy. However, this comes at a cost of lower tractability compared to ARMAs. In general the tractability of ANNs are low in contrast to the high tractability of ARMAs. Kuan & White (1994) give a few example areas where ANNs have proved to be successful. These include: handwriting recognition, complex coordination tasks, decoding deterministic chaos, diagnosing chest pain, and in the game of backgammon. Some financial applications where ANNs have been used include risk rating of mortgages and fixed income investments, index constructions, market behavior simulations, portfolio selection, identification of economic explanatory variables and others (Kaastra & Boyd 1996).

In contrast to previous studies, this paper examines the forecasts on the OMXS30 from several perspectives, with statistical and economic measures. There are many studies that examine AR-

MAs using economic measures (cf. Metghalchi et al. 2008, Sermpinis et al. 2012, Ojah & Karemera 1999) and due to their popularity and history ARMA models are typically used as benchmarks for other models (cf. Trinkle 2005, Metghalchi et al. 2012). However, there are not as many studies that examine ANNs using economic measures (cf. Zhang et al. 2001, Shambora & Rossiter 2007, Fernandez-Rodrıguez et al. 2000). Four other studies have compared ANNs to ARMAs on financial time series (Adebiyi et al. 2014, Lindemann et al. 2004, Sermpinis et al. 2012, Merh et al. 2011). All with different conclusions. None of the studies have been carried out on the Swedish index. Also, two of the studies have compared the models solely from a statistical perspective. We aim to compare the models using both statistical and economic measures. Hence, the research question in this study is: which model creates the best forecast from a statistical and economic standpoint?

The statistical measures rely on root mean squared error, mean absolute error, correct direction of the forecast as well as the tests proposed by Diebold & Mariano (2002) and Giacomini & White (2006). The economic measures rely on Sharpe ratios and out-of-sample R^2 of trading strategies based on the forecasts, and on cumulative returns. The purpose of these measures is to evaluate whether there is economic value in a model that might not perform well according to statistical measurements. For example Cenesizoglu & Timmermann (2012) and Welch & Goyal (2008) find that standard statistical measures do not always imply that there is economic value in the forecast. In a similar fashion, even models that perform poorly in statistical terms might be useful in economic terms. If a trading strategy based on one of the forecasts would outperform the index, this would be in violation of the (weak) efficient market hypothesis. The efficient market hypothesis is defined by Jensen (1978) as "a market is efficient with respect to information set Q if it is impossible to make economic profits by trading on the basis of information set Q". It would also contradict the random walk hypothesis by Fama (1995) who states that returns are random and can not be forecasted. Although such results cannot be used to reject the efficient market hypothesis but rather challenge it as the efficient market hypothesis does not provide a time-frame nor does the study account for market fictions.[1]

The results in this paper are inconclusive, in some cases ARMA forecasts are superior and in other cases ANN forecasts are superior . The performance of the models are highly dependant on whether the evaluation is conducted using statistical or economic measures. The worst performing model statistically is among the top performers economically. Also, evaluating a model from an economic point of view is concluded to be a highly complex task and visualizations of cumulative returns are deemed to be the most informative. Hence, our results show that there is a discrepancy between statistical and economic measures, but also highlights the difficulty of evaluating a forecast from an economic point of view.

The remainder of this paper is structured as follows: Section 2 describes ANNs and section 3 describes ARMAs. Section 4 reviews theory on forecast comparison and economic evaluation of forecasts. Section 5 describes the data collection and transformation. In section 6 the methodology of the study is described together with the results and analysis, which is followed by discussion in section 6 and conclusion in section 7.

[1] This study does not evaluate the models based on general forecasting ability, but rather the forecasts created by the models during the selected time period. The rationale being that model comparison requires testing of the models on a larger sample and/or on pseudo time series, which is not relevant if the aim is to evaluate during the selected time period of study. Further, the study does not consider intra-day changes in the index and trading strategies assume that the OMXS30 future has the same movements as the underlying index. Also, transaction related costs are ignored.

2 Artificial Neural Network

The following section describes the fundamentals of ANNs, including their intuition, structure and fitting. As ANNs are relatively less known compared to ARMAs, we offer a detailed exposition.

Artificial Neural Networks (ANN) are nonparametric models that are commonly used to forecast time series (cf. Zhang et al. 1998, Sharda & Patil 1992, Kaastra & Boyd 1996). Such nonparametric models (as well as parametric models using robustness checks) can be used for processes where the relationship between input and output variables is unknown and, therefore, hard to fit (Darbellay & Slama 2000). ANNs are based on a structure that is meant to imitate how the human brain processes information (Vaisla & Bhatt 2010) and the parameters of the ANN are estimated given a loss function (Kuan & White 1994). One benefit of ANNs is that they require few prior assumptions about data (Khashei & Bijari 2011).

ANNs are structured into layers. A layer refers to a collection of neurons that are working in parallel (neurons can be thought as coefficients in a regression and will be further explained shortly. Figure 1 shows a 4 layer ANN with 3 neurons in the first layer, 5 neurons in the second and third layers, and 1 neuron in the fourth layer. Many biological networks process information using multiple layers of neurons, which inspired the hidden layers of ANNs (Kuan & White 1994). More layers are capable of more complex processing (Kuan & White 1994). For example the cortex, which is one of the more advanced human processing units, has six processing layers and is therefore able to process a lot more information than, e.g., the knee-jerk reflex (Kuan & White 1994). The knee-jerk reflex is an instant reaction that requires no processing – a straight from input to output reaction. A larger number of layers can be viewed as having multiple, multivariate regressions, one regression in each layer. Each layer treats the preceding layer as input and produces output that is then processed by the succeeding layer (Kuan & White 1994). As in any modelling exercise, ANNs are prone to overfitting.

ANNs consist of an input layer, at least one hidden layer and an output layer (Guide 1995). The input layer is where the input variables, the exogenous variables, enter the model and the layer plays no computational role (Gardner & Dorling 1998). The input layer can be either a scalar or a vector of the input variables (Gardner & Dorling 1998). An input variable is a data point that is being used in the regression, for example traded volume, spread, or traded price while forecasting stock prices. The hidden layer(s) is where the processing of the information takes place and what separates the input of the model from the output of the model (Kuan & White 1994). The output layer is simply a vector or a scalar with the final output of the ANN (Gardner & Dorling 1998).

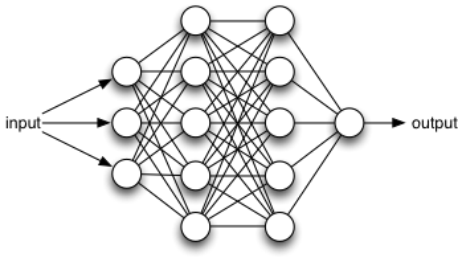

Figure 1: Neural network with four layers, whit 3 neurons in the first layer, 5 neurons in the second layer, 5 neurons in the third layer and 1 neuron in the fourth layer.. Source: Gardner and Dorling, 1998

All hidden layers consists of a set of 'neurons'. A neuron consists of a transfer function (Kaastra & Boyd 1996).[2] The transfer function returns a value or outgoing signal based on the weighted sum of inputs or incoming signals (Guide 1995). Figure 2 illustrates a single neuron.

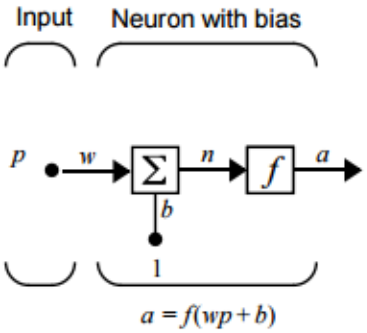

Figure 2: A single neuron. The incoming signal p is adjusted by the weight, w, and a constant, b, is added. These are sent through the transfer function, f, to create the output. Source: Guide 1995

The p represents the input that is multiplied by weight w. Together they form the product wp called a weight function. An optional constant can be added to the weight function (Kaastra & Boyd 1996).The optional constant b and the weight function wp are arguments for the transfer function. The argument is passed through the transfer function and produces output a. The constant b is a weight and works by shifting the function f by a certain amount (Guide 1995), which may be beneficial when fitting the model (LeCun et al. 1998).

Fitting the model is called training in machine learning terms. A network is fitted by minimizing the model's error term through derivation of the transfer functions with respect to the weights. The aim of the fitting is to find weights that produce the lowest errors for each input. Hence, it is similar to estimation of coefficients in a regression. ANNs are usually trained using cross validation (cf. Huang et al. 2004, Kaastra & Boyd 1996, Guide 1995). The use of a cross-validation set helps to weed out overfitting versions of the ANN and improve generalization (Tay & Cao 2001).

Transfer functions can be binary or continuous. Binary transfer functions are simpler to use while continuous functions produce outputs that are more informative. Linear and sigmoidal transfer functions are the most common continuous transfer functions. The sigmoid transfer function is an S-shaped function (Kuan & White 1994) and is what give ANNs their nonlinear ability (LeCun et al. 1998). Another benefit of using a sigmoid transfer function is that it has upper and lower bounds due to its shape, which solves fitting problems that may occur with linear functions (LeCun et al. 1998). The sigmoid transfer function includes the logistic and tanh functions as special cases.

[2]In machine learning terms the constant is called bias, not to be mistaken with biased estimators.

ANNs can be univariate or multivariate, which refers to the number of input variables (Guide 1995). (Kaastra & Boyd 1996) state that raw data should be normalized between the lower and upper bounds of the transfer function, which normally is between -1 and 1. Two of the most common ways of transforming the data is through first differencing and natural logs (Kaastra & Boyd 1996). The most common way to present the input data is by using the neurons of the input layer of the ANN as a rolling window over the time series Huang et al. (2004). As a result, the number neurons in the input layer decides the number of lags. In principle, this provides the same input data as required for an AR(p) regression. As an example, a univariate ANN with five lags on a daily time series could hold days 1-5 in the first sequence and days 2-6 in the next sequence. Sequence refers to how the data is being fed to the ANN. The ANN would therefore constantly be scanning the input data over a time period of five days. A multivariate model works in the same way but for more inputs. Consequently, the number of input neurons decides how much of the past the ANN considers when forecasting.

2.1 Creating and configuring the ANN

Creating and configuring the network refers to choosing architecture of the network. The network can be used for nonlinear regression or pattern recognition (Guide 1995). A network used for nonlinear regression often uses hidden layers of sigmoid neurons and a final hidden layer of linear neurons. A network used for pattern recognition often uses a final hidden layer of a sigmoid transfer function instead since they are more suitable for discrete and binary output values.

An ANN can be static or dynamic. A static ANN, also known as feedforward network, is a network where the output is calculated directly from the input through the connections in the ANN. A dynamic ANN however can, for example, include feedback elements (the output of one layer is fed through the network multiple times), delays, and the input is not necessarily calculated directly through the network. Feedforward ANNs are the most commonly used (Guide 1995, Huang et al. 2004, Kaastra & Boyd 1996).

Deciding on the number of neurons and layers of an ANN is hard (Guide 1995, Huang et al. 2004) and can be compared to deciding the order of ARMA. There are quite a few methods on how to select the starting number of layers and neurons, but there is no consensus (cf. Kaastra & Boyd 1996). To put the number of neurons into perspective Sarle (1994) states that few ANNs have more than a few thousands of neurons while the human brain has about one hundred billion neurons. Huang et al. (2004) advocate trial and error when it comes to the number of hidden layers and neurons.

2.2 Training and validating the ANN

Backpropagation ANN are the most commonly used type (Zhang et al. 1998, Tay & Cao 2001, Kaastra & Boyd 1996). Backpropagation refers to a type of learning algorithm that is often used to fit the ANN (LeCun et al. 1998). Backpropagation works by feeding the network with an input and the resulting output is then compared to a desired output. The comparison is done by calculating an error using a loss function such as mean squared error (MSE). Once the total error has been calculated, the contribution of each neuron to the error is calculated by taking partial derivatives through the network (LeCun et al. 1998). The weight of each neuron is then updated according to its impact on the total error. Backpropagation is thus similar to maximum likelihood estimation or Gaussian mixture model in estimating model parameters through minimizing a cost

function. However, the aim of backpropagation is to minimize the error rather than to maximize the likelihood function through minimizing the error. There are many types of backpropagation algorithms such as Gradient descent, Newton, Quasi-Newton (cf LeCun et al. 1998, Guide 1995). Training or optimization of an ANN may be done in two ways, either with a moving-window or based on the full sample, also called batch traning. The moving-window technique updates the weights after each new input is applied to the network. Hence, estimating the parameters using moving-window technique is similar to using recursive least squares. The batch training is done after the entire sample has been fed through the network. Hence, estimating the parameters using batch training is similar to full sample estimation such as least squares or ordinary least squares. Batch training is faster and normally produces smaller errors (Guide 1995).

Fitting of the model benefits from randomized initialization of weights and biases (Haykin & Network 2004). For example randomizing the weights and biases to numbers between 0 and 1. The randomization ensures that neurons do not receive the exact same output in the beginning, which makes it easier for the backpropagation algorithm to differentiate between the neurons while fitting and as a result makes it easier to find a global minimum.

3 Time-varying mean models

The following section aims to describe the fundamentals of time-varying mean models and more specifically the family of ARMA models, we describe the intuition behind the models, their structure, and fitting.

The autoregressive moving average model (ARMA) was originally proposed by Whittle (1954). Box et al. (1974) developed the model further into the autoregressive integrated moving average (ARIMA). Their idea was to eliminate trend, seasonal and irregular effects by differencing in the beginning of the analysis rather than modelling the different components (of trend, seasonal, and irregular effects) separately (Durbin & Koopman 2012). Box and Jenkins also introduced a coherent and versatile three-stage iterative cycle for identification, estimation and verification known as the Box-Jenkins approach (De Gooijer & Hyndman 2006). Ever since, autoregressive models have been commonly used for forecasting time series (cf. De Gooijer & Hyndman 2006, Du Preez & Witt 2003, Ediger & Akar 2007). There are also many papers applying ARMA to financial time series (cf. Hein & Spudeck 1988, Dhrymes & Thomakos 1998, Downs & Rocke 1983, Öller 1985, Adebiyi et al. 2014, French et al. 1987, Merh et al. 2011, Dhrymes & Peristiani 1988, Metghalchi et al. 2008, Sermpinis et al. 2012, Ojah & Karemera 1999, Trinkle 2005, Metghalchi et al. 2012), which illustrates the popularity and diversity of the model. These papers further show that ARMAs are often used due to being simple and good enough rather than having stellar forecasting ability.

The ARMA model (equation 1) is based on an AR(p) and MA(q) processes, where p is the number of previous lags in the AR processes and q is the number of previous lags in the MA process. The θ represents a constant term.

$$Y_t = \theta + \sum_{i}^{p} \alpha_i Y_{t-i} \sum_{j}^{q} \beta_1 u_{q-1} + \beta_0 u_q \tag{1}$$

The Box-Jenkins approach is widely used for determining what type of process the data follows as well as determining the parameters in the model (De Gooijer & Hyndman 2006). Simplified, the approach is based on three iterative steps, identification, estimation; and diagnostics (Khashei

& Bijari 2011). According to the Box-Jenkins approach the number of lags should be determined by visually analyzing autocorrelation function (ACF) and partial autocorrelation function (PACF) (Gujarati 2009). Other model selection methods than the Box-Jenkins approach are Akaike's information criterion (AIC) described by Shibata (1976), Bayesian information criteria (BIC/SIC) described by Dayhoff et al. (1978) or the minimum description length (MDL) described by Jones (1975), Hurvich & Tsai (1989). The AIC and BIC methods are two of the most used model selection criteria (Yong 2005). Both AIC and BIC are based upon selecting as few model parameters as possible while still producing a good fit. The more complex the model gets the more it will be punished by the AIC/BIC. The underlying rationale is that a more complex model should fit better but the risk of overfitting increases, therefore keeping the models simple is being favoured (Gujarati 2009). Tsay (2005) concludes that there is no evidence that one approach is superior the others, but state that substantive knowledge about the problem under study and simplicity is of importance when determining model specification.

4 Forecast comparison

This section contains a review of the literature regarding forecast comparison deemed relevant to this paper in order to evaluate the forecasts from a economic and statistical view.

As previously stated, forecasting is important when it comes to decision making within economics and finance (Giacomini & White 2006). It is important to distinguish between model and forecast comparison as forecast accuracy may not necessarily reflect the model accuracy (Diebold 2015). Model comparison is more complicated than forecast comparison (cf. Giacomini & White 2006, Diebold 2015). We focus on forecast comparison. Perhaps the most commonly used test, within the area of forecast comparison, is the procedure proposed by Diebold & Mariano (2002). Loss functions, use of samples, as well as statistical and economical performance measures are important aspects when evaluating a financial forecast and we explain them further in the following subsections.

4.1 Loss functions and prediction accuracy

Forecasts are usually evaluated given a chosen loss function. More specifically, Granger (1999) states that the specification of the loss function is an important factor within forecasting. Symmetric quadratic loss functions are often used because of their simplicity (Granger 1999). The optimal forecast of a time series using a symmetric quadratic loss function is the conditional mean or median. However, negative and positive forecast errors are not differentiated by the symmetric quadratic loss function, which may not be reasonable from a financial perspective. In line with Granger's (1999) reasoning, Diebold and Mariano (2002) state that within finance the loss associated with a forecast error is often an asymmetric function of the error and therefore standard statistical tests using symmetric loss functions do not give full insight. For example, if the forecast suggests to go long a stock — i.e., predicts positive returns for the next time period — then the loss of negative forecast errors should be different from the loss of positive forecast errors. Many of the earlier techniques of forecast comparison focus on a specific loss function (cf. Granger & Newbold 2014, Leitch & Tanner 1991), which is limiting if one wants to compare forecasts using different loss functions. The test proposed by Diebold & Mariano (2002), DM-test, compares forecasts by testing for the null hypothesis of zero expected loss differential and is therefore able to compare forecasts using different loss functions. The original DM-test (equation 2 through 5) uses a general

loss function based on squared errors, but can be adjusted for a wide variety of loss functions including asymmetric (equation 6).

$$d = g(e_{1t}) - g(e_{2t}) \tag{2}$$

$$e_{it} = \hat{y}_{it} - y_t \tag{3}$$

$$g(e_{it}) = e_{it}^2 \tag{4}$$

$$H_0 : E(d_t) = 0 \forall t, \quad H_1 : E(d_t) \neq 0 \tag{5}$$

$$g(e_{it}) = e^{\lambda e_{it}} - 1 - \lambda e_{it} \tag{6}$$

4.2 Use of samples

An issue when evaluating performance is that a true out-of-sample performance test, i.e., live testing is often not feasible or realistic as it can take a long time to get results (Diebold 2015). However, in-sample performance does not necessarily imply good out-of-sample performance (Giacomini & Rossi 2008). Also, previous out-of-sample performance from a forecast does not ensure further out-of-sample performance. This may be due to; overfitting, misspecification of the model, or structural or other changes in dynamic properties of the time series. Further, Stock & Watson (2004) show that out-of-sample performance may change if parts of the full out-of-sample are considered in isolation. The DM-test has been further developed by West (1996), Clark & McCracken (2001), Giacomini & Rossi (2008) to better predict out-of-sample performance using pseudo-out-of-sample procedures (POOS). However, many of these updated DM-tests focus on comparing models rather than forecasts. Further, POOS may inform more about particular episodes within the full sample, which is often important within financial economics (Diebold 2015). For example, Pardo (2008) stresses the need for stability throughout the full sample rather than high returns during particular episodes. Giacomini & White (2006) extend the DM-test to include additional variables that may explain the loss differential. Thereby turning the unconditional DM-test into a conditional test. In contrast to DM-test, the Giacomini & White (2006) test also indicates which forecast is better given some significance level.

4.3 Statistical versus economic measures of accuracy

Welch & Goyal (2008) and Cenesizoglu & Timmermann (2012) show that standard statistical accuracy measures such as the mean squared error (MSE) do not necessarily correlate with economic forecasting performance. Cenesizoglu & Timmermann (2012) elaborate on the difference between conventional predictability measures and economic return measures by stating that the economic value, for mean and variance investors, is dependent on the movement in the full return distribution with weights that depend on the utility function. They find that simple models often perform better out of sample when measured using MSE than more complicated models. However, the more complicated models often produce more accurate probability density forecasts. Cenesizoglu & Timmermann (2012) conclude that many of the models they test underperform statistically to the benchmark, but outperform economically in terms of risk-adjusted returns and Sharpe ratios.

The Sharpe ratio measures excess return divided by portfolio volatility (equation 7) (Bodie et al. 2011). A higher Sharpe ratio indicates higher return per unit of risk compared to a lower Sharpe ratio, the ratio can be negative in cases where the portfolio value has decreased or not beaten the risk free return. Cenesizoglu & Timmermann (2012) stress the importance of focusing on economic measures of performance and suggest that return predictability has been too focused on MSE and R^2.

$$S = \frac{E[R_p] - r_f}{SD(R_p)} \tag{7}$$

Cenesizoglu & Timmermann (2012) find a positive correlation with low predictive power between the root mean squared error (RMSE) and the economic value of a forecast. Also, Campbell & Thompson (2008) show that a small increase in R^2 can have a large economic impact and can lead to substantial benefits for investors. Campbell & Thompson (2008) suggest that out-of-sample R^2 can give more meaningful economic insight if compared with the squared Sharpe ratio S^2 as illustrated in equation (9). A positive R^2/S^2 implies that the forecast can be used, by a mean-variance investor, to obtain higher portfolio returns (Campbell & Thompson 2008).

$$R^2_{OS} = 1 - \frac{\sum_{t=1}^{T}(r_i - \hat{r_i}^2)}{\sum_{i=1} T(r_i - \bar{r_i}^2)} \tag{8}$$

$$\frac{R^2_{OS}}{S^2} \tag{9}$$

Out-of-sample R^2 is illustrated in equation (8) where $\hat{r}$ is the fitted value from the predictive regression and $\bar{r}$ is the historical average return over the same period (Campbell & Thompson 2008). If the out-of-sample R^2 is positive, then the forecast has lower average mean-squared prediction error than the historical average returns. Furthermore, Campbell & Thompson (2008) state that since small R^2 can generate large benefits for an investor, large R^2s are unreasonable and most likely signs of spurious regression.

Timmermann (2008) stresses another issue with financial forecasting: the creative self destruction of forecasting models. Timmermann (2008) states that a model may perform well until it is widely discovered and adopted. The issue being that the markets are influenced by the market participants attempt to profit from patterns and thus the market changes over time as types of pattern exploitation get integrated in market prices. For the specific forecasting model this means that it may be profitable for a while, but as it gets more widely adopted its predictive power will cease. Further, Timmermann (2008) concludes that it is difficult to persistently predict stock prices over longer time periods using standard forecasting models. Welch & Goyal (2008) argue that none of the typical predictor variables seem capable of persistently predict stock prices.

5 Data

In this section, we describe the the data set and data collection. The section includes the origin of the data and sample choice as well as the transformation of the original data.

5.1 Dataset and data collection

We use return series from the Swedish OMXS30 index. The index is composed of the 30 stocks with the highest trading volume measured in SEK on NASDAQ Stockholm (Nasdaq, 2016). Notable is that the index is not capped with regards to industries or companies and does not adjust for dividends or cash payouts. However, the OMXS30 is chosen rather than a total return index because the OMXS30 is the underlying for the most traded index future on NASDAQ Stockholm. The price data was retrieved from the Swedish House of Finance.

The sample period is 11 years of daily closing price, starting from 2006-01-02. Seven years of the data is used to fit the models (in-sample) and the remaining 4 years of the data is used for out-of-sample testing. The out-of-sample period is from 2013-01-07 to 2016-12-30. The models are refitted daily during the out-of-sample period using an extending window in order to let the models take new data into account. We use one-period-ahead predictions in our tests. The underlying reason for the amount of data that is used in this study is that the ANNs require a large set of data for fitting while the ARMAs require less. As a result, the ANN becomes the limiting factor when choosing the time frame. However, using a lot of data for fitting may affect the performance of the models negatively as older data does not necessarily reflect current market conditions. More specifically, recent data is preferred since market conditions have been changing since the beginning of the millennium with for example the introduction of electronic market participants. The chosen time frame includes a full business cycle, which ensures that the models experience both bull and bear markets and thus exposes the models to different market conditions.

5.2 Transforming data

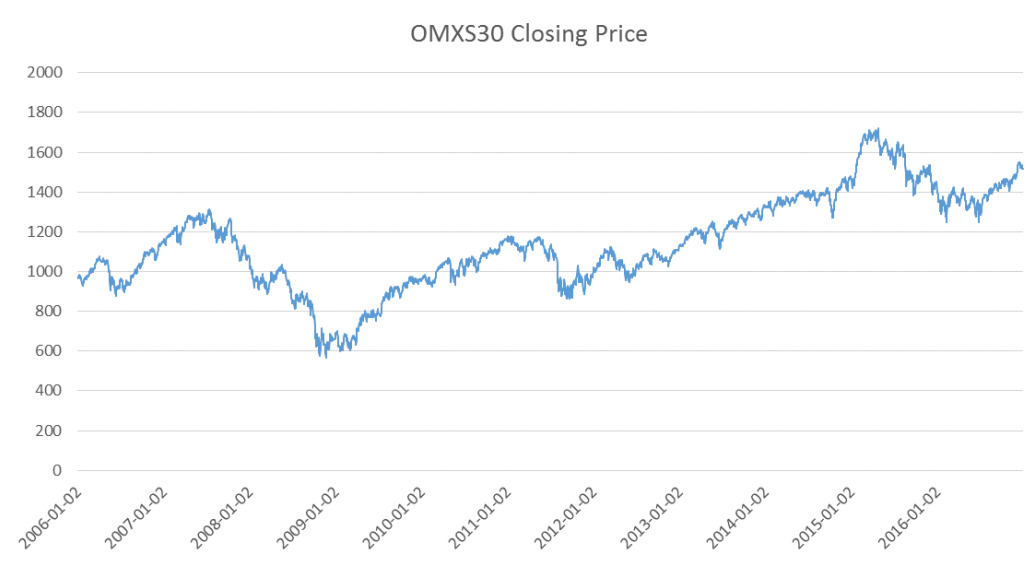

Figure 3: Daily closing price of the OMXS30 index from 2006-01-02 to 2016-12-30.

Figure 3 illustrates that the original data is not stationary. An Augmented Dickey-Fuller test fails to reject the null of a unit root. However, an underlying assumption of the ARMA is that the time series should be stationary. We model log returns as the natural logarithm stabilizes the variance while differencing reduces the effect of trends and cycles.

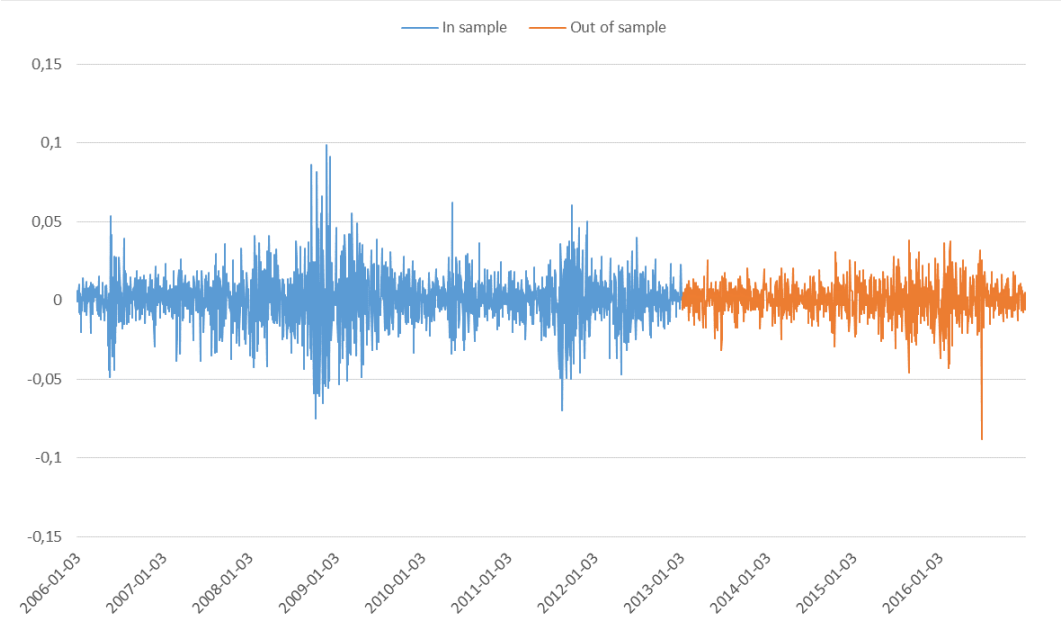

Figure 4: OMXS30 logged values Closing price from 2006-01-03 to 2016-12-30. The period used for initial fitting of the models marked with blue and the period used for the extending-window forecasts is marked with orange

Figure 4 shows the transformed OMXS30 log returns. Figure 5 and 4 indicate that the data is stationary and figure 6 also suggests that there is no clear lag pattern. An Augmented Dickey-Fuller test rejects the null of a unit root (p-value 0.0001). The blue line in figure 4 shows the in-sample data and the orange line shows the data that is used for the out-of-sample extending window forecasts. An F-test shows that there is a significant difference (p-value of 0) between the unconditional variance in the in-sample and out-of-sample periods. In-sample variance is 0.00027 and out-of-sample variance is 0.00012. A Ljung-Box test on returns rejects the null that the returns are not autocorrelated (p-value of 0,001). Further, a Ljung-Box test on squared returns indicates that there are significant ARCH effects (p-value of 0) in the residuals of the returns, and an Engle's ARCH test rejects the null hypothesis of no residual heteroskedasticity (p-value of 0). One way to deal with the decreasing variance over time and conditional heteroskedasticity is to use an ARMA-GARCH or possibly GARCH-M model. Hence, a GARCH-extension is added to the best fitting ARMA in-sample.

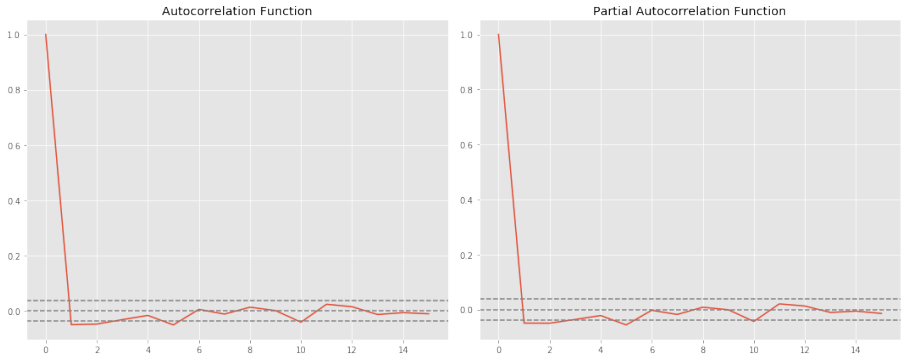

Figure 5: Autocorrelation Function of logged values for 15 days and Partial Autocorrelation Function of logged values for 15 days.

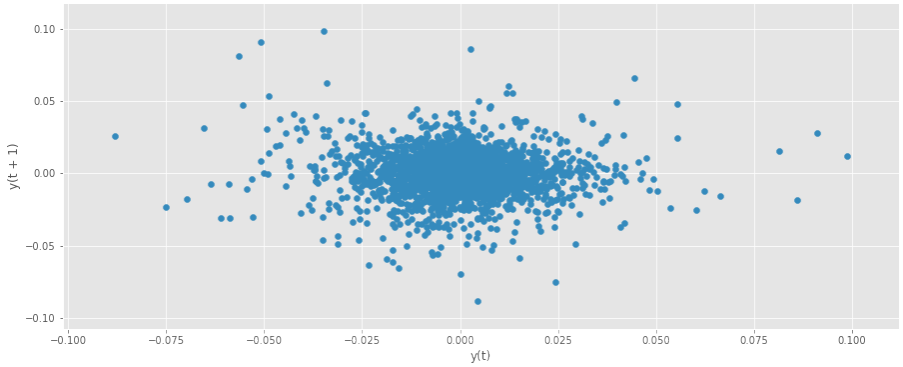

Figure 6: Lag plot of transformed OMXS30

6 Methodology, Results and Analysis

In this section, we describe the methodology, our results and analysis. The standard statistical measures subsection describes how errors are measured throughout the paper. The ANN subsection includes the fitting of the ANN models and the ARMA subsection includes the fitting of the ARMA models. The out-of-sample subsection includes the general results and interpretation from the out-of-sample forecasts followed by the tests conducted on these forecasts. These tests are symmetric DM-test, asymmetric DM-test, unconditional and conditional Giacomini & White (2006). The final subsection includes the results and interpretation from trading strategies based on the forecasts and includes financial measures proposed by Campbell & Thompson (2008) as well as cumulative returns.

6.1 Standard statistical measures

We use root mean squared error (RMSE) and mean absolute error (MAE) to measure the error terms in this paper. MAE and RMSE are among the most commonly used statistical measures, see equation 10 and equation 11.

$$MAE = \frac{1}{n}\sum_{t=i}^{p} \mid x_t - \hat{x}_t \mid \tag{10}$$

$$RMSE = \sqrt{\frac{\sum(e_i)^2}{N}} \tag{11}$$

6.2 ANN

We use three ANNs in this paper – 1 neuron with 1 lag, 10 neurons with 5 lags, 30 neurons with 20 lags as these performed best during in-sample testing. We find that larger networks do not produce quantitatively different results. The ANNs are all feedforward, univariate with two hidden layers – one layer with sigmoidal transfer functions followed by a layer with a linear transfer function, which is in line with Beale et al. (2016) who state that this type is the most common for non-linear regression and is why the non-linear part is important.

It is common to split the data into training and validation sets that contain 90% vs 10% of data respectively (Zhang & Hu 1998), and since there is no general methodology we split the data in a similar manner. The data is split using the 'dividerand' function in the Neural Network Toolbox in MATLAB. 'Dividerand' divides the fitting data into random sets of training and validation. 'Dividerand' is beneficial in the way that the most recent data is used for both training and validation rather than validation only. For example, if the fitting data was divided by blocks using the same ratio of training and validation, then the validation set would equate to the full final year before the out-of-sample testing, which means that the data that is likely most relevant for fitting the model for the out-of-sample testing would be used only for validation. To the best of our knowledge 'dividerand' resamples the data with the dynamic properties of the time series preserved. Furthermore, 'dividerand' decreases the RMSE heavily compared to blockwise fitting.

We use Levenberg-Marquardt algorithm for backpropagation and the training is done in batch-form. The Levenberg-Marquardt is the standard algorithm for moderate-sized feedforward neural networks according to Guide (1995). The choice of the algorithm does not change the results. The ANNs are fitted based on minimizing validation set MSE. The weights and biases are randomized between 0 and 1 so that the initialization is asymmetric.

The training set and validation set RMSE differ every time the model is fitted due to dividerand and randomization of weights and biases. Hence, an average based on 100 iterations of model fitting is presented in table 1 to give an indication of RMSE values. Both of the larger models had higher validation set RMSE than training set RMSE while the 1 neuron 1 lag model had lower validation set RMSE than training set RMSE.

ANN	Full in-sample RMSE	Training set RMSE	Validation set RMSE
1 neuron1 lag	1.65	1.65	**1.63**
10 neuron 5 lag	1.62	1.62	1.67
30 neuron 20 lag	**1.58**	**1.56**	1.78

All numbers are expressed in centesimal

Table 1: Fitting of ANN

In the full sample 30 neurons 20 lags ANN produces the lowest RMSE among all ANNs. Also, the 30 neuron 20 lag ANN has the smallest training set RMSE, but highest validation set RMSE. The 1 neuron 1 lag ANN has the highest full-sample RMSE as well as training set RMSE but lowest validation set RMSE.

6.3 ARMA

We fit six ARMA models: ARMA(1,0), ARMA(1,1)-GARCH(1,1), ARMA(1,1), ARMA(2,1), ARMA(1,2), and ARMA(5,0). The number and types of ARMA models are chosen to ensure a broad spectra of model specifications similar to the ANNs. The main reason is to test if there is a difference between ANNs and ARMAs that have similar lag lengths - ARMA(1,0) and ARMA(5,0) compared to the 1 neuron 1 lag ANN and 10 neuron 5 lag ANN. The ARMA(1,1), ARMA(2,1) and ARMA(1,2) are selected due their low AIC and BIC. The ARMA(1,1)-GARCH(1,1) model is selected to counter varying volatility in the sample. The rationale behind using many models is also that in-sample performance may not correspond with good out-of-sample performance.

Unsurprisingly, the larger ARMA models have lower in-sample RMSE than the smaller models as Table 2 illustrates. However, both AIC and BIC in Table 2 indicate that the ARMA(1,1)-GARCH(1,1) is the best model, followed by the (1,1), (2,1) and (1,2). Noticable is that the ARMA(1,1)-GARCH(1,1) stands out with lower AIC and BIC.

ARMA	In-sample RMSE	AIC	BIC
(1,0)	1.646	-9.47	-9.45
(1,1)-GARCH(1,1)	0.165	**-10.00**	**-9.98**
(1,1)	1.640	-9.48	-9.45
(2,1)	1.640	-9.47	-9.45
(1,2)	1.639	-9.47	-9.45
(5,0)	**1.638**	-9.47	-9.44

In-sample RMSE are expressed in centesimal. AIC and BIC are expressed in thousands.

Table 2: Fitting of ARMA

As table 1 and table 2 show, the 30 neurons 20 lags ANN has the lowest in-sample RMSE of all models. The 10 neuron 5 lag ANN has the second lowest in-sample RMSE of all models followed by the ARMA(5,0),ARMA(2,1),ARMA(1,2),ARMA(1,1) and ARMA(1,0) that are all in a small range. The ARMA(1,1)-GARCH(1,1) has the highest in-sample RMSE of all models.

6.4 Out-of-sample testing

Below are two representative plots of the 1000 day out-of-sample testing of the models. The representative plots show the ARMA(1,1) and the 10 neuron 5 lag ANN to include one of each model types. The forecasts are illustrated in blue and compared to the actual movements of the OMXS30 index in red, for all models see Appendix B. Noticeable is that the forecasts have a narrower range than the actual returns.

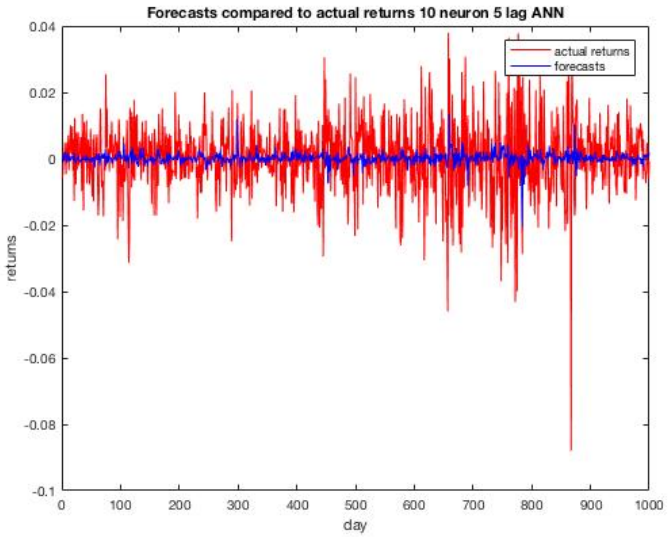

Figure 7: Forecasts compared to actual returns, 10 neurons 5 lag ANN

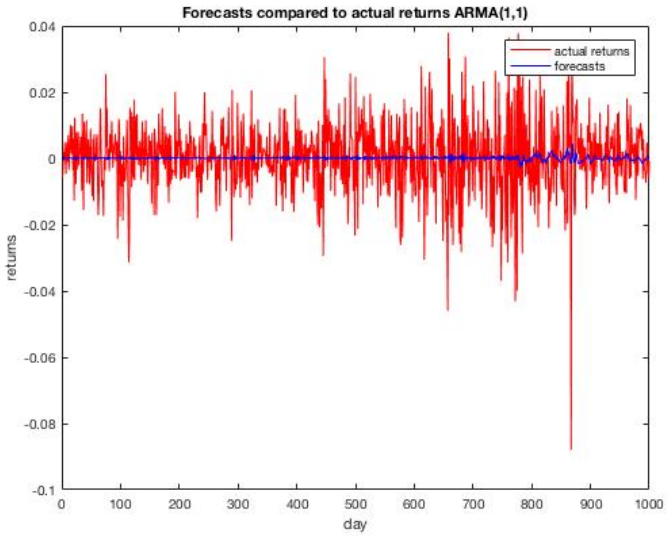

Figure 8: Forecasts compared to actual returns, ARMA(1,1)

6.5 Tests

This subsection compares the performance of the competing forecasts and models using the DM-test proposed by Diebold & Mariano (2002), the test proposed by Giacomini & White (2006), the Sharpe ratio and out-of-sample R^2 test proposed by Campbell & Thompson (2008), a directional test and standard statistical measures such as mean absolute error (MAE) and root mean squared error (RMSE). Two different loss functions are used in the DM-test for robustness purpose, one symmetric and one asymmetric. Further, four trading strategies, one without threshold and three with thresholds are tested based on the direction of the forecasts.

6.5.1 Statistical measures

The directional test is a measure to see how often the forecast has the correct direction. The directional test does not take the magnitude of the forecast or actual movement into consideration, i.e., a forecast predicting a small positive change is considered to be right even when the actual change is larger, and vice versa. It is of interest to measure how many times the models can forecast the direction correctly as it may be argued that the direction of the return is more important than the error of the return for financial time series. Also, directional accuracy is used as a base for trading strategies in subsection 6.5.2.2 and 6.5.2.3 Furthermore, a t-test is used to determine whether there is statistical difference between the directional test scores.

ARMA	Out-of-sample RMSE	Out-of-sample MAE	Correct direction
(1,0)	1.097	0.806	0.478
(1,1)-GARCH(1,1)	1.097	**0.802**	**0,528**
(1,1)	**1.094**	0.803	0.509
(2,1)	1.097	0.806	0.478
(1,2)	1.097	0.806	0.478
(5,0)	1.103	0.811	0.482

Out-of-sample RMSE and out-of-sample MAE are expressed in centesimal. Correct direction is not rescaled.

Table 3: Out-of-sample testing of ARMA

The out-of-sample RMSE and MAE are similar for all ARMA models. However, as table 3 illustrates the ARMA(1,1)-GARCH(1,1) model distinguishes itself with lowest out-of-sample and the best prediction accuracy. The ARMA(1,1)-GARCH(1,1) is significantly better at prediciting the correct direction compared to ARMA(1,0), ARMA(2,1), ARMA(1,2), and ARMA(5,0) at 10% level. However, there is no significant difference compared to the ARMA(1,1). The (5,0) has both the highest out-of-sample RMSE and MAE. However, the higher errors do not affect the model's ability to predict the correct direction as it does not differ statistically to the others with the exception of ARMA(1,1)-GARCH(1,1).

ANN	Out-of-sample RMSE	Out-of-sample MAE	Correct direction
1 neuron per layer, 1 lag	**1.100**	**0.808**	**0.514**
10 neurons per layer, 5 lags	1.106	0.813	0.501
30 neurons per layer. 20 lags	1.142	0.850	0.510

Out-of-sample RMSE and out-of-sample MAE are expressed in centesimal. Correct direction is not rescaled.

Table 4: Out-of-sample testing of ANN

The ANN models differ more when it comes to RMSE and MAE as table 4 illustrates. The 1 neuron 1 lag ANN has the lowest out-of-sample RMSE and out-of-sample MAE of the ANNs. There is no significant difference between ANNs regarding the correct direction.

When comparing the ANNs with the ARMAs, the ARMA(1,1) and ARMA(1,1)-GARCH(1,1) model outperform all other models in terms of RMSE and MAE. Further, in general the ARMAs outperform the ANNs in terms of RMSE and MAE by for example placing the ARMA(1,0), ARMA(1,1), ARMA(1,2) and ARMA(2,1) ahead of the 1 neuron 1 layer ANN in terms of er-

ror. As previously stated the ARMA(1,1)-GARCH(1,1) is significantly better than ARMA(1,0), ARMA(2,1), ARMA(1,2) and ARMA(5,0) regarding the correct direction. However, there are no other significant differences between any of the models regarding the correct direction. A constant forecast of 0% returns is also tested due to the fact that many of the smaller models produce small forecasts (close to 0) and have lower errors than the larger models. Such a model has a RMSE of 1.098 and MAE of 0.805 and hence has the third lowest error of all models after the ARMA(1,1) and ARMA(1,1)-GARCH(1,1).

6.5.1.1 Diebold and Mariano (1995) tests

The DM-test is based on squared errors and should be interpreted as a two-sided hypothesis test where the hypothesis is that both forecasts have equal losses. The alternative hypothesis is that the losses differ between two forecasts. In cases where the null hypothesis can be rejected it merely states that the losses associated with the forecasts are not the same. Hence, further analysis is required to draw conclusions on which forecast is superior e.g. interpreting loss function values. In this thesis, interpreting loss function values is done using the Giacomini & White (2006) test (see subsection 6.5.1.2). For full results from the DM-test see Appendix D.

The DM-test test finds no statistical differences between the ARMAs. However, it does find that there is a statistical difference between the 30 neuron 20 lag and the other ANNs. The difference between the 30 neuron 20 lag ANN and the 10 neuron 5 lag ANN is significant at 5% level and the difference between the 30 neuron 20 lag ANN and the 1 neuron 1 lag ANN is significant at 1% level. Further, the DM-test finds a difference between the 30 neuron 20 lag ANN and all the ARMAs. This is significant at the 5% level. An asymmetric loss function is also used to test the robustness of the DM-test statistic (see Appendix D). However it does not provide any additional information regarding the similarities and differences of the models.

6.5.1.2 Giacomini and White (2006) test

The Giacomini & White (2006) test is similar to the DM-test, but tests the conditional predictive ability and the unconditional predictive ability while the DM-test only compares unconditional predictive ability. The Giacomini & White (2006) test also does not assume that estimation errors are removed and hence allows for non-stationarity (Giacomini & White 2006). A positive sign of the test statistic indicates that forecast B is better than A and a negative sign indicates the opposite. For full results from the Giacomini & White (2006) test see Appendix D.

The unconditional Giacomini & White (2006) test finds that the ARMA(5,0) is statistically worse than ARMA(1,1) at a significance level of 5%. The test also shows that the 30 neuron 20 lag ANN is worse than all other models. The 30 neuron 20 lag ANN is worse than 10 neuron 5 lag ANN at significance level of 5% and the 1 neuron 1 lag ANN at significance level of 1%. Also, all ARMAs beat the 30 neuron 20 lag ANN at significance level of 1%. Further, there is no statistically significant difference between the two smaller ANNs and the ARMAs.

The conditional Giacomini & White (2006) test shows, in addition to results similar to the unconditional test, that the ARMA(1,1) is statistically better than the other ARMAs and that the ARMA(5,0) is statistically worse than all the other ARMAs. However, the ARMA(1,1) does not have a statistical difference to the 10 neuron 5 lag ANN nor the 1 neuron 1 lag ANN.

6.5.1.3 Analysis of statistical measurements

From a statistical point of view the ARMA(1,1), 1 neuron 1 lag ANN and the 10 neuron 5 lag ANN are the best models while the 30 neuron 20 lag is the worst model. We find differences in out-of-sample errors that point towards ARMA(1,1) and ARMA(1,1)-GARCH(1,1). However, the conditional Giacomini & White (2006) test shows that ARMA(1,1) is significantly better than ARMA(1,1)-GARCH(1,1). Further, the formal tests do not find any significant differences between the ARMA(1,1), 1 neuron 1 lag ANN and 10 neuron 5 lag ANN. Also, none of these models have significant differences in accuracy. Further research is needed to determine if these numerical and statistical results change as the sample is increased.

None of the models have significant differences in accuracy except for the ARMA(1,1)-GARCH(1,1) that is statistically superior to ARMA(1,0), ARMA(2,1), ARMA(1,2) and ARMA(5,0). All models predict the correct direction around 50% of the time and are thus similar to tossing a coin. Further, the fact that the 0 forecast is similar to the best performing forecasts error-wise suggests that the models that make low or no predictions perform better and the models that make larger predictions perform worse, which can be extended to state that it is impossible to predict the future and thus assuming that prices follow a random walk.

The reason why the DM-tests, unconditional and conditional Giacomini & White (2006) tests do not find a difference between most of the smaller models may be that the models produce small predictions causing the errors to be small, which in turn makes it hard to differentiate between the models. Further, the standard DM-test and Giacomini & White (2006) tests evaluate models based on squared loss functions, which could partly explain why the tests favor the models with lower forecasts. Since the tests are based on squared errors, forecasts with larger predictions are punished more severely compared to forecasts with lower predictions, given the relative lower volatility in our out-of-sample period. As illustrated in Appendix B there is a large difference in the strength of the forecasts between the 30 neuron 20 lag ANN and the ARMAs. Since the 30 neuron 20 lag ANN make strong predictions it is punished severely by squared errors when it is wrong. The reason why the asymmetric DM-test is not able to draw any further conclusions than the standard DM-test may be explained by that fact that all models tend to underpredict compared to the actual movements of the index i.e. the models rarely overpredict. As such, the contribution of an asymmetric DM-test that punishes underprediction more than overprediction is limited in this study.

The error-argument is strengthened by the fact that the 0-forecast, a constant forecast of 0 percent change every day, would have second lowest errors of all models and neither DM-test nor unconditional Giacomini & White (2006) test can find a statistical difference between the 0-forecast and any of the models except the 30 neuron 20 lag ANN (the 0-forecast is better) as shown in in Appendix C. Further, the conditional Giacomini & White (2006) test shows that the zero forecast is better at a 10% significance level than the ARMA(0,1) and better at a 5% level than the ARMA(5,0) and 30 neuron 20 lag ANN. From a financial perspective the 0-forecast adds no value and the low errors of the 0-forecast indicates a strong need to evaluate forecasts using financial measures rather than statistical measures.

6.5.2 Economic measures

We use Sharpe ratios and Campbell & Thompson (2008) test of two types of trading strategies as economic measures. Sharpe ratio is computed using zero interest rate as risk free. The daily STIBOR 3-month rate adjusted to daily returns over the out-of-sample period is negative, however most investors do not have the possibility to borrow at negative rates, instead zero interest rate is used.

Our trading strategies are based on directional trading and long/short positions meaning that we allow bets in both directions. The trading signal is based on the direction and magnitude of the forecast and the trading strategy is to take a long position when the forecast indicates positive returns and take a short position when the forecast indicates negative returns. The position is opened at the daily closing price and closed at the next day closing price. Hence, the gain or loss of the position is therefore equal to the actual movement of the index the following day. Transaction costs are ignored. The trading strategies are based on full reinvestment in each trade with a starting value of 1. The first strategy takes a long or short position in the OMXS30 every day based on the forecast.

The trading strategies are analyzed using the Campbell & Thompson (2008) test and cumulative returns. The Campbell & Thompson (2008) test includes the out-of-sample R^2, the Sharpe ratio and the R^2/S^2. The R^2 to S^2 ratio gives insight into how the predictive ability of the forecast, out-of-sample R^2, can be used to increase portfolio returns for mean-variance investors. Campbell & Thompson (2008) compare out-of-sample R^2 with Sharpe ratios to specify the percentage that the overall portfolio returns can be increased by including the forecast in the portfolio.

6.5.2.1 Trading system evaluation

When evaluating trading systems there are a few key characteristics of a robust system. Overall profit is not the sole measure of a system, since it may be generated through an insufficient number of trades or the system may have unacceptable drawdowns (Pardo 2008). A successful system should exhibit even distribution of trades, even distribution of profits, acceptable risk, and statistical validity according to Pardo (2008). There are a number of ways to measure these characteristics and in this paper the focus on the visual properties of equity curves (cumulative returns). Equity curve stability refers to the distribution of trades, contribution of each trade, number of trades, and drawdowns of a system. A system where the majority of the overall profit is generated by a single trade is not robust since that trade might be unrepeatable, e.g. shorting the 'Black Monday' in october of 1987[3] (Pardo 2008).

6.5.2.2 No threshold trading strategy

This section includes the results from the Campbell & Thompson (2008) test on the different trading strategies, which includes Sharpe ratio, out-of-sample R^2, and R^2/S^2 ratio. Cumulative returns for each model for all the trading strategies can be seen in Appendix A.

The basic trading strategy based upon taking a directional bet in the same way as the prediction results in that six models have positive Sharpe ratios - ARMA(1,1), ARMA(1,1)-GARCH(1,1),

[3]Black Monday refers to 19th of october 1987, when world markets crashed, the Dow Jones Industrial Average declined 22.61% in a single trading day.

No thresholds									
Model	30N20L	10N5L	1N1L	(1,0)	(1,1)-GARCH(1,1)	(1,1)	(2,1)	(1,2)	(5,0)
Overall return	**1.524**	1.123	1.370	0.852	1.256	1.403	0.888	0.888	1.045
Sharpe Ratio	**0.783**	0.364	0.618	-0.041	0.505	0.670	-0.041	-0.041	0.216
Out-of-sample R^2	-0.718	**-0.523**	-0.691	-1.432	1	-0.544	-1.432	-1.432	-0.711
R^2/S^2	**-296**	-994	-456	-213280	984.91	-305	-213280	-213280	-3870
No. Trades	1000	1000	1000	1000	1000	1000	1000	1000	1000

Table 5: Results from trading strategy with no threshold.

ARMA(5,0) and all of the ANNs. The ARMA(1,1)-GARCH(1,1) has a positive out-of-sample R^2, which yields a positive ratio R^2/S^2 of 984.91 indicating that the overall portfolio returns could be increased by including the forecast. The equity curve of ARMA(1,1)-GARCH(1,1) is similar to that of the OMXS30 – it shows high and stable returns during the first 600 days, however almost all accumulated profit is lost during the following 300 days (see Appendix A). The out-of-sample R^2 is negative for all other forecasts indicating that the return is random, thus also yielding a negative ratio of R^2 to S^2. All strategies based on the ARMA models exhibit unstable equity curves. The equity curves for the ANNs exhibit random patterns with the exception of the 30 neuron 20 lag ANN that shows a positive drift, see Appendix A. The 30 neuron 20 lag ANN also has the highest overall return. As a result the 30 neuron 20 lag ANN is deemed best from this perspective.

The ARMA models have lower out-of-sample RMSE compared to in-sample RMSE. This is usually not the case, however this can be explained by the decrease in variance in the out-of-sample period compared to the in-sample period. One argument behind the decrease in variance is that the variance increased during the financial crisis of 2008 and continued to be high the following years. Another possible explanation is the increase of market participants through computerised trading causing lower volatility, however papers on microstructure tend to disagree on the subject (cf. Zhang 2010, Brogaard et al. 2010). Also, there is no proof for these arguments in this study. Since the ARMA models tend to predict small changes, the error becomes larger in times where the general variance is high and vice versa in times where the general variance is small, which can be seen in the figure 7 and 8 of forecasts against the actual returns.

6.5.2.3 Threshold trading strategies

A threshold for the predicted change is used for increased precision in the long/short directional strategy. This means that the system requires a stronger signal in order to take a position. Our hypothesis behind this strategy is that when the prediction anticipates a sharp move in returns the forecast is more likely to be directionally correct compared to a prediction of a smaller movement, thus leading to higher accuracy and performance. The thresholds are based upon the distribution of absolute values of in-sample forecasts, and set to the 25th, 50th and 75th percentiles.

We set the lowest threshold at the 25th percentile. In this case, all models generate trading signals. However, the ARMA(2,1) and ARMA(1,2) only have 13 and 20 trades respectively. The models that have few trades also experience decreasing activity with time (see for example ARMA(1,0)), indicating inconsistent trading and thus inconsistent returns. All models have positive Sharpe ratios except for ARMA(1,0). None of the models have positive out-of-sample R^2 which according to the Campbell & Thompson (2008) test suggests spurious regression and the returns to be random, the ARMA(1,1)-GARCH(1,1) is an exception with positive out-of-sample

Threshold 25th percentile

Model	30N20L	10N5L	1N1L	(1,0)	(1,1)-GARCH(1,1)	(1,1)	(2,1)	(1,2)	(5,0)
Overall return	**1.562**	1.063	1.327	0.961	1.231	1.452	0.082	0.081	1.124
Sharpe Ratio	0.851	0.255	0.606	-0.087	0.4776	0.821	**0.986**	0.836	0.345
Out-of-sample R^2	-0.637	-0.310	-0.516	-1.083	**0.9959**	-0.313	-0.024	-0.040	-0.438
R^2/S^2	-222	-1197	-354	-35708	**11000**	-117	-6	-14	-930
No. Trades	844	713	803	720	**996**	372	13	20	679

Table 6: Results from trading strategy with threshold at 25th percentile.

R^2. The ARMA(1,1)-GARCH(1,1) is almost identical to that of no threshold as only 4 trades have been removed, but performing slightly worse in terms of overall returns and Sharpe ratio than without thresholds. The equity curves at this threshold are volatile (see Appendix A) except for the ARMA(1,1). The 1 neuron 1 lag ANN shows unstable performance with high returns the first 200 trading days then unstable performance from day 200 until day 850 where it has high returns once again closing at a positive overall return. This suggests a weak trading strategy since short periods of time have large impact on the overall performance. The 30 neuron 20 lag ANN also has a volatile equity curve, but shows a positive drift. The 30 neuron 20 lag ANN exhibits a rather smooth increase of returns until day 650, where it shows poor performance for 100 days and then similar to the 1 neuron 1 lag ANN, performs well the last 250 days. The 30 neuron 20 lag ANN is the best performer of all models given the threshold 25th percentile threshold with a cumulative return of 56%. Second best is the ARMA(1,1) with a cumulative return of 45%.

Threshold 50th percentile

Model	30N20L	10N5L	1N1L	(1,0)	(1,1)-GARCH(1,1)	(1,1)	(2,1)	(1,2)	(5,0)
Overall return	**1.735**	1.311	1.197	1.076	0.980	1.372	1	1	0.880
Sharpe Ratio	**1.085**	0.662	0.460	0.184	-0.410	0.938	0.000	0.000	-0.146
Out-of-sample R^2	-0.473	-0.285	-0.279	-0.292	0.003	-0.245	-0.001	-0.001	**0.018**
R^2/S^2	-101	-164	-332	-2173	5.649	-70	0	0	**213**
No. Trades	**641**	498	581	185	7	96	0	0	362

Table 7: Results from trading strategy with threshold at 50th percentile.

The 50th percentile threshold produces positive Sharpe ratios for all models that are trading except the ARMA(5,0) and ARMA(1,1)-GARCH(1,1). The ARMA(2,1) and ARMA(1,2) never trade given this threshold and ARMA(1,1)-GARCH(1,1) only has 7 trades, which shows that the forecasts produced by these models are very narrow compared to in-sample period. All models have negative out-of-sample R^2, with exception of the ARMA(5,0) and the ARMA(1,1)-GARCH(1,1) which also results in a positive R^2/S^2. However, as previously mentioned the ARMA(5,0) and ARMA(1,1)-GARCH(1,1) have negative Sharpe ratio and therefore it is not relevant to analyze the R^2/S^2 further. Further, the equity curve of the ARMA(5,0) displays instability and negative drift. Noticeable is that the 30 neuron 20 lag ANN shows a cumulative return of 73% and scores the highest Sharpe ratio of all models in this study at 1.08. The equity curve of the 30 neuron 20 lag ANN shows high returns in the last 250 days of trading, which could be due to higher volatility during that period. The 30 neuron 20 lag ANN has 641 trades. The equity curve for the ARMA(1,1) shows positive results, but only has 96 trades Overall the equity curves for the 50th percentile trading strategy shows similarities to the 25th percentile strategy (Appendix A). The 30 neuron 20 lag ANN implies robustness by increased overall return. Further, the ARMA(1,1) has the highest return of all ARMAs but lower compared to 25th percentile strategy. The ARMA(1,1)

and 30 neuron 20 lag ANN are considered the best models at the 50th percentile threshold as they have the highest Sharpe ratios, highest returns and best looking equity curves.

Threshold 75th percentile

Model	30N20L	10N5L	1N1L	(1,0)	(1,1)-GARCH(1,1)	(1,1)	(2,1)	(1,2)	(5,0)
Overall return	**1.347**	1.023	1.034	1	1	1.019	1	1	0.998
Sharpe Ratio	**0.721**	0.156	0.191	0	0	0.153	0	0	0.085
Out-of-sample R^2	-0.271	-0.161	-0.072	-0.001	-0.001	-0.028	-0.001	-0.001	**0.068**
R^2/S^2	-131	-1676	-498	0	0	-297	0	0	**2384**
No. Trades	**387**	219	377	0	0	26	0	0	123

Table 8: Results from trading strategy with threshold at 75th percentile.

The ARMA(1,0), ARMA(1,1)-GARCH(1,1), ARMA(1,2) and ARMA(2,1) are not trading given this threshold. However all trading models have positive Sharpe ratios. All models with the exception of ARMA(5,0) experience decreasing Sharpe ratios at the 75th percentile threshold compared to the 50th percentile threshold. Conversely, the ARMA(5,0) has a positive Sharpe ratio at the 75th percentile compared to a negative Sharpe ratio at 50th percentile. The ARMA(5,0) also has positive out-of-sample R^2 - however, the Sharpe ratio is close to zero and the equity curve displays an uneven distribution of trades and randomness. The ARMA(1,1) produces profit but does not trade a lot. The 30 neurons 20 lags ANN has the best Sharpe ratio at 0.72 but once again negative out-of-sample R^2. The only model that shows equity curve stability among the ANNs is the 30 neurons 20 lags ANN with an overall return close to 34%. However, the 30 neuron 20 lag ANN has worse performance compared to previous thresholds. The best performing model with the given threshold is the 30 neuron 20 lag ANN with the highest Sharpe ratio, highest cumulative returns, most stable equity curve, and relatively large number of trades.

In order to further test the robustness of the forecasts figure 9 shows the arithmetic average return per trade (ARPT) given 99 different threshold strategies for each model. The different strategies represent trading thresholds based on every percentile of the in-sample forecasts. Strategies with less than 100 trades are not included. Given a higher threshold the weak signals are weeded out and arguably the ARPT should increase with the increasing threshold.

Figure 9 shows that two strategies stand out given ARPT with increasing thresholds - the ARMA(1,1) and the 30 neuron 20 lag ANN. The ARMA(1,1) has the second highest initial ARPT and the ARPT is increasing rapidly until the system produces less than 100 trades at the 49th percentile threshold and is stopped. The 30 neuron 20 lag ANN has the highest initial ARPT and a steady increase until around the 50th percentile threshold where the ARPT starts to decrease. However, the ARPT of the 30 neuron 20 lag ANN starts to increase again around the 70th percentile and overall shows an increasing trend. This indicates robustness of the 30 neuron 20 lag ANN and especially the ARMA(1,1). Further, 95% confidence intervals for the ARMA(1,1) show that the ARPT is increasing and is distinguishable from 0 except for the first thresholds. The confidence intervals for the 30 neuron 20 lag ANN however show that the ARPT is not distinguishable from 0 most of the time. The confidence intervals further strengthen the robustness of the ARMA(1,1).

Figure 10 shows cumulative returns rather than average return for the trading strategies based on 1st to 99th percentile of in-sample forecasts. In theory, similarly to the ARPT, the cumulative

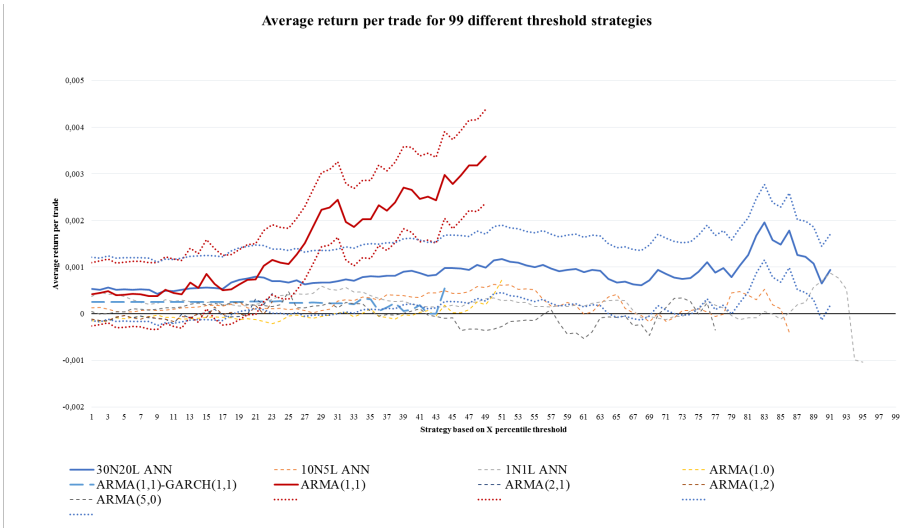

Figure 9: Average return per trade for 99 different threshold strategies. Confidence interval for ARMA(1,1) and 30 Neuron 20 lag ANN are marked with round dots.

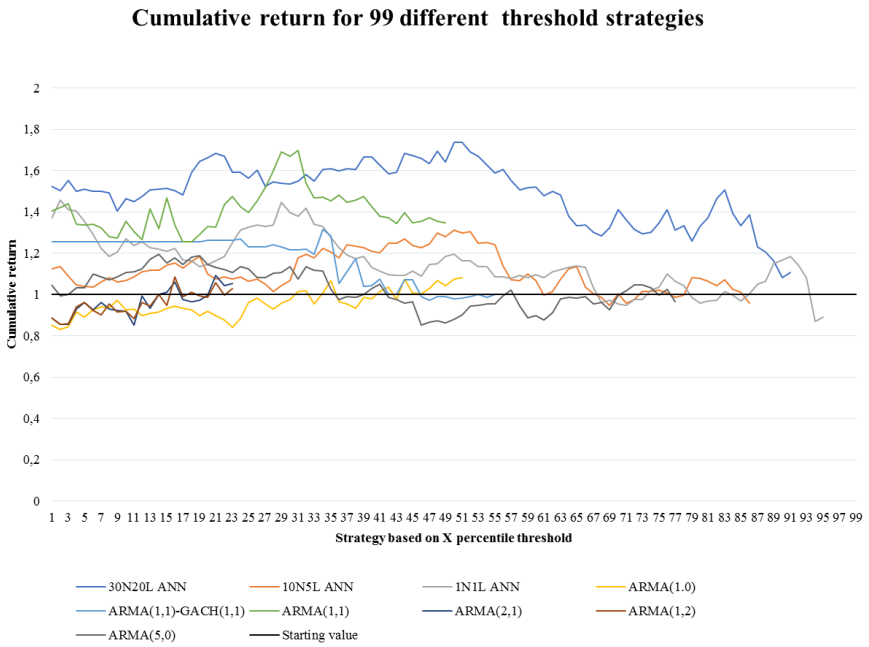

Figure 10: Average return per trade for 99 different threshold strategies.

returns should experience an increase with an increasing threshold (at least initially) as less certain forecasts are weeded out. In addition to figure 9, figure 10 shows that overall the 30 neuron 20 lag ANN would have higher returns than the ARMA(1,1) in our sample. Also, the increased ARPT of ARMA(1,1) shown in figure 9 does not increase the overall cumulative returns as clearly - the cumulative returns of ARMA(1,1) in figure 10 are volatile and have no clear overall trend. Further, the 30 neuron 20 lag ANN has a positive trend of cumulative returns until the 50th percentile where it shifts to a continuous decline instead, which may be explained by the fact that the APRT is not increasing fast enough make up for less trades.

6.5.2.4 Analysis of economic measurements

Based on economic metrics and trading strategies the ARMA(1,1) and the 30 neuron 20 lag ANN are deemed the best performing models. However, it seems as all models have trouble with consistently accumulating returns over time. The trading strategies show that implementing thresholds have varying impact. For some models the thresholds reduce the number of trades while increasing the overall return, thus sorting out the weak (bad) signals. However, for some of the strategies this is not the case. The number of trades are lower but the average returns are also lower - this implies that the forecasts are not robust since 'stronger' signals do not correspond with 'better' signals. If stronger signals would equal better signals the ARPT would increase for all models, which is clearly not the case. For example, the 1 neuron 1 lags ANN that experiences decreasing ARPT over time. Two models that stand out is the ARMA(1,1) where the ARPT increases rapidly and the 30 neuron 20 lag ANN for which we also see a positively increasing ARPT coupled with more trades than the ARMA(1,1). The 30 neuron 20 lag ANN and the ARMA(1,1) are the best models from an ecomomic perspective, however differentiating between the models is hard. One could argue that the ANN is more robust since it has more trades and produces higher overall return than the ARMA(1,1) and higher Sharpe ratios in general. However, the robustness test using ARPT favors the ARMA(1,1) since the 30 neuron 20 lag ANN experiences decreasing ARPT between 50th to the 70th percentile thresholds. Further, the fact that the GARCH(1,1) extension created worse performance from an economic point of view could be interpreted as the original ARMA(1,1) is overfitting. More specifically, the ARMA(1,1) interprets the changing volatility as changes in the mean. As such, the GARCH-extension works as a robustness test and indicates that some of the economic performance of the ARMA(1,1) may be due to luck. Notwithstanding, both the 30 neuron 20 lag ANN and ARMA(1,1) have quite volatile equity curves with positive drift.

Another notable fact (illustrated in Appendix A) is that even though some models outperform the OMXS30 according to final returns, none of the models consistently outperform the index during the 1000 days. Hence, none of the forecasts can be used to challenge the efficient market hypothesis.

7 Discussion

In this section we analyze and discuss the results further and from a higher level of abstraction in order to find the overall best forecast(s). We also discuss discrepancies between financial and statistical measures of performance.

Based on MSE, RMSE, DM-tests and Giacomini & White (2006) tests the ARMA(1,1), 10 neuron 5 lag ANN and 1 neuron 1 lag ANN are deemed the best models while the 30 neuron 20 lag ANN is deemed the worst. Based on economic metrics however the ARMA(1,1) and the 30 neuron 20 lag ANN are deemed the best performing models. The fact that the 30 neuron 20 lag ANN is deemed the worst performing model statistically, but one of the best performing according to Sharpe ratio and equity curve is interesting. In general the DM-tests, the Giacomini & White (2006) tests as well as RMSE and MAE might not give full financial insight when compared to Sharpe ratios and equity curves. Also, the conclusions that can be drawn from out-of-sample R^2 can be questioned when looking at the equity curves based on these forecasts. In the case where the out-of-sample R^2 and the Sharpe ratio are both positive, the Sharpe ratio is low, the equity curve is unstable, or the number of trades are low. Comparing the out-of-sample R^2 measure with

the best performing models considering Sharpe ratio and equity curve stability makes one question the relevance of the measure. For example the ARMA(5,0) at the 50th percentile threshold that scores a positive out-of-sample R^2 and positive Sharpe ratio, but has a low overall profit and unstable equity curve. One problem with the out-of-sample R^2 is that it does not take profitability of the system into consideration, nor equity curve stability which could be argued is the two most interesting parameters when designing a trading strategy. Further, there is often a discrepancy between economic measures (Sharpe ratio and cumulative returns) and the equity curves, which highlights the complexity of evaluating forecasts from a financial perspective.

This study further strengthens the arguments of Welch & Goyal (2008), Cenesizoglu & Timmermann (2012), and Campbell & Thompson (2008) that more meaningful economic measurements are needed to evaluate forecasts and models rather than standard statistical measures, but also shows the difficulty in determining economic performance. For many of the threshold strategies the distribution of trades are not even which is troublesome from a trading perspective as argued by Pardo (2008). The reason why it is troublesome is that it is uncertain if the required conditions would appear again. For example, a system that requires arbitrage possibilities might be obsolete in a modern highly competitive environment. Timmermann (2008) discusses this type of problem where a forecast does not produce consistent returns. In such a case it is costly for the investor to allocate capital to a system that does not trade for extensive periods of time. Whenever a systems activity or performance decreases there is a possibility that the opportunity has ceased to exists according to Timmermann (2008). Therefore a robust trading strategy should have evenly distributed returns and trades over time. Finding consistency in a trading strategy is an argument for why it is important to analyze equity curves. Analyzing equity curves also solves the potential sub-sample issue that Stock & Watson (2004) mention when analyzing out-of-sample performance. Out-of-sample equity curves are deemed to give the most insight into the economic performance of the forecast and minimize the risk of curve fitting - a phenomenon similar to overfitting.

8 Conclusion

The aim of this study is to determine which model produces the best forecasts. We have examined the models with both statistical and economical measures in order to find evidence to support a conclusion. However, our evidence is not homogeneous. When deciding which model that creates the best forecast is conditional on the objective. The statistical measures indicate that ARMA(1,1), 10 neuron 5 lag ANN and 1 neuron 1 lag ANN are the best models. However, if the criteria is economic value it is another issue. The trading strategies show that the 30 neuron 20 lags ANN exhibited the most compelling equity curve in general, and the ARMA(1,1) show positive drift given certain thresholds and highly increasing ARPT with increasing thresholds. These findings suggest that there is limited value in only using statistical measures when analyzing a model if the objective is to create profitable trading strategies, and that it is important to take equity curves and non-statistical measures into consideration when developing trading strategies. The statistical measurements rely heavily on one type of metric – the error, which might not give sufficient information from an economic view. In this study we can conclude that models that have a small predictive range score far better in terms of errors but do not necessarily lead to better economic outcomes. For example, the 0 prediction scored low errors but has no predictive power. The general conclusions that statistical measures do not translate into economic value is supported by previous findings of Timmermann (2008), Welch & Goyal (2008). Further, the study also highlights the complexity in evaluating a forecast from an financial perspective by showing

how economic measures give different results. The out-of-sample equity curve features the reliability of a trading system by displaying distribution of trades and returns, which is something that statistical measures and also most economic measures fail to show. This is an important aspect as a system that either stops trading or has poor performance over extended periods of time might be a sign of creative destruction as Timmermann (2008) argues.

Based on the findings in this study, evaluating a forecast is far more complex than the result of one metric and is dependent on the intended application of the forecast. There are good measures for establishing the statistical performance of a forecast however there are not any good measures for establishing the financial performance. The study suggests that a combination of statistics are needed in order to evaluate forecasts and shows how out-of-sample equity curves may be the best metric for evaluating financial forecasts. Further research is needed to find suitable measures for determining economic value of forecasts.

References

Adebiyi, A. A., Adewumi, A. O. & Ayo, C. K. (2014), 'Comparison of arima and artificial neural networks models for stock price prediction', *Journal of Applied Mathematics* **2014**.

Bodie, Z., Kane, A. & Marcus, A. J. (2011), *Investment and portfolio management*, McGraw-Hill Irwin.

Box, G. E., Jenkins, G. M. & MacGregor, J. F. (1974), 'Some recent advances in forecasting and control', *Applied Statistics* pp. 158–179.

Brogaard, J. et al. (2010), 'High frequency trading and its impact on market quality', *Northwestern University Kellogg School of Management Working Paper* **66**.

Campbell, J. Y. & Thompson, S. B. (2008), 'Predicting excess stock returns out of sample: Can anything beat the historical average?', *Review of Financial Studies* **21**(4), 1509–1531.

Cenesizoglu, T. & Timmermann, A. (2012), 'Do return prediction models add economic value?', *Journal of Banking & Finance* **36**(11), 2974–2987.

Clark, T. E. & McCracken, M. W. (2001), 'Tests of equal forecast accuracy and encompassing for nested models', *Journal of econometrics* **105**(1), 85–110.

Darbellay, G. A. & Slama, M. (2000), 'Forecasting the short-term demand for electricity: Do neural networks stand a better chance?', *International Journal of Forecasting* **16**(1), 71–83.

Dayhoff, M., Schwartz, R. & Orcutt, B. (1978), 22 a model of evolutionary change in proteins, *in* 'Atlas of protein sequence and structure', Vol. 5, National Biomedical Research Foundation Silver Spring, MD, pp. 345–352.

De Gooijer, J. G. & Hyndman, R. J. (2006), '25 years of time series forecasting', *International journal of forecasting* **22**(3), 443–473.

Dhrymes, P. J. & Peristiani, S. C. (1988), 'A comparison of the forecasting performance of wefa and arima time series methods', *International Journal of Forecasting* **4**(1), 81–101.

Dhrymes, P. J. & Thomakos, D. D. (1998), 'Structural var, marma and open economy models', *International Journal of Forecasting* **14**(2), 187–198.

Diebold, F. X. (2015), 'Comparing predictive accuracy, twenty years later: A personal perspective on the use and abuse of diebold–mariano tests', *Journal of Business & Economic Statistics* **33**(1), 1–1.

Diebold, F. X. & Mariano, R. S. (2002), 'Comparing predictive accuracy', *Journal of Business & economic statistics* **20**(1), 134–144.

Downs, G. W. & Rocke, D. M. (1983), 'Municipal budget forecasting with multivariate arma models', *Journal of Forecasting* **2**(4), 377–387.

Du Preez, J. & Witt, S. F. (2003), 'Univariate versus multivariate time series forecasting: an application to international tourism demand', *International Journal of Forecasting* **19**(3), 435–451.

Durbin, J. & Koopman, S. J. (2012), *Time series analysis by state space methods*, Vol. 38, OUP Oxford.

Ediger, V. Ş. & Akar, S. (2007), 'Arima forecasting of primary energy demand by fuel in turkey', *Energy Policy* **35**(3), 1701–1708.

Fama, E. F. (1995), 'Random walks in stock market prices', *Financial analysts journal* **51**(1), 75–80.

Fernandez-Rodrıguez, F., Gonzalez-Martel, C. & Sosvilla-Rivero, S. (2000), 'On the profitability of technical trading rules based on artificial neural networks:: Evidence from the madrid stock market', *Economics letters* **69**(1), 89–94.

French, K. R., Schwert, G. W. & Stambaugh, R. F. (1987), 'Expected stock returns and volatility', *Journal of financial Economics* **19**(1), 3–29.

Gardner, M. W. & Dorling, S. (1998), 'Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences', *Atmospheric environment* **32**(14), 2627–2636.

Giacomini, R. & Rossi, B. (2008), 'Detecting and predicting forecast breakdowns'.

Giacomini, R. & White, H. (2006), 'Tests of conditional predictive ability', *Econometrica* **74**(6), 1545–1578.

Granger, C. W. (1999), 'Outline of forecast theory using generalized cost functions', *Spanish Economic Review* **1**(2), 161–173.

Granger, C. W. J. & Newbold, P. (2014), *Forecasting economic time series*, Academic Press.

Guide, S. T. U. (1995), 'The mathworks, inc. mail 3 apple hill drive natick, ma 01760-2098'.

Gujarati, D. N. (2009), *Basic econometrics*, Tata McGraw-Hill Education.

Haykin, S. & Network, N. (2004), 'A comprehensive foundation', *Neural Networks* **2**(2004), 41.

Hein, S. E. & Spudeck, R. E. (1988), 'Forecasting the daily federal funds rate', *International Journal of Forecasting* **4**(4), 581–591.

Huang, W., Lai, K. K., Nakamori, Y. & Wang, S. (2004), 'Forecasting foreign exchange rates with artificial neural networks: a review', *International Journal of Information Technology & Decision Making* **3**(01), 145–165.

Huang, W., Nakamori, Y. & Wang, S.-Y. (2005), 'Forecasting stock market movement direction with support vector machine', *Computers & Operations Research* **32**(10), 2513–2522.

Hurvich, C. M. & Tsai, C.-L. (1989), 'Regression and time series model selection in small samples', *Biometrika* pp. 297–307.

Jensen, M. C. (1978), 'Some anomalous evidence regarding market efficiency', *Journal of financial economics* **6**(2-3), 95–101.

Jones, R. H. (1975), 'Fitting autoregressions', *Journal of the American Statistical Association* **70**(351a), 590–592.

Kaastra, I. & Boyd, M. (1996), 'Designing a neural network for forecasting financial and economic time series', *Neurocomputing* **10**(3), 215–236.

Khashei, M. & Bijari, M. (2011), 'A novel hybridization of artificial neural networks and arima models for time series forecasting', *Applied Soft Computing* **11**(2), 2664–2675.

Kuan, C.-M. & White, H. (1994), 'Artificial neural networks: an econometric perspective∗', *Econometric reviews* **13**(1), 1–91.

LeCun, Y., Bottou, L., Bengio, Y. & Haffner, P. (1998), 'Gradient-based learning applied to document recognition', *Proceedings of the IEEE* **86**(11), 2278–2324.

Leitch, G. & Tanner, J. E. (1991), 'Economic forecast evaluation: profits versus the conventional error measures', *The American Economic Review* pp. 580–590.

Lindemann, A., Dunis, C. L. & Lisboa, P. (2004), 'Probability distributions, trading strategies and leverage: An application of gaussian mixture models', *Journal of Forecasting* **23**(8), 559–585.

McCulloch, W. S. & Pitts, W. (1943), 'A logical calculus of the ideas immanent in nervous activity', *The bulletin of mathematical biophysics* **5**(4), 115–133.

Merh, N., Saxena, V. & Pardasani, K. R. (2011), 'Next day stock market forecasting: an application of ann and arima', *IUP Journal of Applied Finance* **17**(1), 70.

Metghalchi, M., Chang, Y.-H. & Marcucci, J. (2008), 'Is the swedish stock market efficient? evidence from some simple trading rules', *International Review of Financial Analysis* **17**(3), 475–490.

Metghalchi, M., Marcucci, J. & Chang, Y.-H. (2012), 'Are moving average trading rules profitable? evidence from the european stock markets', *Applied Economics* **44**(12), 1539–1559.

Ojah, K. & Karemera, D. (1999), 'Random walks and market efficiency tests of latin american emerging equity markets: a revisit', *Financial Review* **34**(2), 57–72.

Öller, L.-E. (1985), 'Macroeconomic forecasting with a vector arima model: A case study of the finnish economy', *International Journal of Forecasting* **1**(2), 143–150.

Pardo, R. (2008), 'The evaluation and optimization of trading strategies'.

Sarle, W. S. (1994), 'Neural networks and statistical models'.

Sermpinis, G., Dunis, C., Laws, J. & Stasinakis, C. (2012), 'Forecasting and trading the eur/usd exchange rate with stochastic neural network combination and time-varying leverage', *Decision Support Systems* **54**(1), 316–329.

Shambora, W. E. & Rossiter, R. (2007), 'Are there exploitable inefficiencies in the futures market for oil?', *Energy Economics* **29**(1), 18–27.

Sharda, R. & Patil, R. B. (1992), 'Connectionist approach to time series prediction: an empirical test', *Journal of Intelligent Manufacturing* **3**(5), 317–323.

Shibata, R. (1976), 'Selection of the order of an autoregressive model by akaike's information criterion', *Biometrika* pp. 117–126.

Stock, J. H. & Watson, M. W. (2004), 'Combination forecasts of output growth in a seven-country data set', *Journal of Forecasting* **23**(6), 405–430.

Tay, F. E. & Cao, L. (2001), 'Application of support vector machines in financial time series forecasting', *Omega* **29**(4), 309–317.

Timmermann, A. (2008), 'Elusive return predictability', *International Journal of Forecasting* **24**(1), 1–18.

Trinkle, B. S. (2005), 'Forecasting annual excess stock returns via an adaptive network-based fuzzy inference system', *Intelligent Systems in Accounting, Finance and Management* **13**(3), 165–177.

Tsay, R. S. (2005), *Analysis of financial time series*, Vol. 543, John Wiley & Sons.

Vaisla, K. S. & Bhatt, A. K. (2010), 'An analysis of the performance of artificial neural network technique for stock market forecasting', *International Journal on Computer Science and Engineering* **2**(6), 2104–2109.

Van Gestel, T., Suykens, J. A., Baestaens, D.-E., Lambrechts, A., Lanckriet, G., Vandaele, B., De Moor, B. & Vandewalle, J. (2001), 'Financial time series prediction using least squares support vector machines within the evidence framework', *IEEE Transactions on neural networks* **12**(4), 809–821.

Welch, I. & Goyal, A. (2008), 'A comprehensive look at the empirical performance of equity premium prediction', *Review of Financial Studies* **21**(4), 1455–1508.

West, K. D. (1996), 'Asymptotic inference about predictive ability', *Econometrica: Journal of the Econometric Society* pp. 1067–1084.

Whittle, P. (1954), 'On stationary processes in the plane', *Biometrika* pp. 434–449.

Yong, Y. (2005), 'Can the strengths of aic and bic be shared', *Biometrika* **92**(4), 937–950.

Yule, G. U. (1927), 'On a method of investigating periodicities in disturbed series, with special reference to wolfer's sunspot numbers', *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character* **226**, 267–298.

Zhang, B.-L., Coggins, R., Jabri, M. A., Dersch, D. & Flower, B. (2001), 'Multiresolution forecasting for futures trading using wavelet decompositions', *IEEE Transactions on Neural Networks* **12**(4), 765–775.

Zhang, F. (2010), 'High-frequency trading, stock volatility, and price discovery'.

Zhang, G. & Hu, M. Y. (1998), 'Neural network forecasting of the british pound/us dollar exchange rate', *Omega* **26**(4), 495–506.

Zhang, G., Patuwo, B. E. & Hu, M. Y. (1998), 'Forecasting with artificial neural networks:: The state of the art', *International journal of forecasting* **14**(1), 35–62.

Appendices

A Cumulative returns with trading strategies

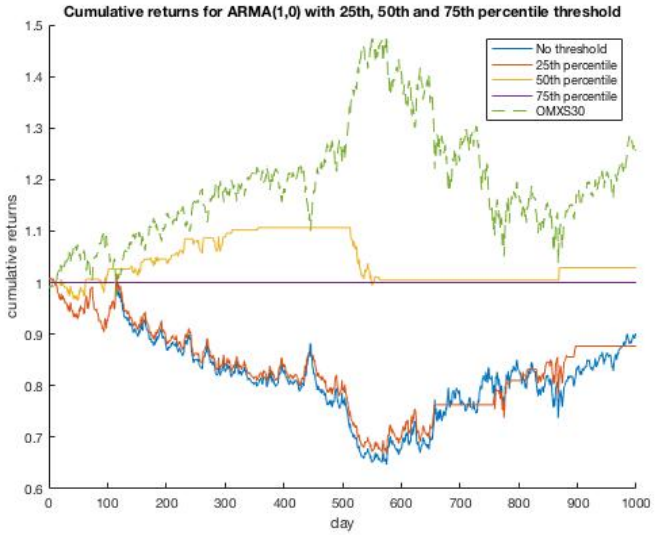

Figure 11: Cumulative returns for ARMA(1,1)-GARCH(1,1), 25th, 50th and 75th percentile threshold.

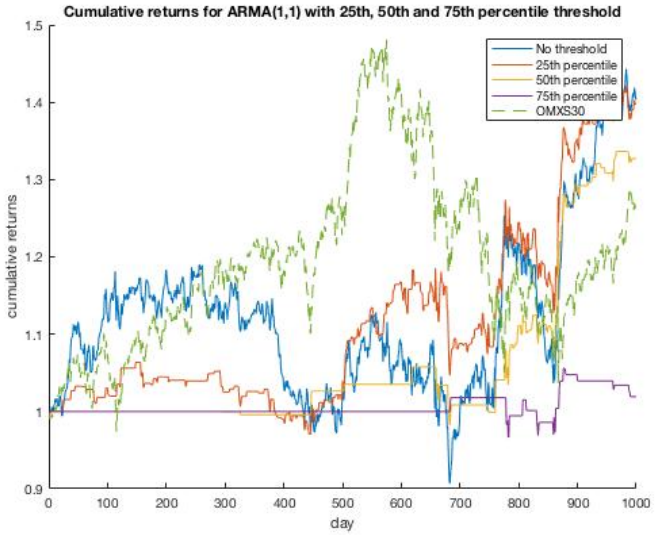

Figure 12: Cumulative returns for ARMA(1,0), 25th, 50th and 75th percentile threshold.

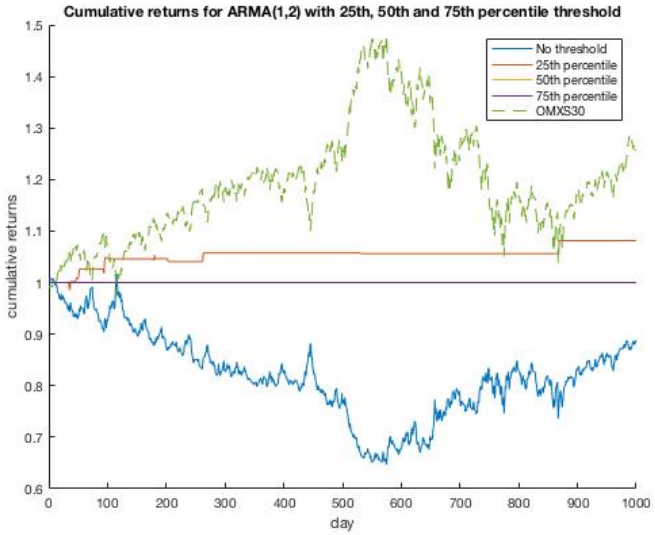

Figure 13: Cumulative returns for ARMA(1,1), 25th, 50th and 75th percentile threshold.

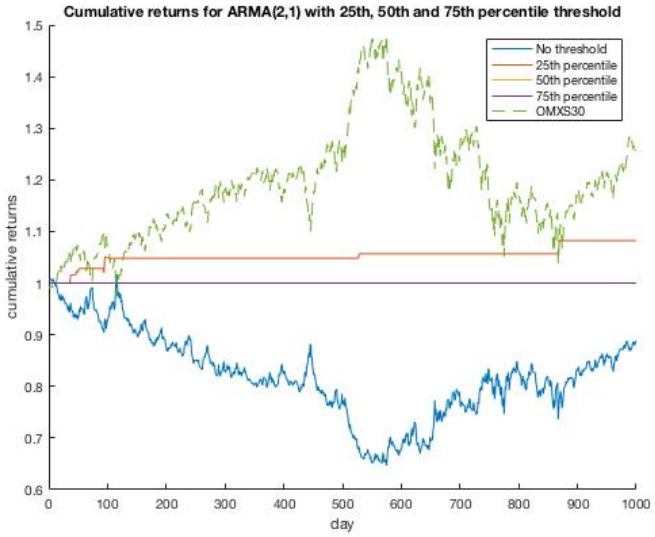

Figure 14: Cumulative returns for ARMA(1,2), 25th, 50th and 75th percentile threshold.

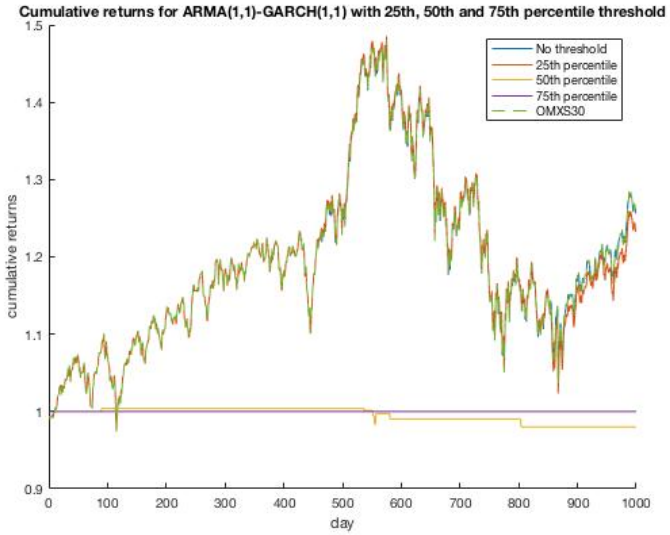

Figure 15: Cumulative returns for ARMA(2,1), 25th, 50th and 75th percentile threshold.

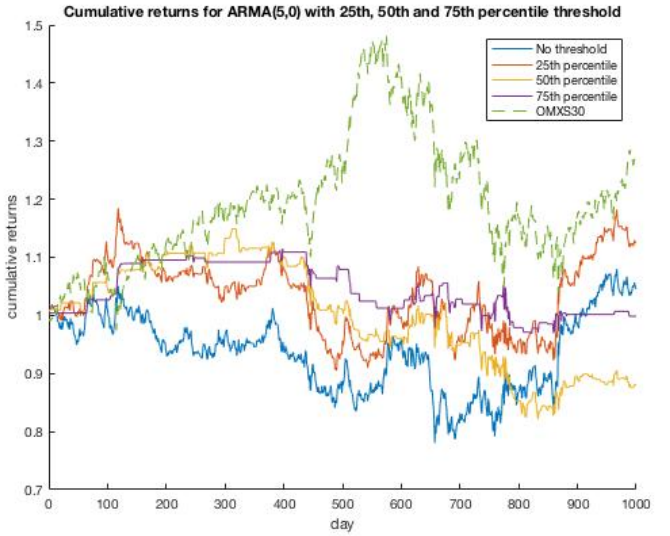

Figure 16: Cumulative returns for ARMA(5,0), 25th, 50th and 75th percentile threshold.

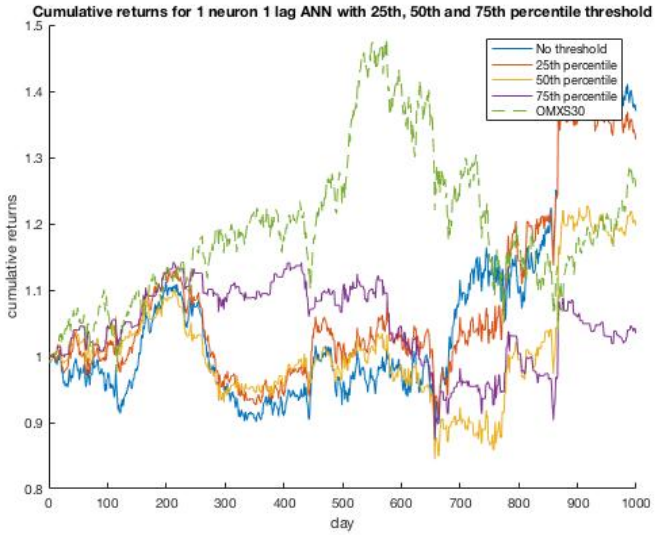

Figure 17: Cumulative returns for 1 neuron 1 lag ANN, 25th, 50th and 75th percentile threshold.

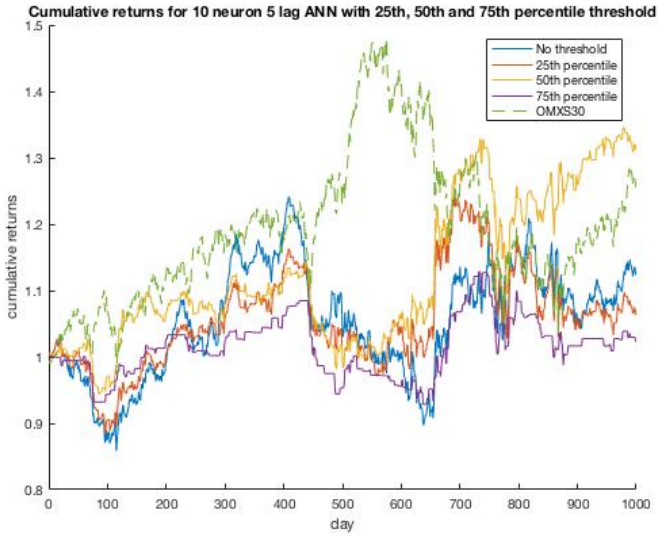

Figure 18: Cumulative returns for 10 neuron 5 lag ANN, 25th, 50th and 75th percentile threshold.

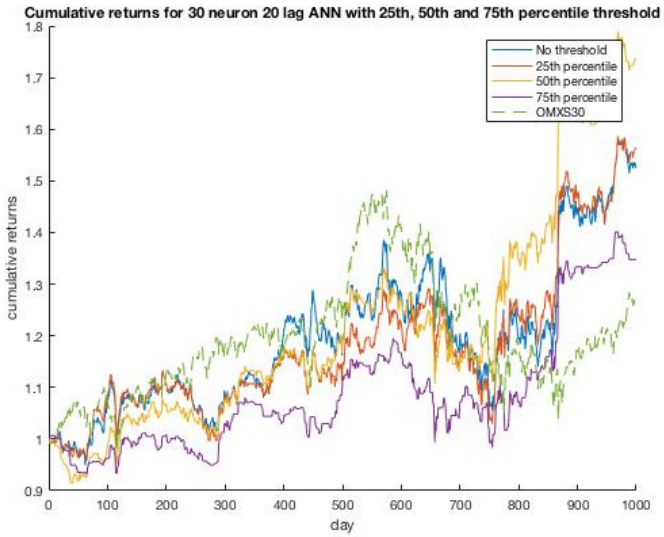

Figure 19: Cumulative returns for 30 neuron 20 lag ANN,, 25th, 50th and 75th percentile threshold.

B Forecasts compared to actual returns

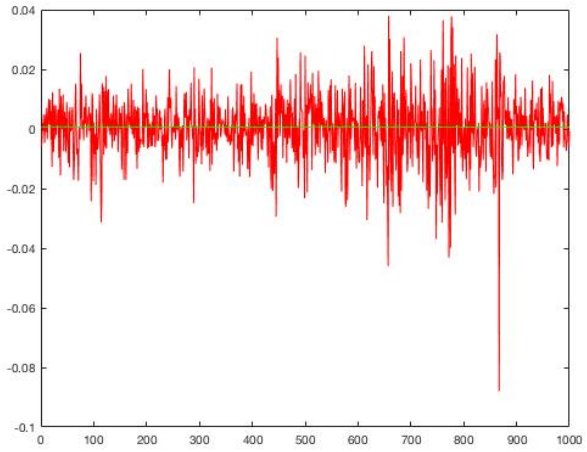

Figure 20: Forecast compared to actual for ARMA(0,1), with indication of max and min forecast.

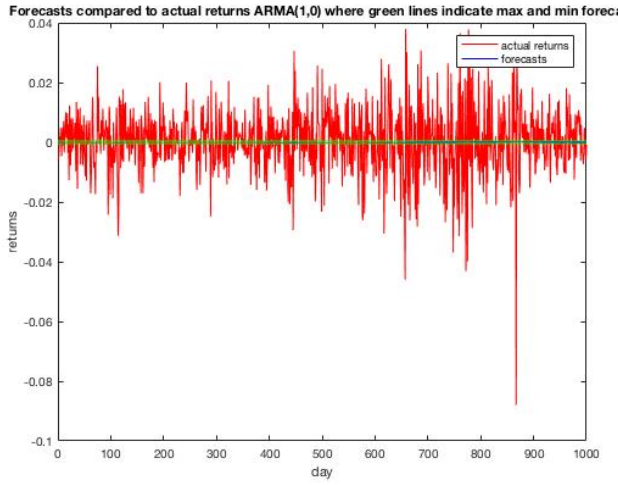

Figure 21: Forecast compared to actual for ARMA(1,0), with indication of max and min forecast.

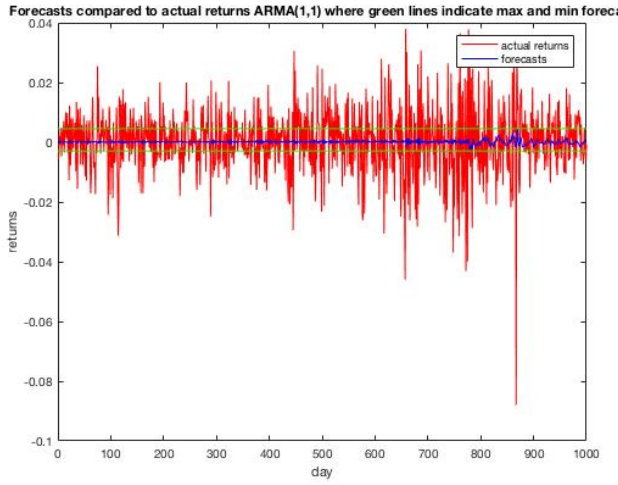

Figure 22: Forecast compared to actual for ARMA(1,1), with indication of max and min forecast.

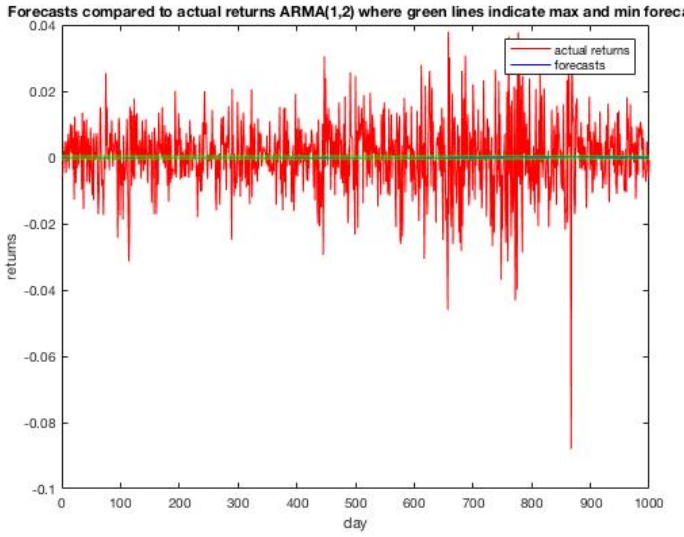

Figure 23: Forecast compared to actual for ARMA(1,2), with indication of max and min forecast.

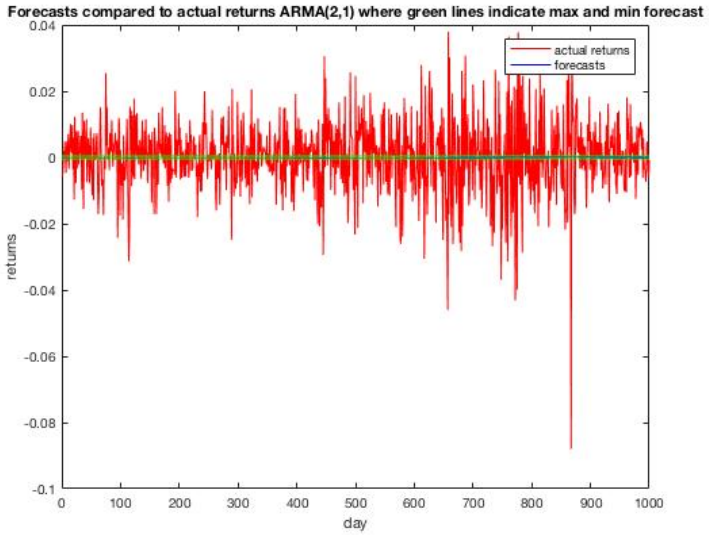

Figure 24: Forecast compared to actual for ARMA(2,1), with indication of max and min forecast.

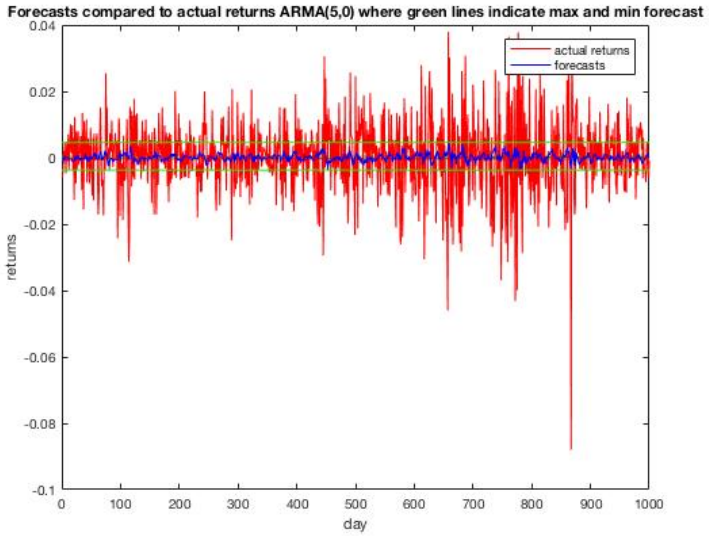

Figure 25: Forecast compared to actual for ARMA(5,0), with indication of max and min forecast.

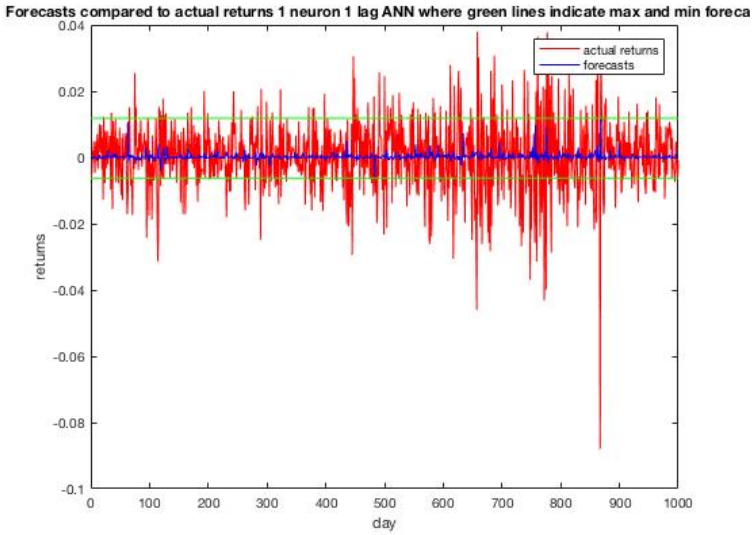

Figure 26: Forecast compared to actual for 1 neuron 1 lag ANN, with indication of max and min forecast.

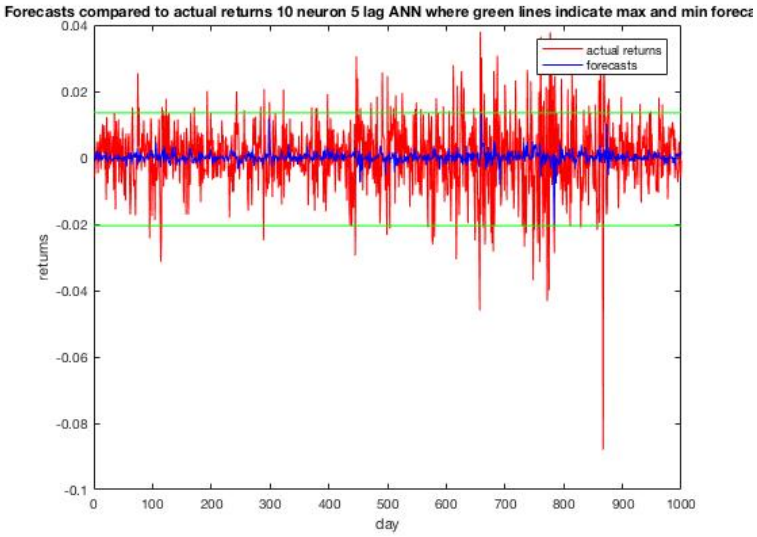

Figure 27: Forecast compared to actual for 10 neurons 5 lags ANN, with indication of max and min forecast.

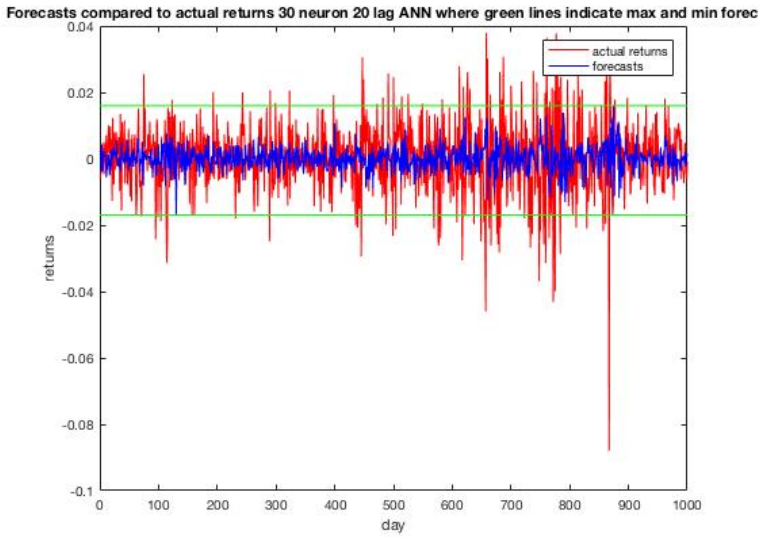

Figure 28: Forecast compared to actual for 30 neurons 20 lags ANN, with indication of max and min forecast.

C DM-test and Giacomini and White-test of 0-forecast

Confidence of DM test of 0-forecast

Model B \ Model A	(1,0)	(1,1)-GARCH(1,1)	(1,1)	(1,2)	(2,1)	(5,0)	1 N 1 L	10 N 5 L	30 N 20 L
0 forecast	0.3116	0.6711	0.8908	0.3180	0.3167	0.1048	0.6711	0.8612	0.9993

Table 9: Confidence of DM test of 0-forecast. Neuron is shortened to 'N', Lag is shortened to 'L'.

Unconditional Giacomini and White (2006) test of 0-forecast

Model B \ Model A	(1,0)	(1,1)-GARCH(1,1)	(1,1)	(1,2)	(2,1)	(5,0)	1 N 1 L	10 N 5 L	30 N 20 L
0 forecast	T-S 0.24(-)	T-S 0.02(+)	T-S 1.51(-)	T-S 0.22(+)	T-S 0.23(+)	T-S 1.57(+)	T-S 0.20(+)	T-S 1.18(+)	T-S 10.09(+)
	P-v: 0.623	P-v: 0.887	P-v: 0.218	P-v: 0.636	P-v: 0.633	P-v: 0.210	P-v: 0.658	P-v: 0.278	P-v: 0.001

Table 10: Unconditional Giacomini and White (2006) test of 0-forecast, + indicates model B better and - indicates model A better . Neuron is shortened to 'N', Lag is shortened to 'L',Test-Statistic is shortened to 'T-S', P-value is shortened to 'P-v'.

Conditional Giacomini and White (2006) test of 0-forecast

Model B \ Model A	(1,0)	(1,1)-GARCH(1,1)	(1,1)	(1,2)	(2,1)	(5,0)	1 N 1 L	10 N 5 L	30 N 20 L
0 forecast	T-S: 1.54(+)	T-S: 2.98(+)	T-S: 3.83(-)	T-S: 1.47(+)	T-S: 1.52(+)	T-S: 9.37(+)	T-S: 0.89(+)	T-S: 3.16(+)	T-S: 10.58(+)
	P-v: 0.462	P-v: 0.225	P-v: 0.147	P-v: 0.480	P-v: 0.469	P-v: 0.009	P-v: 0.642	P-v: 0.206	P-v: 0.005

Table 11: Conditional Giacomini and White (2006) test of 0-forecast, + indicates model B better and - indicates model A better . Neuron is shortened to 'N', Lag is shortened to 'L',Test-Statistic is shortened to 'T-S', P-value is shortened to 'P-v'.

D Comparing tests

D.1 Symmetric DM-test

Confidence of symmetric DM-test

ANN \ ARMA	(1,0)	(1,1)-GARCH(1,1)	(1,1)	(2,1)	(1,2)	(5,0)
1 neuron 1 lag	0.6161	0.6551	0.8349	0.617	0.6172	0.2882
10 neuron 5 lag	0.8422	0.8527	0.922	0.8425	0.8426	0.6286
30 neuron 20 lag	0.9991	0.9994	0.9998	0.9991	0.9991	0.9982

Table 12: Confidence of symmetric DM-test of ANNs and ARMAs.

Table 12 shows the DM-tests between the ARMAs and the ANNs. The DM (1995) test can not find any statistical difference between the forecast of the 1 neuron 1 lag or the 10 neurons 5 lags ANNs and the ARMAs. However, the DM (1995) test shows, with 5% significance, that there is a difference between the 30 neuron 20 lag ANN and all the ARMA models.

Confidence of symmetric DM-test

ANN \ ANN	1 neuron 1 lag	10 neuron 5 lag	30 neuron 20 lag
1 neuron 1 lag	0	-	-
10 neuron 5 lag	0.7563	0	-
30 neuron 20 lag	0.9986	0.9932	0

Table 13: Confidence of symmetric DM-test of ANNs.

As table 13 illustrates the DM-test finds a difference between the 30 neuron 20 lag and the 1 neuron 1 lag ANN on a 1% significance level and a difference on a 5% significance level compared to the 10 neuron 5 lag ANN. The DM-test can not find a difference between the 1 neuron 1 lag ANN and the 10 neuron 5 lag ANN.

Confidence of symmetric DM-test

ARMA \ ARMA	(1,0)	(1,1)-GARCH(1,1)	(1,1)	(1,2)	(2,1)	(5,0)
(1,0)	0	-	-	-	-	-
(1,1)-GARCH(1,1)	0.4827	0	-	-	-	-
(1,1)	0.902	0.8720	0	-	-	-
(1,2)	0.9426	0.5143	0.4851	0	-	-
(2,1)	0.904	0.5141	0.0987	0.4846	0	-
(5,0)	0.1249	0.0083	0.0103	0.1243	0.1070	0

Table 14: Confidence of symmetric DM-test of ARMAs.

As table 14 illustrates the DM-test can not find any statistical difference between the ARMAs.

D.2 Asymmetric DM-test

Confidence of asymmetric DM-test

ARMA \ ARMA	(1,0)	(1,1)-GARCH(1,1)	(1,1)	(2,1)	(1,2)	(5,0)
(1,0)	0					
(1,1)-GARCH(1,1)	0.5078	0				
(1,1)	0.9052	0.8831	0			
(2,1)	0.9049	0.4928	0.5102	0		
(1,2)	0.9435	0.4931	0.0955	0.5097	0	
(5,0)	0.1341	0.0076	0.0104	0.1336	0.1230	0

Table 15: Confidence of asymmetric DM-test of ARMAs.

Confidence of asymmetric DM-test

ANN \ ARMA	(1,0)	(1,1)-GARCH(1,1)	(1,1)	(2,1)	(1,2)	(5,0)
1 neuron 1 lag	0.6215	0.6252	0.8446	0.6223	0.6225	0.2846
10 neuron 5 lag	0.8517	0.8522	0.9275	0.8521	0.8522	0.6405
30 neuron 20 lag	0.9987	0.9990	0.9997	0.9987	0.9987	0.9975

Table 16: Confidence of asymmetric DM-test of ARMAs and ANN.

Confidence of asymmetric DM-test

ANN \ ANN	1 neuron 1 lag	10 neuron 5 lag	30 neuron 20 lag
1 neuron 1 lag	0		
10 neuron 5 lag	0.2327	0	
30 neuron 20 lag	0.0018	0.0096	0

Table 17: Confidence of asymmetric DM-test of ANN.

Conditional Giacomini and White (2006) test.

Model B \ Model A	(1,0)	(1,1)-GARCH(1,1)	(1,1)	(1,2)	(2,1)	(5,0)
(1,0)	T-S: 0 P-v: 1					
(1,1)-GARCH(1,1	T-S: 1.81(+) P-v: 0.405	P-v: 1 P-v: 0				
(1,1)	T-S: 4.86(+) P-v: 0.088	T-S: 4.71(+) P-v: 0.095	T-S: 0 P-v: 1			
(1,2)	T-S: 3.60(+) P-v: 0.165	T-S: 1.78(-) P-v: 0.411	T-S: 4.84(-) P-v: 0.089	T-S: 0 P-v: 1		
(2,1)	T-S: 3.10(+) P-v: 0.212	T-S: 1.79(+) P-v: 0.073	T-S: 4.84(-) P-v: 0.089	T-S: 2.63(-) P-v: 0.269	T-S: 0 P-v: 1	
(5,0)	T-S: 9.07(-) P-v: 0.011	T-S: 10.72(-) P-v: 0.005	T-S: 11.41(-) P-v: 0.003	T-S: 9.08(-) P-v: 0.011	T-S: 9.07(-) P-v: 0.011	T-S: 0 P-v: 1

Table 18: Conditional Giacomini and White (2006) test between ARMAs,+ indicates model B better and - indicates model A better. Test-Statistic is shortened to 'T-S', P-value is shortened to 'P-v'.

Unconditional Giacomini and White (2006) test.

Model B \ Model A	(1,0)	(1,1)-GARCH(1,1)	(1,1)	(1,2)	(2,1)	(5,0)
(1,0)	T-S: 0 P-v: 1					
(1,1)-GARCH(1,1)	T-S: 0.00(-) P-v: 0.965	P-v: 1 P-v: 0				
(1,1)	T-S: 1.67(+) P-v: 0.196	T-S: 1.29(+) P-v: 0.256	T-S: 0 P-v: 1			
(1,2)	T-S: 2.48(+) P-v: 0.115	T-S: 0.00(+) P-v: 0.970	T-S: 1.66(-) P-v: 0.198	T-S: 0 P-v: 1		
(2,1)	T-S: 1.70(+) P-v: 0.192	T-S: 0.00(+) P-v: 0.969	T-S: 1.66(-) P-v: 0.198	T-S: 0.11(-) P-v: 0.745	T-S: 0 P-v: 1	
(5,0)	T-S: 1.32(-) P-v: 0.250	T-S: 1.54(-) P-v: 0.214	T-S: 5.34(-) P-v: 0.021	T-S: 1.33(-) P-v: 0.249	T-S: 1.33(-) P-v: 0.249	T-S: 0 P-v: 1

Table 19: Unconditional Giacomini and White (2006) test between ARMAs,+ indicates model B better and - indicates model A better. Test-Statistic is shortened to 'T-S', P-value is shortened to 'P-v'.

Conditional Giacomini and White (2006) test. + indicates model B better and - indicates model A better

Model B \ Model A	(1,0)	(1,1)-GARCH(1,1)	(1,1)	(1,2)	(2,1)	(5,0)
1 N 1 L	T-S: 0.65(-) P-v: 0.721	T-S: 0.85(-) P-v: 0.655	T-S: 4.01(-) P-v: 0.135	T-S: 0.66(-) P-v: 0.720	T-S: 0.66(-) P-v: 0.720	T-S: 1.05(+) P-v: 0.593
10 N 5 L	T-S: 2.65(-) P-v: 0.266	T-S: 4.18(-) P-v: 0.124	T-S: 3.54(-) P-v: 0.170	T-S: 2.65(-) P-v: 0.265	T-S: 2.65(-) P-v: 0.266	T-S: 1.64(-) P-v: 0.440
30 N 20 L	T-S: 10.28(-) P-v: 0.006	T-S: 10.64(-) P-v: 0.005	T-S: 12.74(-) P-v: 0.002	T-S: 10.28(-) P-v: 0.006	T-S: 10.28(-) P-v: 0.006	T-S: 8.38(-) P-v: 0.015

Table 20: Conditional Giacomini and White (2006) test between ARMAs,+ indicates model B better and - indicates model A better. Test-Statistic is shortened to 'T-S', P-value is shortened to 'P-v'.

Unconditional Giacomini and White (2006) test.

Model B \ Model A	(1,0)	(1,1)-GARCH(1,1)	(1,1)	(1,2)	(2,1)	(5,0)
1 N 1 L	T-S: 0.12(-)	T-S: 0.16(-)	T-S: 1.03(-)	T-S: 0.13(-)	T-S: 0.13(-)	T-S: 0.31(+)
	P-v: 0.724	P-v: 0.690	P-v: 0.310	P-v: 0.722	P-v: 0.722	P-v: 0.576
10 N 5 L	T-S: 1.05(-)	T-S: 1.10(-)	T-S: 2.04(-)	T-S: 1.06(-)	T-S: 1.06(-)	T-S: 0.11(-)
	P-v: 0.305	P-v: 0.295	P-v: 0.153	P-v: 0.304	P-v: 0.304	P-v: 0.743
30 N 20 L	T-S: 9.77(-)	T-S: 10.27(-)	T-S: 12.38(-)	T-S: 9.78(-)	T-S: 9.78(-)	T-S: 8.41(-)
	P-v: 0.002	P-v: 0.001	P-v: 0.000	P-v: 0.002	P-v: 0.002	P-v: 0.004

Table 21: Unconditional Giacomini and White (2006) test between ARMAs,+ indicates model B better and - indicates model A better. Test-Statistic is shortened to 'T-S', P-value is shortened to 'P-v'.

Unconditional Giacomini and White (2006) test.

Model B \ Model A	1 neuron 1 lag	10 neuron 5 lag	30 neuron 20 lag
1 neuron 1 lag	T-S: 0		
	P-v: 1		
10 neuron 5 lag	T-S: 0.48(-)	T-S: 0	
	P-v: 0.487	P-v: 1	
30 neuron 20 lag	T-S: 8.87(-)	T-S: 6.06(-)	T-S: 0
	P-v: 0.003	P-v: 0.014	P-v: 1

Table 22: Unconditional Giacomini and White (2006) test between ARMAs,+ indicates model B better and - indicates model A better. Test-Statistic is shortened to 'T-S', P-value is shortened to 'P-v'.

Conditional Giacomini and White (2006) test.

Model B \ Model A	1 neuron 1 lag	10 neuron 5 lag	30 neuron 20 lag
1 neuron 1 lag	T-S: 0		
	P-v: 1		
10 neuron 5 lag	T-S: 2.30(-)	T-S: 0	
	P-v: 0.316	P-v: 1	
30 neuron 20 lag	T-S: 11.07(-)	T-S: 6.42(-)	T-S: 0
	P-v: 0.004	P-v: 0.040	P-v: 1

Table 23: Conditional Giacomini and White (2006) test between ARMAs,+ indicates model B better and - indicates model A better. Test-Statistic is shortened to 'T-S', P-value is shortened to 'P-v'.