Expert Systems with Applications 36 (2009) 7355–7362

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Improving forecasts of GARCH family models with the artificial neural networks: An application to the daily returns in Istanbul Stock Exchange

Melike Bildirici [a,*], Özgür Ömer Ersin [b,1]

[a] Yıldız Technical University, Faculty of Economics and Business Administration, Department of Economics, Barbaros Bulvari, Besiktas, İstanbul, Turkey
[b] Yeditepe University, Faculty of Commerce, Department of International Trade and Business, Kadikoy, Kayisdagi, İstanbul, Turkey

A R T I C L E I N F O

Keywords:
Volatility
Stock returns
ARCH/GARCH
EGARCH
TGARCH
PGARCH
APGARCH
Artificial neural networks

A B S T R A C T

In the study, we discussed the ARCH/GARCH family models and enhanced them with artificial neural networks to evaluate the volatility of daily returns for 23.10.1987–22.02.2008 period in Istanbul Stock Exchange. We proposed ANN-APGARCH model to increase the forecasting performance of APGARCH model. The ANN-extended versions of the obtained GARCH models improved forecast results. It is noteworthy that daily returns in the ISE show strong volatility clustering, asymmetry and nonlinearity characteristics.

1. Introduction

The importance of volatility has led to the development and application of many significant econometric models and found important application areas in financial markets as a result of both the need of modeling the uncertainty and the risk in financial asset returns. The three most significant characteristics of returns in financial assets could be stated as the following: *volatility clustering property*, as a result of the volatility changes over time in magnitude and in cases where prices hardly change but volatility increases in sizes of large clusters; *asymmetric relation property* of volatility to past return shocks (Engle & Ng, 1993; Glosten, Jagannathan, & Runkle, 1993; Nelson, 1991, 1992); and *nonlinearity property*, the path of volatility reacts differently in different regimes (Klaassen, 2002; Kramer, 2006).

Engle (1982) ARCH models and Bollerslev (1986) GARCH models found many important applications in the financial markets. As a result, GARCH models respond to the need felt for a foresight method that takes into account various properties of the probability distribution of return series of financial variables and had been used intensively in academic studies. Due to this effect, asymmetric GARCH models have rapidly expanded (Nelson, 1991). While Nelson (1991) developed Exponential GARCH (EGARCH) model, Zakoian (1994) and Glosten et al., 1993 working independent of one another developed the GJR-GARCH model. Zakoian (1994) introduced the Threshold GARCH (TGARCH) model and Sentana (1995) introduced the Quadratic GARCH (QGARCH) model.

GARCH models of Taylor (1986) and Schwert (1989) relate the conditional standard deviation of a series and past standard deviations of a different property compared to other models. The model was generalized by Ding, Granger, and Engle (1993) as Power GARCH. This study can be considered as the basis of the APGARCH literature. Hentschel (1995) has applied his study, in which he proposes a more general model of the Power ARCH model, to US stock market data. Tse and Tsui (2002) determined the APGARCH model. Brooks, Faff, McKenzie, and Mitchell (2000) show the leverage effect in the model and the usefulness of including a free power term.

In this study, we aim to analyze the volatility of stock return behavior of Istanbul Stock Exchange ISE 100 Index for the 23.10.1987–22.02.2008 period. The study will compare and combine a general class of Autoregressive Conditional Heteroscedasticity (G)ARCH family models (Bollerslev, 1986; Engle, 1982; Nelson, 1991) of GARCH, EGARCH, GJR-GARCH, TGARCH, NGARCH, SAGARCH, PGARCH, APGARCH, NPGARCH with Artificial Neural Network models and discuss and compare them in accordance with their forecast capabilities. The study is organized into the following sections: *Theory* and *Data Characteristics*, *Econometric Results* and lastly, *Conclusion*.

2. Theory

ANN models have found many important applications especially in the field of financial modeling and forecasting in the recent literature. Among many, Abhyankar, Copeland, and Wang (1997), Castiglione (2001), Freisleben (1992), Kim and Chun

* Corresponding author.
E-mail addresses: bildiri@yildiz.edu.tr (M. Bildirici), oersin@yeditepe.edu.tr (Ö.Ö. Ersin).
[1] Tel.: +90 216 578 0956.

0957-4174/$ - see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2008.09.051

(1998), Liu and Yao (2001), Phua, Zhu, and Koh (2003), Refenes, Zapranis, and Francies (1994) and Zhu, Wang, Xu, and Li (2007) applied nonlinear forecast methods to financial markets. Further, Black and McMillan (2004), Donaldson and Kamstra (1997), Jasic and Wood (2004), Kanas (2001), Kanas and Yannopoulos (2001) and Shively (2003) applied regarding the stock prices; whereas, Dunis and Huang (2002) and Hamid and Iqbal (2004) provide insightful applications regarding the stock volatility.

Dutta and Shektar (1988) and Surkan and Xingren (1991) applied ANN models to rate bonds, Kamijo and Tanigava (1990) to stock prices, Tam and Kiang (1992) to forecasting bank failures, Hutchinson, Lo, and Poggio (1994) to forecasting option prices and as an assessment to hedging, Grudnitsky and Osburn (1993) to gold futures prices and S&P 500 forecasting, Leung, Chen, and Daouk (2000) to FOREX forecasting, Barr and Mani (1994) to investment management, Saad, Prokhorov, and Wunsch (1998) to stock trend predicton, Udo (1993) and Wilson and Sharda (1997) to bankruptcy classification, Kaski and Kohonen (1996) to economic rating and Kong and Martin (1995) and Thiesing, Middleberg, and Vornberger (1995) to sales forecasting.

In the study, we investigated the conditional volatility by applying a hybrid modeling approach that combines various GARCH family models. Furthermore, we used the Back Propagation Artificial Neural Network model for forecasting.

2.1. Multilayer perceptron models

Neural networks represent an important class of nonlinear approximation and classification models relating a set of input variables to one or more output target variables that contain nonlinear latent units to achieve significant flexibility (Kay and Titterington, 1999: p. 2). Rosenblatt (1962) discussed the single hidden layer feedforward neural networks and called ANN models with threshold activation functions as the perceptron. The details of perceptron are analyzed in Block (1962). A similar model had been introduced and discussed in detail by Widrow and Hoff (1960) called ADALINES (ADaptive LINear Elements) that refers to a single hidden unit with threshold nonlinearity. See Bishop (1995) and Widrow and Lehr (1990) for detail.

Rosenblatt (1962) perceptron is stated as

$$o_t = f\left(\sum_{j=0}^{s} w_j \phi_j(x)\right), \quad j = 0, 1, \ldots, s \tag{1}$$

where φ_j is the activation function given in vector form $\varphi_0, \ldots, \varphi_s$; $\mathbf{x}$ is the input variable matrix, w_j is the weight, f is the output function and o is the output of the neuron. The perceptron model of Rosenblatt uses the threshold activation function of the form, $f(a) = \begin{cases} -1 & \text{if} \quad a < 0 \\ +1 & \text{if} \quad a \geq 0 \end{cases}$ The function is bounded between $[-1,+1]$.

In the NN literature, several activation functions are applied. The threshold function or step function can be expressed to be bounded between $[0,1]$ having the form, $f(a) = \begin{cases} 0 & \text{if} \quad a < 0 \\ 1 & \text{if} \quad a \geq 0 \end{cases}$ expressed similar to the *step* function. A commonly applied form of the output function f is the logistic form to achieve a bounded, continuous, sigmoidal and twice differentiable in the log-sigmoid form, $f(a) = 1/(1 + e^{-a})$. The tanh activation function is employed to achieve practical advantage in certain applications and bounded between $[-1,+1]$ similar to Rosenblatt threshold which is defined as $f(a) = \tanh = \frac{e^a - e^{-a}}{e^a + e^{-a}}$, continuous and sigmoid (Bishop, 1995: p. 98).

The multilayer perceptron model is achieved by a weighted linear combination of the d input values in the form

$$a_j = \sum_{i=0}^{d} w_{ji}^{(1)} x_i \tag{2}$$

By employing an activation function $g(\cdot)$; $z_j = g(a_j)$. The network is achieved by associating the activations of the hidden units to the second layer. For each output unit k

$$a_j = \sum_{i=0}^{M} w_{kj}^{(2)} z_j \tag{3}$$

using a nonlinear activation function, $y_k \cong g(a_k)$. The complete function of MLP can be expressed by combining (2) and (3) to give

$$y_k = \tilde{g}\left(\sum_{j=0}^{M} w_{kj}^{(2)} g\left(\sum w_{ji}^{(1)} x_i\right)\right) \tag{4}$$

If the output function is taken linear, $\tilde{g}(a) = a$, the model reduces to

$$y_k = \sum_{j=0}^{M} w_{kj}^{(2)} g\left(\sum w_{ji}^{(1)} x_i\right) \tag{5}$$

There are several training methods for Neural Networks. In NN literature, the most common method of model estimation is Backpropagation (Rumelhart, Rubin, Golden, & Chanvin, 1995), where parameters are updated so that the tuning of parameters is in accordance with the quadratic loss function; hence, the resulting *weight decay* method aims to estimate weights iteratively to achieve the lowest error. Alternative methods include Genetic Algorithms for nonlinear optimization and training of Neural Networks (Goldberg, 1989). Other second-order derivative based optimization algorithms are the Conjugate Gradient Descent, Quasi-Newton, Quick Propagation, Delta-Bar-Delta and Levenberg–Marquardt (Marquardt, 1963), which are faster and effective algorithms but more exposed to over-fitting, an important phenomenon in Neural Networks. To overcome over-learning, we applied two methods. First, the *early stopping* and second, the *algorithm cooperation.* The early stopping approach aims to stopping the training once the selection error starts to rise. The ANN models are retrained with conjugate gradient descent algorithm after the training with backpropagation.[2]

2.2. NN-GARCH models

In addition, we will investigate the negative and positive impacts of shocks on volatility by using the APGARCH model that utilizes the power parameter and then apply if for forecasting. Third, we will investigate the forecast efficiency of GARCH, EGARCH, TGARCH, GJR-GARCH, SAGARCH, PGARCH, NPGARCH, and APGARCH models.

In NN-GARCH model, the learning process is applied to GARCH(1,1) process by including the input variables defined as $\sigma_{t-1}^2 = \gamma_1 \sigma_{t-1}^2$ and $\varepsilon_{t-1}^2 = \beta_1 \varepsilon_{t-1}^2$.

On the other hand, in the NN-EGARCH model, $\ln \sigma_{t-1}^2 = \gamma_1 \ln \sigma_{t-1}^2$ and for the leverage effect,$L(leverage) = \delta \frac{\varepsilon_{t-1}}{\sigma_{t-1}}$ so that $LE(leverage\_effect) = \gamma \left| \frac{\varepsilon_{t-1}}{\sigma_{t-1}} - \sqrt{\frac{2}{\pi}} \right|$ was obtained by including β, γ and δ parameters to define L and LE.

[2] The methodology we followed has two stages to overcome local minima:

(I) Back propogation
- (i) The sample is divided into training/test/selection subsets.
- (ii) Both training and test samples are trained with large steps.
- (iii) The minimization criteria (RMSE) is checked every epoch.
- (iv) Training is early stopped if the RMSE starts to increase for the test subsample even if the same does not hold for the training sample (generalization principle).

(II) Conjugate Gradient Descent
- (v) Training is repeated with conjugate gradient descent (stages i.–iv.) but with small steps (low learning rate in ii.).Model is accepted if the global minimum is reached (see Patterson, 1996; Haykin, 1994; Fausett, 1994).

Donaldson and Kamstra (1997) developed the NN-GJR-GARCH model for financial forecasting. Following the modeling methodology proposed by Donaldson and Kamstra (1997), we introduce the hybrid models that combine a certain class of GARCH family models and ANN models based on the multilayer perceptron models to augment the forecasting capabilities of conditional variance models.

2.2.1. NN-GARCH model

The NN-GARCH(p,q,s) model is an augmented GARCH(p,q) model with a single hidden layer ANN with (s) hidden neurons and defined as

$$\sigma_t^2 = C + \sum_{i=1}^{q} \alpha_i \varepsilon_{t-i}^2 + \sum_{i=1}^{p} \beta_j \sigma_{t-j} + \sum_{h=1}^{s} \xi_h \psi(z_t \lambda_h) \tag{6}$$

$$\psi(z_t\lambda_h) = \left[1 + \exp\left(\lambda_{h,d,w} + \sum_{d=1}^{1}\left[\sum_{w=1}^{m} \lambda_{h,d,w} z_{t-d}^{w}\right]\right)\right]^{-1} \tag{7}$$

$$z_{t-d} = [\varepsilon_{t-d} - E(\varepsilon)]/\sqrt{E(\varepsilon^2)} \tag{8}$$

$$(1/2)\lambda_{h,d,w} \sim uniform[-1,+1] \tag{9}$$

where $\psi(z_t\lambda_h)$ is considered as logistic activation function of the form $1/(1+\exp(-x))$. Following the methodology proposed in Donaldson and Kamstra (1997), the model is augmented with the ANN model. The ANN expression in the GARCH-NN model in Eq. (6) is analoguous to the MLP model in Eq. (5) with different notations; the weight vector $\xi = w$; $\psi = g$ logistic activation function and input variables $z_t\lambda_h = x_i$, where λ_h is defined in Eq. (9).

2.2.2. NN-EGARCH model

The NN-EGARCH model is defined as

$$\log\sigma_t^2 = \alpha + \sum_{i=1}^{p} \beta_i \log\sigma_{t-1}^2 + \sum_{j=1}^{q} \gamma_j [\delta(\varepsilon/\sigma)_{t-j} + |(\varepsilon/\sigma)_{t-j}| - \sqrt{z/\pi}] + \sum_{h=1}^{s} \xi_h \psi(z_t\lambda_h) \tag{10}$$

$$\psi(z_t\lambda_h) = \left[1 + \exp\left(\lambda_{h,d,w} + \sum_{d=1}^{1}\left[\sum_{w=1}^{m} \lambda_{h,d,w} z_{t-d}^{w}\right]\right)\right]^{-1} \tag{11}$$

$$z_{t-d} = [\varepsilon_{t-d} - E(\varepsilon)]/\sqrt{E(\varepsilon^2)} \tag{12}$$

$$(1/2)\lambda_{h,d,w} \sim uniform[-1,+1] \tag{13}$$

σ_t^2 is an asymmetric function of ε_t, $\log(\cdot)$ denotes natural logarithms, $\psi(z_t\lambda_h)$ and z_{t-d} are defined in Eqs. (2) and (3). The model uses a logarithmic specification and the terms in square brackets to account for asymmetric effects in lagged ε As a result of the logarithmic transformation, conditional variance is nonnegative.

2.2.3. NN-TGARCH model

The NN-TGARCH model has the form

$$\sigma_t^y = \alpha_0 + \sum_{i=1}^{q} \beta_i^- I(\varepsilon_{t-1} > 0)|\varepsilon_{t-1}|^y + \alpha_i^- I(\varepsilon_{t-1} \leqslant 0)|\varepsilon_{t-i}|^y] + \sum_{j=1}^{p} \gamma_j \sigma_{t-j}^{\gamma} + \sum_{h=1}^{s} \xi_h \psi(z_t\lambda_h) \tag{14}$$

provided that $I(\cdot)$ is the indicator function. Similarly, for $\psi(z_t\lambda_h)$ and z_{t-d}, Eqs. (11)–(13) hold. If $y = 1$, the model is stated as NN-TGARCH model; whereas if $y = 2$, the model equals to NN-GJR-GARCH model of Donaldson and Kamstra (1997).

2.2.4. NN-GJR-GARCH (Donaldson & Kamstra, 1997)

The NN-TGARCH model stated could be illustrated as Glosten et al., 1993 sign-ARCH model with $y = 2$.Donaldson and Kamstra (1997) augmented the forecasting power of GJR-GARCH model stated as in the form

$$\sigma_t^2 = \alpha + \sum_{i=1}^{q} \beta_i \varepsilon_{t-i}^2 + \sum_{j=1}^{q} \gamma_j \sigma_{t-j}^2 + \sum_{k=1}^{i} \phi_k D_{t-k} \varepsilon_{t-k}^2 \tag{15}$$

The NN-GJR-GARCH model of Donaldson and Kamstra (1997) is stated as

$$\sigma_t^2 = \alpha + \sum_{i=1}^{p} \beta_i \sigma_{t-i}^2 + \sum_{j=1}^{q} \gamma_i \varepsilon_{t-j}^2 + \sum_{k=1}^{r} \phi_k D_{t-k} \varepsilon_{t-k}^2 + \sum_{h=1}^{s} \xi_h \psi(z_t\lambda_h) \tag{16}$$

$$D_{t-k} = \begin{cases} 1 & \text{if} \quad \varepsilon_{t-k} < 0 \\ 0 & \text{if} \quad \varepsilon_{t-k} \leq 0 \end{cases} \tag{17}$$

$$\psi(z_t\lambda_h) = \left[1 + \exp\left(\lambda_{h,d,w} + \sum_{d=1}^{1}\left[\sum_{w=1}^{m} \lambda_{h,d,w} z_{t-d}^{w}\right]\right)\right]^{-1} \tag{18}$$

$$z_{t-d} = [\varepsilon_{t-d} - E(\varepsilon)]/\sqrt{E(\varepsilon^2)} \tag{19}$$

$$(1/2)\lambda_{h,d,w} \sim uniform[-1,+1] \tag{20}$$

where $\psi(z_t\lambda_h)$ is the logistic activation function. In the model, D_{t-k} is the dummy variable that equals 1 if ε_{t-k} is negative and equals zero if ε_{t-k} is nonnegative.

2.2.5. NN-SAGARCH model

Neural Network-Simple Asymmetric GARCH is obtained as

$$\sigma_t^2 = \alpha + \sum_{i=1}^{p} \beta_i \sigma_{t-i}^2 + \sum_{j=1}^{q} \gamma_i \varepsilon_{t-j}^2 + \sum_{k=1}^{r} \phi_k \varepsilon_{t-k} + \sum_{h=1}^{s} \xi_h \psi(z_t\lambda_h) \tag{21}$$

where $\psi(z_t\lambda_h)$ is defined in Eq. (7) and z_{t-d} in Eq. (8). The negative values of φ_k implies that positive shocks will result in smaller increases in future volatility than negative shocks of the same absolute magnitude.

2.2.6. NN-PGARCH model

Neural Network-Power GARCH model is obtained as

$$\sigma_t^1 = \alpha_0 + \sum_{i=1}^{p} \beta_i (|\varepsilon_{t-i}| - \gamma_i \varepsilon_{t-i})^{\delta} + \sum_{j=1}^{q} \phi_j \sigma_{t-i}^{\delta} + \sum_{h=1}^{s} \xi_h \psi(z_t\lambda_h) \tag{22}$$

where the conditional variance is similar to the NN-APGARCH model with $\delta = 1$.

2.2.7. NN-NGARCH model

The NN-NGARCH model is stated as follows:

$$\sigma_t^{\delta} = \alpha + \sum_{k=1}^{r} \phi_k (|\varepsilon_{t-k}|)^{\delta} + \sum_{t=1}^{q} \beta_i \sigma_{t-i}^{\delta} + \sum_{h=1}^{s} \xi_h \psi(z_t\lambda_h) \tag{23}$$

$$\psi(z_t\lambda_h) = \left[1 + \exp\left(\lambda_{h,d,w} + \sum_{d=1}^{1}\left[\sum_{w=1}^{m} \lambda_{h,d,w} z_{t-d}^{w}\right]\right)\right]^{-1} \tag{24}$$

$$z_{t-d} = [\varepsilon_{t-d} - E(\varepsilon)]/\sqrt{E(\varepsilon^2)} \tag{25}$$

$$(1/2)\lambda_{h,d,w} \sim uniform[-1,+1] \tag{26}$$

Further, the NN-APGARCH model reduces to NN-NGARCH model if $\gamma_k = 0$ as discussed below.

2.2.8. NN-APGARCH model

Further, the APGARCH model given in Eq. (10) is modified to obtain the NN-APGARCH model

$$\sigma_t^{\delta} = \alpha + \sum_{k=1}^{r} \phi_k (|\varepsilon_{t-k}| - \gamma_k \varepsilon_{t-k})^{\delta} + \sum_{t=1}^{q} \beta_i \sigma_{t-i}^{\delta} + \sum_{h=1}^{s} \xi_h \psi(z_t\lambda_h) \tag{27}$$

$$\psi(z_t\lambda_h) = \left[1+\exp\left(\lambda_{h,d,w} + \sum_{d=1}^{1}\left[\sum_{w=1}^{m}\lambda_{h,d,w}z_{t-d}^{w}\right]\right)\right]^{-1} \tag{28}$$

$$z_{t-d} = [\varepsilon_{t-d} - E(\varepsilon)]/\sqrt{E(\varepsilon^2)} \tag{29}$$

$$(1/2)\lambda_{h,d,w} \sim uniform[-1,+1] \tag{30}$$

Here, Eq. (17) is the NN-APGARCH model modified with the ANN and the logistic activation function $\psi(z_t\lambda_h)$ in Eq. (18). The NN-APGARCH model reduces to the standard NN-GARCH model for $\delta = 2$ and $\gamma_k = 0$, the NN-NGARCH model for $\gamma_k = 0$, and the NN-GJR-GARCH model for $\delta = 2$ and $0 \le \gamma_k \le 1$; the NN-TGARCH model for $\delta = 1$ and $0 \le \gamma_k \le 1$ (for further discussion in GARCH family models, see: Bollersev (2008)).

2.2.9. NN-NPGARCH

Similar to the NN-APGARCH model, the NN-Nonlinear Power GARCH model is achieved as

$$\sigma_t^{\delta} = \alpha + \sum_{k=1}^{r}\phi_k(|\varepsilon_{t-k}|)^{\delta} + \sum_{t=1}^{q}\beta_i\sigma_{t-i}^{\delta} + \sum_{h=1}^{s}\xi_h\psi(z_t\lambda_h) \tag{31}$$

where the model Eqs. (22)–(24) hold for the $\psi(z_t\lambda_h)$ and z_{t-d}.

2.2.9.1. Forecast comparison methodology. In forecast problems, the importance of forecast evaluation is an important process that aims to improve the quality of forecasts. It is also important to decide on the loss function since it also means that an appropriate measure is selected. The second-order quadratic function means that the MSE is taken as the loss function during the analysis. For forecast performances, the following model selection criteria (MSC) are calculated.

Mean square error (MSE):

$$\mathrm{MSE} = \frac{1}{N-1}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2 \tag{32}$$

and RMSE is obtained as the root of MSE so that

$$\mathrm{RMSE} = \sum_{i=1}^{N}\sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2} \tag{33}$$

where $\hat{y}_i$ is the predicted value, y_i is the observed value and N is the number of observations.

3. Data characteristics

The study concentrates on the work days of the period of 23.10.1987–22.02.2008. The data used in the study consist of the daily closing values of the ISE National-100 Index. The data are obtained from the Central Bank of Turkey Electronic Database. The level values of the index are transformed as the percent change $((t - t_{-1})/t_{-1}) * 100$ and taken as the measure of volatility of the daily return.

In this part of the study, we will try to test the impact of all (domestic and international) good and bad news on the Stock Exchange. Positive effects, as much as adverse effects, had been affective on the return volatility in these periods.

The statistics of daily percentage returns of ISE 100 Index is given in Table 1. Daily returns are given in Fig. 1 in the Appendix. The kurtosis and skewness statistics show that there had been deviation from normality in the series and Jarque–Berra test confirms the result.

4. Econometric results

The study covers 5274 daily observations; the period covers 23.10.1987–22.02.2008. We employed the daily close price of the ISE 100 Stock Index and calculated daily percentage returns. In the following part of the study, we calculated 9 members of ARCH/GARCH family models, namely, GARCH, EGARCH, TGARCH,

Table 1
Statistics of daily percentage returns, ISE 100 Stock Index

Mean	Median	Maximum	Minimum	Standard deviation	Skewness	Kurtosis	Jarque–Berra (prob.)
0.206	0.12	24.25	−18.11	2.84	0.29	7.22	3990.41 (0.0000)

Table 2
ARCH/GARCH family results

Estimated models	Model parameters					
1. GARCH	Arch .148(18.63)	Garch .836(105.07)	*c* .203(8.13)			Log likelihood −12458.04
2. EGARCH	Earch* −.010(−1.88)	Earch-*a* .303(24.21)	Egarch .958(227.07)	*c* .089(10.60)		Log likelihood −12453.87
3. TGARCH	Abarch .156(20.83)	Atarch* −.011(−1.84)	Sdgarch .851(123.38)	*c* .093(9.20)		Log likelihood −12464.75
4. GJR-GARCH	Arch .164(16.71)	Tarch −.025(−2.63)	Garch .833(103.32)	*c* .210(8.18)		Log likelihood −12456.18
5. SAGARCH	Arch .151(18.69)	Saarch −.038(−2.26)	Garch .833(103.31)	*c* .211(8.20)		Log likelihood −12456.7
6. PGARCH	Parch .154(20.08)	Pgarch .845(109.57)	Power 1.582(16.31)	*c* .142(6.90)		Log likelihood −12454.26
7. NGARCH	Narch .151(18.69)	Narch-*k* .125(2.23)	Garch .833(103.26)	*c* .209(8.16)		Log likelihood −12456.66
8. APGARCH	Aparch .156(20.08)	Aparch-*e* −.041(−2.35)	Pgarch .842(107.00)	Power 1.602(15.78)	*c* .149(6.75)	Log likelihood −12452.7
9. NPGARCH	Nparch .156(20.14)	Nparch-*k** .112(1.95)	Pgarch .841(107.53)	Power 1.591(15.84)	*c* .147(6.83)	Log likelihood −12453.13

Note: *z*-values are reported in parantheses. All parameters are significant at 1% level except: parameters with (*) are significant at 10% significance level.

Table 3
Descriptive statistics of conditional variance

Models	Error mean	Error standard deviation	Absolute error mean	Standard deviation ratio[a]	Pearson correlation coefficient
1. NN-GARCH	0.062	2.690	1.373	0.314	0.950
2. NN-EGARCH	−0.017	2.538	1.382	0.348	0.938
3. NN-TGARCH	−0.025	2.245	1.306	0.296	0.955
4. NN-GJR-GARCH	0.016	2.976	1.392	0.336	0.942
5. NN-SAGARCH	−0.065	3.022	1.355	0.339	0.941
6. NN-PGARCH	−0.007	2.646	1.334	0.317	0.949
7. NN-NGARCH	−0.029	2.978	1.345	0.334	0.943
8. NN-APGARCH	0.031	2.771	1.432	0.331	0.943
9. NN-NPGARCH	0.005	2.695	1.373	0.321	0.947

[a] The ratio of the prediction error standard deviation to the original output data standard deviation. A lower standard deviation ratio indicates a better prediction. This is equivalent to 1 minus the explained variance of the model. A s.d. ratio of 0.33 represents that two-thirds of the data are explained by NN model after the removal of the linear part, which depicts strong nonlinearity.

Table 4
Summary of estimated models and training performance

Models[a]	Train performance	Select performance	Test performance	Train RMSE	Select RMSE	Test RMSE
1. NN-GARCH	0.316	0.322	0.291	0.019	0.016	0.015
2. NN-EGARCH	0.344	0.358	0.337	0.020	0.022	0.021
3. NN-TGARCH	0.283	0.295	0.324	0.018	0.019	0.018
4. NN-GJR-GARCH	0.316	0.406	0.286	0.020	0.025	0.016
5. NN-SAGARCH	0.527	0.476	0.589	0.399	0.401	0.401
6. NN-PGARCH	0.309	0.319	0.311	0.018	0.020	0.017
7. NN-NGARCH	0.300	0.379	0.366	0.021	0.025	0.025
8. NN-APGARCH	0.348	0.300	0.318	0.021	0.018	0.016
9. NN-NPGARCH	0.311	0.350	0.303	0.018	0.021	0.016

Note: We used a two step estimation process to overcome local minima problem. First, the ANN model learns with backpropagation (BP) with large steps. Second, after intialization with BP algorithm, model is reestimated with conjugate gradient descent (CG) method following the methodology in the Section 2.1.

[a] Each NN model is estimated for 10 times; summing up to a total of 90 models. Only the best models with the lowest errors are reported. All networks are single hidden layer MLP networks. All models have three layers. The hidden layer activation function is log-sigmoid, whereas the input and the output layers are linear activation functions.

GJR-GARCH, SAGARCH, PGARCH, NGARCH, APGARCH and NPGARCH. The results are given in Table 2.

The models are extended to ANN models. The econometric results of ANN models are given in Table 3. In the study, we analyzed ANN models based on the multilayer perceptrons forecast evaluation to improve the forecasting performance of GARCH family models. The obtained descriptive statistics for the predictions are given below.

For estimation purposes, the training of the network is done in two stages with two algorithms to avoid local minima. First, the back propagation algorithm used for 100 epochs that search the gradient area as an overall search. At the second stage, the methodology continues with the conjugate gradient descent algorithm for i epochs. Algorithm is terminated at the ith epoch where the RMSE of the selection set started to rise as the RMSE of the training set is continuing to decrease, which is taken as a strong sign of overlearning (overfitting). The results for training are given in Table 4.

We compared the estimated models according to the calculated RMSE. The results for estimated GARCH family models and NN-GARCH models are given in Table 5. We compared the percentage decrease in RMSE values. We regarded a model to have an improved forecasting power if the model provides a decrease in RMSE of equal or less than −5% in RMSE values. On the other hand, models with comparatively more RMSE values are not rejected, since their RMSE values are within the 5% interval.

In-sample forecast comparisons are given in Table 5. It is observed that RMSE values of ANN models are lower than the RMSE values of their GARCH versions. The highest value of forecast augmentation is achieved for the NN-TGARCH models compared to the TGARCH model, where the RMSE is calculated as 2.89721

Table 5
Forecast comparisons of the models

Model	RMSE	Model	RMSE	% Decrease in RMSE[a] after ANN specification
1. GARCH	2.89713	1. NN-GARCH	2.65467	−8.37
2. EGARCH	2.89721	2. NN-EGARCH	2.50434	−13.56
3. TGARCH	2.89720	3. NN-TGARCH	2.21608	−23.51
4. GJR-GARCH	2.89731	4. NN-GJR-GARCH	2.90481	0.26[a]
5. SAGARCH	2.89729	5. NN-SAGARCH	2.98097	2.89[a]
6. PGARCH	2.89713	6. NN-PGARCH	2.58586	−10.74
7. NGARCH	2.89729	7. NN-NGARCH	2.93682	1.36[a]
8. APGARCH	2.89729	8. NN-APGARCH	2.73516	−5.6
9. NPGARCH	2.89728	9. NN-NPGARCH	2.65879	−8.23

A decrease (increase) in RMSE of 5% or more (less) is regarded as an improvement (worsening) in forecast performance.

[a] The difference in RMSE is insignificant to conclude that the relevant latter NN version has less forecasting power against the former.

Table 6
Out-of-sample comparisons and RMSE values

Models	RMSE values for the forecast horizon, $T+h$						
	$T+2$	$T+6$	$T+11$	$T+21$	$T+41$	$T+81$	$T+161$
1. NN-GARCH	0.79591	1.51803	1.41859	1.97721	2.21470	2.59342	2.69419
2. NN-EGARCH	0.81949	1.18782	1.30525	1.90916	2.15762	2.65002	2.86726
3. NN-TGARCH	0.69753	0.93871	0.94287	1.46257	1.65966	2.43475	2.52748
4. NN-GJR-GARCH	0.91363	1.34098	1.63724	2.26277	2.57423	2.81441	2.97441
5. NN-SAGARCH	0.96436	1.39401	1.59485	2.00909	2.08462	2.73431	2.47025
6. NN-PGARCH	0.79980	1.16111	1.40639	1.98718	2.19131	2.55386	2.66025
7. NN-NGARCH	0.73144	0.97121	1.15343	1.71512	2.04479	2.59372	2.70624
8. NN-APGARCH	1.00330	1.30065	1.43154	1.83962	1.89423	2.69614	2.39682
9. NN-NPGARCH	1.08392	1.45762	1.61526	2.06759	2.15154	2.67374	2.50722

Table 7
Percentage (%) increase in out-of-sample RMSE values

Models	$T+2$	$T+6$ (%)	$T+11$ (%)	$T+21$ (%)	$T+41$ (%)	$T+81$ (%)	$T+161$ (%)
1. NN-GARCH	–	90.73	−6.55	39.38	12.01	17.10	3.89
2. NN-EGARCH	–	44.95	9.89	46.27	13.01	22.82	8.20
3. NN-TGARCH	–	34.58	0.44	55.12	13.48	46.70	3.81
4. NN-GJR-GARCH	–	46.77	22.09	38.21	13.76	9.33	5.69
5. NN-SAGARCH	–	44.55	14.41	25.97	3.76	31.17	−9.66
6. NN-PGARCH	–	45.18	21.12	41.30	10.27	16.54	4.17
7. NN-NGARCH	–	32.78	18.76	48.70	19.22	26.85	4.34
8. NN-APGARCH	–	29.64	10.06	28.51	2.97	42.33	−11.10
9. NN-NPGARCH	–	34.48	10.81	28.00	4.06	24.27	−6.23

Note. $T+2$ RMSE values are taken as the base value. A positive (negative) value represents percentage increase (decrease) compared to the previous forecast.

for the latter and 2.21608 for the former. The percentage decrease in RMSE after ANN specification is −8.37% for the NN-GARCH, −13.56% for NN-EGARCH, −23.51% for NN-TGARCH, +0.26% for NN-GJR-GARCH, +2.89% for NN-SAGARCH, −10.74% for NN-PGARCH, 1.36% for NN-NGARCH, −5.6% for NN-APGARCH and −8.23% for NN-NPGARCH models. In the forecasting literature, an appreciation or devaluation in RMSE to be accepted significant, it is common to use a 5 and 10 confidence interval in model comparisons. Since the positive increases in RMSE values are 0.26%, 2.89% and 1.36% for NN-GJR-GARCH, NN-SAGARCH and NN-NGARCH models, we failed to reject their relative significance.

Accordingly, the out-of-sample forecast comparisons are given in Table 6. We compared the percentage increase in RMSE of forecast at $T+h$ horizon and the previous horizon. Comparisons are given in Table 7. One important point is that the RMSE values of the NN models are comparatively lower than expected. Starting from $T+2$ days to $T+81$ days, all models provided lower RMSE values, noticeably lower than even the in-sample values.

Thus, the RMSE values are comparatively lower even for the most distant horizon of $T+161$ days. Furthermore, we compared the percentage increase in RMSE values of out-of-sample forecasts in Table 7. The highest increase in the RMSE is observed for $T+6$ for all models. Following the $T+6$ days forecasts, RMSE values followed a stable path until $T+161$ days.

On the other hand, the increase in RMSE lowered after $T+21$ forecasts and decreased between $T+81$ and $T+161$ days forecasts for NN-APGARCH, NN-NPGARCH and NN-SAGARCH models. Thus, the highest RMSE is obtained by NN-GJR-GARCH with 2.97 and NN-EGARCH model with 2.86. According to the results, the RMSE values obtained for long horizons for the ANN models are lower than their GARCH family and almost equal for the ANN models with the highest out-of-sample RMSE values.

We noticed that the training methodology we followed consisting of two stages utilizes the early stopping, a procedure to allow us to obtain good generalization (forecasting) power. We should note the fact that comparatively lower in-sample RMSE values could have been achieved without early stopping in cases where the optimization had been stopped before the training set reached its lowest RMSE; however, the resulting model might have lost its comparative power in out-of-sample forecasts. Accordingly, we conclude that the ANN models discussed in the study improved the generalization and forecasting power of the GARCH models.

5. Conclusions

In the study, we analyzed and compared a general class of Autoregressive Conditional Heteroscedasticity (G)ARCH family models (Bollerslev, 1986; Engle, 1982; Nelson, 1991) of GARCH, EGARCH, SAGARCH, PGARCH, NPGARCH, TARCH, APGARCH, GJR-GARCH, NPGARCH and combined them with Artificial Neural Network models (Bishop, 1995; Haykin, 1999).

The main aim of the study is to augment the forecasting power of GARCH models. Further, we followed the NN-GJR-GARCH modeling methodology proposed in Donaldson and Kamstra (1997) and developed the NN-GARCH, NN-EGARCH, NN-TGARCH, NN-SAGARCH, NN-PGARCH, NN-NGARCH, NN-APGARCH and NN-NPGARCH models. We observed that the conditional variance models augmented with artificial neural networks propose an improvement in modeling as long as an improved model captures the volatility more efficiently.

Our results suggest that the ANN models discussed in the study performed especially for out-of-sample values. The training methodology we followed consisting of two stages uses the early stopping procedure to allow us to obtain good generalization (forecasting) power. According to the results obtained, ANN models provide significant improvement in forecasts.

Appendix

See Fig. 1.

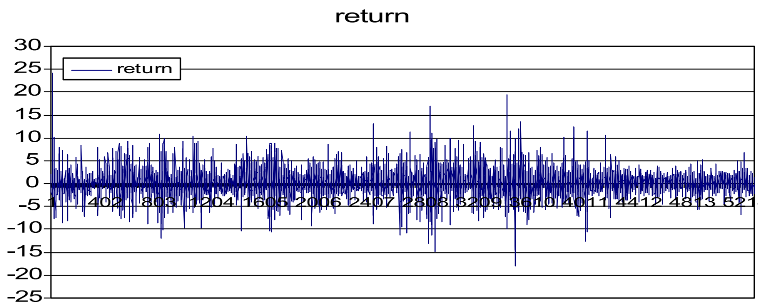

Fig. 1. ISE 100 daily returns, 1987–2008.

References

Abhyankar, A., Copeland, L. S., & Wong, W. (1997). Uncovering nonlinear structure in real-time stock-market indexes: The S&P 500, the DAX, the Nikkei 225, and the FTSE-100. *Journal of Business & Economic Statistics, 15*(1), 1–14.

Barr, D. S., & Mani, G. (1994). Using neural nets to manage investments. *AI Expert, 9*, 16–21.

Bishop, C. M. (1995). *Neural networks for pattern recognition* (1st ed.). New York: Oxford University Press.

Black, A. J., & McMillan, D. G. (2004). Long run trends and volatility spillovers in daily exchange rates. *Applied Financial Economics, 14*(12), 895–907.

Block, H. D. (1962). The perceptron: A model for brain functioning. *Reviews of Modern Physics, 34*(1), 123–135.

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

Bollersev, T. (2008). Glossary to ARCH(GARCH). In T. Bollerslev, J. R. Russell & M. Watson (Eds.), *Festschrift in Honor of Robert F. Engle*.

Brooks, R. D., Faff, R. W., McKenzie, M. D., & Mitchell, H. (2000). A multi-country study of power ARCH models and national stock market returns. *Journal of International Money and Finance, 19*(3).

Castiglione, F. (2001). Forecasting price increments using an artificial neural network. *Advances in Complex Systems, 4*(1), 45–56.

Ding, Z., Granger, C. W. J., & Engle, R. F. (1993). A long memory property of stock market returns and a new model. *Journal of Empirical Finance, 1*, 83–106.

Donaldson, R. G., & Kamstra, M. (1997). An artificial neural network-GARCH model for international stock return volatility. *Journal of Empirical Finance, 4*, 17–46.

Dutta, S., & Shekar, S. (1988). Bond-rating: A non-conservative application of neural networks. In *Proceedings of IEEE international conference on neural networks*, pp. 443–450.

Dunis, C. L., & Huang, X. (2002). Forecasting and trading currency volatility: An application of recurrent neural regression and model combination. *Journal of Forecasting, 13*, 317–354.

Engle, R. F. (1982). Autoregressive conditional heteroskedasticity with estimates of the variance of United Kingdom inflation. *Econometrica, 50*, 987–1007.

Engle, R. F., & Ng, V. K. (1993). Measuring and testing the impact of news on volatility. *Journal of Finance, 48*, 1749–1778.

Freisleben, B. (1992). Stock market prediction with back propagation networks. In *Proceedings of the 5th international conference on industrial and engineering application of artificial intelligence and expert systems* (pp. 451–460).

Glosten, L. R., Jagannathan, R., & Runkle, D. (1993). On the relation between the expected value and the volatility of the nominal excess return on stocks. *Journal of Finance, 48*, 1779–1801.

Goldberg, D. E. (1989). *Genetic algorithms in search, optimization and machine learning* (1st ed.). Boston, MA: Kluwer Academic Publishers.

Grudnitsky, G., & Osburn, L. (1993). Forecasting S&P 500 and gold futures prices: An application of neural networks. *Journal of Futures Markets, 13*, 631–643.

Hamid, S. A., & Iqbal, Z. (2004). Using neural networks for forecasting volatility of S&P 500 Index futures prices. *Journal of Business Research, 57*, 1116–1125.

Hentschel, L. (1995). All in the family: Nesting symmetric and asymmetric GARCH models. *Journal of Financial Economics, 39*, 71–104.

Haykin, S. (1999). *Neural networks a comprehensive foundation* (2nd ed.). New Jersey: Prentice Hall.

Hutchinson, J. M., Lo, A. W., & Poggio, T. (1994). A nonparametric approach to pricing and hedging derivative securities via learning networks. *Journal of Finance, 49*, 851–859.

Jasic, T., & Wood, D. (2004). The profitability of daily stock market indices trades based on neural network predictions: Case study for the S&P 500, the DAX, the TOPIX and the FTSE in the period 1965–1999. *Applied Financial Economics, 14*, 285–297.

Kanas, A. (2001). Neural network linear forecasts for stock returns. *International Journal of Finance & Economics, 13*, 317–354.

Kanas, A., & Yannopoulos, A. (2001). Comparing linear and nonlinear forecasts for stock returns. *International Review of Economics & Finance, 10*, 383–398.

Kamijo, K., & Tanigava, T. (1990). Stock price recognition - a recurrent neural net approach. In *Proceedings of the international joint conference on neural networks.* San Diego: SOS Printing, I:589.

Kaski, S., & Kohonen, T. (1996). Exploratory data analysis by the self-orginizing map: Structures of welfare and poverty in the world. In *Proceedings of the third international conference on neural networks in the capital markets* (pp. 498–507).

Kay, J. W., & Titterington, D. M. (1999). *Statistics and neural networks: Advances at the interface* (1st ed.). Oxford: Oxford University Press.

Kim, S. H., & Chun, S. H. (1998). Graded forecasting using an array of bipolar predictions: Application of probabilistic neural networks to a stock market index. *International Journal of Forecasting, 14*, 323–337.

Kong, J. H. L., & Martin, G. M. (1995). A backpropagation neural network for sales forecasting. In *Proceedings of the IEEE international conference on neural networks* (Vol. 2, pp. 1007–1011).

Klaassen, F. (2002). Improving GARCH volatility forecasts with regime-switching GARCH. *Empirical Economics, 27*, 363–394.

Kramer, W. (2006). *Long memory with Markov-switching GARCH*. Working papers 6, Business and Social Statistics Department, University Dortmund, revised October 2006.

Leung, M. T., Chen, A. S., & Daouk, H. (2000). Forecasting exchange rates using general regression neural networks. *Computers and Operations Research, 27*(11), 1093–1110.

Liu, Y., & Yao, X. (2001). Evolving neural networks for Hang Seng stock index forecast. In *Proceedings of the 2001 congress on evolutionary computation* (Vol. 1, pp. 256–260).

Marquardt, D. W. (1963). An algorithm for the least-squares estimation of nonlinear parameters. *SIAM Journal of Applied Mathematics, 11*(2), 431–441.

Nelson, D. B. (1991). Conditional heteroscedasticity in assets returns: A new approach. *Econometrica, 55*, 703–708.

Nelson, D. B. (1992). Filtering and forecasting with misspecified ARCH models I: Getting the right variance with the wrong model. *Journal of Econometrics, 52*, 61–90.

Phua, P. K. H., Zhu, X., & Koh, C. (2003). Forecasting stock index increments using neural networks with trust region methods. In *Proceedings of IEEE international joint conference on neural networks* (Vol. 1, pp. 260–265).

Refenes, A. N., Zapranis, A., & Francies, G. (1994). Stock performance modeling using neural networks: A comparative study with regression models. *Neural Networks, 5*, 961–970.

Rosenblatt, F. (1962). *Principles of neuro-dynamics: Perceptrons and the theory of brain mechanisms* (1st ed.). Washington, DC: Spartan.

Rumelhart, D. E., Rubin, R., Golden, R., & Chauvin, Y. (1995). Backpropagation: The basic theory. In Y. Chauvin & D. E. Rumelhart (Eds.), *Backpropagation: Theory, architectures, and applications* (pp. 1–34). Hillsdale NJ: Lawrence Erlbaum.

Saad, E. W., Prokhorov, D. V., & Wunsch, D.C. II. (1998). Comparative study of stock trend prediction using time delay, recurrent and probabilistic neural networks. *IEEE Transactions on Neural Networks, 9*(6), 1456–1470.

Schwert, G. W. (1989). Stock volatility and crash of '87. *Review of Financial Studies, 3*, 77–102.

Shively, P. A. (2003). The nonlinear dynamics of stock prices. *Quarterly Review of Economics and Finance, 13*, 505–517.

Sentana, E. (1995). Quadratic ARCH models. *Review of Economic Studies, 62*, 639–661.

Surkan, A. J., & Xingren, Y. (1991). Bond rating formulas derived through simplifying a trained neural network. In *Proceedings of the IEEE international conference on neural networks* (Vol. 2, pp. 1028–1031).

Tam, K. Y., & Kiang, M. Y. (1992). Managerial applications of neural networks: The case of bank failure predictions. *Management Science, 38*, 926–947.

Thiesing, F. M., Middleberg, U., & Vornberger, O. (1995). Short term predictions of sales in supermarkets. In *Proceedings of the IEEE international conference on neural networks* (Vol. 2, pp. 1028–1031).

Taylor, S. (1986). *Modelling financial time series*. New York: John Wiley & Sons.

Tse, Y. K., & Tsui, A. K. C. (2002). A multivariate GARCH model with time-varying correlations. *Journal of Business and Economic Statistics, 20*, 351–362.

Udo, G. (1993). Neural network performance on the bankrubtcy classification problem. *Computers & Industrial Engineering, 25*, 377–380.
Widrow, B., & Hoff, M. E. (1960). Adaptive switching circuits. In *IRE WESCON convention record* (Vol. 4, pp. 96–104).
Widrow, B., & Lehr (1990). 30 years of adaptive neural networks: Perceptron, madaline, and backpropagation. *Proceeding of the IEEE, 78*(9), 1415–1442.
Wilson, R., & Sharda, R. (1997). Business failure prediction using neural networks. *Encyclopedia of Computer Science and Technology, 37*(22), 193–204.
Zakoian, J. M. (1994). Threshold heteroskedastic models. *Journal of Economic Dynamics and Control, 18*, 931–955.
Zhu, X., Wang, H., Xu, L., & Li, H. (2007). Predicting stock index increments by neural networks: The role of trading volume under different horizons. *Expert Systems with Applications*. doi:10.1016/j.eswa.2007.06.023. <www.elsevier.com/locate/eswa>.