Data analytics enhanced component volatility model

Author:

Yuan Yao
Institute of Management Science and Engineering, Business School,
Henan University, 475004, Jinming District, Kaifeng, Henan Province, China
prof.yuanyao@gmail.com

Jia Zhai
Salford Business School,
University of Salford,
43 Crescent, Salford M5 4WT, UK
j.zhai@salford.ac.uk
jia.zhai1982@gmail.com

Yi Cao
Department of Business Transformation and Sustainable Enterprise,
Surrey Business School, University of Surrey,
Guildford, Surrey, GU2 7XH,
United Kingdom
Jason.caoyi@gmail.com

Xuemei Ding
School of Computing and Intelligent Systems,
Ulster University, Magee campus,
Northland Rd, Londonderry
Northern Ireland, UK, BT48 7JL
x.ding@ulster.ac.uk

Faculty of Software, Fujian Normal University,
Upper 3rd Rd, Cangshan, Fuzhou,
Fujian Province, 350108, China
xuemeid@fjnu.edu.cn

Junxiu Liu,
School of Computing and Intelligent Systems,
Ulster University, Magee campus,
Northland Rd, Londonderry
Northern Ireland, UK, BT48 7JL
j.liu1@ulster.ac.uk

Yuling Luo
Guangxi Key Lab of Multi-Source Information Mining & Security,
Faculty of Electronic Engineering, Guangxi Normal University,
Diecai, Guilin, Guangxi, China, 541000
yuling0616@mailbox.gxnu.edu.cn

Corresponding author:
Jia Zhai
Salford Business School,
University of Salford,
43 Crescent, Salford M5 4WT, UK
+44(0) 161 295 8147
j.zhai@salford.ac.uk
jia.zhai1982@gmail.com

Data analytics enhanced component volatility model

Abstract

Volatility modelling and forecasting have attracted many attentions in both finance and computation areas. Recent advances in machine learning allow us to construct complex models on volatility forecasting. However, the machine learning algorithms have been used merely as additional tools to the existing econometrics models. The hybrid models that specifically capture the characteristics of the volatility data have not been developed yet. We propose a new hybrid model, which is constructed by a low-pass filter, the autoregressive neural network and an autoregressive model. The volatility data is decomposed by the low-pass filter into long and short term components, which are then modelled by the autoregressive neural network and an autoregressive model respectively. The total forecasting result is aggregated by the outputs of two models. The experimental evaluations using one-hour and one-day realized volatility across four major foreign exchanges showed that the proposed model significantly outperforms the component GARCH, EGARCH and neural network only models in all forecasting horizons.

1. Introduction

Volatility is considered the "barometer for the vulnerability of financial markets and the economy" (Jiang, Ahmed, & Liu, 2016) (Poon & Granger, 2003) and is crucial for asset pricing, derivative valuation and risk management. Volatility modelling and forecasting is a much devoted area of research and have attracted many attentions of researches. A number of econometrics and machine learning models have been developed in past years. Conventional models include generalized autoregressive conditional heteroskedastic (GARCH) model, developed by (Tim, 1986) and (Bollerslev, 1990), was accepted as one of the most popular volatility models for years. At the same time, autoregressive fractionally integrated moving average (ARFIMA) model has been proposed to capture the long memory property in realized volatility (Granger C. W., 1980) (Granger & Joyeux, 1980). In 1999, (Lee & Engle, 1999) introduced a component GARCH model, which decomposes volatility into two components: a permanent long run trend component and a transitory short-run one that is mean-reverted to the long-run trend (Harris, Stoja, & Yilmaz, 2015). After that, more empirical evidences suggested that two-component model can capture the volatility structure much better than the one-component models. For example, (Brandt & Jones, 2006) showed that daily volatility can be well characterized by two-component model structure with one highly persistent component and one strongly stationary component. A number of literatures have found that two-component volatility models perform better than the one-component models in explaining stock as well as exchange rate volatilities. Started from component GARCH model, introduced by (Lee & Engle, 1999), those two-component models usually consider the volatilities as a composition of a permanent long-run trend component and a transitory short-run component, both of which follow a mean-reverting process with a slow reversion speed in long-run trend and quick reversion speed in the short-run one.

Additional to the traditional econometrics models, machine-learning algorithms have been widely used in financial modelling (Jia, Yi, Yuan, Xuemei, & Yuhua, 2017) (Zhai, Cao, Yao, Ding, & Li, 2017) (Yi, Yuhua, Sonya, Ammar, & Martin, 2015) in recent years. (Boyacioglu & Avci, 2010) proposed an adaptive neural fuzzy inference system (ANFIS) for predicting earning per shares on Istanbul stock market and concluded that their method performed well on monthly forecasting. This ANFIS model was also effectively applied in predicting closing price of Zagreb Stock Market index (Svalina, Galzina, Lujić, & Šimunović, 2013). Two hybrid models were proposed in (Hajizadeh, Seifi, Zarandi, & Turksen, 2012), where explanatory input variables were selected based on GARCH, EGARCH and GJR-GARCH models and were fed into neural network model for volatility forecasting. The experimental evaluations showed that the hybrid models outperforming traditional models effectively. A hybrid model of self-organized fuzzy neural network and ARIMA model has been proposed in (McDonald S. , Coleman, McGinnity, & Li, 2013) and applied on financial markets. This hybrid model was thoroughly compared with other traditional forecasting models in (McDonald S. , Coleman, McGinnity, Li, & Belatreche, 2014). The results showed that hybrid model achieved better forecasting results in average. (Kristjanpoller, Fadic, & Minutolo, 2014) developed a hybrid model composed of neural network and GARCH model, where the volatility is modelled and forecasted by GARCH model at the first step and the output of the GARCH as well as the original volatility data were then fed into a neural network model for forecasting the volatility. It showed that the hybrid model significantly outperformed the traditional ARFIMA and GARCH model. The hybrid model was extended in (Kristjanpoller & Minutolo, 2016) for spot and future oil price and showed 30% increase on precision over previous models. Among the structures of most hybrid models, artificial neural network is widely used for modelling the non-linear part of the underlying variable. Artificial neural network is also widely used in time series forecasting in recent studies (Rubio, Elias, Cruz, & Pacheco, 2016) (Aljarah, Faris, Mirjalili, & Al-Madi, 2016) (Rubio, 2016) (Restrepo, Manotas, & Lozano, 2016) (Jesús, 2016) (Liu, et al., 2016). Those studies show that careful feature selection and certain neural network structure with tailor-made algorithms can reach a stable, convergent, and better accuracy when compared with the traditional neural network for the prediction of time-series in different areas, i.e., mechatronic processes, brain signal, and some other biomedical data. (Kristjanpoller, Fadic, & Minutolo, 2014) and (Kristjanpoller & Minutolo, 2016) further show that particularly designed hybrid model composed of the artificial neural network and traditional GARCH models achieves significantly better performance than the traditional models, i.e., GARCH, EGARCH, and ARFIMA in forecasting the volatility of foreign exchange and commodity future. The hybrid models are trained by carefully selected features that are related to the volatility. In addition, the work in (Sharda & Patil, Oct 1992) (Gorr, June 1994) (Zhang & Qi, 16 January 2005) show that autoregressive neural network (ARNN) outperforms other structure of artificial neural network in trend and seasonality forecasting of a non-linear time series, where the trend is defined as removing the short-term components from the raw time series and the seasonality is defined as removing the long-term components. (Zhang & Qi, 16 January 2005) and (Nelson, Hill, Remus, & O'Connor, 1999) show that prior data processing, removing either the trend or the seasonality, can dramatically reduce forecasting errors and is critical to build an adequate ARNN based forecasting model.

Inspired by previous work of two-component models in econometrics area and hybrid models in machine learning area, we propose an alternative, very simple-structured model to capturing and forecasting volatility across short and long forecasting horizons. Our measure of volatility is based on the realized volatility. We decompose the volatility to long-term and short-term components using a low-pass Hodrick-Prescott filter, and then model the long-term component using an autoregressive neural network and the short-term component as a stationary autoregressive process around the long-term component. The structure of a low-pass filter, a first order autoregressive process and an autoregressive neural network is simple but effective to capture the dynamics of two components of the realized volatility. Since neural network is one of the most popular data analytics algorithms, we name it as “data analytics enhanced component volatility model”. We evaluate the model’s out-of-sample forecasting performance based on the one-hour and one-day realized volatility constructed using EUR/USD, GBP/EUR, GBP/JPY and GBP/USD high frequency exchange rates over the period 27 September 2009 to 12 August 2015, which includes around 7.7 billion observations across 2145 days. As a benchmark, we compare the forecasting accuracy of the proposed model with those of the two-component GARCH model (Lee & Engle, 1999) and the traditional EGARCH model as well as the neural network only model. We evaluate the performances up to 500 time points, which are 50 days for one-hour volatility and roughly two years for one-day volatility. Consistent with the findings of (Lee & Engle, 1999) and (Brandt & Jones, 2006), our experimental evaluations also show that in almost all the cases, our proposed model provides a significant improvement in forecasting performance over Component GARCH and EGARCH as well as the neural network only model. The improvement is stable across short and long horizons. In particular, the forecasting accuracy of our proposed model is roughly consistent across all horizons while the performance of the Component GARCH and EGARCH model deteriorate significantly in long horizon forecasting. Overall, the experimental evaluations show that our proposed model achieves a stable and much better forecasting performance than most of the traditional volatility models.

The outline of the remainder of this paper is as follows. In Section 2, we present the structure and details of the proposed model. Section 3 describes the data used in the empirical analysis and the forecasting evaluation criteria. Section 4 presents the empirical results and Section 5 provides a summary, concluding remarks and some future research directions.

2. Methodology

2.1 Model structure

In this paper, we follow this idea and the model format in (Lee & Engle, 1999), we assume the realized volatility follows a two-component process given by

$$\mathrm{\sigma}_{\mathrm{t}} = L_t + S_t \tag{1}$$

$$S_t = \alpha S_{t-1} + \mathrm{\varepsilon}_{\mathrm{t}} \tag{2}$$

where L_t and S_t are the long and short term component of the realized volatility respectively and $\mathrm{\varepsilon}_{\mathrm{t}}$ is the random error term with zero mean and constant variance. The short term component S_t is an autoregressive, AR(1), process with the parameter α, which measures the speed of the short term revision to long term trend. This structure of the volatility is following the two-component characteristics in previous literatures, for example (Lee & Engle, 1999), (Alizadeh, Brandt, & Diebold, 2002), and (Brandt & Jones, 2006). However, the implementation of our two-component model is different from traditional Component GARCH model.

We implement the two-component model given in equation (1) and (2) in three steps. Firstly, we decompose and extract the long-term component L_t from the realized volatility using low-pass Hodrick-Prescott filter

(Hodrick & Prescott, 1997). After getting L_t, the short term component of the realized volatility can be obtained by $S_t = \mathrm{\sigma_t} - L_t$. In the second step, we train an artificial neural network (ANN) using the long-term component L_t to obtain a forecasting model. A future value of L_{t+n} at time $t + n$ can be forecasted by the trained ANN. In the last step, the AR(1) process model $S_t = \alpha S_{t-1} + \mathrm{\varepsilon_t}$ is estimated using the short term component S_t, which is obtained from the first step. A future value of S_{t+n} at time $t + n$ can be then forecasted by the estimated AR(1) model. Therefore, we can calculate the future value of the realized volatility at time $t + n$ by $\sigma_{t+n} = L_{t+n} + S_{t+n}$.

2.2 Volatility decomposition

In the first step, the long-term component L_t is extracted from the realized volatility. To do this, the low-pass Hodrick-Prescott filter (Hodrick & Prescott, 1997) is applied to extract a low frequency non-linear component from a time-series. That low frequency component represents the trend of the long-term component of the realized volatility. Hodrick-Prescott filter (Hodrick-Prescott filter) is widely used in applied macroeconomics (Stock & Watson, 1999) (McElroy, 2008) (Stock & Watson, 2016) for removing the short-term cyclical component of a time series from raw data. Given the value of the smoothing parameter λ, the long term trend component shall solve

$$\min_{\tau}\left(\sum_{t=1}^{T}(y_t - \tau_t)^2 + \lambda\sum_{t=2}^{T-1}[(\tau_{t+1} - \tau_t) - (\tau_t - \tau_{t-1})]^2\right)$$

where the smoothing parameter λ penalizes the variations in the growth rate of the trend component. The larger the value λ, the higher is the penalty. We follow the studies in (Baxter & King, 1999) (Ravn & Uhlig, 2002) (Harris, Stoja, & Yilmaz, 2015) to set the λ as the widely used empirical value of 100 multiplied by the squared frequency of the data, which for one hour realized volatility (assuming 360 trading days per year and 8 hours per day) is $\lambda = 100 * (360 * 8)^2 = 829440000$.

2.3 Autoregressive Neural Network

2.3.1 Network structure

In the second step of implementing our two-component model, we use an autoregressive neural network (ARNN), a special format of artificial neural network, to model and forecasting the long-term component L_t of realized volatility. As the discussion in Section 1, the work in (Sharda & Patil, Oct 1992) (Gorr, June 1994) (Zhang & Qi, 16 January 2005) show that ARNN outperforms other structure of artificial neural network in trend and seasonality forecasting of a non-linear time series. (Zhang & Qi, 16 January 2005) and (Nelson, Hill, Remus, & O'Connor, 1999) further show that prior data processing, removing either the trend or the seasonality, can dramatically reduce forecasting errors and is critical to build an adequate ARNN based forecasting model. Therefore, in our paper, we follow the existing study and employ the ARNN to model and forecasting the trend, the long-term component L_t of realized volatility. The reason of selecting ARNN is twofold. First, main strand finance literature usually assumes the volatility is composed of two autoregressive process, long term and short term. We follow the traditional assumption with augmented adjustment: ARNN, which models the lags as a nonlinear function. Secondly, compared with other artificial neural network, ARNN can achieve even better accuracy in "deseasonalized" financial time-series forecasting by a relatively simple structure, where appropriate lags instead of a number of additional features is crucial for the forecasting performance. ARNN is also proved to have advantages over recurrent feed-forward neural network and is less sensitive to the problem of long-term dependence (Mustafaraj, Lowry, & Chen, 2011). Compared with traditional feed-forward artificial neural network, the ARNN has interconnection from the lagged input to the output layer, which enhances its capability of forecasting more than one-step as a time-series predictor. We follow the traditional ARNN structure:

$$Y_t = \alpha_0 + \sum_{i=1}^{n} \alpha_i Y_{t-i} + \sum_{j=1}^{h} \Psi\left(\gamma_{0j} + \sum_{i=1}^{n} \gamma_{ij} Y_{t-i}\right) \beta_j + \varepsilon_t$$

where n is the number of lags, h is the number of hidden neurons, $\Psi(.)$ is the activation function, γ is the weights between input and hidden neurons, β is the weights between hidden and output neurons. The non-linear part contains h hidden neurons transforms the input variables, defined as the n-lagged long-term component L_t of realized volatility, weighted by parameters γ_{ij} plus a bias γ_{0j}, via a non-linear activation function $\Psi(.)$. If representing the number of lags and hidden neurons as i and j, a hidden neuron can be denoted by

$$\Psi\left(\gamma_{0j} + \sum_{i=1}^{n} \gamma_{ij} Y_{t-i}\right)$$

each of which is weighted by a parameter β_j before it produces the output layer. Since we want to model and forecasting the long-term component of volatility, ARNN operates as a non-linear regression function at the end as

$$Y_t = G(Y_{t-1}, Y_{t-2}, \dots, Y_{t-n}) + \varepsilon_{t,ARXNN}$$

which maps the unknown relation, $G(\cdot)$, between the input variable and the target function Y_t and the error term $\varepsilon_{t,ARXNN}$. For the purpose of this study, the hidden layer uses the hyperbolic tangent sigmoid transfer function, while the output layer uses a linear transfer function. The ARNN structure with only one hidden layer is considered since it operates as a non-linear regression function and can be trained to approximate most non-linear function arbitrarily well (Siegelmann, Horne, & Giles, 1997), (Andreou, Charalambous, & Martzoukos, 2008) (Mustafaraj, Lowry, & Chen, 2011). Following this structure, in this paper, to forecasting long-term component L_t of the realized volatility, an autoregressive neural network (ARNN) with three layers structure is used: an input layer that includes lagged L_t inputs to the network; a hidden layer with hyperbolic tangential activation function, and an output layers with a linear activation function. In previous studies, the estimation of the model parameters usually follows the back-propagation algorithm. It is shown in (Charalambous, June 1992) that the back-propagation algorithm is often unable to converge rapidly to the optimal solution. Therefore, we utilize the modified Levenberg-Marquardt (LM) algorithm, which is much more sophisticated and efficient in terms of time capacity and accuracy (Hagan & Menhaj, 1994). When training the ARNN, we divide the training dataset into three subsets, 80% of the data for training the ARNN, 10% for validation, and the last 10% for testing. During the training process, the errors on the training dataset and validation dataset are monitored at the same time. When the validation error rises while the training error maintains, the ARNN begins to overfit the data. The weights and bias at the minimum of the validation error are saved as the trained ARNN

2.3.2 Network parameters

Under this structure of ARNN, the number of lags n and the number of hidden neurons h are the crucial parameters for constructing the ARNN model. We follow the widely used configuration for the ARNN in (Siegelmann, Horne, & Giles, 1997) and (Mustafaraj, Lowry, & Chen, 2011), and investigate the performance using the number of lags from 2 to 5, which is $n = 2,3,4,5$. We firstly use $h = 10$ as the preliminary configuration for the number of hidden neurons and finds the appropriate lags. We use the long-term component L_t of one hour realized EURUSD rate from 27 Sep 2009 to 06 Dec 2012 to train the ARNN model under $n = 2,3,4,5$ and use 500 long-term component values of EURUSD rate from 07 Dec 2012 to 15 Jan 2013 to test the trained model. The forecasting error is defined as $\hat{L}_t - L_t$, the difference between the original value L_t and the forecasted one. The average errors across the forecasting horizons is listed in Table 1, where we can observe that the best average error is at $n = 4$, although the errors do not show a significant difference among different configuration of lag numbers. Four lags also conform to the configuration suggested in (Siegelmann, Horne, & Giles, 1997) and (Mustafaraj, Lowry, & Chen, 2011). Therefore, in this paper, we use four lags autoregressive neural network (ARNN) following the suggested configuration and our empirical investigation.

Table 1 This table contains the average forecasted error across the horizon from 07 Dec 2012 to 15 Jan 2013 under configurations of lag number from 2, 3, 4 and 5

	$n = 2$	$n = 3$	$n = 4$	$n = 5$
Average error	1.22807E-05	1.22647E-05	1.21385E-05	1.2293E-05

The following **Error! Reference source not found.**Figure 1 shows an example of using ARNN model to forecasting the long-term component L_t of one hour realized EURUSD rate from 07 Dec 2012 to 15 Jan 2013. The ARNN model was constructed using one hidden layer with 10 neurons and was trained using the

L_t of EURUSD rate from 27 Sep 2009 to 06 Dec 2012. The forecasting results $\hat{L}_t$ include 500 long-term component values of EURUSD rate from 07 Dec 2012 to 15 Jan 2013. The forecasting error, defined as $\hat{L}_t - L_t$, between the original value L_t and the forecasted one $\hat{L}_t$ are shown in the bottom sub-figure, from which we can observe that out-of-sample forecasting errors are around 10^{-5}.

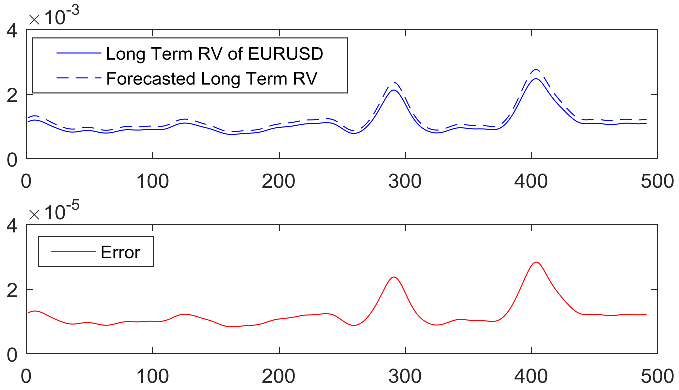


Figure 1 This figure shows an example of out-of-sample forecasting long term component $\boldsymbol{L_t}$ of EURUSD using ARNN. The top figure shows the long term component of EURUSD from 07 Dec 2012 to 05 Feb 2013 decomposed by Hodrick-Prescott filter. The middle figure shows the forecasted value $\boldsymbol{\hat{L}_t}$ of the long term component of EURUSD from 07 Dec 2012 to 15 Jan 2013 using ARNN model. The bottom figure shows the differences $\boldsymbol{\hat{L}_t - L_t}$ between the original long term component of EURUSD and the forecasted one.

2.4 Autoregressive model

In the third step, we estimate an autoregressive model for the short-term component S_t: $S_t = \alpha S_{t-1} + \varepsilon_t$. The n-step ahead forecasting of the short-term component S_t is therefore given by

$$\hat{S}_{t+n} = \alpha^n S_t + \sum_i^{n-1} \alpha^i \, \varepsilon_{t-i} \tag{3}$$

We can obtain the forecasted the long-term component $\hat{L}_{t+n}$ of the realized volatility through ARNN model. Therefore, the n-step ahead forecasting of the one hour realized volatility is given by Equation (1) as $\hat{\sigma}_{t+n} = \hat{L}_{t+n} + \hat{S}_{t+n}$. We name the proposed model as autoregressive neural network enhanced two-component model (shorted as: NNE2C) due to the implementation of the model structure. To estimate the autoregressive model for the short-term component S_t, we utilize the method of moment through Yule–Walker equations, named for Udny Yule and Gilbert Walker (Yule, 1927) (Walker, 1931):

$$\gamma_m = \sum_{k=1}^{p} \alpha_k \gamma_{m-k} + \sigma_\epsilon^2 \delta_{m,0}$$

where γ_m is the autocovariance function of S_t, σ_ϵ^2 is the variance of the input noise process and the $\delta_{m,0}$ is the Kronecker delta function. For the case $p = 1$, one lag autoregressive process AR(1), the α_1 can be obtained by γ_1/γ_0.

2.5 Model flow

We summarize the NNE2C model workflow when applied in realized volatility forecasting. Different from other hybrid models, which usually use the neural network as the dominate one to model the non-linear part of the underline financial variable, i.e. volatility or foreign exchange and use the ARIMA (McDonald S. , Coleman, McGinnity, Li, & Belatreche, 2014) or GARCH model (Kristjanpoller, Fadic, & Minutolo, 2014) as the pre-processing part for modelling the linear part of the underline financial variable, the NNE2C model follows the traditional financial theory to consider the volatility as a combination of the long and short term while producing an enhanced mechanism through explicitly decomposing the two components and modelling them separately. The simple NNE2C model follows and enhances the financial theory by capturing the volatility structure explicitly and effectively.

Algorithm 1 Workflow of proposed NNE2C model

1	Given the realized volatility σ_t of selected financial security, i.e. FX, we assume it is composed of long and short term components: $\sigma_t = L_t + S_t$
2	We use Hodrick-Prescott filter as a low-pass filter to explicitly extract the long term component L_t. The smoothing parameter λ of Hodrick-Prescott filter is selected according to the empirical study: $\lambda = 100 * (360 * 8)^2 =$ 829440000. $\min_{\tau}\left(\sum_{t=1}^{T}(\sigma_t - \tau_t)^2 + \lambda \sum_{t=2}^{T-1}[(\tau_{t+1} - \tau_t) - (\tau_t - \tau_{t-1})]^2\right)$
3	The short term component S_t is then obtained by $S_t = \sigma_t - L_t$
4	Modelling L_t and S_t simultaneously:

	Using L_t to train the ARNN with 4 lags and 10 hidden neurons, n step ahead L_{t+n} is forecasted by the trained ARNN model; $L_t = \alpha_0 + \sum_{i=1}^{n} \alpha_i L_{t-i} + \sum_{j=1}^{h} \Psi\left(\gamma_{0j} + \sum_{i=1}^{n} \gamma_{ij} L_{t-i}\right)\beta_j + \varepsilon_t$ Using S_t to estimate the AR(1) model, n step ahead S_{t+n} is forecasted by the estimated AR(1) model; $S_t = \alpha S_{t-1} + \mathrm{\varepsilon_t}$
5	n step ahead realized volatility is obtained by $\mathrm{\sigma_{t+n}} = L_{t+n} + S_{t+n}$

3. Data and forecasting evaluation

3.1 Data

We use the neural network enhanced two-component model defined in Section 2 to forecasting the volatility of the EUR/USD, GBP/EUR, GBP/JPY and GBP/USD exchange rates. High frequency exchange rate data (tick data) were obtained from Oricode Inc for the period 27 September 2009 to 12 August 2015 and included around 7.7 billion observations across 2145 days. The unobserved true volatility, in principle, can be estimated arbitrarily accurately using a measure of realized volatility calculated through the intraday returns (Harris, Stoja, & Yilmaz, 2015). It is proved by (Torben, Tim, & Francis, 2009) that the sum of squared intraday returns converges to the unobserved true volatility with the intraday interval approaching to zero. In this study, we use realized volatility as the proxy of the unobserved true volatility. This is obtained by aggregating the intraday squared returns using the approach given by (Andersen & Bollerslev, 1998):

$$\hat{\sigma}_{rv,t}^2 = \sum_{n=1}^{N} r_{t,n}^2$$

where $\hat{\sigma}_{rv,t}^2$ is the realized volatility for time t, and $r_{t,n}^2$ is the squared log return on time t for interval n ($n = 1,2,\dots,N$). In this paper, we use one-hour and one-day realized volatilities, both of which are constructed from 10 millisecond log return, which is the highest frequency in our data.

The one-hour realized volatility is obtained by aggregating 360,000 10-millisecond log returns $r_{t,n}^2$ in each hour. The data from 27 September 2009 to 07 December 2012 (20000 observations) is used for the initial estimation of the model, while the data from 08 December 2012 to 12 August 2015 (16681) is used for out-of-sample evaluation. The one-day realized volatilities is constructed by the sum of 3,600,000 log returns in each trading day. For the one-day realized volatilities, the data from 27 September 2009 to 07 December 2012 (1000 observations) is used for the initial estimation of the model, while the remained data (837 observations) is used for out-of-sample evaluation.

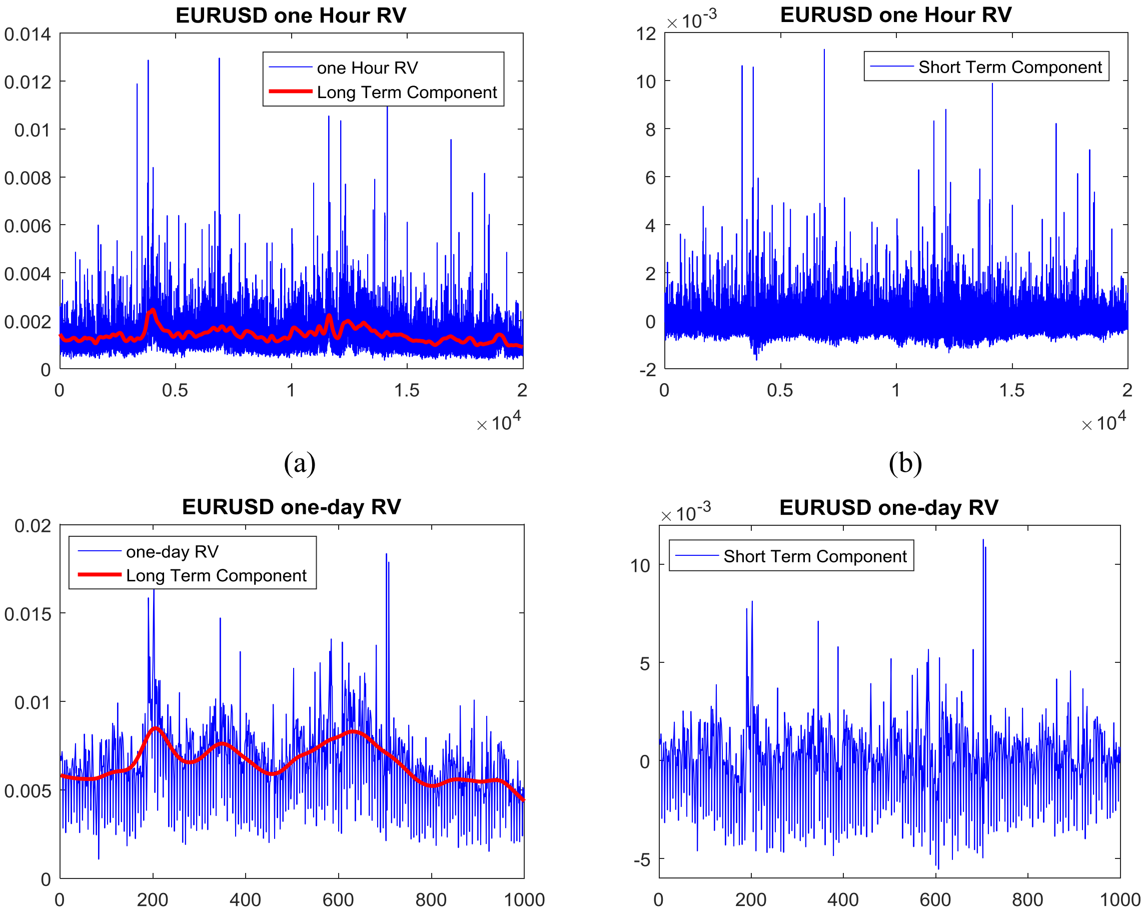
Table 2 reports summary statistics for the one-hour and one-day realized volatilities of full observations of four exchange rates. Panel A reports the mean, standard deviation, skewness and excess kurtosis and Bera–Jarque statistic and Panel B reports the first six autocorrelation coefficients and the Ljung–Box Q statistic for autocorrelation of six lags for the realized volatilities. P-values are also reported in parentheses. In the Ljung–Box Q tests, the null hypothesis that the residuals of the returns are not autocorrelated is rejected in both one-hour and one-day realized volatilities. Therefore the two realized volatilities are all highly autocorrelated.

Table 2 Summary statistics and autocorrelations

	mean	Standard deviation	Skewness	Excess kurtosis	Bera-Jarque
Panel A: Summary statistics					
GBP/USD 1 Hr	1.6657E-06	9.1010E-06	1.0231E+02	1.1661E+04	2.0743E+11
GBP/JPY 1 Hr	3.6200E-06	7.6346E-06	4.5817E+01	3.4566E+03	1.8227E+10
GBP/EUR 1 Hr	1.8514E-06	3.1560E-06	2.5651E+01	1.1853E+03	2.1401E+09
EUR/USD 1 Hr	2.1207E-06	5.6940E-06	4.4465E+01	3.8006E+03	2.2055E+10
GBP/USD Daily	3.2698E-05	7.2746E-05	3.5359E+01	1.4143E+03	1.5292E+08
GBP/JPY Daily	7.0981E-05	6.8848E-05	7.5062E+00	1.0538E+02	8.2050E+05
GBP/EUR Daily	3.6060E-05	2.4165E-05	2.4964E+00	1.6728E+01	1.6342E+04
EUR/USD Daily	4.1188E-05	5.5114E-05	2.1497E+01	7.0608E+02	3.8019E+07

Panel B: Autocorrelations							
	1	2	3	4	5	6	Ljung–Box Q
GBP/USD 1 Hr	0.6638	0.2696	0.0387	0.0210	0.0216	0.0225	1.8952E+04 (0.0000)
GBP/JPY 1 Hr	0.2870	0.1879	0.1897	0.1569	0.1559	0.1502	1.4856E+04 (0.0000)
GBP/EUR 1 Hr	0.2381	0.1711	0.1363	0.1149	0.0903	0.0784	5.8392E+03 (0.0000)
EUR/USD 1 Hr	0.3683	0.2986	0.2442	0.2361	0.1952	0.1588	1.7166E+04 (0.0000)
GBP/USD Daily	0.0569	0.0255	0.0214	0.0218	0.0753	0.1139	6.8927E+01 (2.7236E-07)
GBP/JPY Daily	0.5392	0.2908	0.2817	0.2527	0.2763	0.3728	2.8913E+03 (0.0000)
GBP/EUR Daily	0.4351	0.1860	0.1641	0.1646	0.3203	0.5706	3.3945E+03 (0.0000)
EUR/USD Daily	0.1811	0.1007	0.0914	0.0894	0.1412	0.2041	6.3254E+02 (0.0000)

In the example in the following Figure 2(a-b), Hodrick-Prescott filter is applied to one hour realized volatility with smoothing parameter λ of 829440000 (blue curve in Figure 2(a)) of EURUSD exchange rate from 27 Sep 2009 to 07 Dec 2012 to extract the long term component L_t (red curve in Figure 2(b)). Figure 2(c-d) shows an example of one-day realized volatility with long term component extracted by Hodrick-Prescott filter with smoothing parameter λ of 12960000 (assuming 360 trading days per year, the λ is calculated by 100 multiplied by the squared frequency of the data).



(a)

(b)

(c)

(d)

Figure 2 (a) Blue curve in the figure: One-hour realized volatility of EURUSD exchange rate from 27 Sep 2009 to 07 Dec 2012; Red curve: the long term component $\boldsymbol{L_t}$of the realized volatility extracted by low-pass Hodrick-Prescott filter; (b) Short term component of one-hour realized volatility of EURUSD exchange rate; (c) Blue curve in the figure: one-day realized volatility of EURUSD exchange rate from 27 Sep 2009 to 07 Dec 2012; Red curve: the long term component $\boldsymbol{L_t}$of the realized volatility extracted by low-pass Hodrick-Prescott filter; (d) Short term component of one-day realized volatility of EURUSD exchange rate.

3.2 Forecasting evaluation

The proposed neural network enhanced two-component model is used to calculate out-of-sample forecastings of the realized volatilities of up to 500 hours or days ahead across the evaluation period for the one-hour or one-day realized volatility respectively. As the benchmark, we select one-factor EGARCH and two-factor Component GARCH of (Lee & Engle, 1999). In addition, we also employ four lags autoregressive neural network applied directly on realized volatility as one of the benchmark models. We therefore estimate four models: a) four lags autoregressive neural network enhanced two-component model (NNE2C); b) one-component EGARCH model; c) two-component GARCH model; and d) four lags autoregressive neural network model (NNOnly). The four models are evaluated using one-hour and one-day realized volatilities respectively. For one-hour and one-day data, four models are initially estimated using the first 20,000 and 1,000 observations from 27 Sep 2009 to 07 Dec 2012 respectively and then the volatilities at different estimation periods are calculated. Following (Michael & Christopher, 2006), we consider forecasting horizons of 5, 20, 100, 200, 360, and 500 hours or days ahead for one-hour or one-day volatility respectively. We choose the root mean square error (RMSE) with respect to the true realized volatility as the measure of the forecasting performance:

$$RMSE = \left[\frac{1}{T}\sum_{t=1}^{T}\left(\sigma_t(\tau_t, \tau_{t+T}) - \hat{\sigma}_t(\tau_{i,t}, \tau_{t+T})\right)^2\right]^{\frac{1}{2}} \quad (4)$$

For the shorter evaluation horizons of 5, 20, 100, we calculate the RMSE over the forecasting horizon, i.e. $(\tau_t, \tau_{t+T}) = (1,5), (1,20)$ and $(1,100)$. For the three longer horizons, we calculate the RMSE over 100 time points, i.e. $(\tau_t, \tau_{t+T}) = (100,200), (260,360)$ and $(400,500)$. For the one-hour realized volatility, the RMSE is to evaluate the average performance over 10 days (10 trading hours in each day). For the one-day realized volatility, the RMSE is to evaluate the average performance over 3 month ahead.

4. Results

4.1 Model parameter

In theory, three-layer autoregressive neural network can approximate most of the functions as long as a sufficient number of hidden neurons is provided. In this paper, we construct the NNE2C with 5, 10, and 15 neurons in the hidden layer and test the forecasting performance of the proposed model. The results are illustrated in Table 3 and Table 4 for one-hour and one-day realized volatility data respectively. For the comparison purpose, we also included the forecasting results with 10 hidden neurons in Table 3 and Table 4. From Table 3, it is very clear that for the NNE2C, the forecasting accuracies are roughly the same by using different number of hidden neurons. The forecasting accuracy does not increase with the number of hidden neuron increases. This result is consistent in all forecasting horizons across four currencies. The only case that the forecasting accuracy rises with the increase of the number of hidden neurons is highlighted in Table 3 as GBP/USD rate at $(100{,}200)$ horizon. However, the accuracy increase is as subtle as around 10E-8. For all the cases in Table 4, the forecasting accuracy differences by different number of hidden neurons are also tiny. The only three cases that the forecasting accuracy increases (although tiny) with the rise of the number of hidden neurons are highlighted in Table 4: GBP/JPY rate at horizon $(260{,}360)$, $(400{,}500)$ and GBP/EUR at horizon $(400{,}500)$. In addition, the forecasting accuracies of NNOnly model with different number of hidden neurons do not show significant differences in Table 3 and Table 4 as well. Our results also conform to the previous researches in (Kristjanpoller & Minutolo, 2016) and (Kristjanpoller, Fadic, & Minutolo, 2014). Therefore, based on our experimental evaluations in Table 3 and Table 4, we believe our proposed NNE2C can achieve the best forecasting performance under the structure of three-layer autoregressive neural network with 10 hidden neurons.

Table 3 Forecasting performance of one-hour realized volatilities. This table reports the Root mean square error (RMSE) for the autoregressive neural network enhanced two-component model constructed by one hidden layer with 5 and 15 neurons (NNE2C) and autoregressive neural network model constructed by one hidden layer with 5, 10 and 15 neurons (NNOnly).

τ_1	τ_2		NNE2C (**5 neurons**)	NNE2C (**10 neurons**)	NNE2C (**15 neurons**)	NNOnly (**5 neurons**)	NNOnly (**10 neurons**)	NNOnly (**15 neurons**)
1	5	EUR / USD	2.9838E-06	3.0167E-06	2.9828E-06	5.8882E-02	5.8289E-02	5.8018E-02

		GBP / EUR	9.8432E-06	9.8434E-06	9.8435E-06	7.5266E-02	7.4758E-02	7.4355E-02
		GBP / JPY	4.9453E-05	4.9452E-05	4.9452E-05	1.4364E-01	1.3907E-01	1.4022E-01
		GBP / USD	1.1083E-05	1.1081E-05	1.1087E-05	5.5426E-02	5.2369E-02	5.4421E-02
1	20	EUR / USD	5.7274E-05	5.7239E-05	5.7276E-05	1.3165E-01	1.3015E-01	1.3121E-01
		GBP / EUR	4.8869E-05	4.8870E-05	4.8870E-05	1.2097E-01	1.2032E-01	1.2054E-01
		GBP / JPY	5.3459E-05	2.3459E-05	5.3459E-05	1.5310E-01	1.4879E-01	1.5035E-01
		GBP / USD	2.6065E-05	2.6063E-05	2.6069E-05	7.4815E-02	7.2129E-02	7.3733E-02
1	100	EUR / USD	3.1612E-05	3.1565E-05	3.1614E-05	1.0184E-01	1.0078E-01	1.0153E-01
		GBP / EUR	3.0537E-05	1.0537E-05	3.0538E-05	1.0201E-01	1.0147E-01	1.0153E-01
		GBP / JPY	4.4369E-05	4.4369E-05	4.4369E-05	1.4828E-01	1.4397E-01	1.4558E-01
		GBP / USD	2.6203E-05	2.6201E-05	2.6207E-05	7.4542E-02	7.1763E-02	7.3194E-02
100	200	EUR / USD	2.8230E-05	2.8183E-05	2.8229E-05	9.4983E-02	9.4030E-02	9.4606E-02
		GBP / EUR	3.3874E-05	3.2874E-05	3.3874E-05	1.0056E-01	1.0006E-01	1.0004E-01
		GBP / JPY	6.0882E-05	6.0882E-05	6.0882E-05	1.8717E-01	1.8326E-01	1.8483E-01
		GBP / USD	2.3845E-05	2.3844E-05	2.3843E-05	7.8703E-02	7.6050E-02	7.7510E-02
260	360	EUR / USD	3.8015E-05	3.8012E-05	3.8012E-05	1.5028E-01	1.4931E-01	1.4995E-01
		GBP / EUR	2.0364E-05	2.0363E-05	2.0364E-05	8.7187E-02	8.6600E-02	8.6657E-02
		GBP / JPY	4.8720E-05	4.8719E-05	4.8720E-05	1.7955E-01	1.7590E-01	1.7743E-01
		GBP / USD	1.0392E-04	1.0396E-04	1.0393E-04	5.5737E-01	5.5654E-01	5.5698E-01
400	500	EUR / USD	6.1426E-05	6.1437E-05	6.1424E-05	1.7433E-01	1.7332E-01	1.7402E-01
		GBP / EUR	3.9610E-05	3.9608E-05	3.9608E-05	1.2262E-01	1.2199E-01	1.2209E-01
		GBP / JPY	7.0257E-05	7.0253E-05	7.0258E-05	2.0427E-01	2.0026E-01	2.0216E-01
		GBP / USD	4.7603E-05	4.7615E-05	4.7608E-05	1.9838E-01	1.9656E-01	1.9750E-01

Table 4 Forecasting performance of one-day realized volatilities. This table reports the Root mean square error (RMSE) for the autoregressive neural network enhanced two-component model constructed by one hidden layer with 5 and 15 neurons (NNE2C) and autoregressive neural network model constructed by one hidden layer with 5 and 15neurons (NNOnly)

τ_1	τ_2		NNE2C (**5 neurons**)	NNE2C (**10 neurons**)	NNE2C (**15 neurons**)	NNOnly (**5 neurons**)	NNOnly (**10 neurons**)	NNOnly (**15 neurons**)
1	5	EUR / USD	1.2852E-04	1.2854E-04	1.2860E-04	4.4915E-01	4.5038E-01	4.5214E-01
		GBP / EUR	2.3046E-04	2.3082E-04	2.3053E-04	5.0792E-01	5.0309E-01	5.0854E-01
		GBP / JPY	2.7913E-04	2.7904E-04	2.7938E-04	7.6417E-01	7.6045E-01	7.6562E-01
		GBP / USD	3.3468E-05	3.1831E-05	3.3231E-05	4.0386E-01	4.0349E-01	4.0417E-01
1	20	EUR / USD	1.7691E-04	1.7692E-04	1.7702E-04	5.7042E-01	5.7113E-01	5.7226E-01
		GBP / EUR	1.3060E-04	1.3087E-04	1.3088E-04	4.4148E-01	4.3658E-01	4.4204E-01
		GBP / JPY	2.2931E-04	2.2910E-04	2.3074E-04	7.8099E-01	7.7768E-01	7.8039E-01
		GBP / USD	3.4261E-04	3.4074E-04	3.4227E-04	1.4302E+00	1.4300E+00	1.4300E+00
1	100	EUR / USD	1.8115E-04	1.8101E-04	1.8164E-04	5.6324E-01	5.6411E-01	5.6537E-01
		GBP / EUR	1.9050E-04	1.9037E-04	1.9162E-04	5.8821E-01	5.8323E-01	5.8888E-01
		GBP / JPY	2.9117E-04	2.9235E-04	2.9621E-04	9.1889E-01	9.1522E-01	9.1853E-01
		GBP / USD	1.8631E-04	1.8424E-04	1.8571E-04	7.5809E-01	7.5764E-01	7.5780E-01
100	200	EUR / USD	1.7611E-04	1.7565E-04	1.7784E-04	5.5212E-01	5.5295E-01	5.5436E-01
		GBP / EUR	1.7733E-04	1.7777E-04	1.7590E-04	5.6214E-01	5.5714E-01	5.6286E-01
		GBP / JPY	3.0586E-04	3.0894E-04	3.0814E-04	9.5076E-01	9.4728E-01	9.5029E-01
		GBP / USD	1.7191E-04	1.6691E-04	1.7008E-04	5.4682E-01	5.4626E-01	5.4663E-01

260	360	EUR / USD	1.3838E-04	1.3791E-04	1.4283E-04	4.2934E-01	4.3042E-01	4.3157E-01
		GBP / EUR	1.6731E-04	1.6787E-04	1.6194E-04	5.2557E-01	5.2051E-01	5.2631E-01
		GBP / JPY	2.0534E-04	2.0251E-04	1.7687E-04	6.4310E-01	6.3951E-01	6.4250E-01
		GBP / USD	1.3880E-04	1.3370E-04	1.3637E-04	4.2614E-01	4.2548E-01	4.2597E-01
400	500	EUR / USD	1.0489E-04	1.0359E-04	1.1499E-04	3.2422E-01	3.2540E-01	3.2634E-01
		GBP / EUR	1.4010E-04	1.3893E-04	1.3327E-04	4.3346E-01	4.2840E-01	4.3424E-01
		GBP / JPY	1.6133E-04	1.4925E-04	1.0101E-04	5.0550E-01	5.0165E-01	5.0518E-01
		GBP / USD	1.0201E-04	9.7793E-05	9.9039E-05	3.1800E-01	3.1736E-01	3.1824E-01

4.2 Experimental results

Following the configurations, we employ our experiments using NNE2C with 4 lags and 10 neurons. Table 5 and Table 6 report the RMSE of the one-hour and one-day realized volatilities given by equation (4) for four models over six forecasting intervals for four currencies respectively. Overall, for all models in the experiments in Table 5 and Table 6, the RMSE measures fall at the first with the forecasting horizon and then rise for four currencies. This is due to the reason that initially the forecasting interval increases from five to 20 and then to 100 time points. After the horizon $(\boldsymbol{\tau_t}, \boldsymbol{\tau_{t+T}}) = (\mathbf{1}, \mathbf{100})$, the forecasting interval is fixed at 100 time points, and therefore the forecasting error rises as the horizon increases.

In one-hour realized volatility evaluations in Table 5, the NNE2C model shows the highest forecasting accuracy in 23 out of 24 cases (four currencies with each having 6 horizons) followed by CGARCH model. The only exception is the highlighted case in Table 5: CGARCH achieved RMSE of 3.2067E-05 at horizon $(\mathbf{100}, \mathbf{200})$ on GBP/EUR and was more accurate than the NNE2C, which had 3.2874E-05 RMSE at the same case. At horizon of $(\mathbf{1}, \mathbf{100})$, the performance of CGARCH model is lower but close to the performance of the NNE2C in all four currencies. In other horizons, the NNE2C is significantly more accurate than CGARCH model. Particularly, the decline of the forecasting accuracy of the CGARCH model as well as EGARCH and NNOnly model is obvious from horizon $(\boldsymbol{\tau_t}, \boldsymbol{\tau_{t+T}}) = (\mathbf{100}, \mathbf{200})$ to $(\mathbf{260}, \mathbf{360})$ and then to $(\mathbf{400}, \mathbf{500})$. It shows that at longer forecasting horizon, the NNE2C has sufficiently stronger forecasting capability than the traditional models.

Table 5 Forecasting performance of one-hour realized volatilities. This table reports the Root mean square error (RMSE) for the autoregressive neural network enhanced two-component model constructed by one hidden layer with 10 neurons (NNE2C 10

neurons), Component GARCH model, EGARCH model, and autoregressive neural network model (NNOnly), for the forecasting interval $\boldsymbol{\tau_1}$ to $\boldsymbol{\tau_2}$, where $\boldsymbol{\tau_1 = max(1, \tau_2 - 100)}$, for the one-hour realized volatilities of four currencies.

τ_1	τ_2		NNE2C (10 neurons)	CGARCH	EGARCH	NNOnly (10 neurons)
1	5	EUR / USD	3.0167E-06	5.1341E-04	8.1872E-04	5.8289E-02
		GBP / EUR	9.8434E-06	1.7790E-04	5.8565E-04	7.4758E-02
		GBP / JPY	4.9452E-05	1.9269E-04	1.0531E-03	1.3907E-01
		GBP / USD	1.1081E-05	3.2305E-04	6.4933E-04	5.2369E-02
1	20	EUR / USD	5.7239E-05	1.6669E-04	1.7328E-04	1.3015E-01
		GBP / EUR	4.8870E-05	1.7436E-04	1.3082E-04	1.2032E-01
		GBP / JPY	2.3459E-05	2.9617E-05	3.5046E-04	1.4879E-01
		GBP / USD	2.6063E-05	1.1403E-04	4.5078E-04	7.2129E-02
1	100	EUR / USD	3.1565E-05	2.2018E-05	2.4161E-04	1.0078E-01
		GBP / EUR	1.0537E-05	1.9650E-05	2.1353E-04	1.0147E-01
		GBP / JPY	4.4369E-05	6.9260E-05	4.0864E-04	1.4397E-01
		GBP / USD	2.6201E-05	8.3107E-05	3.9228E-04	7.1763E-02
100	200	EUR / USD	2.8183E-05	5.1364E-05	3.0277E-04	9.4030E-02
		GBP / EUR	3.2874E-05	3.2067E-05	2.5238E-04	1.0006E-01
		GBP / JPY	6.0882E-05	1.7346E-04	4.3271E-04	1.8326E-01
		GBP / USD	2.3844E-05	5.2547E-05	3.2808E-04	7.6050E-02
260	360	EUR / USD	3.8012E-05	2.2140E-04	3.0277E-04	1.4931E-01
		GBP / EUR	2.0363E-05	1.2524E-04	2.6884E-04	8.6600E-02
		GBP / JPY	4.8719E-05	1.0468E-04	1.2865E-03	1.7590E-01
		GBP / USD	1.0396E-04	1.2240E-03	2.9355E-04	5.5654E-01
400	500	EUR / USD	6.1437E-05	5.3047E-04	1.6394E-04	1.7332E-01
		GBP / EUR	3.9608E-05	1.7219E-04	2.3168E-04	1.2199E-01
		GBP / JPY	7.0253E-05	4.0934E-04	8.0361E-04	2.0026E-01
		GBP / USD	4.7615E-05	5.8900E-04	1.5979E-04	1.9656E-01

In Table 6, the NNE2C was significantly more accurate than other three models in all 24 cases. In the highlighted cases in Table 6, which include GBP/EUR rate at $(1,5)$ and $(1,20)$ horizons, and GBP/EUR rate at $(400,500)$ horizon, CGARCH model achieved the forecasting accuracies lower but close to the performance of the NNE2C. In other cases, the NNE2C remarkably outperforms all other models.

It is worth noting that the NNOnly model performs the worst in all cases while the NNE2C achieves the significantly accurate performance. Our results conform to that of **(Zhang & Qi, 2005)** and **(Nelson, Hill, Remus, & O'Connor, 1999)**, which concluded that neural network is not able to model volatility directly but neural networks built with deseasonalized data could produce significantly more accurate forecasting than with non-deseasonalized data.

Table 6 Forecasting performance of one-day realized volatilities. This table reports the Root mean square error (RMSE) for the autoregressive neural network enhanced two-component model constructed by one hidden layer with 10 neurons (NNE2C 10 neurons), Component GARCH model, EGARCH model, and autoregressive neural network model constructed by one hidden layer with 10 neurons (NNOnly 10 neurons), for the forecasting interval $\boldsymbol{\tau_1}$ to $\boldsymbol{\tau_2}$, where $\boldsymbol{\tau_1 = max(1, \tau_2 - 100)}$, for the one-day realized volatilities of four currencies.

τ_1	τ_2		NNE2C (10 neurons)	CGARCH	EGARCH	NNOnly (10 neurons)
1	5	EUR / USD	1.2854E-04	5.7251E-04	8.4922E-04	4.5038E-01
		GBP / EUR	2.3082E-04	3.7841E-04	1.2353E-03	5.0309E-01
		GBP / JPY	2.7904E-04	1.1783E-03	2.2118E-03	7.6045E-01
		GBP / USD	3.1831E-05	3.9678E-04	1.2697E-03	4.0349E-01
1	20	EUR / USD	1.7692E-04	5.4301E-04	1.7308E-03	5.7113E-01
		GBP / EUR	1.3087E-04	2.1771E-04	7.0773E-04	4.3658E-01
		GBP / JPY	2.2910E-04	1.5067E-05	1.3846E-04	7.7768E-01
		GBP / USD	3.4074E-04	3.1626E-03	5.9010E-04	1.4300E+00
1	100	EUR / USD	1.8101E-04	9.4802E-04	1.0034E-04	5.6411E-01
		GBP / EUR	1.9037E-04	1.1753E-03	1.1850E-03	5.8323E-01
		GBP / JPY	2.9235E-04	1.5569E-03	5.2037E-04	9.1522E-01
		GBP / USD	1.8424E-04	1.8094E-03	1.4454E-03	7.5764E-01
100	200	EUR / USD	1.7565E-04	9.4161E-04	7.8610E-04	5.5295E-01
		GBP / EUR	1.7777E-04	1.0306E-03	1.1111E-03	5.5714E-01
		GBP / JPY	3.0894E-04	1.9363E-03	3.8808E-04	9.4728E-01
		GBP / USD	1.6691E-04	1.7113E-03	1.7661E-03	5.4626E-01
260	360	EUR / USD	1.3791E-04	2.9667E-04	2.2160E-03	4.3042E-01
		GBP / EUR	1.6787E-04	6.3274E-04	1.0282E-03	5.2051E-01
		GBP / JPY	2.0251E-04	9.6631E-04	1.5438E-03	6.3951E-01
		GBP / USD	1.3370E-04	6.2537E-04	8.5126E-04	4.2548E-01
400	500	EUR / USD	1.0359E-04	1.3025E-03	3.3809E-03	3.2540E-01
		GBP / EUR	1.3893E-04	2.1427E-04	2.4499E-03	4.2840E-01
		GBP / JPY	1.4925E-04	2.3339E-03	3.5560E-03	5.0165E-01
		GBP / USD	9.7793E-05	4.3271E-04	3.0735E-04	3.1736E-01

5. Conclusions

The fact that volatility comprises both a long-term trend component and a strongly oscillation short-term component has crucial implications for modelling and forecasting volatility over both short and long horizons. In this paper, we develop a simple but effective volatility-forecasting model. The model is based on a decomposition of intraday realized volatility into the long and short-term components using the low-pass Hodrick-Prescott filter. The three-layer autoregressive neural network with 10 hidden neurons models the long-term component and the short-term component is modelled as a simple autoregressive process.

Therefore, we name the proposed model as “neural network enhanced two-component volatility model”. The model was thoroughly evaluated using high frequency tick data of four currencies across six forecasting horizons. The out-of-sample forecasting results consistently and significantly outperform the Component GARCH and EGARCH models as well as the autoregressive neural network model, which is applied on modelling the volatility directly.

The results reported in this paper are based on two simple structures: the low-pass Hodrick-Prescott filter uses a fixed smoothing parameter and the short-term component follows a first order autoregressive progress. Future work would be to consider using an optimized smoothing parameter that decomposes a stationary short-term component. A higher order ARMA process would provide a better fit for the decomposed short-term component and may bring improved out-of-sample forecasting performance. Moreover, a higher smoothing parameter brings more smoothed long-term component, which can be easier to forecasting by neural network with higher accuracy, and decomposes a more volatile short-term component, which may not follow a stationary process. Indeed, it would be a sufficiently further improvement if considering the forecasting model as an optimization framework to find an optimal parameter, which decomposes a stationary short-term process while maintaining a smooth long-term component, which can be forecasted by autoregressive neural network with high accuracy.

Acknowledgement

This research is partially supported by the Scientific Research Funds for the Returned Overseas Chinese Scholars from State Education Ministry, Nature and Science Funds from Fujian Province, China (Grant No. 2015J01236), and the Funds for Young Key Program of Education Department from Fujian Province, China (Grant No. JZ160425)

6. References

Alizadeh, S., Brandt, M. W., & Diebold, F. X. (2002). Range-based estimation of stochastic volatility models. *Journal of Finance*, 1047-1091.

Aljarah, I., Faris, H., Mirjalili, S., & Al-Madi, N. (2016). Training radial basis function networks using biogeography-based optimizer. *Neural Computing and Applications*, 1-25.

Andersen, T., & Bollerslev, T. (1998). Answering the skeptics: Yes, standard volatility models do provide accurate forecasts. *International economic review*, 885-905.

Andreou, P. C., Charalambous, C., & Martzoukos, S. H. (2008). Pricing and trading European options by combining artificial neural networks and parametric models with implied parameters. *European Journal of Operational Research*, 1415–1433.

Baxter, M., & King, R. G. (1999). Measuring Business Cycles Approximate Band-Pass Filters for Economic Time Series. *Review of Economics and Statistics*, 575-593.

Bollerslev, T. (1990). Modelling the coherence in short-run nominal exchange rates: a multivariate generalized ARCH model. *The review of economics and statistics*, 498-505.

Boyacioglu, M. A., & Avci, D. (2010). An Adaptive Network-Based Fuzzy Inference System (ANFIS) for the prediction of stock market return: The case of the Istanbul Stock Exchange. *Expert Systems with Applications*, 7908–7912.

Brandt, M., & Jones, C. (2006). Volatility forecasting with range-based EGARCH models. *Journal of Business and Economic Statistics*, 61–74.

Charalambous, C. (June 1992). Conjugate gradient algorithm for efficient training of artificial neural networks. *IEE Proceedings G - Circuits, Devices and Systems*, 301 - 310.

Gorr, W. L. (June 1994). Research prospective on neural network forecasting. *International Journal of Forecasting, 10*(1), 1-4.

Granger, C. W. (1980). Long memory relationships and the aggregation of dynamic models. *Journal of econometrics 14.2*, 227-238.

Granger, C. W., & Joyeux, R. (1980). An introduction to long-memory time series models and fractional differencing. *Journal of time series analysis*, 15-29.

Hagan, M. T., & Menhaj, M. B. (1994). Training feedforward networks with the Marquardt algorithm. *IEEE transactions on Neural Networks*, 989-993.

Hajizadeh, E., Seifi, A., Zarandi, M. F., & Turksen, I. (2012). A hybrid modeling approach for forecasting the volatility of S&P 500 index return. *Expert Systems with Applications*, 431–436.

Harris, R. D., Stoja, E., & Yilmaz, F. (2015). A cyclical model of exchange rate volatility. *Journal of Banking & Finance*, 3055-3064.

Hodrick, R. J., & Prescott, E. C. (1997). Postwar US business cycles: an empirical investigation. *Journal of Money, credit, and Banking*, 1-16.

Jesús, R. J. (2016). A method with neural networks for the classification of fruits and vegetables. *Soft Computing*, 1–14.

Jia, Z., Yi, C., Yuan, Y., Xuemei, D., & Yuhua, L. (2017). Coarse and fine identification of collusive clique in financial market. *Expert Systems with Applications*, 225–238.

Jiang, Y., Ahmed, S., & Liu, X. (2016). Volatility forecasting in the Chinese commodity futures market with intraday data. *Review of Quantitative Finance and Accounting*, 1-51.

Kristjanpoller, & Minutolo. (2016). Forecasting volatility of oil price using an artificial neural network-GARCH model. *Expert Systems With Applications, 65*, 233–241.

Kristjanpoller, W., Fadic, A., & Minutolo, M. (2014). Volatility forecast using hybrid Neural Network models. *Expert Systems with Applications*, 2437–2442.

Lee, G. G., & Engle, R. F. (1999). A Permanent and Transitory Component Model of Stock Return Volatility. In R. Engle, & H. White, *Cointegration, Causality, and Forecasting: A Festschrift in Honor of Clive W. J. Granger* (pp. 475-497). Oxford: Oxford: Oxford University Press.

Liu, Q., Yin, J., Leung, V. C., Zhai, J.-H., Cai, Z., & Lin, J. (2016). Applying a new localized generalization error model to design neural networks trained with extreme learning machine. *Neural Computing and Applications*, 59–66.

McDonald, S., Coleman, S., McGinnity, Li, Y., & Belatreche, A. (2014). A comparison of forecasting approaches for capital markets. *2104 IEEE Conference on Computational Intelligence for Financial Engineering & Economics (CIFEr).* London: IEEE.

McDonald, S., Coleman, S., McGinnity, T. M., & Li, Y. (2013). A hybrid forecasting approach using ARIMA models and self-organising fuzzy neural networks for capital markets. *The 2013 International Joint Conference on Neural Networks (IJCNN).* Dallas: IEEE.

McElroy, T. (2008). Exact Formulas for the Hodrick-Prescott Filter. *Econometrics Journal*, 209–217.

Mustafaraj, Lowry, & Chen. (2011). Prediction of room temperature and relative humidity by autoregressive linear and nonlinear neural network models for an open office. *Energy and Buildings*, 1452–1460.

Nelson, M., Hill, T., Remus, W., & O'Connor, M. (1999). Time series forecasting using NNs: Should the data be deseasonalized first? *Journal of Forecasting* , 359–367.

Poon, S.-H., & Granger, C. W. (2003). Forecasting Volatility in Financial Markets: A Review. *Journal of Economic Literature, 41(2)*, 478-539.

Ravn, M. O., & Uhlig, H. (2002). On adjusting the Hodrick–Prescott filter for the frequency of observations. *Review of Economics and Statistics*, 371-375.

Restrepo, A. R., Manotas, D. F., & Lozano, C. A. (2016). Self-generation of electricity, assessment and optimization under the new support schemes in colombia. *IEEE Latin America Transactions*, 1308 - 1314.

Rubio. (2016). Least square neural network model of the crude oil blending process. *Neural Networks*, 88–96.

Rubio, J. d., Elias, I., Cruz, D. R., & Pacheco, J. (2016). Uniform stable radial basis function neural network for the prediction in two mechatronic processes. *Neurocomputing*, 122–130.

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

Siegelmann, H., Horne, B., & Giles, C. (1997). Computational capabilities of recurrent NARX neural networks. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 208 - 215.

Stock, J. H., & Watson, M. W. (1999). Forecasting Inflation. *Journal of Monetary Economics*, 293–335.

Stock, J., & Watson, M. (2016). Dynamic Factor Models, Factor-Augmented Vector Autoregressions, and Structural Vector Autoregressions in Macroeconomics. In J. B. Taylor, & H. Uhlig, *Handbook of Macroeconomics* (pp. Volume 2, 415–525). ELSEVIER.

Svalina, I., Galzina, V., Lujić, R., & Šimunović, G. (2013). An adaptive network-based fuzzy inference system (ANFIS) for the forecasting: The case of close price indices. *Expert Systems with Applications*, 6055–6063.

Tim, B. (1986). Generalized autoregressive conditional heteroskedasticity. *Journal of econometrics 31.3*, 307-327.

Torben, A., Tim, B., & Francis, D. (2009). Parametric and nonparametric measurements of volatility. In Y. Aït-Sahalia, & L. P. Hansen, *Handbook of Financial Econometrics.* Amsterdam: North-Holland.

W, B. M., & S, J. C. (2006). Volatility forecasting with range-based EGARCH models. *Journal of Business & Economic Statistics*, 470-486.

Walker, G. (1931). On Periodicity in Series of Related Terms. *Proceedings of the Royal Society of London*, 518–532.

Yi, C., Yuhua, L., Sonya, C., Ammar, B., & Martin, M. (2015). Adaptive Hidden Markov Model With Anomaly States for Price Manipulation Detection. *IEEE Transactions on Neural Networks and Learning Systems*, 318 - 330.

Yule, G. U. (1927). On a Method of Investigating Periodicities in Disturbed Series, with Special Reference to Wolfer's Sunspot Numbers. *Philosophical Transactions of the Royal Society*, 267–298.

Zhai, J., Cao, Y., Yao, Y., Ding, X., & Li, Y. (2017). Computational intelligent hybrid model for detecting disruptive trading activity. *Decision Support Systems*, 26–41.

Zhang, G., & Qi, M. (16 January 2005). Neural network forecasting for seasonal and trend time series. *European Journal of Operational Research*, 501–514.

Zhang, P., & Qi, M. (2005). Neural network forecasting for seasonal and trend time series. *European Journal of Operational Research*, 501–514.