Mathematical Theory and Modeling www.iiste.org
ISSN 2224-5804 (Paper) ISSN 2225-0522 (Online)
Vol.3, No.14, 2013

Application Of Kalman Filter To Artificial Neural Networks Prediction For Foreign Exchange Rates

Bonventure Macharia. M, Waititu, A. G, Wanjoya, A. K

Department of Statistics and Actuarial Sciences, Jomo Kenyatta University of Agriculture and Technology,

P.O Box 62000-00200, Nairobi, Kenya

* E-mail of the corresponding author: bonventuremacharia@gmail.com

Abstract

In recent years, artificial neural networks have received increasing attention as a decision making tool when prediction of financial time series is concerned. Modeling issues associated with artificial neural network model like the size of sample data and the general architecture of the model affect the performance of the model. For this reason, artificial neural networks outputs are prone to over-fitting or under-learning resulting to large mean squared errors which affect the accuracy of the prediction. In this paper, we investigate if the application Kalman filter algorithm to artificial neural networks model output can improve the model accuracy through the reduction of the mean squared error. Performance measures for prediction accuracy were used to compare the two models over the datasets for dollar, Euro and Pound exchange rates in Kenya Shilling for a period of five years. In the entire cases artificial neural networks model performed better than artificial neural networks with Kalman filter model.

Keywords: Artificial Neural Networks, Currency, Exchange Rates, Kalman Filter, Multi-Layer Perceptron

1. Introduction

Artificial neural networks (ANN's) possess more advantages over the traditional methods such as Box-Jenkins. First, ANNs are data-driven self-adaptive methods in that there are few a prior assumptions about the models for problem under study. Second, ANNs can be generalized that is they can often correctly infer the unseen part of the population even if the sample data contain noisy information. Finally, ANNs are universal functional approximaters. It has been shown that a network can approximate any continuous function to any desired accuracy, (Yao and Tan, 2000, Qi and Wu, 2002 and Pacelli et al., 2011).Since, ANNs model are data-driven and model free can suffer high variance in the estimation. Previous research has suggested the use of post-processing methods to be applied on ANN's model output so as to reduce the large mean squared error, (Zhang et al., 1998). One of the successful post-processing used on model output that exhibit systematic error is Kalman filter (KF). In 1960, Rudolph Emil Kalman, published the famous paper describing a recursive solution to the discrete-data linear filtering problem. Since then Kalman filter has been the subject of extensive research and application due to advances in digital computing. The filter consists of a set of mathematical equations that provide an efficient computational solution for least square method. Kalman filter implements predictor-corrector type estimator that is optimal in the sense that it minimizes the estimated error covariance. The main advantage of Kalman filter is the easy adaptation to the observations as well as the fact that it may utilize short series of background information. The filter consists of a set of mathematical equations that provide an efficient computational solution for least square method. The main advantage of this methodology is the easy adaptation to the observations as well as the fact that it may utilize short series of background information. Kalman filter algorithm has been found to be a supplementary tool to improve the direct model output which exhibit systematic errors in the forecast (Ganalis et al., 2008).

2. Artificial Neural Networks

Artificial neural networks were originally developed to mimic basic biological neural systems. The human neural system is composed of a number of interconnected simple processing elements called neurons or nodes. Each node receives an input signal which is the total information from other node or external stimuli, processes it locally through an activation or transfer function and produces a transformed output signal to other nodes or external outputs. The neural networks are typically organized into several layers of nodes. The first layer is the input layer which receives external variables inform of inputs. The last layer is the output layer which gives the output variables. There exist intermediate layer(s) called hidden layer(s) which connect the input and output layer. The nodes are fully connected by direct arcs which have numeric weights. This type of network is called a feed-forward network, or multilayer feed forward network. Mathematically, a three-layer Multilayer feed-forward network with *N* input nodes, *H* hidden nodes, and one output node, can be expressed in the following formula:

General Output Equation

$$y = S_1 \sum_{h=1}^{H} (O_h w_h) + w_o \qquad [1]$$

Hidden Node Output Equation

$$O_h = S_2 \sum_{n=1}^{N} (x_n w_{nh} + w_{oh}) \qquad [2]$$

Where $x_n, n = 1,2,\ldots,N$ and y are network input and output respectively, O_h is the output value of the h^{th} hidden node, w_h and w_{nh} are connection weights between nodes of the hidden and output layer and input layer. The bias input x_o which stabilizes the inputs and has the weights w_o and w_{oh}. The activation function S_1 and S_2 will be used to introduce a degree of non-linearity to the model and prevent the output from reaching large values that can paralyze the model and inhibit training. The back propagation algorithm (Rumelhart and McClelland, 1986) will be used in layered feed-forward ANN's. This means that the nodes are organized in layers, and send their signals "forward", and then the errors are propagated backwards. The algorithm uses supervised learning, which means that we provide the algorithm with examples of the inputs and outputs we want the network to compute, and then the error is calculated. The idea of the back propagation algorithm is to reduce this error, until the ANN learns the training data. The training begins with random weights, and the goal is to adjust them so that the error will be minimal.

3. Kalman Filter

The filter consists of a set of mathematical equations that provide an efficient computational solution of the least square method. Let $y_t, y_{t-1}, \ldots, y_1$ denote the observed values of a variable of interest at time $t, t-1, \ldots, 1$. Assume that y_t depends on an unobservable quantity x_t known as the state vector. The relationship between y_t and x_t is linear and is specified by the observation equation

$$y_t = H_t x_t + v_t \qquad [3]$$

Assume that the change of the unknown state x_t from time $t-1$ to t is given by the system equation

$$x_t = G_t x_{t-1} + w_t \qquad [4]$$

The approach is based on the nonlinear correction of prediction bias using Kalman filter algorithm. The focus is on the study of a single ANN's model output in time, based on the estimation of the bias of this parameter as a function of the forecasting model direct output. Since, the ANN's model output is one-dimensional, so every entity in the model is a numerical value, not a matrix. Let the system matrix G_t be equal to 1. Then, the new model becomes

$$y_t = H_t x_t + v_t \qquad [5]$$

$$x_t = x_{t-1} + w_t \qquad [6]$$

Kalman filter gives a method of the recursive estimation of the unknown state x_t based on all observation value of y up to time t .The filter evolves from time $t-1$ to t using prediction equations and update equations.

Prediction Equations

$$x_{t|t-1} = H_t x_{t-1} \qquad [7]$$

$$P_{t|t-1} = H_t P_{t-1} H_t^T + W_t \qquad [8]$$

Update Equations

$$x_t = x_{t|t-1} + K_t (y_t - H_t x_{t|t-1}) \qquad [9]$$

$$K_t = P_{t|t-1} H_t (H_t P_{t|t-1} H_t^T + W_t) \qquad [10]$$

$$P_t = (1 - K_t H_t) P_{t|t-1} \qquad [11]$$

Where H_t is the observation matrix, v_t is the observation noise with mean 0 and variance V_t, G_t is the system matrix, w_t is the system noise with mean 0 and variance W_t is the $x_{t|t-1}$ state at inter-mediate time step, x_{t-1} is the state at previous time step, P_t is the error covariance, $P_{t|t-1}$ is the error covariance matrix at inter-mediate time step, P_{t-1} is the error covariance matrix at previous time step and K_t Kalman gain.

4. Methodology

The dataset of three major foreign currencies, namely United States Dollar (USD), Pound and Euro, was used in this study. The dataset comprised of daily exchange rates, which have been collected during the period 1st January 2008 to 31st December 2012, which amount to 1253 records; five days a week which was extracted from the Central Bank of Kenya website http://www.centralbank.go.ke as shown in Figure 1.In this work, the missing observations were dropped and assumed not to exist in the sample.

Table 1 represents summary statistics of the foreign currency rate for the five years. The average exchange rates for each individual currency against Kenya Shilling (KES) are USD/KES 79.86, Pound/KES 129.4 and Euro/KES 109.29 respectively. This means that the exchange rate for the last five years stood near at about this exchange rate. The Pound/KES data is highly dispersed from the mean with a standard deviation of 10.018 followed closely by the Euro/KES with 9.269 and lastly by the USD/KES with 7.816. The Pound/KES and Euro/KES data sets are skewed to the right with 0.7447 and 1.3859 indication of existence of extreme values larger than the mean. The USD/KES is skewed to the left with -0.1065 indication of existence of extreme values less than the mean. The Euro/KES data set has high kurtosis value of 1.7485 compared with 0.731 and 0.6424 for USD/KES and Euro/KES data set respectively.

Time series data have a high correlation between the current value and previous value. For this reason, in order to improve the quality of forecast in this study we have analyzed the logarithmic rates. Logarithmic transformation help stabilize the time series, since it reduces positive skewness because it compresses the upper tail of the distribution while stretching out the lower tail.

Table 2 shows the summary statistics of the first log difference data. The USD/KES and Pound/KES data sets are negatively skewed while the Euro/KES is positively skewed. The first log differenced data set for USD/KES has a high kurtosis of 13.16 while the Euro/KES and Pound/KES have 3.9 and 2.99.

4.1 Testing the predictability of the data

Before forecasting time series we want to know whether the data under study is predictable. If the time series is random, then all methods are expected to fail. For this reason, we need more information on the degree of predictability of the data. The Hurst Exponent (H) provides a measure for long-term memory and fractality of a time series and can be used as a numerical estimate of the predictability of a time series**.** It is defined as the relative tendency of a time series to either regress to a longer term mean value or cluster in a direction. Moreover, Hurst Exponent is an estimate and not a definitive measure because the algorithm operates under the assumption that the time series is a pure fractal, which is not entirely true for most financial time series. The value of Hurst exponent ranges between 0 and 1. There are three distinct classifications for the Hurst exponent:

i. $H = 0.5$: indicates a random series.
ii. $0 \leq H \leq 0.5$: indicates an anti-persistent or ergodic time series with frequent reversals and high volatility.
iii. $0.5 \leq H < 1.0$: indicates a persistent series.

Table 3 contains values of Hurst exponent for the first log difference of the three currencies. From the results all the currencies are persistent or trend reinforcing series since the *H* value is greater than 0.5, which is characterized by long memory effects indication of high quality predictions. The USD/KES has the highest value of 0.61897, displaying higher level of predictability followed by the Pound/KES 0.56394 and lastly the Euro/KES with the least value of 0.56342.

4.2 Selection of ANN model input variables

The selection of input variables for the ANN's model is crucial as it facilitates the neural networks understand the movements of the time series. Any time series forecasting model assumes that there is an underlying process from which data are generated and the future value of a time series is solely determined by the past and current observations. In this work, lagged observations of the time series being forecasted were used as the input variables for the ANN model. The model adopted was non-linear (auto) regression model of the form

$$y_t = f(y_{t-1}, y_{t-2}, ..., y_{t-p}) \qquad [12]$$

Where, y_t the observation at time t and *p* is the dimension of the input vector or the number of past observations related to the current value. The challenge therefore is to determine the number of lags since with less of lags between each input variable may result to over fitting while increase in the lag between input variable may result to under-learning/fitting. To handle with this dilemma of over fitting or under-learning, we selected the number of lags to be 5 since the data comprised of five days daily exchange rates (i.e. p = 5) to test the performance of the neural network on a trial and error basis.

4.3 Training and testing of data

The logarithmic rates were divided into training set from January 2008 to December 2011 and testing/out-of sample sets from January to December 2012. The training set was used to optimize the weights and the bias of the network, while test set was used to determine the ability of the network to generalize. The transformed data was scaled using a linear function (equation 13) to an interval of 0 and 1, where X_t is the observation at time t and X_t^* is the scaled observation at time *t*. Data scaling is important since ANNs are pattern recognizers and the performance largely depends on the quality of the data used as it influences the accuracy of the results.

$$X_t^* = \frac{X_t - \min(X_t)}{\max(X_t) - \min(X_t)} \qquad [13]$$

The general issues on ANN's model which include the choice of input variable, the architecture and activation function and the criteria for selecting the best model were put into consideration. The number of input nodes is the most important factor in neural network analysis of a time series since it corresponds to the number of past lagged observations related to future values. The number of hidden nodes allows neural networks to capture nonlinear patterns and detect complex relationships in the data. But networks with too many hidden nodes may cause over fitting problems, leading to poor forecasting ability. In this work we used one hidden layer to help reduce computation time and danger of over-fitting which lead to poor out-of-sample forecasting performance. The default choice of the activation function at the output stage is the logistic function, which constrains the output in the range of 0 and 1. However, the number of nodes in the hidden layer ranged from 1 to 10, though research show that neural network performance is insensitive to the number of hidden nodes (Zhang and Hu, 1998). The R http://www.r-project.org/ software has a package *nnet* designed to handle neural networks with a single hidden layer and the choice of architecture is about choosing the number of neurons in the hidden layer.

4.4 Model selection criterion

The Akaikes Information Criterion (AIC) and Bayesian Information Criterion (BIC) are more satisfactory for choosing a 'best' model for ANN from a candidates models having different number of parameters. In both criteria, the first term measures the fit and the rest is a penalty term to prevent over-fitting, where BIC penalizes more severely the extra parameter than AIC does. The lower value obtained by the criterion, the better the model.

$$AIC = n\log\left\{\frac{\sum (y-\hat{y})^2}{n}\right\} + 2k \qquad [14]$$

$$BIC = n\log\left\{\frac{\sum (y-\hat{y})^2}{n}\right\} + k + k\log(n) \qquad [15]$$

Where n the number of observations and k the number of parameters being estimated.

Table 4 shows the values for AIC and BIC for all the ANN models out-of-sample output. The model with lags 1, 2, 3, 4 and 5 performed poorly for all the currencies with the maximum AIC and BIC values. The model with 1 and 5 lags model performed better than other models in the prediction of all the currencies future rates.

4.5 ANN's model with Kalman filter

Kalman filter algorithm was applied to ANN's model output for out-of-sample forecast and the performance measures computed. The R software package *sspir* designed to handle dynamic generalized model provides a platform for Kalman filter and Kalman smoother for models within Gaussian, Poisson and binomial families was used.

4.6 Performance measures

There can be many performance measures of an ANN's forecaster like the modeling time and training time, the ultimate and the most important measure of performance is the prediction accuracy it can achieve beyond the training data. An accuracy measure is defined in terms of forecasting error which is the difference between the actual and predicted value. There are a number of measures of accuracy in the forecasting literature and each has advantages and limitations (Makridakis et al., 1983). In this study the following performance measures will be used;

i. Mean Absolute Error (MAE)

$$\frac{\sum |y_t - \hat{y}_t|}{N} \qquad [16]$$

ii. Mean Squared Error (MSE)

$$\frac{\sum (y_t - \hat{y}_t)^2}{N} \qquad [17]$$

iii. Root Mean Squared Error (RMSE)

$$\sqrt{MSE} \qquad [18]$$

iv. Mean Absolute Percentage Error (MAPE)

$$\frac{1}{N}\left\{\sum \frac{|y_t - \hat{y}_t|}{y_t}\right\}100 \qquad [19]$$

Where $\hat{y}_t$ the predicted output y_t is the actual output and N is the number of output.

5. Out-of-Sample Prediction Accuracy Results

The predictions accuracy for ANN's and ANN's with KF (ANN-KF) for the three currencies is represented in Table 5 to 7.The model with lags 1, 2, 3, 4 and 5 has the large values for the performances measure for all the currencies; indication that it's the worst out-of-sample prediction model. The ANN's model with lags 1 and 5 has the least performance measures values for all the currencies out-of-sample predictions. A general observation is that ANN's model performs better than the model with ANN's with Kalman filter algorithm.

6. Conclusion

The main objective of the study was to investigate if the application of Kalman filters to ANN model output improves the accuracy of the predictions of foreign exchange rates. The findings from the study show that application of Kalman filter algorithm to ANN model output does not improve the accuracy of the ANN's model predictions. A general observation is that ANN's model is a better predictor in situations where the time series data is characterized by long memory effects indication of high quality predictions.

References

Kalman, R.E. (1960), A New Approach to Linear Filtering and Prediction Problems, *Journal of Basic Engineering*, **82** : 34 - 45.

Makridakis, S., Wheelwright, S.C., McGee, V.E., 1983. *Forecasting Methods and Applications*, John Wiley, New York.

Rumelhart, D.E., Hinton, G.E., Williams, R.J. (1986), *Parallel Distributed Processing: Explorations in the Micro-structures of Cognition*, **1** : 318 - 362.

Kuan, C.M., Liu, T. (1995), Forecasting exchange rates using feedforward and recurrent neural networks, *Journal of Applied Econometrics*, **10**, (4) : 347 – 364.

Faraway, J., Chatfield, C. (1998), Time series forecasting with neural networks: a comparative study using airline data, *Appl. Statistic*, **47** : 231 - 250.

Zhang, G., and Hu, M.Y. (1998), Neural network forecasting of the British pound/US dollar exchange rate. *International Journal of Management Science*, **26** : 495 - 506.

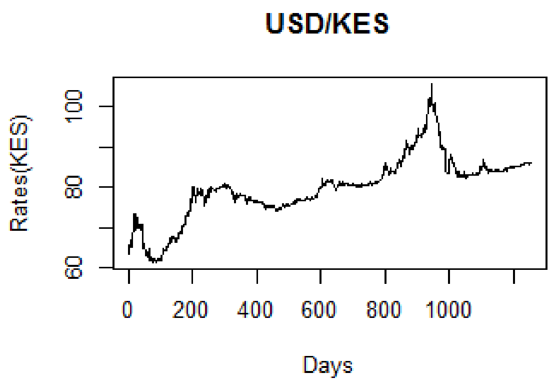
Zhang, G., Patuwo, B.E., Hu, M.Y. (1998), Forecasting with artificial neural networks: The state of the art, *International journal of forecasting*, **14** : 35 - 62.

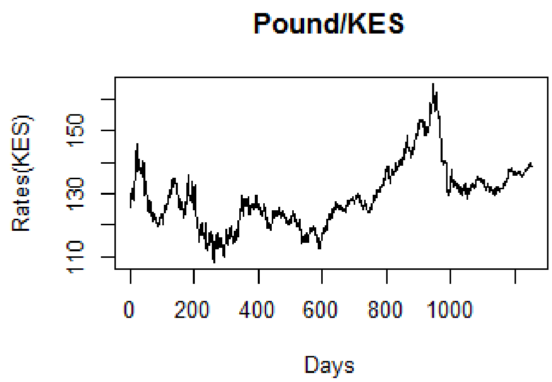
Yao, J.T., Tan, L.C. (2000), A case study on using neural networks to perform technical forecasting of forex, *Neurocomputing*, **34** : 79 - 98.

Qi, M., Wu, Y., (2002), Nonlinear prediction of exchange rates with monetary fundamentals, *Journal of empirical finance*,**10** : 623 - 640.

Galanis, G., Luoka, P., Katsafados, P., Kallos, G., Pytharoulis, I. (2006), Applications of Kalman filters based on non-linear functions to numerical weather predictions, *Annales Geophysicae*, **24** : 2451 - 2460.

Pacelli1, V., Bevilacqua, V., Azzollini, M. (2011), An Artificial Neural Network Model to Forecast Exchanges rates, *Journal for Intelligent Learning Systems and Applications*, **3** :57 - 69.





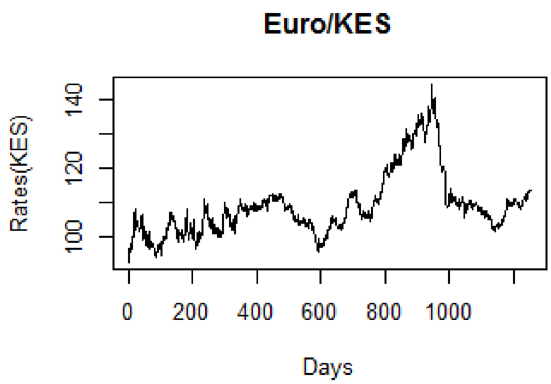


Figure 1: Plots for data sets

Table 1: Summary statistics for dataset

	USD/KES	**Pound/KES**	**Euro/KES**
Mean	79.86	129.4	109.29
Standard Deviation	7.816	10.018	9.269
Skewness	-0.1065	0.7447	1.3859
Kurtosis	0.731	0.6424	1.7485
Minimum	61.51	108.5	92.91
Maximum	105.96	165.3	144.60

Table 2: Summary statistics for data first log difference

	USD/KES	Pound/KES	Euro/KES
Mean	0	0	0
Standard Deviation	0.01	0.01	0.01
Skewness	-0.05	-0.11	0.25
Kurtosis	13.16	3.9	2.99
Minimum	-0.05	-0.05	-0.04
Maximum	0.04	0.05	0.05

Table 3: The table of H values

	H
USD/KES	0.61897
Pound/KES	0.56394
Euro/KES	0.56342

Table 4: ANN's model AIC and BIC values

Lags	USD/KES		Pound/KES		Euro/KES	
	AIC	BIC	AIC	BIC	AIC	BIC
1,2,3,4,5	-1464.18	-1450.58	-1238.18	-1224.58	-1302.14	-1288.54
1,2,3,4	-1732.12	-1718.62	-1480.5	-1466.9	-1562.79	-1549.19
1,2	-1732.83	-1719.23	-1486.2	-1472.6	-1550.83	-1537.23
1,5	-1979.26*	-1965.66*	-1580.45*	-1566.85*	-1676.93*	-1663.33*
1,2,3,5	-1485.25	-1471.65	-1255.75	-1242.15	-1300.92	-1287.32
1,2,4,5	-1612.59	-1598.99	-1446.89	-1433.29	-1475.98	-1462.38
1,2,5	-1716.11	-1702.51	-1532.13	-1518.53	-1543.38	-1529.98
1,3,5	-1967.19	-1953.59	-1485.77	-1472.17	-1603.13	-1589.53
1,4,5	-1762.85	-1749.25	-1519.79	-1506.19	-1574.07	-1560.47

* indicates the minimum value for AIC and BIC

Table 5: Bias performance measures for ANN model and ANN with KF model for USD/KES

Lags	MSE		RMSE		MAE		MAPE	
	ANN	ANN-KF	ANN	ANN-KF	ANN	ANN-KF	ANN	ANN-KF
1,2,3,4,5	0.003011	0.011732	0.054795	0.108257	0.035076	0.064632	0.041488	0.07651
1,2,3,4	0.001058	0.012136	0.032416	0.110153	0.024524	0.064421	0.02902	0.076243
1,2	0.000991	0.011956	0.031351	0.109334	0.022673	0.063388	0.026829	0.075016
1,5*	0.000386	0.014997	0.018297	0.12245	0.017359	0.070214	0.02059	0.083084
1,2,3,5	0.002771	0.012169	0.052566	0.110274	0.032699	0.065175	0.038696	0.077151
1,2,4,5	0.00171	0.011915	0.041233	0.109115	0.028775	0.065124	0.034022	0.077064
1,2,5	0.001059	0.012086	0.03232	0.109922	0.023988	0.064184	0.028385	0.07596
1,3,5	0.00046	0.015099	0.020821	0.122863	0.017583	0.069973	0.020833	0.082776
1,4,5	0.000977	0.014766	0.030941	0.121504	0.024886	0.070947	0.029441	0.083924

*indicates the 'best' ANN model for out-of-sample prediction

Table 6: Bias performance measures for ANN model and ANN with KF model for Pound/KES

Lags	MSE		RMSE		MAE		MAPE	
	ANN	ANN-KF	ANN	ANN-KF	ANN	ANN-KF	ANN	ANN-KF
1,2,3,4,5	0.007313	0.073952	0.085417	0.271924	0.059791	0.19848	0.044806	0.148852
1,2,3,4	0.003293	0.070785	0.057219	0.266052	0.053567	0.197442	0.039917	0.147913
1,2	0.003097	0.07047	0.055402	0.265459	0.051766	0.196835	0.038584	0.147468
1,5*	0.00208	0.079512	0.044592	0.281976	0.040961	0.207497	0.030466	0.155514
1,2,3,5	0.007047	0.076801	0.083924	0.277122	0.058787	0.203057	0.043982	0.152241
1,2,4,5	0.003339	0.072594	0.057218	0.269426	0.045706	0.198826	0.034099	0.149031
1,2,5	0.002606	0.071895	0.050421	0.268131	0.041998	0.198051	0.031311	0.148439
1,3,5	0.002837	0.083414	0.052916	0.288813	0.046027	0.212065	0.034273	0.158928
1,4,5	0.002445	0.080787	0.048467	0.284228	0.041385	0.208964	0.03081	0.156629

*indicates the 'best' ANN model for out-of-sample prediction

Table 7: Bias performance measures for ANN model and ANN with KF model for Euro/KES

Lags	MSE		RMSE		MAE		MAPE	
	ANN	ANN-KF	ANN	ANN-KF	ANN	ANN-KF	ANN	ANN-KF
1,2,3,4,5	0.005785	0.076238	0.075944	0.276108	0.054445	0.206164	0.050435	0.190197
1,2,3,4	0.00192	0.072181	0.043771	0.268665	0.039388	0.198324	0.036181	0.182744
1,2	0.001962	0.072233	0.044174	0.268762	0.039784	0.198426	0.036541	0.182835
1,5*	0.001669	0.078437	0.040716	0.280064	0.034555	0.206179	0.031861	0.190053
1,2,3,5	0.005988	0.075881	0.076381	0.275415	0.05737	0.205739	0.053081	0.189773
1,2,4,5	0.003146	0.075048	0.056011	0.273947	0.041239	0.201958	0.038297	0.186323
1,2,5	0.002194	0.073488	0.046826	0.271086	0.040634	0.201318	0.037323	0.185508
1,3,5	0.002098	0.081481	0.04574	0.285445	0.034292	0.209615	0.031841	0.193329
1,4,5	0.002213	0.079933	0.046937	0.282724	0.033067	0.206068	0.030835	0.19009

*indicates the 'best' ANN model for out-of-sample prediction