An ensemble strategy for multi-step-ahead time series forecasting

JunYong Jeong
Department of Industrial and Management Engineering
POSTECH, Pohang, Korea
Tel: (+82) 54-279-2197, Email: june0227@postech.ac.kr

Chi-Hyuck Jun †
Department of Industrial and Management Engineering
POSTECH, Pohang, Korea
Tel: (+82) 54-279-2197, Email: chjun@postech.ac.kr

Abstract. A large and growing body of literature has investigated on multi step ahead time series forecasting. Because no single model defeats the others for all circumstances, a hybrid strategy has drawn attention. In this paper we propose a hybrid strategy based on ensemble method to improve performance of multi-step ahead time series forecasting. Least absolute shrinkage and selection operator regression excludes non-significant forecasts and determines the weights to avoid over-fitting. Experiment results on 60 series from NN3 competition showed that the proposed method improved forecasting accuracy over single models and a simple mean ensemble strategy.

Keywords: Multi-step-ahead time series forecasting, Auto-regressive moving average (ARIMA), Least square support vector regression (LSSVR), Least absolute shrinkage and selection operator (LASSO) regression, Ensemble strategy

1. INTRODUCTION

A large and growing body of literature has investigated on multi-step-ahead time series forecasting. Multi-step-ahead forecasting has more practical implications and applications than one-step ahead forecasting; however, the increased uncertainty and lack of information make it difficult to generate multi-step-ahead forecasts. (Bao et al., 2014).

Traditionally, linear models such as Auto-regressive integrated moving average (ARIMA) and Exponential smoothing (ES) have been used for multi-step-ahead time series forecasting. However, Artificial neural network (ANN) and Support vector regression (SVR) have drawn attention in the time series forecasting community (Sapankevy and Sankar, 2009; Zhang, 2012) because the linear models have a lack of explanation power (De Gooijer and Hyndman, 2006). However, there is no algorithm that is always suitable for all circumstances. Thus, a hybrid strategy that combines several models could be a good candidate. A considerable amount of literature has been published on the hybrid strategy on time series forecasting (Zhang, 2003; Yu et al., 2005; Ren et al., 2015).

An ensemble strategy is a well-known and traditional approach in the hybrid strategy. The main issue in the ensemble strategy is how to determine the weights of different forecasting models. Previous studies solve this problem by utilizing simple mean (Andrawis et al., 2011), optimization technique (Rather et al., 2015), forecasting error (Wichard, 2011) and forecasting model (Wang and Hu, 2015). However, they have a limited ability to exclude forecasting models that degrade forecasting accuracy.

We propose an ensemble strategy based on least absolute shrinkage and selection operator (LASSO) regression to determines the weights and exclude non-significant forecasting models. The proposed method was compared to several benchmark models over 60 long time series from NN3 competition.

2. METHODOLOGY

2.1 ARIMA

ARIMA describes the future value of time series as a linear function of past values and error terms (Box and Jenkins, 1976). An ARIMA(p, d, q) model of degree of AR (d), difference (d) and MA (q) is as follow:

$$x_t = \theta_0 + \varphi_1 x_{t-1} + \varphi_2 x_{t-2} + \cdots + \varphi_p x_{t-p} + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} - \cdots - \theta_q \varepsilon_{t-q} \quad (1)$$

where x_t is the actual value obtained by differencing d times, ε_t is the random error at time t, p and q are the numbers of auto-regressive and moving average terms in the ARIMA model, and φ_i $(i = 1, \dots, p)$ and θ_j $(j = 1, \dots, q)$ are the model parameters to be estimated.

2.2 LSSVR

Least square support vector regression (LSSVR) is a variant of standard Support vector regression to reduce training time (Suykens et al., 1999). LSSVR solves a linear problem instead of a quadratic problem by modifying its constraints.

Given the training set $\mathbf{x}_i, y_i \ \ i = 1, \dots, N$ with input $\mathbf{x}_i$ and output y_i, LSSVR solves the following problem:

$$\min J(\mathbf{w}, \mathbf{e}) = \frac{1}{2}\mathbf{w}^T\mathbf{w} + \gamma \frac{1}{2}\sum_{i=1_0}^{N} e_i^2 \quad (2)$$

Subject to

$$y_i = \mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_i) + b + e_i, \qquad i = 1, \dots, N \quad (3)$$

where $\mathbf{w}$ is the weight vector, γ is the penalty parameter, $\mathbf{e} = (e_1, \dots, e_N)^T$ is the vector of the approximation error , $\boldsymbol{\phi}(\cdot)$ is the nonlinear mapping function and b is the bias term. The above problem is solved by introducing the Lagrangian multipliers α_i

$$L(\mathbf{w}, \mathbf{e}, \boldsymbol{\alpha}, b) = J(\mathbf{w}, \mathbf{e}) - \sum_{i=1}^{N} \alpha_i \mathbf{w}^T \boldsymbol{\phi}(x_i) + b + e_i - y_i \quad (4)$$

where $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_N)^T$ is the vector of the Lagrangian multipliers.

Then, the Karush-Kuhn-Tucker conditions are applied to optimize the Lagrangian.

$$\begin{cases} \dfrac{\delta L}{\delta \mathbf{w}} = 0 \rightarrow \mathbf{w} = \sum_{i=1}^{N} \alpha_i \phi(x_i) \\ \dfrac{\delta L}{\delta b} = 0 \rightarrow \sum_{i=1}^{N} \alpha_i = 0 \\ \dfrac{\delta L}{\delta e_i} = 0 \rightarrow \alpha_i = \gamma e_i \\ \dfrac{\delta L}{\delta \alpha_i} = 0 \rightarrow \mathbf{w}^T \phi(x_i) + b + e_i - y_i = 0 \end{cases} \quad (5)$$

After the elimination of w and e_i, the equations could be transformed into

$$\begin{bmatrix} b \\ \boldsymbol{\alpha} \end{bmatrix} = \begin{bmatrix} 0 & \mathbf{I}_v^T \\ \mathbf{I}_v^T & \boldsymbol{\Omega} + \gamma^{-1}\mathbf{I} \end{bmatrix} \begin{bmatrix} 0 \\ \mathbf{y} \end{bmatrix} \quad (6)$$

where $\mathbf{y} = [y_1, \dots, y_N]$, $\mathbf{I}_v = [1, \dots, 1]^T$, and the Mercer condition has been applied to the matrix $\boldsymbol{\Omega}$ with $\Omega_{km} = \phi(\mathbf{x}_k)^T \phi(\mathbf{x}_m)$, $k, m = 1, \dots, N$. Thus, the LSSVR becomes

$$y(\mathbf{x}) = \sum_{i=1}^{N} \alpha_i K(\mathbf{x}, \mathbf{x}_i) + b \quad (7)$$

2.3 LASSO regression

LASSO regression aims to perform feature selection and regression simultaneously (Tibshirani, 1960). Lasso does that by introducing a penalty term in its objective function

Given the training set $\mathbf{x}_i, y_i \ \ i = 1, \dots, N$ with input $\mathbf{x}_i$ and output y_i. LASSO model can be written as follows:

$$y_i = \beta_0 + \boldsymbol{\beta}_1^T \mathbf{x}_i + e_i \quad (8)$$

where β_0 is the intercept and $\boldsymbol{\beta}_1$ is the slope. Let $\boldsymbol{\beta} = [\beta_0, \boldsymbol{\beta}_1^T]^T$ be the coefficient of model. The coefficient $\boldsymbol{\beta}$ is computed by solving the following problem

$$\min_{\boldsymbol{\beta}} \left\{ \frac{1}{2}\sum_{i=1}^{N} (y_i - \beta_0 - \boldsymbol{\beta}_1^T \mathbf{x}_i)^2 + \lambda \parallel \boldsymbol{\beta}_1 \parallel_1 \right\} \quad (9)$$

where λ is the regularization parameter that controls trade-off between data-fitting term $\sum_{i=1}^{N} (y_i - \beta_0 - \boldsymbol{\beta}_1^T \mathbf{x}_i)^2$ and model complexity term $\parallel \boldsymbol{\beta}_1 \parallel_1$.

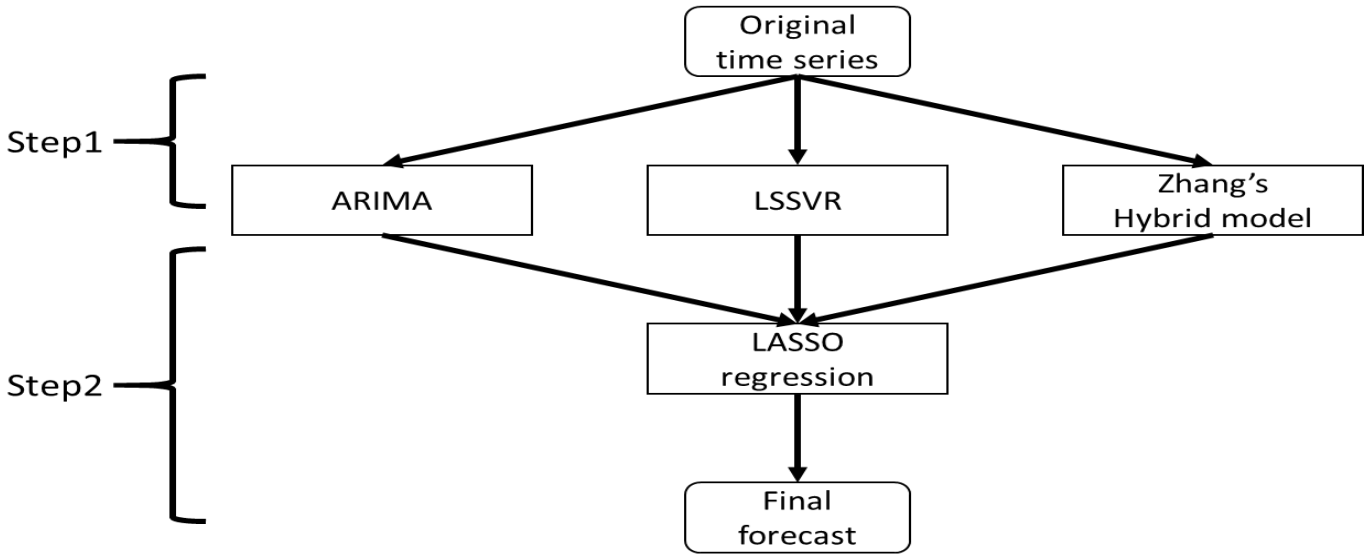

Figure 1: The procedure of the proposed method

2.3 Zhang's hybrid model with LSSVR

Zhang's hybrid model (Zhang, 2003) assumes that time series x_t is composed of the sum of a linear component and a nonlinear component

$$x_t = L_t + N_t \tag{10}$$

where L_t is the linear component and N_t is the nonlinear component. First, fit ARIMA to model linear component and generate fitted series $\hat{L}_t$ and residuals $\varepsilon_t = x_t - \hat{L}_t$. These residuals are considered to contain only the nonlinear relationship. Then, model the residuals using LSSVR to discover the nonlinear relationship

$$\varepsilon_t = f(\varepsilon_{t-1}, \varepsilon_{t-2}, \dots, \varepsilon_{t-n}) + e_t \tag{11}$$

where f is the nonlinear function built by the LSSVR and e_t is the random. The forecast from (11) is considered as $\hat{N}_t$. The combined forecast is given by

$$\hat{x}_t = \hat{L}_t + \hat{N}_t \tag{12}$$

2.4 Proposed method

In the ensemble strategy, a main problem is determining the weights of forecasts. Suppose L forecasts $\hat{x}_t^1, \hat{x}_t^2, \dots, \hat{x}_t^L$ are given. The general form of the ensemble strategy can be expressed as follows:

$$\hat{x}_t = \sum_{j=1}^{L} w^j \hat{x}_t^j \tag{13}$$

where w^j is the assigned weight of forecasts $\hat{x}_t^j$.

We applied LASSO regression to find the weight w^j. LASSO regression selects the weights that avoid overfitting by excluding non-significant forecasts. If one forecasting model is significantly outperformed by the others, LASSO regression would exclude it from the final forecast.

The figure 1 presents the proposed method.

> ***Step1***: Build ARIMA, LSSVR and Zhang's hybrid model with LSSVR in recursive strategy (Taieb and Atiya, 2016) to generate multi-step ahead forecast.
>
> ***Step2***: Learn LASSO regression to determine the weights in ensemble strategy based on the fitted values

Only one-step-ahead model is trained for each forecasting model because the proposed method adopts the recursive strategy for multi-step-ahead forecasting. Thus LASSO regression is learned based on one-step-ahead fitted values.

3. EXPERIMENT

3.1 The data description

The NN3 competition dataset[1] was selected for the evaluation of the proposed method. It was drawn from

[1]Available at http://www.neural-forecasting-competition.com/NN3/

Table 1: Experiment result of forecasting models

Forecasting model	Forecasting horizon h				Average			
	1	6	12	18	1-6	7-12	13-18	1-18
SMAPE								
LSSVR	10.52	17.38	23.90	25.86	12.99	17.94	21.24	17.39
ARIMA	12.78	23.11	23.29	20.50	17.47	19.82	19.56	18.95
Zhang's hybrid	9.77	20.99	21.46	19.34	15.14	18.20	18.25	17.20
Simple mean	9.94	19.23	22.61	19.95	14.15	17.96	18.16	16.76
Proposed method	10.55	16.81	18.26	18.42	12.99	15.65	16.72	15.12
MASE								
LSSVR	0.64	2.50	16.88	4.76	1.37	5.48	3.29	3.38
ARIMA	0.75	1.62	1.76	2.65	1.18	1.83	2.16	1.72
Zhang's hybrid	0.61	1.50	1.63	2.59	1.07	1.72	2.08	1.62
Simple mean	0.62	1.62	6.66	3.16	1.10	2.88	2.29	2.09
Proposed method	0.66	1.50	1.57	2.58	1.03	1.69	2.06	1.59

business area with monthly frequency and positive observations. The most of series have seasonal and trend behavior and high-level of noise. Among 111 time series, only 60 long series where the training length of series is more than 84 were used to ensure enough training examples for the learning of the proposed method. Our objective is to forecast next 18 months based on the given history.

3.2 Performance evaluation

We applied two error measures to evaluate the performance of the proposed method: symmetric mean absolute percentage error (SMAPE) and mean absolute scaled error (MASE); both error measures are scale-independent.

$$SMAPE_h = \frac{1}{M}\sum_{m=1}^{M}\left|\frac{x_{N+h}^m - \hat{x}_{N+h}^m}{x_{N+h}^m + \hat{x}_{N+h}^m}\right| \times 200 \qquad (14)$$

$$MASE_h = \frac{1}{M}\sum_{m=1}^{M}\left|\frac{x_{N+h}^m - \hat{x}_{N+h}^m}{\frac{1}{N-1}\sum_{i=2}^{N}\left|x_i^m - x_{i-1}^m\right|}\right| \qquad (15)$$

where $\hat{x}_{N+h}^m$ is the h-step ahead forecast for time series m, x_{N+h}^m is the actual value for time series m, H is the prediction horizon ($H = 18$) and M is the number of time series ($M = 60$).

3.3 Model learning

Partial auto-correlation was adopted for input selection method because of its simplicity. Considering that the data set is monthly, we set the maximum number of lags to 12.

An automatic model selection procedure in *forecast* package in R (Hyndman and Khandakar, 2008) was chosen to fit ARIMA model. It is based on statistical tests and an information criterion. The RBF kernel function was chosen for LSSVR. The hyper-parameters of LSSVR were selected by Nelder-Mead simplex algorithm (Nelder and Mead, 1965) to minimize mean squared error of 10-fold cross validation. The starting points of simplex were determined by Coupled simulated annealing. The regularization parameter λ for LASSO regression was decided by minimizing mean squared error of 10-fold cross validation.

The proposed method was compared to single models and the simple average of them. The single models consisted of LSSVR, ARIMA and Zhang's hybrid model.

Table 2: The number of selected series

The forecasting model	The number of selected series
ARIMA	37
LSSVR	47
Zhang's hybrid	50

3.4 Result

Table 1 provides the performance of the forecasting models based on the accuracy measures and forecasting horizon. Zhang's hybrid model showed the lowest error among the single models on both SMAPE and MASE. The simple mean ensemble strategy only improved SMAPE. In contrast, the proposed method succeeded to reduce both SMAPE and MASE. The improvement of the proposed method over the best single model, Zhang's hybrid, on SMAPE $(17.20 - 15.12)/17.20 = 12.09\%$ was larger than the one on MASE $(1.62 - 1.59)/1.62 = 1.85\%$. This

results could be explained by the following analysis on the weights in the proposed method.

Table 2 summarizes the weights of the single models in the proposed method. The first column indicates the forecasting model. The number of the series of having the non-zero weight was provided in the second column. We found that the trend in the weights was related to only SMAPE, not MASE. In other words, a chance to have the non-zero weight was only decreasing in SMAPE. ARIMA, for example, showed the worst performance on SMAPE; it had the lowest chance to be selected by LASSO regression. Thus, the improvement of the proposed method was larger on SMAPE than the one on MASE.

4. CONCLUSION

In this paper, we proposed ensemble strategy based on LASSO regression to improve forecasting performance of multi-step-ahead time series forecasting. The experiment on 60 time series from NN3 competition showed that the proposed method improved both SMAPE and MASE over the benchmark models by excluding the non-significant forecast and determining the weights that avoid over-fitting.

The major limitation of the proposed method is that it is based on only general supervised learning theory. It doesn't consider the property of time series. Thus, future research should be undertaken to incorporate property of time series into the proposed method. One way to do that is applying decomposition method such as Seasonal trend decomposition using Loess (Cleveland et al., 1990) and Empirical mode decomposition (Huang et al., 1996) to deal with trend and seasonality.

ACKNOWLEDGMENTS

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MEST) (NRF-2013-R1A2A2A0306832)

REFERENCES

Andrawis, R. R., Atiya, A. F., and El-Shishiny, H. (2011) Forecast combinations of computational intelligence and linear models for the NN5 time series forecasting competition. *International Journal of Forecasting*, **27**, 672-688.

Bao, Y., Xiong, T., and Hu, Z. (2014) Multi-step-ahead time series prediction using multiple-output support vector regression. *Neurocomputing,* **129**, 482-493

Box, G. E. and Jenkins, G. M. (1976) *Time series analysis: forecasting and control,* Holden-day, San Francisco.

Cleveland, R. B., Cleveland, W. S., McRae, J. E., and Terpenning, I. (1990) STL: A seasonal-trend decomposition procedure based on loess. *Journal of Official Statistics*, **6**, 3-73.

De Gooijer, J. G. and Hyndman, R. J. (2006) 25 years of time series forecasting. *International Journal of Forecasting,* **22**, 443-473.

Huang, N. E., Shen, Z., Long, S. R., Wu, M. C., Shih, H. H., Zheng, Q., Yen, N.-C., Tung, C. C., and Liu, H. H. (1998) The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis. *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, **454**, 903-995.

Hyndman, R. J. and Khandakar, Y. (2008) Automatic Time Series Forecasting: The forecast Package for R. *Journal of Statistical Software,* **27**, 22.

Nelder, J. A. and Mead, R. (1965) A Simplex Method for Function Minimization. *The Computer Journal,* **7**, 308-313.

Rather, A. M., Agarwal, A., and Sastry, V. N. (2015) Recurrent neural network and a hybrid model for prediction of stock returns. *Expert Systems with Applications*, **42**, 3234-3241.

Ren, Y., Suganthan, P. N., and Srikanth, N. (2015) Ensemble methods for wind and solar power forecasting—A state-of-the-art review. *Renewable and Sustainable Energy Reviews,* **50**, 82-91.

Sapankevych, N. I. and Sankar, R. (2009) Time Series Prediction Using Support Vector Machines: A Survey. *IEEE Computational Intelligence Magazine,* **4**, 24-38.

Suykens, J. A. K. and Vandewalle, J. (1999) Least Squares

Support Vector Machine Classifiers. *Neural Processing Letters,* **9**, 293-300.

Taieb, S. B. and Atiya, A. F. (2016) A Bias and Variance Analysis for Multistep-Ahead Time Series Forecasting. *IEEE Transactions on Neural Networks and Learning Systems*, **27**, 62-76.

Tibshirani, R. (1996) Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, **58**, 267-288.

Wang, J. and Hu, J. (2015) A robust combination approach for short-term wind speed forecasting and analysis – Combination of the ARIMA (Autoregressive Integrated Moving Average), ELM (Extreme Learning Machine), SVM (Support Vector Machine) and LSSVM (Least Square SVM) forecasts using a GPR (Gaussian Process Regression) model. *Energy*, **93**, Part 1, 41-56.

Wichard, J. D. (2011) Forecasting the NN5 time series with hybrid models. *International Journal of Forecasting*, **27**, 700-707.

Yu, L., Wang, S. and Lai, K. K. (2005) A novel nonlinear ensemble forecasting model incorporating GLAR and ANN for foreign exchange rates. *Computers and Operations Research,* **32**, 2523-2541.

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

Zhang, G. P. (2012) Neural Networks for Time-Series Forecasting. *In G. Rozenberg, T. Bäck and J. N. Kok (Eds.), Handbook of Natural Computing (Berlin: Heidelberg),* chapter 14, 461-477.