A NEW STRATEGY OF HYBRID MODELS USING ARIMA, ANN, AND DWT IN TIME SERIES MODELLING

TSUNG-LIN LI and CHEN-AN TSAI

Division of Biometry
Department of Agronomy
National Taiwan University
Taipei 10617
Taiwan
e-mail: paul831201@gmail.com
catsai@ntu.edu.tw

Abstract

Time series forecasting is a challenging task of interest in many disciplines. A variety of techniques have been developed to deal with the problem through a combination of different disciplines. Although various researches have proved successful for hybrid models, none of them carried out the comparisons with solid statistical test.

© 2021 Scientific Advances Publishers

This paper proposes a new stepwise model determination method for artificial neural network (ANN) and a novel hybrid model combining autoregressive integrated moving average (ARIMA) model, ANN and discrete wavelet transformation (DWT). Simulation studies are conducted to compare the performance of different models, including ARIMA, ANN, ARIMA-ANN, DWT-ARIMA-ANN and the proposed method, ARIMA-DWT-ANN. Also, two real data sets, Lynx data and cabbage data, are used to demonstrate the applications. Our proposed method, ARIMA-DWT-ANN, outperforms other methods in both simulated datasets and Lynx data, while ANN shows a better performance in the cabbage data. We conducted a two-way ANOVA test to compare the performances of methods. The results showed a significant difference between methods.

As a brief conclusion, it is suggested to try on ANN and ARIMA-DWT-ANN due to their robustness and high accuracy. Since the performance of hybrid models may vary across data sets based on their ARIMA alike or ANN alike natures, they should all be considered when encountering a new data to reach an optimal performance.

1. Introduction

Since many real-world databases involve temporal data, time series analysis has been widely applied to accommodate the diversity of temporal data across different fields of research. Statistical methods based on conventional regression models have been developed to deal with time series, and the most well-known one is the autoregressive integrated moving average (ARIMA) model. Some curve-fitting methods such as generalized autoregressive conditional heteroscedasticity model (GARCH) is also widely used. In the topic of engineering, time series are often treated as signals. Researchers try to erase the noise from the data to extract the information masked behind. Several popular methods such as short time Fourier transform (STFT) and discrete wavelet transform (DWT) have been developed for this task. As computer science makes great progress in the past three decades, people start using machine learning methods to deal with time series data, such as artificial neural network (ANN) and recurrent neural network (RNN). The results of machine learning methods are often with higher accuracy but lack of explanatory ability as statistical models.

Although different methods are frequently used to deal with all kinds of time series data in different research fields, researchers are always wondering which one of them will have the best performance. Makridakis et al. [9] conducted an extensive comparison of twenty-four time series methods for forecasting accuracy. Although there is not an out performing method, they pointed out that the greater the randomness in the data, the less important the use of statistical sophisticated methods is. Hence, this statement explains why some simple methods, such as exponential smoothing or ARIMA, still perform well for a common practice. The paper also claimed that by combining different methods, the performance could be better than individual methods. Smith and Demetsky [12] made a comparison of four forecasting methods in order to build a traffic volume forecasting models for two sites. While neural network is slightly better than traditional ARIMA model, nearest neighbour nonparametric regression outperforms the others. Sfetsos [11] presented a comparison of different methods via mean hourly wind speed data. The researcher concluded that the ANN based models outperformed the respective linear ones, while the neural logic network combined with logic rules produced the smallest root mean square error (RMSE) among them. Several neural networks and conditional heteroscedastic models were used to forecast five time series of exchange rates by Dhamija and Bhalla [3]. It is shown that the neural network can be effectively employed to estimate the conditional volatility of exchange rate series, while they are found to outcompete conditional heteroscedastic models.

Combining several methods together is an intuitive solution to improve model accuracy. One of the most popular hybrid models is proposed by Zhang [16], a hybrid model combining ARIMA and ANN models. The results showed that the hybrid model had the smallest mean square error (MSE) in four experimental data sets. Conejo et al. [1] proposed a combination of ARIMA models and wavelet transformation which corrects the ill-behaved data. The hybrid method outperformed the original ARIMA model and the effects of non-constant mean or variance,

outliers, and seasonal effects could be reduced. A useful application was performed on a real data set and showed a well performance in the study. Guresen et al. [4] conducted a comparison between multilayer perceptron (MLP), dynamic artificial neural network (DAN2) and the hybrid neural networks using GARCH for extraction of new input variables was conducted using real exchange daily rate values of NASDAQ stock exchange index. The results showed that for the classical ANN model, MLP outperforms DAN2 and GARCH-MLP with a little difference, while GARCH input shad a noise effect on DAN2 model. The paper suggested ANN as a more reliable method for forecasting stock movement. Wang et al. [14] presented another hybrid model combining with ARIMA and ANN. After implementation on four series, the results showed that the proposed model was superior to ARIMA, ANN and Zhang's hybrid model in most situations. Khandelwal et al. [7] tried to integrate DWT into ARIMA-ANN hybrid model. The proposed method and others were tested by four time series data. As a result, the Khandelwal's hybrid model had the best performance among all. A noticeable discovery is that Zhang's model can only perform a slightly better work than each component models and Khandelwal's model has a much more advancement in forecasting than the others. Pannakkong et al. [10] integrated DWT into ARIMA-ANN model in a different way and combined the usage of restricted Boltzmann machine to perform pre-training for generating initial weights and biases based on inputs feature for ANN. The result of testing on sunspot, Lynx, exchange rate time series indicated that the proposed method had the best performance in all datasets.

　　Hybrid models were proven to be one of efficient ways for improving the forecasting accuracy. When there are so many models available for prediction, more combinations are possible to be discovered. A worth noting situation is that most researches are based on MSE comparison. However, this could be problematic for ANN models which could produce different outputs across different time. In this paper, we focus on the usage of different hybrid models to find a best combination of ARIMA,

ANN and DWT methods with a minimum prediction error. In addition, a statistical test is employed to ensure the significance of the benefits of hybrid models.

2. Preliminaries

2.1. Autoregressive integrated moving average (ARIMA) model

ARIMA model can be viewed as an extension of the conventional linear model. The model assume that a time series data, x_t, can be considered as a combination of past values and errors. A simplified version of ARIMA is autoregressive moving average model (ARMA), which is a combination of a autoregressive model (AR) and a moving average model (MA). An $ARMA(p, q)$ is given as

$$x_t = \varphi_1 x_{t-1} + \varphi_2 x_{t-2} + \cdots + \varphi_p x_{t-p} + w_t + \theta_1 w_{t-1} + \cdots + \theta_q w_{t-q},$$

or

$$\varphi(B)X_t = \theta(B)W_t,$$

where w_t is the error at time t, φ_i and θ_j are the parameters of AR and MA respectively, p and q are the orders of ARMA. If any differencing operation is applied to keep the original time series stationary, an ARMA model becomes an ARIMA model. An $ARIMA(p, d, q)$ is written as

$$(1 - B)^d \varphi(B)x_t = \theta(B)w_t.$$

The backshift operator B is used in performing differencing. It is represented as $B^k x_t = x_{t-k}$ and k is any constant smaller than t. The purpose of using ARIMA model instead of ARMA model is to fix nonstationarity of the time series with differencing operations. ARIMA models are usually implemented step by step according to Box-Jenkins method, which is proposed by George Box and Gwilym Jenkins in 1970.

2.2. Artificial neural networks (ANN)

ANN is a combination of several perceptrons, where each input node and each hidden node are fully connected. Since it has been proven by the universal approximation theorem that a single layer ANN with a finite number of hidden nodes is capable of modeling any continuous function, an $m \times n \times 1$ ANN is quite useful to predict time series. An ANN for one-step-ahead time series prediction with p previous values is actually similar to an $AR(p)$, while the coefficients are now calculated in different ways and the model could be non-linear depending on the selected activation functions. A single layer ANN predicting one step-ahead time series data can be mathematically written as $x_t = f(b_o + \sum_j^n w_j g(b_{ij} + \sum_i^m x_{t-i} w_{ij}))$, where x_t is the value at time t, b_{ij} and b_o are biases of hidden layer and output layer, respectively, w_{ij} and w_j are weights between nodes, $g(*)$ and $f(*)$ are activating functions of hidden nodes and output node, respectively, m and n are the numbers of input nodes and hidden nodes, respectively. The determination of the model structure could be annoying in time series analysis. While some researchers apply an AR to search for a suitable number of input nodes, carrying out a grid search for node determination is useful though time consuming. On the other hand, hidden nodes determination has little clue comparing with input nodes determination. Some rules of thumb and pruning methods were proposed to give less time consuming solutions comparing with grid search.

This paper applies ANN with single hidden layer. Since the purpose is to compare different hybrid models, activating function and weight optimizing method are simply set as sigmoid function and standard backpropagation, respectively. The newly proposed model determination method will be discussed later in Section 3.

2.3. Zhang's hybrid model (ARIMA-ANN)

It is believed that ARIMA can not capture non-linear relationship while ANN can not handle both linear and non-linear patterns equally well. Since the underlying pattern of the data is hard to be recognized as pure linear or non-linear, it is not guaranteed to have a best model among all. Zhang [16] proposed a hybrid model to combine ARIMA and ANN together. Suppose that a time series can actually be separated into two components

$$y_t = L_t + N_t,$$

where L_t stands for the linear part of the data and N_t is the non-linear part. An ARIMA is carried out first to forecast for linear part, yielding predictions $\hat{L}_t$ and residuals $e_t = y_t - \hat{L}_t$. Residuals are usually checked if there is any linear correlation left in conventional ARIMA, where non-linear patterns are usually ignored and would pass the diagnosis. Zhang tried to look for non-linear pattern hiding in residuals by applying ANN, which is given as

$$e_t = f(e_{t-1}, \ldots, e_{t-n}) + \varepsilon_t,$$

where $f(*)$ is the discovered non-linear function and ε_t is the random error. ANN is considered to handle the pattern missed by ARIMA. With the help of ANN, a forecast for non-linear part $\widehat{N}_t$ is acquired. The final prediction is the sum of the two forecasts, $\hat{y}_t = \hat{L}_t + \widehat{N}_t$. The hybrid model was considered to be more powerful than ARIMA and ANN in pattern capturing, the results of the empirical tests also confirmed this guess. The purpose of fitting ARIMA first rather than ANN was to avoid the over-fitting problem usually caused by ANN (see Figure 1). A multiplicative version of Zhang's hybrid model was later proposed by Wang et al. [14], the only difference was to change the additive form into multiplicative form as $y_t = L_t \times N_t$. The paper showed that the multiplicative version

was better than the original hybrid model. Since other hybrid models were mostly based on Zhang's version, the issue of multiplicative models will not be included in this paper.
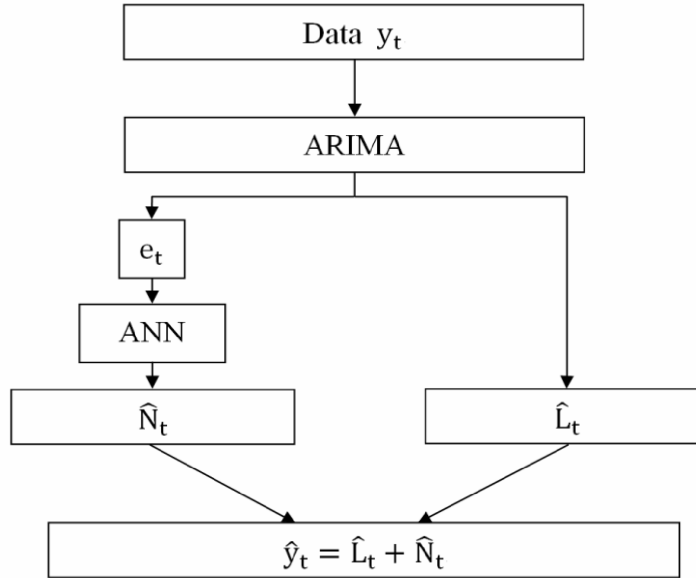
Figure 1. Zhang's ARIMA-ANN hybrid model.

## 2.4. Discrete wavelet transformation (DWT)

DWT is a common method in industrial engineering area, it is benefit for its time localizing ability. It is usually carried out to decompose a signal into several components. By then, components that relatively cause no influence to the whole data can be neglected. If so, the remaining part is considered to be more reliable for specialty recognition or forecasting. By applying fast wavelet transform algorithm, the original data would pass through predefined high pass filter and low pass filter then apply a down sampling of order 2 to acquire components responsible for high frequency and low frequency, respectively. Components responsible for low frequency are called *approximation transforms* while components responsible for high frequency are called *detail transforms*. The next level

components that are responsible for lower frequencies can be decomposed from approximation transforms with the same procedure. After deleting some negligible transforms, the inverse discrete wavelet transformation (IDWT) of the remaining transforms can receive a result approximating the original data. DWT can be mathematically written as $x_t = A_N(t) + \sum_i^N D_i(t),$ where $A_N(t)$ is the IDWT of the N-th level approximation transforms at period t and $D_i(t)$ is the IDWT of the i-th level detail transforms at period t. Usually, when the purpose is to remove some noises from the data, one or two levels of decomposing would accomplish the task.

2.5. Ina Khandelwal's hybrid model (DWT-ARIMA-ANN)

Ina Khandelwal et al. [7] brought out the idea of combining Zhang's hybrid model with DWT technique. The purpose is to obtain a prior decomposition of a time series into high and low frequency components. After applying a one level DWT to get approximation coefficients and detail coefficients, an IDWT is followed up to obtain original data points corresponding to both coefficients. High frequency components are denoted by y^D and low frequency components are denoted by y^A. The study assumed that the high frequency components are more responsible for the underlying linear components of the data. ARIMA is carried out in the high frequency components to acquire linear predictions. The residuals of the ARIMA model and the low frequency components are fitted to an ANN to obtain non-linear forecasts. The final predictions are the sum of linear predictions and non-linear predictions (see Figure 2).
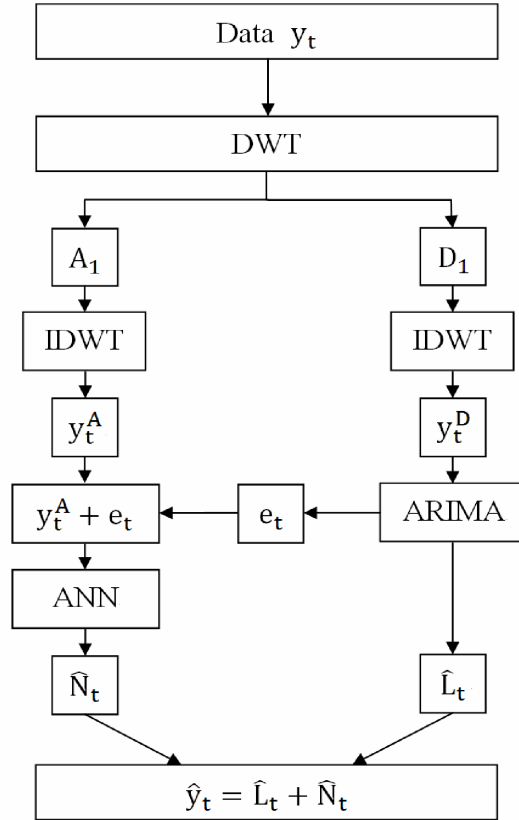
Figure 2. Khandelwal's DWT-ARIMA-ANN hybrid model.

3. Methods and Materials

3.1. Proposed model (ARIMA-DWT-ANN)

DWT and IDWT are applied to decompose the data and then to analyze the components instead of the original data in Ina Khandelwal's method. The process implies that there might be different patterns behind the high frequency components and the low frequency components. Consider that a common usage of DWT in signal analysis is to compress the data, which is unfortunately not acquired in previous hybrid models. The novel hybrid model is proposed to utilize the

compressing ability of DWT. Recall that in ARIMA models, error terms are considered and presumed in the model. On the other hand, ANN has no trick to deal with noises, this is the reason why ANN usually encounters over-fitting problem more often than ARIMA does. The proposed model continues the idea from Zhang to model linear and non-linear parts by ARIMA and ANN, respectively. A novel introduction of DWT to eliminate undesirable noises before applying ANN is the specialty of the proposed model. After getting predictions for the linear part of the data by ARIMA, one level DWT and IDWT are carried out to decompose the residuals into components with high and low frequencies. While the high frequency components are treated as noises and are eliminated, the low frequency components are fitted by an ANN to acquire the predictions for non-linear part of the data (see Figure 3).
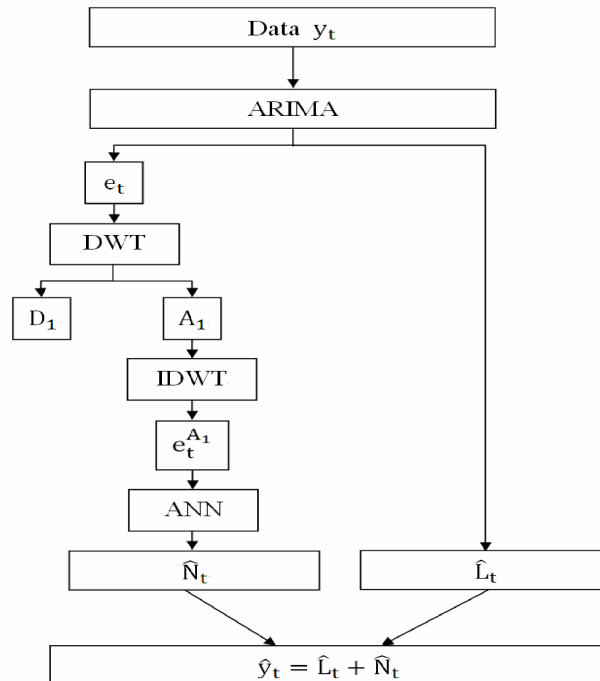
Figure 3. ARIMA-DWT-ANN hybrid model.

3.2. Model structure determination

In previous researches, ARIMA model structures are usually determined based on other studies using the same data set. As for ANN model structures, while some papers follow model structure of other researches just as cases in ARIMA, others perform pretests for model structure determination. However, these pretests are usually mentioned little and commonly applied without validation.

3.2.1. ARIMA model determination

The decision for an ARIMA model structure usually based on the observation of autocorrelation function (ACF) plot, partial autocorrelation function (PACF) plot, the result of unit root test and a following up model comparison. This decision presumes that the upcoming data will follow the same model structure and no longer change. This assumption is usually true though suspicious. In this study, we loosen this assumption. Each time a new data occur, model structure is re-determined to avoid the influence from model structure changing. This paper follow the auto determination method proposed by Hyndman and Khandakar [6].

3.2.2. ANN model determination

Previous researches usually determine their model structure based on other researches using the same data set or a grid search. None of them consider the usage of pruning methods. A comparison between optimal brain damage (OBD), optimal brain surgeon (OBS), statistical stepwise method (SSM) and grid search is carried out by the author to confirm the usage of pruning methods. The result shows that grid search is the method possessing the highest accuracy though being the most time consuming method at the same time, which is consistent with the decisions in previous studies. However, since a thoroughly grid search is extremely exhausting, a novel partial grid search is proposed in this study to reduce the time consumption. Based on the rule of thumb proposed by Tang and Fishwick [13], the number of hidden nodes is

considered to be the same as the number of input nodes. After separating data into training data (0-50% of the data), validating data (50-75% of the data), and testing data (75-100% of the data). A stepwise grid search is applied. Models with different structures are trained with training data, their performances in validating data are compared to search for the best model structure. At first, models $(2 \times 2 \times 1), (3 \times 3 \times 1), \dots, (10 \times 10 \times 1)$ are calculated for 10 times each. The model $(k \times k \times 1)$ with the lowest average cost is selected to apply follow up searches. The purpose of the first step is to select the number of input nodes. Models with more than 10 input nodes are not considered since the dependent time lags of past values are usually small. Although this statement is not proven, the selected input number is seldom 10 or even 9 in this study. This statement is a rule of thumb and it is adjustable. The second step is to select a suitable number of hidden nodes. By comparing 10 times average costs of the models $(k \times 1 \times 1), (k \times 2 \times 1), \dots, (k \times 10 \times 1)$, the most accurate model $(k \times s \times 1)$ is the determined model structure. A follow up parameters searching for the weight optimizing method is carried out, which is the same as two comparisons applied previously. After determining the model structure and suitable parameters for the weight optimizing method, the model is trained by the combination of training and validating data. Testing data is used to observe the final performance for comparison.

3.3. Model details

A stepwise automatic ARIMA is done by the R package "forecast". Each time facing a new data, the first data of the training set is trimmed to keep the training size fixed for the next prediction. The procedure of one step prediction stops when getting the last prediction of the testing data. As for ANN, the traditional sigmoid function is picked as activation function and the standard backpropagation is chosen as the weight optimizing method. The iteration time is set to 1000 in order to accelerate the model pruning steps. The application of ANN is done by the R

package "RSNNS". Note that, training size of ANN is also fixed as ARIMA does. The last but not the least, DWT. Haar wavelet is applied due to its advantage in discovering signals with sudden transitions and its common usage. Both DWT and IDWT in this paper are done by the R package "wmtsa".

3.4. Numerical studies

3.4.1. Simulated data sets

Three simulating models are used to test the performances between different methods. One of the models is a combination of the other two to mimic a data structure changing occasion. The settings of the parameters are according to Zhang et al. [15]. All simulations are done with a burn-in size 1000 and are added 100 to avoid ill-behaved mean absolute percentage error (MAPE). The first simulating model is bilinear model (BL), which is set as $x_t = 0.4x_{t-1} - 0.3x_{t-2} + 0.5x_{t-1}w_{t-1} + w_t$, where $w_t \sim N(0, 1)$. The second simulating model is smooth transition autoregressive model (STAR), which is set as $x_t = 0.8x_{t-1} - 0.8x_{t-1}$ $(1 + \exp\{-10x_{t-1}\})^{-1} + w_t$, where $w_t \sim N(0, 1)$. The third simulating model is model structure changing model (MSC). A cut point is randomly generated to separate the whole data into two parts. While previous data points are simulated by BL, data points after the cut point are simulated by STAR.

3.4.2. Real data sets

Two real data sets are used to carry out the comparison between performances of hybrid models and to test their practical reliabilities. The first data set is Lynx data, which is an annual record of the number of the Canadian Lynx trapped in the Mackenzie river in Canada. The data was recorded from 1821 to 1934 with total 114 data points. The data set is built in R. A common procedure applied to Lynx data before analysis goes on is a logarithm transformation to a base 10. This procedure is also

conserved to keep the data set more stationary (see Figure 4). The second data set is cabbage data, which is the monthly average price (NTD) per kilo of cabbages in Taipei wholesale market from January of 1996 to May of 2019. The data is collected from URL: https://amis.afa.gov.tw. It is an official outsourced website recording market prices and trading volumes (see Figure 5).
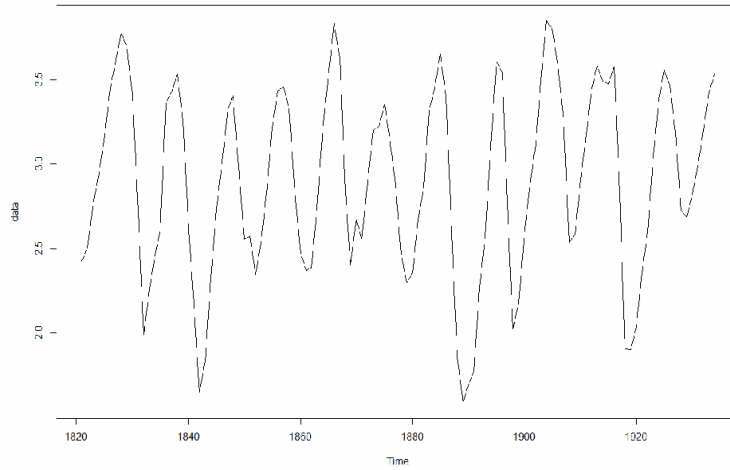
Figure 4. Lynx data (log10 transformed).
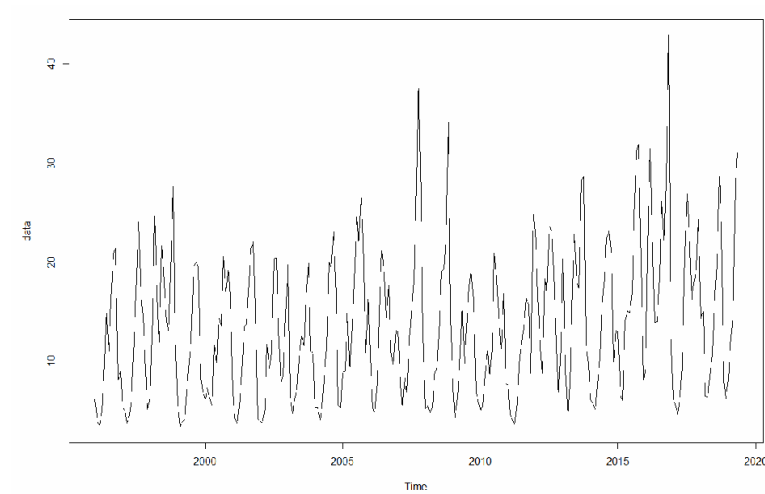
Figure 5. Cabbage data.

3.5. Experimental design

The evaluation for model performance is carried out by calculating MSE and MAPE defined as

$$\text{MSE} = \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{n},$$

and

$$\text{MAPE} = \frac{\sum_{i=1}^{n} \left| \frac{y_i - \hat{y}_i}{y_i} \right|}{n} \times 100\%,$$

where y_i is the i-th testing value and $\hat{y}_i$ is the predicted value of it with total n testing data points. While MSE is usually applied to compare for different methods on the same data set, MAPE is more reliable for comparison of models on different data sets.

In this research, model treatment, data size and validation are the experimental factors. While the effects of model types and data size are tested through simulated data sets, the real data sets are used to recheck for model performances and test for the effect of validation.

3.5.1. For simulated data sets

Since model treatment difference is the main purpose of the paper, the experimental design mainly focus on the ability of identifying the existence of method effect. For each simulating function, five simulations are made for data size equaling 100, 200, and 400 each, with total fifteen simulations. All simulations are applied with ARIMA, ANN, ARIMA-ANN, DWT-ARIMA-ANN and the proposed method, ARIMA-DWT-ANN. A RCBD is applied to test for difference between models, where simulations are considered as blocks. A follow up Duncan's multiple range test with $\alpha = 0.01$ is carried out to show the difference between methods. After the effect of different methods is showed, a two-way ANOVA test is applied to test for sample size effect, method effect and their interaction effect.

Table 1. Detail of separation of each data sets

Time series	Size		
	Training (50% of total)	Validating (25% of total)	Testing (25% of total)
Sample size 100 simulations	50	25	25
Sample size 200 simulations	100	50	50
Sample size 400 simulations	200	100	100
Lynx data	57	28	29
Cabbage data	140	70	71

*For cases without validation, training data and validating data are combined together as training set.

3.5.2. For real data sets

In previous researches, some of them recommend the usage of validating data to mimic real tasks, which is impossible to consider testing data into model pruning procedure. On the other hand, some other researches use testing data to prune the model directly and re-run the model to get model evaluations. Real data sets are used to test if there is validation effect. Also, the method effect is rechecked to assure their difference in real world. Each real data is applied with all methods mentioned in simulated cases with five replicates. A two-way ANOVA is carried out to test for method effect and validation effect. Duncan's multiple rang test with $\alpha = 0.01$ is applied for different methods and validation if the factors are significant.

4. Results and Discussions

Both RCBD results for the five candidate methods' MSE and MAPE confirm the significance of method effect. Since method effect is significant, Duncan's multiple rang test with $\alpha = 0.01$ is applied. The results of both MSE and MAPE show that DWT-ARIMA-ANN is significantly worse than others. A further RCBD eliminating ARIMA-DWT-ANN is applied to assure the difference between ARIMA, ANN,

ARIMA-ANN, and ARIMA-DWT-ANN. Both results for MSE and MAPE show that the method effect is still significant. Further extended Duncan's tests are applied for the reduced methods' MSE and MAPE for instance. As the Duncan's tests showed, ARIMA-DWT-ANN is dominant in both MSE and MAPE results. Confirming the usage of the proposed model (see Table 2 and Figure 6).

Table 2. Results of simulating data sets

Methods	Average MSE	Average MAPE	Duncan's test grouping of first comparison	Duncan's test grouping of second comparison
ARIMA	1.427281	0.9306678	b (b)	a (a)
ANN	1.475889	0.9472932	b (b)	a (a)
ARIMA-ANN	1.475018	0.9377025	b (b)	a (a)
DWT-ARIMA ANN	2.678607	1.1365884	a (a)	Eliminated
ARIMA-DWT-ANN	1.239068	0.8518634	b (b)	b (b)

*values in the brackets are the results for MAPE.

*alphabets in Duncan's test results are arranged from higher error to lower error.
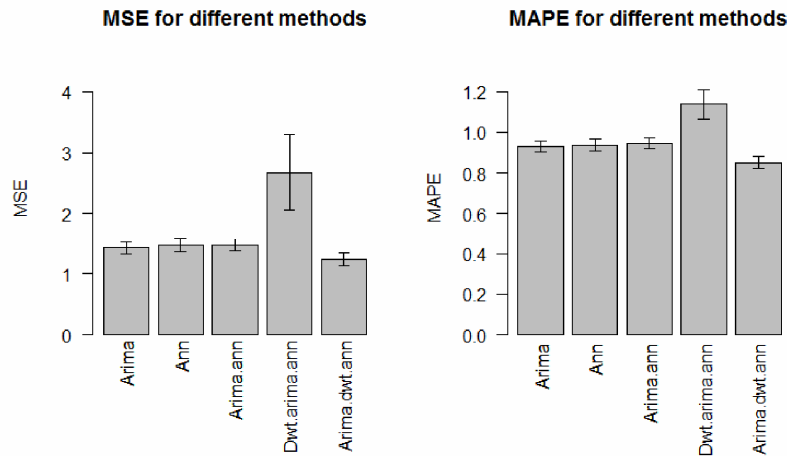
Figure 6. Bar plots of MSE and MAPE in simulating data sets.

Sample size effect, method effect and their interaction effect are tested by a two-way ANOVA test. The results show that all effects are significant, the extended Duncan's tests on method effect and sample size effect also strengthen this statement. However, if we withdraw DWT-ARIMA-ANN then apply another two-way ANOVA test, the results show that none of the effects are significant anymore. These results imply that DWT-ARIMA-ANN is the method mostly affected by sample size. These tests point out the importance of data sample size when comparing different predicting models.

Real data sets are used to check method effect and validation effect. The two-way ANOVA tables for Lynx data's MSE and MAPE both show that the effects for method, validation and interaction are all significant. Duncan's tests are applied to check for difference between methods and validation. As for the Duncan's tests responsible for method effect in Lynx's data. Both results conclude ARIMA-DWT-ANN as the method with lowest error, which is consistent with the result from simulations. On the other hand, the Duncan's test checking for validation effect shows that applying methods without validating data would underestimate their prediction errors (see Table 3).

Table 3. Results of Lynx data

Methods	Average MSE	Average MAPE	Duncan's test grouping of method effect	Results of validation effect
ARIMA	0.06163278	7.409021	b (b)	Validate
ANN	0.05195298	5.761306	c (c)	0.06799258 (7.106881)
ARIMA-ANN	0.06575318	7.649821	b (b)	Without validate
DWT-ARIMA ANN	0.11508250	9.342198	a (a)	0.06207600 (6.971839)
ARIMA-DWT-ANN	0.03075001	5.034454	d (d)	

*values in the brackets are the results for MAPE.

*alphabets in Duncan's test results are arranged from higher error to lower error.

　　　Different from the Lynx data, the ANOVA tables for cabbage data's MSE and MAPE show a mix result. In MSE part, the result shows that both method effect, validation effect and interaction effect are all significant. However, MAPE part claims that only method effect is significant. Although validation effect and interaction effect are not significant in MAPE part, we still apply a Duncan's test for it. The follow up Duncan's tests comparing different methods in MSE and MAPE consistently point out that ANN is the best model among all. While DWT-ARIMA-ANN is the one with second last performance in Lynx data, it performs quite well as the second best in cabbage data. Duncan's tests focusing on validation effect are carried out. In MSE part, it turns out that neglecting validation would cause the error to be significantly lower. On the other hand, the result for MAPE part continued the conclusion from its ANOVA table, which has no significance. The results show that validation in cabbage data is not as important as it does in Lynx data.

Table 4. Results of cabbage data

Methods	Average MSE	Average MAPE	Duncan's test grouping of method effect	Results of validation effect
ARIMA	0.06163278	7.409021	b (b)	Validate
ANN	0.05195298	5.761306	c (c)	0.06799258 (7.106881)
ARIMA-ANN	0.06575318	7.649821	b (b)	Without validate
DWT-ARIMA ANN	0.11508250	9.342198	a (a)	0.06207600 (6.971839)
ARIMA-DWT-ANN	0.03075001	5.034454	d (d)	

*values in the brackets are the results for MAPE.
*alphabets in Duncan's test results are arranged from higher error to lower error.

　　　When acquiring a brand new data, it is clear that researchers can never use future data to determine the model structure. Determining model structure based on testing data is absurd for instance. However, many researchers did not apply validation for model structure decision.

The neglect of validation would lead to an underestimation of errors. As for results of cabbage data, the underestimation is negligible. Concerning that validation is crucial in Lynx data and one can never foresee how validation influence the predictions, the process should not be skipped.

It is observed that the forecasts for ARIMA-ANN and ARIMA-DWT-ANN are highly correlated with ARIMA predictions. Since ARIMA performs better in Lynx data than it does in cabbage data, the performance ranking for two methods also change with ARIMA's rank. The differences between the variations and ARIMA usually locate at the arc of the prediction patterns. Those locations are the places where ARIMA's errors usually occur. This phenomenon implies the usage of ARIMA alike hybrid models. In both data sets, ANN earns considerably stable performance rankings, which are the second place in Lynx data and the first place in cabbage data. This observation shows the strong and robust usage of ANN in real data tasks. Combining results from simulations and real data sets, there is no such the best model for forecasting.

In this paper, only Haar wavelet is considered in DWT and IDWT. Although Haar wavelet is widely used to eliminate noises in data, other wavelet types such as Daubechies wavelet should be introduced and tested for their application. Another simplified issue should be mentioned is the determination for suitable activating function. Since all compared methods are using the same activating function with the same learning parameter decision process, the comparison results of this paper should hold. However, it is still suggested to use other activating function in further applications for sigmoid function is already an old fashion method. Also, in the part of learning parameter decision, a grid search is done from 0.01 to 0.99, the values out of scale are not considered and could slightly affect the prediction results. The last but not the least, the model structure decision in this paper is done by mainly two methods. While ARIMA is done by Hyndman and Khandakar's method, ANN is

determined by the proposed stepwise comparing method. The proposed method is basically an improved and simplified version of grid search. Since the performances of ANN models are acceptable, the usage of the method is confirmed. However, these two methods for model structure determination could be switched depending on the need. Researchers are free to determine model structure based on their own background knowledge.

5. Conclusion

Forecasting has always been an important issue in statistics and computer science. While researchers develop several methods for forecasting in different areas, more accurate predictions can sometimes be observed by combining different methods. This study focuses on different combinations between ARIMA, ANN and DWT and confirms the usage of hybrid models. In the results of simulated data sets, the proposed ARIMA-DWT-ANN is shown to perform the best among all methods and the sample size effect does influence model performance. While ARIMA-DWT-ANN out competes other methods in Lynx data, it no longer forecast the best in cabbage data. In cabbage data, ANN is the best method among all while DWT-ARIMA-ANN occupies the second place. The necessity of validation is confirmed by both real data sets to prevent from underestimating errors. Predictions for ARIMA-ANN and ARIMA-DWT-ANN are observed to be highly correlated with ARIMA's predictions, where their differences mainly occur at the arcs of the series. According to the results from simulated data sets and two real data sets, ANN and ARIMA-DWT-ANN are more robust in comparison to others. As for further researches, they are suggested to be applied first when encountering novel series.

Acknowledgement

References

[1]   A. J. Conejo, M. A. Plazas, R. Espinola and A. B. Molina, Day-ahead electricity price forecasting using the wavelet transform and ARIMA models, IEEE Transactions on Power Systems 20(2) (2005), 1035-1042.

DOI: https://doi.org/10.1109/TPWRS.2005.846054

[2]   M. Cottrell, B. Girard, Y. Girard, M. Mangeas and C. Muller, Neural modeling for time series: A statistical stepwise method for weight elimination, IEEE Transactions on Neural Networks 6(6) (1995), 1355-1364.

DOI: https://doi.org/10.1109/72.471372

[3]   A. K. Dhamija and V. K. Bhalla, Financial time series forecasting: Comparison of neural networks and ARCH models, International Research Journal of Finance and Economics 49 (2010), 194-212.

[4]   E. Guresen, G. Kayakutlu and T. U. Daim, Using artificial neural network models in stock market index prediction, Expert Systems with Applications 38(8) (2011), 10389-10397.

DOI: https://doi.org/10.1016/j.eswa.2011.02.068

[5]   B. Hassibi and D. G. Stork, Second order for network pruning: Optimal brain surgeon, Advances in Neural Information Processing Systems 5 (1994), 164-171.

[6]   R. J. Hyndman and Y. Khandakar, Automatic time series forecasting: The forecast package for R, Journal of Statistical Software 27(3) (2008), 1-22.

DOI: https://doi.org/10.18637/jss.v027.i03

[7]   I. Khandelwal, R. Adhikari and G. Verma, Time series forecasting using hybrid ARIMA and ANN models based on DWT decomposition, Procedia Computer Science 48 (2015), 173-179.

DOI: https://doi.org/10.1016/j.procs.2015.04.167

[8]   Y. Le Cun, J. S. Denker and S. A. Solla, Optimal brain damage, Advances Neural Information Process System 2 (1990), 598-605.

[9]   S. Makridakis, A. Anderson, R. Carbone, R. Fildes, M. Hibon, R. Lewandowski, J. Newton, E. Parzen and R. Winkler, The accuracy of extrapolation (time series) methods: Results of a forecasting competition, Journal of Forecasting 1(2) (1982), 111-153.

DOI: https://doi.org/10.1002/for.3980010202

[10]   W. Pannakkong, S. Sriboonchitta and V.-N. Huynh, An ensemble model of ARIMA and ANN with restricted Boltzmann machine based on decomposition of discrete wavelet transform for time series forecasting, Journal of Systems Science and Systems Engineering 27(5) (2018), 690-708.

DOI: https://doi.org/10.1007/s11518-018-5390-8

[11]   A. Sfetsos, A comparison of various forecasting techniques applied to mean hourly wind speed time series, Renewable Energy 21(1) (2000), 23-35.

DOI: https://doi.org/10.1016/S0960-1481(99)00125-1

[12]   B. L. Smith and M. J. Demetsky, Traffic flow forecasting: Comparison of modeling approaches, Journal of Transportation Engineering 123(4) (1997), 261-266.

DOI: https://doi.org/10.1061/(ASCE)0733-947X(1997)123:4(261)

[13]   Z. Tang and P. A. Fishwick, Feedforward neural nets as models for time series forecasting, ORSA Journal on Computing 5(4) (1993), 374-385.

DOI: https://doi.org/10.1287/ijoc.5.4.374

[14]   L. Wang, H. Zou, J. Su, L. Li and S. Chaudhry, An ARIMA-ANN hybrid model for time series forecasting, Systems Research and Behavioral Science 30(3) (2013), 244-259.

DOI: https://doi.org/10.1002/sres.2179

[15]   G. P. Zhang, B. E. Patuwo and M. Y. Hu, A simulation study of artificial neural networks for nonlinear time-series forecasting, Computers & Operations Research 28(4) (2001), 381-396.

DOI: https://doi.org/10.1016/S0305-0548(99)00123-9

[16]   P. G. Zhang, Time series forecasting using a hybrid ARIMA and neural network model, Neurocomputing 50 (2003), 159-175.

DOI: https://doi.org/10.1016/S0925-2312(01)00702-0

∎