Ekoist: Journal of Econometrics and Statistics, 37, 53-68

İSTANBUL ÜNİVERSİTESİ YAYINEVİ

DOI: 10.26650/ekoist.2022.37.1108411
http://ekoist.istanbul.edu.tr

Ekoist: Journal of Econometrics and Statistics

Received: 25.04.2022
Revision Requested: 14.07.2022
Last Revision Received: 04.08.2022
Accepted: 10.10.2022

RESEARCH ARTICLE / ARAŞTIRMA MAKALESİ

Hybrid Approaches in Financial Time Series Forecasting: A Stock Market Application

Finansal Zaman Serileri Tahmininde Hibrit Yaklaşımlar: Bir Hisse Senedi Piyasası Uygulaması

Canberk Bulut[*], **Burcu Hüdaverdi**[**]

Abstract

The hybrid approach in time series forecasting is one of the key methodologies in selecting the most accurate model when compared to the single models. Applications of machine learning algorithms in hybrid modeling for stock market forecasting have been developing rapidly. In this paper, we propose hybrid modeling through machine learning approach for four stock market data; two from the developed stock markets (NASDAQ and DAX) and the other two from the emerging stock markets (NSE and BIST). A stock market is known with its volatile structure and has an unstable nature, so we propose several combinations for the hybrid models considering volatility to reach the most accurate time series forecasting model. In hybrid modeling, first ARIMA (Autoregressive Integrated Moving Average) models combined with GARCH models (Generalized Autoregressive Conditional Heteroscedasticity) are used for modeling of time series, then intelligent models such as SVM (support vector machine) and LSTM (Long-Short term memory) are used for nonlinear modeling of error series. We also compare their performances with single models. The proposed hybrid methodology markedly improves the prediction performances of time series models by combining several models which reflect the time series data characteristics best.

Keywords

Hybrid Time Series, Machine Learning, ARIMA, GARCH, SVM, LSTM, Stock Market

Öz

Zaman serisi tahmininde hibrit yaklaşım, tekli modellerle karşılaştırıldığında en doğru modeli seçmede anahtar metodolojilerden biridir. Hisse senedi piyasası tahmini için hibrit modellemede makine öğrenmesi algoritmalarının uygulanması hızla gelişmektedir. Bu çalışmada, ikisi gelişmiş hisse senedi piyasasından (NASDAQ ve DAX) ve diğer ikisi yükselen hisse senedi piyasasından (NSE ve BIST) olmak üzere dört hisse senedi verisi için makine öğrenimi yaklaşımıyla hibrit modellemesi uygulanmıştır. Bir hisse senedi piyasası, değişken yapısıyla bilinir ve istikrarsız bir yapıya sahiptir, bu nedenle, bu çalışmada, en doğru zaman serisi tahmin modeline ulaşmak için oynaklığı dikkate alan çeşitli hibrit modeller önerilmektedir. Hibrid modellemede, öncelikle GARCH (Generalized Autoregressive Conditional Hetereoscedastic) ile birleştirilen ARIMA (Autoregressive Integrated Moving Average) modelleri zaman serilerinin modellemesinde, ardından SVM (support vector machine) ve LSTM (Long-Short term memory) gibi zeki modeller hata serilerinin doğrusal olmayan modellemesinde kullanılmaktadır. Ayrıca, hibrit modellerin performansları mevcut metodolijiler kullanılarak tekli modeller ile karşılaştırılmaktadır. Önerilen hibrit metodoloji, zaman serisi verisinin özelliklerini en iyi yansıtan birkaç modeli birleştirerek tahmin performanslarını önemli ölçüde iyileştirmektedir.

Anahtar Kelimeler

Hibrit Yaklaşımlar, Makine Öğrenimi, ARIMA, GARCH, SVM, LSTM, Hisse Senedi Piyasası

* Canberk Bulut (Ph.D. Student), Dokuz Eylul University, Faculty of Sciences, Department of Statistics, Buca, Izmir, Turkiye. E-mail: canberk.bulut@ogr.deu.edu.tr ORCID: 0000-0001-8203-4770

** **Corresponding author:** Burcu Hüdaverdi (Prof. Dr.), Dokuz Eylul University, Faculty of Sciences, Department of Statistics, Buca, Izmir, Turkiye. E-mail: burcu.hudaverdi@deu.edu.tr ORCID: 0000-0002-6939-9668

To cite this article: Bulut, C., & Hudaverdi, B. (2022). Hybrid approaches in financial time series forecasting: A stock market application. *EKOIST Journal of Econometrics and Statistics*, 37, 53-68. https://doi.org/10.26650/ekoist.2022.37.1108411

Introductıon

Time series forecasting is one of the main tasks in many areas, especially finance, economics, logistics, supply chain, etc. It is generally hard to predict financial time series data due to the unpredictable changes in the economy. Market volatility in recent years has caused serious problems for time series forecasting. Therefore, in this study, we have two main tasks: improving forecasting accuracy and model selection. We investigate which time series forecasting methods have the best predictions concerning lower forecast errors.

One of the most used methods in time series forecasting is the Autoregressive Integrated Moving Average (ARIMA) where Autoregressive (AR) and Moving Average (MA) models are combined. See also, Box and Jenkins methodology (Box, Jenkins, Reinsel, & Ljung, 2015; Zhang, 2003). Moreover, a financial market is a complex, evolutionary, and high volatile system (Chiang, Urban, & Bailridge, 1996), and as a result of this, a financial time series is noisy and non-stationary (Yaser, & Atiya, 1996). This means forecasting of financial time series is challenging. Also, in order to model the volatility, Engle (1992) suggested the ARCH (Autoregressive Conditional Heteroscedasticity) and also Bollerslev (1986) introduced GARCH (Generalized Autoregressive Conditional Heteroscedasticity) models.

Machine learning methods have introduced new approaches in time series modelling. Especially, deep-learning based algorithms can define and model non-linear or more complex time-series data.

An artificial neural network (ANN) is a type of machine learning algorithm and widely applied in various fields since it has fewer assumptions and restrictions compared to many conventional statistical methods. ANNs are able to approximate the nonlinearities in the data. Zhang et al. (1998) presented a state-of-the-art survey of ANN applications in forecasting. Generally, ANN is a single hidden layer feedforward network for time series modelling and forecasting. Recurrent neural networks (RNNs) can also be used to learn temporal patterns, so they can model effective sequential data and can be used in time series analyses. As a special type of RNN, recently, the Long-Short term memory (LSTM) method which is a deep-learning based forecasting algorithm has gained interest since generally it has good accuracy and precision in forecasting. Maknickiene and Maknickas (2012) used LSTM to improve prediction performance over feedforward neural networks for financial time series. Chen et al. (2015) presented that LSTM has better performance for predicting returns in Chinese stock market when compared to the random prediction method. With LSTM, there is no need to identify a model for forecasting.

Another method mostly used is called support vector machine (SVM) method which was developed by Vapnik (1995). SVMs are used for pattern recognition problems (Scholkopf, Burges, & Vapnik,1995). Recently, SVMs have been developed to solve nonlinear regression estimation problems, and they generally have a good performance (Muller, Smola, & Scholkopf, 1997). Kim (2003) used SVM in financial time series forecasting. In financial applications, SVM is preferable since the method uses risk function consisting of empirical error and regularized term which is based on the structural risk minimization principle. However, in modelling both nonlinear and linear patterns for time series data, these methods may not be sufficient alone.

In this study, hybrid systems combining linear and nonlinear structures of the time series data are used to improve the forecast accuracy of the model. The hybrid approach employs residual-modeling in order to increase the model performance. (Zhang, 2003; Panigrahi, & Behera,2017). Domingos et al. (2019) used a hybrid approach using ARIMA with Multi-Layer Perceptron (MLP) and Support Vector Regression (SVR) for time series forecasting. Zhang (2003) also used ARIMA with neural network models for time series forecasting. Kim and Won (2018) presented a hybrid LSTM model to forecast the stock price. In our hybrid modelling, we consider the time series data composed of linear autocorrelation and nonlinear component. First, we use ARIMA or ARIMA-GARCH models for modeling first component, so the residuals obtained from this model represent the nonlinear structure of the time series data. Second, we use LSTM or SVM for residual modelling in order to detect the nonlinear time-series structure of the data. The hybrid models have advantages because both linear and nonlinear parts of the time-series data are modelled separately by using various models, and then the models are combined to improve the forecasting performance. Several combinations of the forecasting models through the hybrid approach are used to reach the most accurate forecasting result.

The study is organized as follows. In Section 2, the methods used in time series forecasting are given. The hybrid methodology is also given in this section. The stock market analysis and comparisons of the methods are given in Section 3. And Section 4 is devoted to the conclusion.

Time Series Forecasting Models

ARIMA

The autoregressive integrated moving averages (ARIMA) model combines the autoregressive model (AR) and the moving averages model (MA) for stationarized time series. This means that the stationary time series is modelled by a linear function of the past observations and the random errors.

The AR part of the model at the p level can be simply expressed as:

$$x_t = c + \sum_{i=1}^{p} \varphi_i x_{t-i} + \epsilon_t, \tag{1}$$

where x_t is the stationary variable, c is constant, φ_i, i=1,2,...,p are autocorrelation coefficients in lags and residuals ϵ_t are the Gaussian white noise series.

MA part of the model at the q level can be simply expressed as:

$$x_t = \mu + \sum_{i=0}^{q} \theta_i \epsilon_{t-i} \ , \tag{2}$$

where μ is the expectation of x_t (usually considered equal to 0), θ_i, i=1,2,...,q is the weight given to the past and present values in the stochastic period in the time series, and θ_0=1. ϵ_t is assumed as a Gaussian white noise series. We can combine these two models as follows:

$$x_t = c + \sum_{i=1}^{p} \phi_i x_{t-i} + \epsilon_t + \sum_{i=0}^{q} \theta_i \epsilon_{t-i} \ , \tag{3}$$

where $\phi_i \neq 0$, $\theta_i \neq 0$, and $\sigma_\epsilon^2 > 0$. The parameters p and q are the orders of AR and MA, respectively. ARIMA metholodology is also known as Box and Jenkins (1970) methodology. Box and Jenkins (1970) presented a useful approach for the ARIMA models. This method is also successful when dealing with non-stationary time series because it is "integrated". The "integrated" component actually includes the operation of taking the difference, which makes the non-stationary time series stationary. The general form of the ARIMA method in terms of machine learning approach is given by Sima et al. (2018).

GARCH

A GARCH model which has an additional value of conditional variance by restricting the parameters is used to forecast time series volatilities. The negative correlation between future return and volatility is not considered by a GARCH model. The GARCH (1, 1) model is given as in Eq. (7),

$$\sigma_t^2 = \alpha_0 + \alpha_1 \epsilon_{t-1}^2 + \beta_1 \sigma_{t-1}^2 \ , \tag{7}$$

By Eq. (7), we obtain Eq. (8) :

$$\sigma_t^2 = \alpha_0 + \alpha_1 \epsilon_{t-1}^2 + \beta_1 \sigma_{t-1}^2 \ , \tag{8}$$

The EGARCH is modelled by Nelson (1991). This model also does not require all coefficients of the distribution equation to be negative. Also, this model may include the leverage effect, which reflects the asymmetric effects of negative and positive effects of the same magnitude. We can define the EGARCH model as below:

$$r_t = X_t M + \epsilon_t \tag{9}$$

$$ln\sigma_t^2 = \alpha'_0 + \beta ln\sigma_{t-1}^2 + \omega(\frac{\epsilon_t - 1}{\sigma_t - 1}) + \gamma|\frac{\epsilon_t - 1}{\sigma_t - 1}| \ , \tag{10}$$

where r_t in Eq. (9) is the linear combination of independent variable X, parameter M and error term ϵ_t. Here, the conditional variance is always positive even the parameter estimation is negative. The parameter ω measures the leverage effect . So, for $\frac{\epsilon_t - 1}{\sigma_t - 1} < 0$, $\gamma - \omega$ is obtained, and for $\frac{\epsilon_t - 1}{\sigma_t - 1} > 0$, $\gamma + \omega$, is obtained, that is reflected asymmetrically. (Kim & Won, 2018)

LSTM

LSTM is a special type of Recursive Neural Network (RNN) with additional features for memorizing the data string. As a gradient-based method, LSTM has been first introduced by Hochreiter and Schmidhuber (1997). Each LSTM consists of a set of cells or system modules where data are captured and stored. The basic concept of LSTM contains cell states and gates. The cell state provides a communication line and network memory which carries meaningful information across cells for predictions. This information which the cell state carries is determined through the gates.

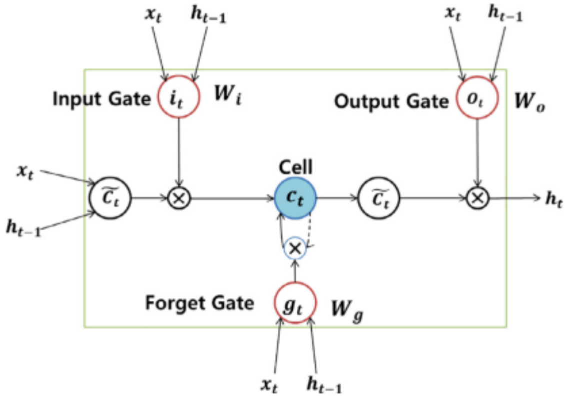

Figure 1. An LSTM flow.

In Figure 1, an LSTM flow is given. An LSTM cell is composed of memory cell (C_t), input gate (i_t), forget gate (g_t) and output gate (o_t). For a given time t, x_t is the input, h_t is the hidden state, and $\tilde{c}_t$ determines how much new information is received in the cell state. These elements are obtained as Eq.11-16 given below:

$$g_t = \sigma\left(U_g x_t + W_g h_{t-1} + b_f\right) \tag{11}$$

$$i_t = \sigma(U_i x_t + W_i h_{t-1} + b_i) \tag{12}$$

$$\tilde{c}_t = tanh(U_c x_t + W_c h_{t-1} + b_c) \tag{13}$$

$$c_t = g_t \times c_{t-1} + i_t \times \tilde{c}_t \tag{14}$$

$$o_t = \sigma(U_o x_t + W_o h_{t-1} + b_o) \quad (15)$$

$$h_t = o_t \times tanh(c_t) \quad (16)$$

In these equations, U and W are weight matrixes, b is a bias term, and σ (·) is a sigmoid function. See also, Kim, & Won (2018).

Forget Gate decides which information is necessary or to be ignored. The information from the input and the information from the hidden state $h_{t-1}$are passed through the sigmoid function.The sigmoid function produces values between 0 and 1. It is decided whether it is necessary according to the closeness of these generated values to 0 and 1.

Input Gate decides which of the new incoming data will be stored. First, a sigmoid layer which is called as input gate layer chooses which data to change/update. The tanh layer then constructs a vector of new candidate values that could be added to this state.

Output Gate determines the value of the next hidden state. This hidden state contains information from previous entries. The current state and the previous latent state pass through the sigmoid function. The new cell state passes through the tanh function. These two outputs are multiplied by each other. Based on the final value, the network decides what value the hidden state (h_t) will take. This hidden state is used for prediction.

SVM

The support vector machine is an extension of the support vector classifier resulting from extending the feature function in a certain way using Kernel functions. The main idea of the SVM method is to map the original vectors to a higher space and look for a separator hyperplane with a maximum margin in this space. Moreover, two parallel hyperplanes are obtained on either side of the hyperplane that divides it into classes. The separating hyperplane will be the main hyperplane where the distance is maximized between two parallel hyperplanes. For the linearly separable case, we can use following equations:

$$y = w_0 + w_1 x_1 + w_2 x_2 + w_3 x_3 \ , \quad (17)$$

where y is the output, x_i s are attributes, and w_is are weights. In Equation 17, the hyperplane is determined by weights. The maximum margin hyperplane is given as:

$$y = b + \sum \alpha_i y_i x(i).x \ , \quad (18)$$

where y_i can be determined as the $x(i)$ class value , "." represents the dot product. x-vector is determined as a test example and $x(i)$-vectors represent the support vectors. a_i and b both determine a hyperplane given in Equation 19.

Then, SVM builds a linear model to implement nonlinear classes by converting inputs to high-dimensional feature space. For the high-dimensional case, Equation 18 can be re-obtained as

$$y = b + \sum \alpha_i y_i K(x(i)x) \ , \tag{19}$$

where $K(x(\text{i})x)$ is the Kernel function. There are different kernels that generate inputs for construction of machines with different types of nonlinear decision surfaces. Choosing the model from different kernels can minimize the estimation error of the models.

The Hybrid Methodology

All the models mentioned in Section 2 are successful in linear and nonlinear domains. However, none alone works best for all situations. ARIMA or ARIMA-GARCH models may be more suitable for linear problems. On the other hand, LSTM and SVM models can yield complex results for linear problems, moreover, the performance of these models may depend on sample size and noise level. See, Markham and Rakes (1998). Also, ARIMA models may not be adequate to explain nonlinear data structure. Since it is really hard to know the characteristics of the time series structure of a real data, hybrid approaches can be used by combining several methodology to obtain a better predicted model. The hybrid models that combine linear and nonlinear time series models have improved the performances in terms of accuracy in various applicational fields. The proposed hybrid methodology is based on some previous literature. See, Zhang (2003), Kashei and Bijari (2010), Kashei and Bijari (2011), Bildirici and Ersin (2009), Pérez –cruz et al. (2003) and Kim, & Won (2018),etc.

A time series may be a combination of both linear and nonlinear model such as

$$y_t = L_t + N_t \ , \tag{20}$$

where L_t is a linear component and N_t is a nonlinear component. These two components are predicted from the data. First, the linear component is predicted from ARIMA or ARIMA-GARCH models. Then the residuals e_t are obtained as

$$e_t = y_t - \hat{L} \ , \tag{21}$$

where $\hat{L}$ is the prediction for time t. These residuals contain nonlinear dependence structure of the data. So, e_t values are used to model nonlinear structure with the machine learning algorithms like LSTM, SVM such as

$$e_t = f(e_{t-1}, e_{t-2}, \ldots e_{t-n}) + \epsilon_t \ , \tag{22}$$

where f is a nonlinear function part evaluated by the LSTM or SVM and ϵ_t is defined as the random error. $\widehat{N}_t$ is predicted from (20), the forecast is obtained as

$$\hat{y}_t = \hat{L}_t + \widehat{N}_t \ , \tag{23}$$

In brief, there are two steps to achieve the proposed hybrid methodology. The first step, ARIMA or ARIMA-EGARCH models are used for constructing the relationship by processing historical data. The second step, the residuals that are obtained from ARIMA or ARIMA-EGARCH models are used for nonlinear pattern recognition by developing LSTM or SVM models. So, linear and nonlinear parts are modelled separately with different time series models, and then they can be combined to obtain more accurate forecast values.

Emprical Results

Datasets

As an empirical study, the daily closing values of the NASDAQ (USA), DAX (Germany), NSE (India), and BIST100 (Turkey) stock market indices between January 1, 2017 and October 18, 2021 are used. Some significant economic developments in the United States or Germany which have the largest economies in the world affect the emerging economies. The main purpose in the selection of these datasets are to compare time series forecasting models between world's leading financial markets and emerging markets. We investigate whether these models can differ from each other or not in terms of time series characteristics. Time series plots of daily closing values of the NASDAQ (USA), DAX (Germany), NSE (India), and BIST100 (Turkey) stock market indices are given in Figure 2.

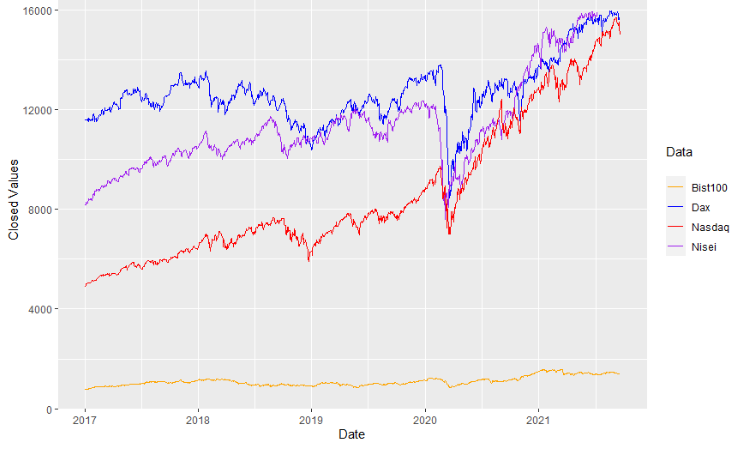

Figure 2. Time series plots of daily closing values of the NASDAQ (USA), DAX (Germany), NSE (India), and BIST100 (Turkey) stock market indices

In this application, it is aimed to select the most adequate model for the estimation of stock market index values. In the datasets, we work on log-return values of the data. We investigate the forecasting results of the comparative models, in terms of the performance accuracy measures. These models are ARIMA, EGARCH, LSTM, and SVM as single models and ARIMA-SVM, ARIMA-EGARCH-SVM, ARIMA-EGARCH-LSTM and ARIMA-LSTM as hybrid models. For the first step of constructing hybrid models, in ARIMA-EGARCH modelling, several models are obtained, but all the stock data are fitted with ARIMA(1,1,1) and EGARCH(1,1) for better results. In model selection part, we use the most commonly preferred and highly celebrated criterion, the Akaike's Information Criterion (AIC), and also we consider the significant parameter estimates. We investigate the significance of the parameter estimates of the identified model, then the lowest AIC in model selection. The GARCH (1, 1) model is particularly easy to handle and superior to other financial time-series models (Bollerslev, 1987).

As a second step, LSTM and SVM are applied. For the forecasting performances of these models, each data is split into two parts called test data and training data. The training data is used for the development of the model and the test part is used to evaluate the predicted model. We split the dataset 70 % train and 30 % test. The train dataset started on January 1, 2017, and the test dataset started on May 5, 2020. The test size and train size are 362 rows and 845 rows respectively for the four log-return data set. The models are trained on the training data set, and then the model performances are given based on the test dataset.

For LSTM, we obtain the results for Epoch=25, batch size=1 and Units=100 by using "Adam" for optimization. For SVM, we apply different types of kernels and obtain the best result for radial kernel function and regularization parameter C=1.

We compare model performances by some forecasting accuracy measures such as Mean Square Error (MSE), Root Mean Square Error (RMSE), Mean Absolute Error (MAE) and Mean Absolute Percentage Error (MAPE) which are given in Table 1. Hyndman (2006) investigated the forecast accuracy metrics and compared these metrics for univariate time series forecasts. Model selection is based on the criterion that the evaluation metrics with the lowest values give the best forecasting models. We choose the forecasting models according to these accuracy metrics for which the model is selected the most.

Table 1

Accuracy Metrics

MSE	$\frac{1}{N}\sum_{i=1}^{N}(x_i - \hat{x}_i)^2$
RMSE	$\sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \hat{x}_i)^2}$
MAE	$\frac{1}{N}\sum_{i=t}^{N}\frac{\mid x_i - \hat{x}_i \mid}{\mid x_i \mid}$
MAPE	$\frac{100}{N}\sum_{i=t}^{N}\frac{\mid x_i - \hat{x}_i \mid}{\mid x_i \mid}$

For the developed markets, we give the forecasting accuracy measures for NASDAQ and DAX log-return data in Table 2 and Table 3, respectively.

According to the results, for NASDAQ data set, ARIMA-GARCH-SVM model shows better performance compared to the other models. Except from MSE, all the accuracy measures (MAE, MAPE, RMSE) are the smallest for the selected model. For DAX data set, we select ARIMA-SVM model according to the values of MAE and RMSE. MSE is quite sensitive to the outliers since it heavily weights large errors more than small ones, which can be a disadvantage in some situations. (Hyndman, 2006) In order to bring MSE to the unit of the data, we take the square root which is the RMSE.

We can see the prediction performances in Figure 3 and Figure 4 for the selected models ARIMA-EGARCH-SVM and ARIMA-SVM. We can conclude that the model has a good fit to the return data.

Table 2

Forecasting Accuracy Measures for NASDAQ log-return data

Model	**MAE**	**MAPE**	**MSE**	**RMSE**
LSTM	0.01447	3.91177	0.00038	0.01963
ARIMA-EGARCH	0.00986	59.06250	**0.00017**	0.01329
ARIMA	0.01001	1.52985	0.00018	0.01344
SVM	0.01009	1.33651	0.00020	0.01435
ARIMA_SVM	0.003322	1.33289	2.19455	0.00468
ARIMA-EGARCH_SVM	**0.00244**	**0.59846**	1.34361	**0.00366**
ARIMA-EGARCH_LSTM	0.00993	1.98392	0.00018	0.01341
ARIMA_LSTM	0.01306	4.29118	0.00031	0.01772

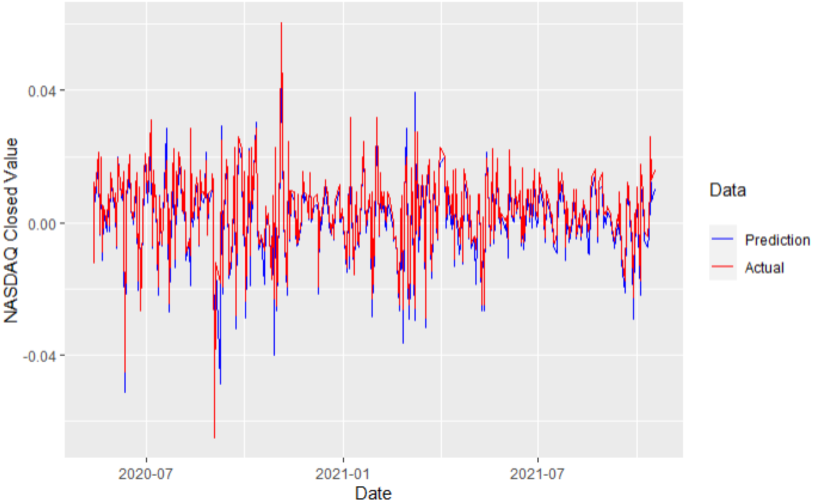

Figure 3. ARIMA-EGARCH-SVM prediction of NASDAQ log-return data.

Table 3

Forecasting accuracy measures for DAX log-return data

Model	**MAE**	**MAPE**	**MSE**	**RMSE**
LSTM	0.01305	8.34821	0.00032	0.01802
ARIMA-EGARCH	0.00827	76.18057	**0.00014**	0.01204
ARIMA	0.00832	**1**	**0.00014**	0.01209
SVM	0.00901	1.17050	0.00017	0.01308
ARIMA_SVM	**0.00158**	1.23456	7.67252	**0.00276**
ARIMA-EGARCH_SVM	0.00166	1.15763	7.83712	0.00279
ARIMA-EGARCH_LSTM	0.00830	2.82786	**0.00014**	0.01205
ARIMA_LSTM	0.01286	4.23613	0.00031	0.01765

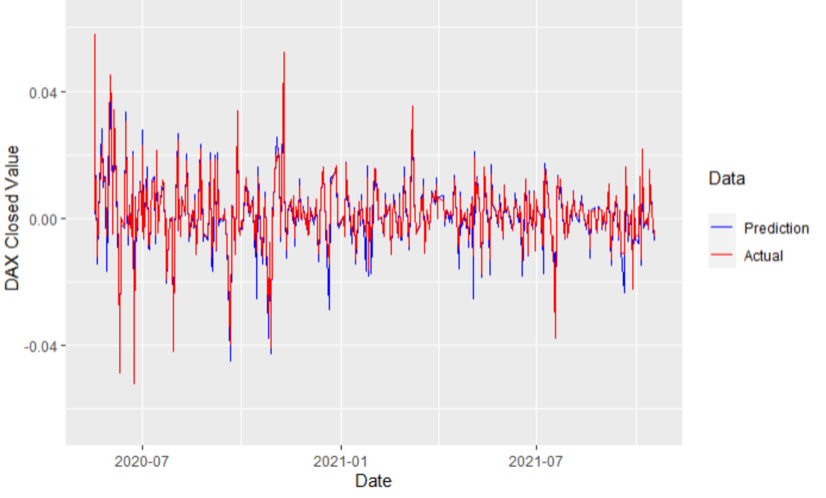

Figure 4. ARIMA-SVM prediction of DAX log-return data

Similarly, again for the emerging markets, we give the forecasting accuracy measures for NSE and BIST log-return data in Table 4 and Table 5, respectively. According to the values MAE, MAPE, and RMSE, for both data set, ARIMA-SVM model shows better performance compared with the other models. Also, we can see the prediction performances in Figure 5 and Figure 6 for the selected model ARIMA-SVM which has a quite good fit to the data.

Table 4

Forecasting Accuracy Measures for NSE log-return data

Model	**MAE**	**MAPE**	**MSE**	**RMSE**
LSTM	0.01036	4.45507	0.00020	0.01420
ARIMA-EGARCH	0.00765	24.83050	**0.00010**	0.01027
ARIMA	0.00776	1.23036	**0.00010**	0.01035
SVM	0.00713	1.59433	0.00011	0.01060
ARIMA_SVM	**0.00206**	**0.48953**	1.60821	**0.00401**
ARIMA-EGARCH_SVM	0.00222	0.52988	1.70980	0.00413
ARIMA-EGARCH_LSTM	0.00765	1.24970	**0.00010**	0.01025
ARIMA_LSTM	0.00988	15.10841	0.00018	0.01366

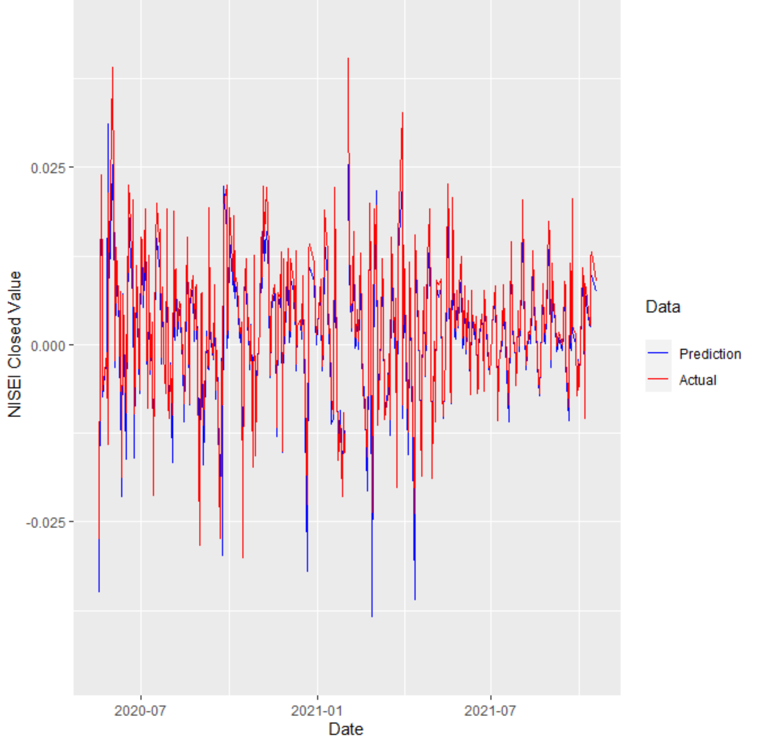

Figure 5. ARIMA-SVM prediction of NSE log-return data

Table 5

Forecasting Accuracy Measures for BIST100 log-return

Model	MAE	MAPE	MSE	RMSE
LSTM	0.01331	Inf	0.00034	0.01866
ARIMA-EGARCH	0.00930	99.03596	**0.00017**	0.01339
ARIMA	0.00938	NA	**0.00017**	0.01341
SVM	0.00930	Inf	0.00019	0.01404
ARIMA_SVM	**0.00183**	**0.82378**	9.14861	**0.00302**
ARIMA-EGARCH_SVM	0.00191	0.93336	9.65127	0.00310
ARIMA-EGARCH_LSTM	0.00952	1.51085	0.00018	0.01349
ARIMA_LSTM	0.01353	5.82403	0.00035	0.01893

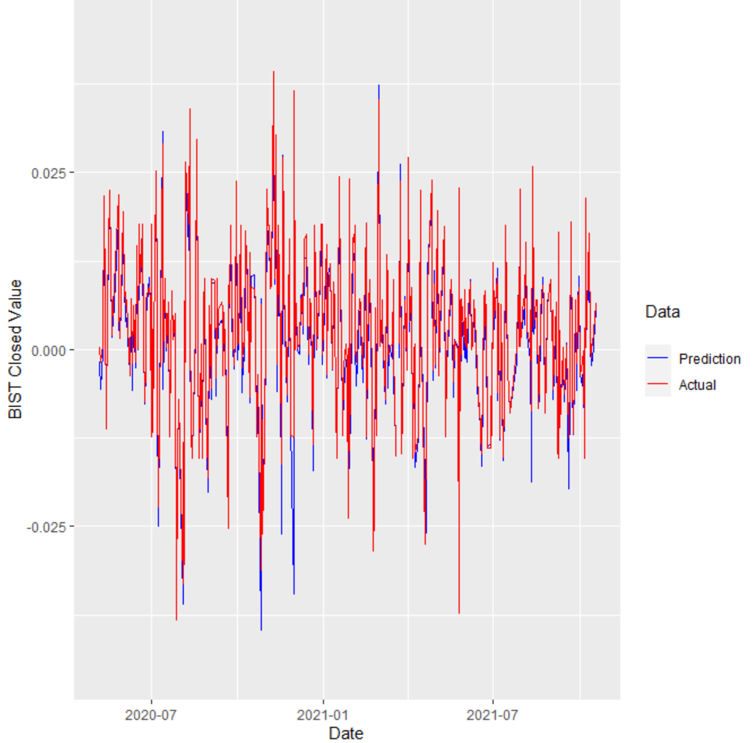

Figure 6. ARIMA-SVM prediction of BIST100 log-return data.

In Table 6, we give the one-day ahead point forecast results of the stock price data and also forecast intervals at 95% and 80% levels according to the selected models.

Table 6

One-day ahead Forecasts of the selected models

Method	Point Forecast	95% CI	80 %CI
ARIMA_EGARCH_NASDAQ	15545.35838	[15512.98, 15577.72]	[15524.21, 15566.49]
ARIMA_SVM_DAX	15388.38658	[15377.26, 15399.51]	[15381.13, 15395.65]
ARIMA_SVM_BIST	1427.08163	[1426.76, 1427.39]	[1426.87, 1427.28]
ARIMA_SVM_NISEI	18641.56773	[18579.6,18703.54]	[18601.1, 18682.04]

Conclusıon

The hybrid approaches in time series modelling mainly aim to achieve more accurate results. In this study, we propose some hybrid models combining linear and nonlinear approaches for financial time series data considering its volatility. Some intelligent methods like LSTM or SVM combined with ARIMA methodology, which predicts the future values of data by extracting linear relationships from historical data, are outstanding at learning high-level time dependent patterns in time series data. When financial time series data is considered, volatility forecasting should also be taken into account, so some GARCH models are also combined with these models. The most accurate models are selected according to the accuracy measures (MAE, MSE, RMSE, and MAPE).

Experimental results show that the hybrid models give more accurate results in time series modeling of a stock price index and have very good fit to the data since the forecasting errors are quite small. One of the developed markets NASDAQ has the same time series forecasting model ARIMA-EGARCH-SVM where the volatility is modelled with EGARCH, besides the emerging markets which are NSE and BIST and the other developed market DAX have the same forecasting model ARIMA-SVM. But generally, we can conclude that these two models are not too much different from each other according to the performance results of accuracy measures.

Peer-review: Externally peer-reviewed.
Conflict of Interest: The author has no conflict of interest to declare.
Grant Support: The author declared that this study has received no financial support.
Author Contributions: Conception/Design of study: B.H., C.B.; Data Acquisition: B.H., C.B.; Data Analysis/Interpretation: B.H., C.B.; Drafting Manuscript: B.H., C.B.; Critical Revision of Manuscript: B.H., C.B.; Final Approval and Accountability: B.H., C.B.
Acknowledgement: We thank the anonymous referees and the editor for their helpful suggestions which improved the presentation of the paper.

Hakem Değerlendirmesi: Dış bağımsız.
Çıkar Çatışması: Yazar çıkar çatışması bildirmemiştir.
Finansal Destek: Yazar bu çalışma için finansal destek almadığını beyan etmiştir.
Yazar Katkısı: Çalışma Konsepti/Tasarımı: B.H., C.B.; Veri Toplama: B.H., C.B.; Veri Analizi /Yorumlama: B.H., C.B.; Yazı Taslağı: B.H., C.B.; İçeriğin Eleştirel İncelemesi: B.H., C.B.; Son Onay ve Sorumluluk: B.H., C.B.
Teşekkür: Bildirinin sunumunu iyileştiren yararlı önerileri için anonim hakemlere ve editöre teşekkür ederiz.

References / Kaynakça

Bildirici, M. & Ersin, Ö. Ö. (2009). Improving forecasts of GARCH family models with the artificial neural networks: An application to the daily returns in Istanbul Stock Exchange, *Expert Systems with Applications, 36*(4), 7355-7362. doi:10.1016/j.eswa.2008.09.051.

Bollerslev, T. (1986). Generalized autoregressive conditional heteroskedasticity. *Journal of Econometrics, 31*(3), 307–327.

Box, G., & Jenkins, G.M., (1970). Time series analysis: forecasting and control. Holden-Day, San Francisco,CA.

Box, G.E., Jenkins, G.M., Reinsel, G.C., & Ljung, G.M. (2015), Time Series Analysis: Forecasting and Control, John Wiley and Sons.

Chen, K., Zhou, Y., Dai, F. (2015). A LSTM-based method for stock returns prediction: A case study of China stock market. *In Proceedings of the 2015 IEEE international conference on big data (Big Data) IEEE*, (pp. 2823–2824).

Chiang, W.C.,Urban, T.L., & Baildridge, G.(1996), A neural network approach to mutual fund net asset value forecasting, *Omega 24* (2), 205–215.

de Mattos Neto, P.S., Cavalcanti, G.D., & Madeiro, F.(2017), Nonlinear combination method of forecasters applied to PM time series, Pattern Recognit. Lett. 95, 65–72.

Domingos S.O., Oliveira de J.F.L., & Mattos Neto de P.S.G., (2019), An intelligent hybridization of ARIMA with machine learning models for time series forecasting, *Knowledge-Based Systems 175*, pp.72-86

Engle, R. F. (1982). Autoregressive conditional heteroscedasticity with estimates of he variance of United Kingdom inflation. *Econometrica: Journal of the Econometric Society*, 987–1007.

Fayyad, R. Uthurusamy (Eds.), Proceedings of the First International Conference on Knowledge Discovery and Data Mining, *AAAI Press*, Menlo Park, CA.

Hyndman, Rob. 2006. "Another Look at Forecast Accuracy Metrics for Intermittent Demand." *Foresight: The International Journal of Applied Forecasting*, 4, 43–46.

Khashei, M., & Bijari, M.(2010), An artificial neural network (p, d, q) model for time series forecasting, *Expert Syst. Appl. 37*(1), 479–489.

Khashei, M., Bijari, M.(2011), A novel hybridization of artificial neural networks and ARIMA models for time series forecasting, *Appl. Soft Comput. 11*(2), 2664–2675.

Kim K-J., (2003) Financial time series forecasting using support vector machines, *Neurocomputing 55*, pp.307 – 319.

Kim, H.Y., & Won, C.H. (2018), Forecasting the volatility of stock price index: A hybrid model integrating LSTM with multiple GARCH-type models, *Expert Systems With Applications 103*, 25–37.

Maknickiené N., & Maknickas, A. (2012, May). Application of neural network for orecasting of exchange rates and forex trading, *In Proceedings of the 7th international scientific conference on business and management* pp. 10–11.

Markham, L.S., & Rakes T.R. (1998), The effect of sample size and variability of data on the comparative performance of artificial neural networks and regression, *Comput. Oper. Res. 25* 251–263.

Muller, K.R., Smola, J.A., & Scholkopf, B.(1997), Prediction time series with support vector machines, *Proceedings of International Conference on Artificial Neural Networks, Lausanne*, Switzerland, pp. 999–1004.

Nelson, D. B. (1991). Conditional heteroskedasticity in asset returns: A new approach. *Econometrica: Journal of the Econometric Society*, pp. 347–370.

Pérez-cruz, F., Afonso-rodríguez, J. A. & Giner, J. (2003), Estimating GARCH models using support vector machines, *Quantitative Finance, 3(3),* pp.163-172. doi:10.1088/1469-7688/3/3/302

Panigrahi, S., & Behera, H.(2017), A hybrid ETS–ANN model for time series forecasting, *Eng. Appl. Artif. Intell*. 66, 49–59.

Scholkopf, B., Burges, C., &Vapnik, V.(1995), Extracting support data for a given task, in: U.M.

Sima, S.N., Neda, T., &Akbar, S.N. (2018), A Comparison of ARIMA and LSTM in Forecasting Time Series,*2018 17th IEEE International Conference on Machine Learning and Applications* (ICMLA)

Vapnik, V.N.(1995), The Nature of Statistical Learning Theory, Springer, New York .

Yaser, S.A.M., & Atiya, A.F. (1996), Introduction to financial forecasting, *Appl. Intell. 6*, 205–213.

Zhang G.P., Patuwo E.B., & Hu M.Y., (1998) Forecasting with artificial neural networks: the state of the art, *Int. J. Forecasting 14,* pp.35–62.

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