An investigation on the performance of long and short memory models in forecasting realized volatility of emerging stock market returns

[1]Altaf Hossain, M. Arifur Rahman[2] and M. Nasser[1]

[1]Department of Statistics, Rajshahi University, Rajshahi – 6205, Bangladesh

[2]Department of Finance and Banking, Rajshahi University, Rajshahi – 6205, Bangladesh

E-mails: rasel_stat71@yahoo.com, mnasser.ru@gmail.com and marifurr@yahoo.com

Abstract – Measuring and modeling financial volatility is the key to asset and derivative pricing, asset allocation and risk management. Volatility of emerging stock market returns poses especial challenges in this regard. Given that there are several estimators of volatility, we use daily realized volatility estimated from intraday returns as a proxy for the true daily volatility. In this paper, we endeavour to apply three models, finite mixture of ARMA – GARCH, BPNN and SVR vis-à-vis the long memory model ARFIMA model in forecasting (directly) the realized volatility in the context of an emerging stock market. Based on the deviation criteria, the SVR together with ARFIMA models perform better than the ARMA – GARCH and BP models. With respect to the direction criteria, the ARMA – GARCH model tends to forecast better than the competing models. The memory property of these forecasting techniques is also examined using the behavior of forecasted values vis-à-vis the original values. The SVR model together with long memory ARFIMA model shows long memory property in forecasting. All the remaining models give constant forecasts after a short period of time.

**This paper was accepted in the conference entitled "High frequency data models with application to Finance" organized at Stevens Institute of Technology, New Jersey, July 10 – 12, 2009. Online available:*

http://bergenbier.math.stevens.edu/conference2009/index.php/program/menu-talk/72-abstract-hossain.html

Keywords – Finite Mixture of ARMA – GARCH, Back Propagation Neural Network (BPNN), Support Vector Regression (SVR), Realized Volatility and Fractionally Integrated ARMA (ARFIMA)

1. Introduction

As volatility is not a directly observable variable, large research areas have emerged that attempts to best address this problem. By far the most popular approach is to obtain volatility estimates using the statistical models that have been proposed in the autoregressive conditional heteroskedasticity (ARCH) and Stochastic Volatility literature. Stochastic volatility models with separate dynamic structure for the volatility process have been in the focus of the mathematical finance literature; see Heston (1993) and Bates (2000), while parametric generalized autoregressive conditional heteroskedasticity (GARCH) – type models for the returns of the underlying (s) have been intensively analyzed in financial econometrics. Another method of extracting information about volatility is to formulate and apply economic models that link the information contained in the options to the volatility of the underlying asset. All these approaches have in common that the resulting volatility measures are only valid under the specific assumptions of the models used and it is generally uncertain which or whether any of these specifications provide a good description of actual volatility. Moreover, the evaluation of the predictive ability of volatility models is quite important in empirical applications. However, the latent character of the volatility poses a problem. To what measure should the volatility forecasts be compared to? Conventionally, the forecasts of daily volatility models, such as GARCH – type or stochastic volatility models have been evaluated with respect to absolute or squared daily returns. In view of the excellent in – sample performance of these models, the forecasting performance, however, seems to be disappointing. A model – free measure of volatility is the sample variance of returns. Using daily data, for instance, it may be freely estimated using returns spanning over any number of days and, as such, one can construct a time series of model – free variance estimates. When one chooses the observation frequency of this series, an important trade – off has to be made, however. When the variances are calculated using a large number of observations (e.g. the returns over an entire year), many interesting properties of volatility tend to disappear (the volatility clustering and leverage effect, for instance). On the other hand, if only very few observations are used, the measures are subject to great error. At the extreme, only one return observation is used for each daily variance estimate.

The availability of ultra – high – frequency data opens the door for a refined measurement of volatility and model evaluation. The approach taken in this dissertation is to calculate the daily volatility from the sample variance of intraday returns, the 'realized' volatility. These are free of the assumptions necessary when the statistical or economic approaches are employed and, as we have an (almost) continuous record of returns for each day, we can calculate the interdaily variances with little or perhaps negligible error. The realized volatility turns out to be very useful in the assessment of the validity of volatility models. For instance, reconciling evidence in favor of the forecast accuracy of GARCH – type models is observed when using realized volatility as a benchmark rather than daily squared returns. Moreover, the availability of the realized volatility measure initiated the development of a new and quite accurate class of volatility models. Almost all of the work on daily volatility is within the confines of ARCH/GARCH and Stochastic Volatility models or derivative pricing formulas. There are exceptions, however.

Studies focusing on realized volatility start with Andersen and Bollerslev (1998) who constructs the realized volatility by summing the squared intraday returns and the first paper that proposes the realized volatility to be a volatility measure. The paper shows the dramatic improvement of the forecast performance of a daily GARCH model by using the new volatility measure. This study can be regarded as the seminal paper on using high frequency data in volatility forecasting.

A number of further studies by the same authors focus further on the forecasting of realized volatility and its properties. ABDL (1999) first recommends without application, the Fractionally Integrated Autoregressive Moving Average (ARFIMA) model for forecasting realized volatility after studying the properties of the distributions of realized volatility and realized covariance. ABDL (2000) supports the argument that realized volatility is an efficient estimator of integrated volatility by showing the empirical results that the returns standardized by realized volatility are nearly normally distributed. The main contribution of ABDL (2001a) is that it recognizes that realized volatility can benefit forecasting if it is directly modeled by a parametric model rather than simply used as an evaluation of other models' forecasting behavior. The findings of the above studies constitute the theoretical base of directly using realized volatility in volatility forecasting. Martens (2001) compares daily volatility forecasts constructed from multiple volatility forecasts of intraday intervals, with a daily model and a daily model extended by intraday information. It finds that the higher the intraday frequency is used, the better are the out-of-sample daily volatility forecasts. The daily model, including intraday information, has similar performance to the method that models intraday returns directly. Martens and Zein (2002) provides the evidence that using high frequency data can improve both the accuracy of measurement and the performance of forecasting. It also shows that the long memory model contributes to its improvement as well. Hol and Koopman (2002) adopt the high frequency return series of the S&P100 stock index and use two kinds of models: realized volatility models and daily time-varying volatility models to compare their predictive power. The results of the out-of-sample evaluation show that the ARFIMA model estimated by realized volatility gives the most accurate forecast. Pong et al. (2004) compares among option implied volatility and the forecasts obtained from the short memory model--ARMA, the long memory model---ARFIMA, and the daily GARCH model. It finds that the most accurate historical forecasts come from the use of high frequency returns, not from a long memory specification.

Again volatility of emerging stock market returns poses especial challenges in this regard. In sharp contrast to the well developed stock markets, emerging markets are generally characterized by high volatility. In addition, high volatility in these markets is often marked by frequent and erratic changes, which are usually driven by various local events (such as political developments) rather than by the events of global importance (Bekaer and Harvey (1997), Aggarwal et al. (1999)). In this paper, we attempt to assess the performance of alternative models of forecasting future volatility in the context of Indian stock market. Given that there are several estimators of volatility, we use the daily realized variance estimated from intraday returns as a proxy for the true daily variance.

Unlike most other estimators, this estimator has been shown to be model – free by Andersen et al. (2001a, 2001b). The realized volatility models should be able to capture the strong persistence in the sample correlation function. One approach is to assume that the long memory is generated by fractional integrated process as originally introduced by Granger and Joyeux (1980) and Hosking (1981). In the GARCH literature this has led to the development of the fractionally integrated GARCH model (Baillie et al. (1996)). For the realized volatility, a fractionally integrated autoregressive moving average (ARFIMA) process is generally used (see for example, Andersen et al. (2003) and Hardle et al. (2008)). In ARFIMA, the error term is usually assumed to be a Gaussian white noise process. Several extensions of the realized volatility ARFIMA model have been proposed for non – Gaussianity of realized volatility (see Corsi et al. (2008)). The financial returns usually have fatter tails than the normal distribution and reveal significant excess kurtosis. The severity of this problem is particularly true of emerging stock market returns (Bekaer et al. (1998)). Although GARCH – type models often quite successful in dealing with excess kurtosis, they can not completely capture this property in the real data. To improve the forecasting ability of GARCH model, the neural networking method (NN) can be used to model volatility that does not rely too much on the empirical assumptions of error distributions, such as normality. However, NN suffers from a number of weaknesses including the need for a large number of controlling parameters, difficulty in obtaining a global solution and the danger of over-fitting. More recently, a novel NN algorithm, called support vector machine (SVM), was developed by Vapnik and his co-authors (1998, 1995). The SVM implements the structural risk minimization (SRM) principle which seeks to minimize an upper bound as opposed to the empirical risk minimization (ERM) that minimizes the error on the in-sample estimating data.

In this paper, our contribution is to investigate the memory property of ARMA – GARCH, BP neural network and SVR models vis-à-vis the long memory model (ARFIMA) in forecasting (directly) the realized volatility in the context of an emerging stock market. The memory property of these forecasting techniques is also examined using the behavior of forecasted values vis-à-vis the original values.

The rest of the paper is organized as the following way. Motivation of our paper is given in Section 2. The methodology is discussed in Section 3. The data source and their properties are introduced in Section 4. The results and discussion are presented in Section 5. Finally, our summary and conclusion are in Section 6.

2. Motivation

Wong et al. (1998) proposed the finite mixture of AR-GARCH model and it performed better than pure GARCH in financial prediction. Again Tang et al. (2003) extended the mixture of AR-GARCH model to the mixture of ARMA-GARCH model and it yield better prediction results than the mixture of AR-GARCH models. Chen et al. (2008) applied SVR in return square volatility forecasting. The experiment showed that the SVR-based GARCH model outperforms the MA, the recurrent NN and the parametric GARCH based on the criteria of mean absolute error (MAE) and directional accuracy (DA); they did not account of the finite mixture ARMA-GARCH. Motivated by Wong et al. (1998), Tang et al. (2003) and Chen et al. (2008), we use SVR with ARMA - GARCH and NN methods in forecasting (directly) realized volatility.

3. Methodology

3.1 Fractional Integration and Long Memory Process

3.1.1 Introduction

This section introduces an alternative method of modeling long term memory in a time series, through the use of a fractional value d. Fractionally integrated extensions of ARMA models are developed and methods of testing for fractional integration and estimation such models are introduced. We have so far considered only integer values of d. If d is non-integer, however, x_t is said to be fractionally integrated, and model for such values of d are referred to as ARFIMA by Diebold and Rudebusch (1989).

To make the concept operational, we may use the binomial series expansion

$$\nabla^d = (1-B)^d = \sum_{k=0}^{\infty} \begin{bmatrix} d \\ k \end{bmatrix} (-B)^k$$

$$= 1 - dB - \frac{d(1-d)}{2!} B^2 - \frac{d(1-d)(2-d)}{3!} B^3 - \dots \dots \dots \qquad (1)$$

How does the ARFIMA model incorporate 'long memory' behavior? For $0 < d < \frac{1}{2}$, it can be shown that its ACF declines hyperbolically to zero, i.e., at a much slower rate than the exponential decay of a standard ARMA (d = 0) process. For $d \geq \frac{1}{2}$, the variance of x_t is infinite, and so the process is non-stationary.

Examples of how autocorrelation vary with d are provided in Hosking (1981). Typically, autocorrelations from ARFIMA processes remain noticeably positive at very high lags, long after the autocorrelations from I(0) processes have declined to (almost) zero.

3.1.2 Realized Volatility Models

Realized volatility models should be able to capture the strong persistence in the sample autocorrelation function. One approach is to assume that the long memory is generated by Granger and Joyeux (1980) and Hosking (1981). In the GARCH literature this has lead to the development of the fractionally integrated GARCH model as, e.g., proposed by Baillie et al. (1996). For realized volatility the use of a fractionally integrated autoregressive moving average (ARFIMA) process was advocated, for example, by Andersen et al. (2003).

The ARFIMA (p, q) model is given by

$$\phi(L)(1-L)^d (x_n - \mu) = \psi(L)u_n, \quad \text{……………….. (2)}$$

with $\phi(L) = 1 - \phi_1 L - ... - \phi_p L^p, \psi(L) = 1 + \psi_1 L + ... + \psi_q L^q$, and d denoting the fractional difference parameter. Moreover, u_n is usually assumed to be a Gaussian white noise process, and x_n denotes either the realized volatility (see Koopman et al. (2005)) or its logarithmic transformation. Several extensions of the realized volatility ARFIMA model have been proposed, accounting, for example, for leverage effects (see Martens et al. (2004)), for non-Gaussianity of (log) realized volatility or for time – variation in the volatility of realized volatility (see Corsi et al. (2008)). Generally the empirical results show significant improvements in the point forecasts of volatility when using ARFIMA rather than GARCH – type models.

Since it is the parameter d that enables long – term persistence to be modeled, the value chosen for it is obviously crucial in any empirical application. Typically, this value will be unknown and must therefore be estimated. See Mills (1993) for more in details. A popular approach is proposed by Geweke and Porter – Hudak (1983). Having obtained an estimate of $\tilde{d}$, and hence d, x_t can be transformed by the long memory filter (1), truncated at each point to the available sample. The transformed series is then modeled as an ARMA process. Sowell (1992a, 1992b) discusses joint maximum likelihood estimation of d and the ARMA parameters.

3.2. Finite Mixture of ARMA-GARCH for Time Series Forecasting

The linear time series model like AR and ARMA can be combined with GARCH to model the dynamics of stock indices and their volatilities. The mixture of ARMA-GARCH model is similar to the mixture of AR-GARCH model proposed by Wong et al. (1998). Specifically, each component of the mixture model can be denoted as a normal ARMA series

$$y_{t,j} = \sum_{r=1}^{R} b_{rj} y_{t-r,j} + \sum_{s=1}^{S} a_{sj} \in_{t-s,j} + \in_{t,j} , \quad (3)$$

Furthermore, each residual term $\in_{t,j}$ is assumed Gaussian white noise with variance denoted by the GARCH model

$$\sigma_{t,j}^2 = \delta_{0j} + \sum_{q=1}^{Q} \delta_{qj} \in_{t-q,j}^2 + \sum_{p=1}^{P} \beta_{pj} \sigma_{t-p,j}^2 , \quad (4)$$

where $\delta_{qj} > 0$ for q = 1, …, Q and $\beta_{pj} > 0$ for p = 1, …, P.

Mathematically, the finite mixture of ARMA – GARCH model can be denoted as a K – component Gaussian mixture model

$$P(y_t) = \sum_{j=1}^{K} \alpha_j G(y_t; \hat{y}_{t,j}, \sigma_{t,j}^2), \tag{5}$$

$$\hat{y}_{t,j} = \sum_{r=1}^{R} b_{rj} y_{t-r,j} + \sum_{s=1}^{S} a_{sj} \in_{t-s,j}, \tag{6}$$

where $\alpha_j > 0$ and $\sum_{j=1}^{K} \alpha_j = 1$.

Once the model has been learned, one – step ahead prediction can be done via taking expectation of y_t

$$E(y_t) = \alpha_1 \hat{y}_{t,1} + \ldots + \alpha_K \hat{y}_{t,K} \tag{7}$$

A generalized expectation-maximization (GEM) algorithm is used to learn the mixture model. See Tang et al. (2003) for details.

3.3. ANNs for Time Series Forecasting

An Artificial NN (ANN) is a biologically inspired form of distributed computation. It simulates the functions of the nervous system by a composition of interconnected simple elements (artificial neurons) operating in parallel. An element is a simple structure that performs three basic functions: input, processing and output. ANNs can be organized into several different connection topologies and learning algorithms (Lippmann (1987)). In case of time-series, the series at different lags act as inputs of ANNs. The number of inputs to the network is constrained by the problem type, whereas the number of neurons in the output layer is constrained by the number of outputs required by the problem type.

However, the number of hidden layers and the size of the layers are decided by the designer.

There are several different BP training algorithms with a variety of different computation and storage requirements. Kamruzzaman and Sarker (2003) used three algorithms, Standard Backpropagation (SBP), Scaled Conjugate Gradient (SCG) and Backpropagation with Baysian Regularization (BPR), in training the ANN. It was found that SCG based model performs best when measured on the two most commonly used metrics and showed competitive results when compared with BPR based model on other three metrics. BP is one of the most commonly used algorithms in financial research. Chen and Wu (2006) therefore used only the SBP and mentioned, "No single algorithm is best suited to all the problems." So we also consider only the SBP for training the ANN models in our paper.

3.4. SVMs for Time Series Forecasting

3.4.1 Introduction to SVMs

The SVM, originally developed as an implementation of Vapnik's Structural Risk Minimization (SRM) principle (Vapnik (1995)), is now being used to solve a variety of learning, classification and prediction problems. It has the following advantages over ANN: (1) it can obtain the global optimum and (2) the overfitting problem can be easily controlled.

The SVM deals with the classification and regression problems by mapping the input data into the higher-dimensional feature spaces. Its central feature is that the regression surface can be determined by a subset of points or Support-Vectors (SV); all other points are not important in determining the surface of the regression. Vapnik introduced a ε -

insensitive zone in the error loss function (Figure 1). Training vectors that lie within this zone are deemed correct, whereas those that lie outside the zone are deemed incorrect and contribute to the error loss function. As with classification, these incorrect vectors also become the support vector set. Vectors lying on the dotted line are SV, whereas those within the ε -insensitive zone are not important in terms of the regression function.

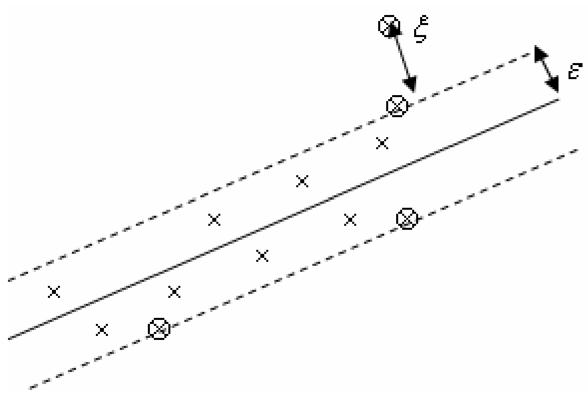

Figure 1: Approximation function (solid line) of the SV regression using a ε -insensitive zone

3.4.2. Support-vector Regression

The SVR algorithm tries to construct a linear function such that training points lie within a distance ε (Figure 1).

Given a set of training data $\{(x_1, y_1),...,(x_l, y_l)\} \subset X \times R$, where X denotes the space of the input patterns, the goal of SVR is to find a function $f(x)$ that has at most ε deviation from the targets y_i for all the training data and, at the same time, is as flat as possible.

Let the linear function f takes the form:

$$f(x) = \langle w, x \rangle + b; w \in X, b \in R \tag{8}$$

The optimal regression function is given by the minimum of the functional,

$$\Phi(w,\xi) = \frac{1}{2}\|w\|^2 + C\sum_i (\xi_i^- + \xi_i^+), \tag{9}$$

where C is pre-specified value, and ξ^-, ξ^+ are slack variables representing upper and lower constraints on the outputs of the system. Flatness in (8) means a smaller $\|w\|$. Using an ε-insensitive loss function,

$$L_\varepsilon(y) = \begin{cases} 0 & for\,|f(x) - y| < \varepsilon \\ |f(x) - y| - \varepsilon & otherwise \end{cases} \tag{10}$$

the solution is given by,

$$\begin{aligned} &\max_{\alpha,\alpha^*} W(\alpha,\alpha^*) = \max_{\alpha,\alpha^*} -\frac{1}{2}\sum_{i,j=1}^{l}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*) < x_i, x_j > \\ &+ \sum_{i=1}^{l} \alpha_i (y_i - \varepsilon) - \alpha_i^* (y_i + \varepsilon) \end{aligned} \tag{11}$$

with constraints,

$$\begin{aligned} &0 \le \alpha_i, \alpha_i^* \le C, i = 1,2,...,l \\ &\sum_{i=1}^{l} (\alpha_i - \alpha_i^*) = 0. \end{aligned} \tag{12}$$

Solving equation of (11) with constraints equation (12) determine the Lagrange multipliers, α, α^* and the regression function is given by (8), where

$$\begin{aligned} &\overline{w} = \sum_{i=1}^{l} (\alpha_i - \alpha_i^*) x_i \\ &\overline{b} = -\frac{1}{2} < \overline{w}, (x_r + x_s) > . \end{aligned} \tag{13}$$

w is determine by training patterns x_i, which are SVs. In a sense, the complexity of the SVR is independent of the dimensions of the input space because it only depends on the number of SV.

To enable the SVR to predict a non-linear situation, we map the input data into a *feature space*. The mapping to the feature space F is denoted by

$$\Phi : \Re^n \rightarrow F$$
$$x \mapsto \Phi(x)$$

The optimization equation (8) can be written as

$$\max_{\alpha,\alpha^*} W(\alpha,\alpha^*) = \max_{\alpha,\alpha^*} -\frac{1}{2}\sum_{i,j=1}^{l}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*) < \Phi(x_i), \Phi(x_j) >$$
$$+\sum_{i=1}^{l}\alpha_i(y_i - \varepsilon) - \alpha_i^*(y_i + \varepsilon)$$

The decision can be computed by the inner products, $< \Phi(x_i), \Phi(x_j) >$ without explicitly mapping to a higher dimension which is a time – consuming task. Hence the kernel function is as follows:

$$K(x,z) \equiv < \Phi(x), \Phi(z) >$$

By using a kernel function, it is possible to compute the SVR without explicitly mapping in the feature space. The condition for choosing kernel functions should confirm to Mercers condition, which allows the kernel substitutions to represent dot products in some Hilbert space.

3.5 Preprocessing and Design of the Models

Each of the four data sets is partitioned into three subsets according to the time sequence in the ratio 95: 2.5: 2.5 in forecasting realized volatility. The first part is used for training; the second is a validating set that selects optimal parameters for the SVR, identifies the orders of ARMA-GARCH and ARFIMA models finally and prevents the overfitting found in the BP neural networks, and the last is used for testing.

In Standard BP models, we use one hidden layer of different units for each market. We select the number of inputs for the ANNs, and units for the hidden layer of that ANNs which give minimum MSE on the validation sets. Regression techniques like SVR or RBF nets can be used to estimate the prediction function on the basis of time – delay coordinates. For stationary dynamical systems the embedding parameters can be found e.g. by the method of Liebert et al. (1991).

3.6. Performance Criteria

Although the MSE is a perfectly acceptable measure of performance, in practice the ultimate goal of any testing strategy is to confirm that the results of models are robust and capable of measuring the profitability of a system. It is important, therefore, to design a test from the outset. According to Tay and Cao (2001) and Thomason (1999a, b), the prediction performance is evaluated using the following statistics: MSE, Normalized Mean Squared Error (NMSE), Mean Absolute Error (MAE), Directional Symmetry (DS) and Weighted Directional Symmetry (WDS). We only use two statistics, MAE and DS, and these two criteria are defined by

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|a_i - p_i\right|$$

$$DS = \frac{100}{n}\sum_{i=1}^{n} d_i$$

$$d_i = \begin{cases} 1 & (a_i - a_{i-1})(p_i - p_{i-1}) \geq 0 \\ 0 & otherwise \end{cases}$$

See Chen and Wu (2006) for more details.

4. Data and Their Properties

The data series are two leading stock market indices of India - the S&P-CNX Nifty and BSE SENSEX, which are both value-weighted indices of 50 and 30 most actively traded stocks of the National Stock Exchange and Bombay Stock Exchange of India, respectively. Our sample period is from 03 December 2007 to 20 April 2009.

The daily realized volatility is calculated based on 15-minute period. Throughout our experiment, we use the realized volatility multiplied by 100 as a realized volatility. The realized volatility series of each market is plotted in the same Figure 2. As also shown by the corresponding summary statistics in Table 1, the realized volatility series are highly volatile themselves and reveal significant excess kurtosis. The realized volatility of BSE SENSEX is to some extent more volatile than that of Nifty.

Table 1: Summary Statistics of Realized Volatility of Returns for Emerging Markets

	TIME PERIOD	N	MEAN	STDEV	SKEWNESS	KURTOSIS
BSE	2007.12.03-2009.04.20	328	0.06252	0.06809	2.53	**8.32**
NIFTY	2007.12.03-2009.04.20	328	0.06337	0.06933	2.38	**6.76**

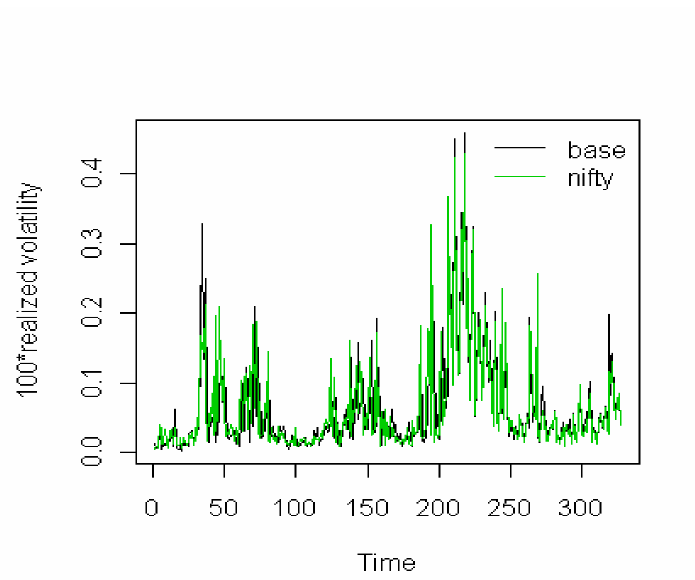

Figure 2: Realized Volatility Series of Index Returns for Two Emerging Markets

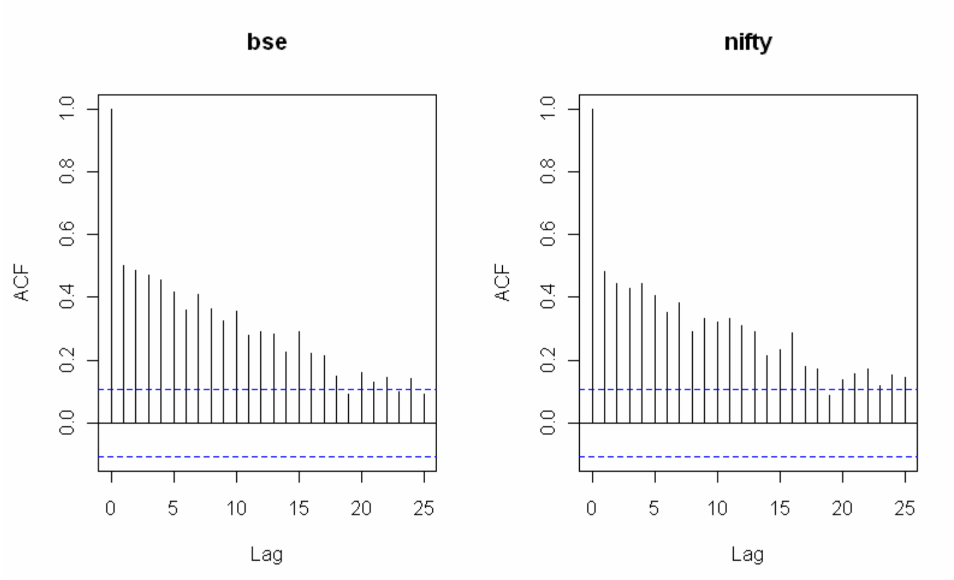

Figure 3: ACFs of Realized Volatility of Returns for Emerging Markets

Figure 3 of ACFs shows that the realized volatility is strongly positively autocorrelated with high persistence, which typical of the emerging stock markets. The plot shows that the ACF decays relatively slowly providing hints on the existence of long range dependence which is well captured by fractionally integrated processes.

5. Results and Discussion

5.1. BP and SVR Implementation

The BP models used in this experiment are implemented using the R-2.8.1-win32's tsDyn Package. We use the conjugate gradient learning methods both in forecasting the returns and realized volatility. The architecture of the BP model is as follows: the optimum number of hidden nodes is 2 for each market in forecasting realized volatility that minimize the error rate on the validation sets are determined market – wise. The activate function of the hidden layer is sigmoid and the output node uses the linear transfer function.

We apply Vapnik's SVM for regression by using the R-2.8.1-win32's e1071 Package. We use the radial basis kernel function $k(x, y) = \exp(-\gamma|x - y|^2)$ from several typical functions as it performs well under general smoothness assumptions. The set of parameters to be determined is (γ, C, ε). Method of choosing these parameters is an area of active research (Duan et al. (2003); Rossi and de Carvallo (2008)). We choose them following a version of the most popular method of cross validation. The optimum sets ((.005, 1, 0.001) for NIKE 255 and (.001, 1, 0.01) for S & P 500 index, and (.001, 1, 0.001) for US - JP and (.01, 1, 0.001) for US - UK exchange rates) in forecasting returns that minimize the error rate on the validation sets are determined market – wise.

5.2. Comparison and Discussion

For such a realized volatility process generally $d \in (0,0.5)$ and $d \geq 0.5$ indicates the process is non – stationary. The estimated $d \geq 0.5$ for both the series we analyze. We have fitted ARFIMA to each of our volatility series. Resulting skewness and kurtosis of residuals are reported in Table 2. The table shows excess skewness and kurtosis for the ARFIMA model.

Table 2: Skewness and Kurtosis of the Residuals for ARFIMA and GARCH Models

		SKEWNESS	KURTOSIS
BSE	ARFIMA	1.78	**6.67**
	ARMA - GARCH	2.72	**9.74**
NIFTY	ARFIMA	1.56	**4.75**
	ARMA - GARCH	2.16	**6.05**

To excess kurtosis, we have fitted ARMA – GARCH model and resulting residuals statistics are also given in the same table (Table 2). It shows that the ARMA – GARCH models have failed to capture excess kurtosis; rather this type of models exhibit greater excess kurtosis than the ARFIMA type models.

Next, we introduce the back propagation (standard) NN and simple SVR models to see how they fare with ARFIMA and ARMA – GARCH models in forecasting realized volatility. The mean absolute error (MAE) and directional accuracy (DA) obtained are reported in Table 3. The table shows that, based on the MAE criteria, the SVR together with ARFIMA models perform better than the GARCH type and BP models but again ARFIMA model forecasts relatively better than the SVR model. However, when we compare alternative models based on DA criterion, the GARCH type models tend to forecast better than the competing models.

Table 3: One-Period-Ahead Forecasting Accuracy

		ARFIMA	ARMA - GARCH	BP	SVR
BSE	MAE	0.025832	0.041180	0.032178	0.029582
	DS	26.666	93.333	40	26.666
NIFTY	MAE	0.024450	0.044471	0.026303	0.026241
	DS	26.666	93.333	33.333	26.666

Note: in ARFIMA, estimated d = 0.69842 for BSE, .67883 for NIFTY

Finally, we have plotted the forecasted realized volatility from all four models along with actual realized volatility for each market in Figure 4. It can be easily observed that ARFIMA being the long memory model forecasts consistently over the whole period (including validation period) with time varying property for each market. Also, the SVR model behaves much like the long memory ARFIMA model. However, on the other hand, both the GARCH-type and BP models give only the constant forecasts.

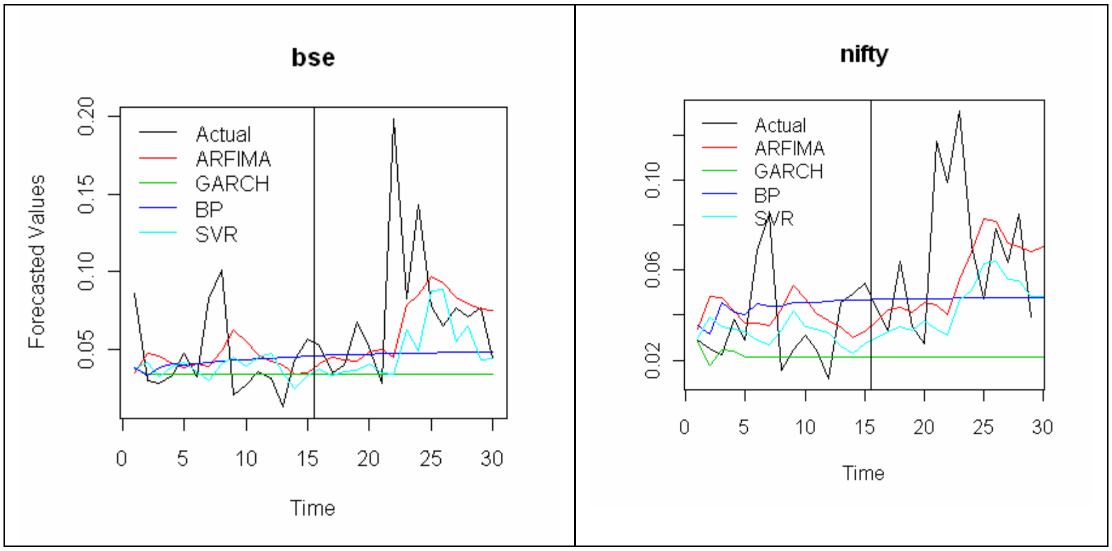

Figure 4: Volatility Forecasts of Each Market (Vertical Line in Each Panel Divides the Forecasting Period into Validation and Test Periods)

6. Summary and Conclusion

Based on the deviation criteria, the SVR together with ARFIMA models perform better than the ARMA – GARCH and BP models. With respect to the directional criteria, the ARMA – GARCH model tends to forecast better than the competing models. It can also be easily observed that ARFIMA being the long memory model forecasts consistently over the whole period (including validation period) with time varying property for each market. Also, the SVR model behaves much like the long memory ARFIMA model. However, on the other hand, both the GARCH – type and BP models give only the constant forecasts.

The statistical methods, ARMA and ARMA-GARCH, require large number of sample size for better forecasting and these models drastically reduce the original sample size when the high order model is fitted. Though these statistical methods have not performed well in deviation performance criteria, our results support that these models have both the important interpretability and intrinsic properties. To improve the forecasting ability of GARCH type model, we can use NN methods as our results show that NN performs well in both the deviation and direction performance criteria. But the ANN structures contain hidden layers which are not interpretable. And the selection of optimal parameters for ANN is very difficult, that is, in the selection procedure, there is no art. In this thesis, we have simply used autoregressive (AR) model based SVR model. So, SVR model is quite simple and has the interpretability properties as opposed to the complex GARCH type and NN models. The simple AR model is estimated by maximum likelihood estimation (MLE) (which is usually affected by potential outliers), while the SVR model is estimated by robust estimation procedure. And the selection of optimal parameters for SVR is easier than NN. Rather, SVR model additionally shows long memory property supporting the statement that SVR model have the property of generalization power. In summary, our result suggest that the simple SVR model together with ARFIMA model could be used fairly successfully as long memory models in forecasting realized volatility series of the emerging market.

Acknowledgment

We would like to express our special thanks to the authors of fGarch, tsDyn and e1071 packages of R which we have used to implement ARMA-GARCH, ANN and SVM modeling and forecasting. In addition, we would like to thank the reviewers of the conferences of ICCIT, 2008 as well as IASC2008, audiences for giving valuable suggestions.

Reference:

Aggarwal, R., Inclan, C. and Leal, R. (1999), "Volatility in Emerging Stock Markets", *Journal of Financial and Quantitative Analysis* **34(1):** 33-55.

Andersen, T., and Bollerslev, T. (1998), "Answering the skeptics: Yes, standard volatility models do provide accurate forecasts", *International Economic Review*, **39**, **4**, 885–905.

Andersen, T.G., Bollerslev, T., Diebold, F.X. and Labys, P. (1999), "(Understanding, Optimization, Using and Forecasting) Realized Volatility and Correlation", New York University, Leonard N. Stem School Finance Department *Working Paper*.

Andersen, T.G., Bollerslev, T., Diebold, F.X. and Labys, P. (2000), "Exchange Rate Returns Standardized by Realized Volatility are (nearly) Gaussian", Multinational Finance Journal, **4**: 159 – 179.

Andersen, T.G., T. Bollerslev, F.X. Diebold and P. Labys(2001a), "The distribution of Realized Exchange Rate Volatility", *Journal of American Statistical Association* **96:** 42-55.

Andersen, T.G., T. Bollerslev, F.X. Diebold and H. Ebens (2001b), "The distribution of Stock Return Volatility", *Journal of Financial Economics* **61(1):** 43-76.

Andersen, T.G., Bollerslev, T., Diebold, F.X. and Labys, P. (2003), "Modeling and Forecasting Realized Volatility", *Econometrica*, **71**: 579 – 625.

Bates, D.S. (2000), "Post-87 crash fears in S&P 500 futures options", *Journal of Econometrics*, **94**, 181–238.

Baillie, R.T., Bollerslev, T. and Mikkelsen, H.O.(1996), "Fractionally Integrated Generalized Autoregresive Conditional Heteroskedasticity", *Journal of Econometrics* **74**: 3 – 30.

Beckers, S. (1993), "Variances of security price returns based on high, low and closing prices", *Journal of Business*, **56:** 97–112.

Bekaert, G., C.B. Erb, C.R. Harvey and T.E. Viskanta (1998), "Distributional Characteristics of Emerging Market Returns and Asset Allocation", *Journal of Portfolio Management* **Winter**: 102-116.

Chen, W-H., Shih, J-Y. and Wu, S. (2006), "Comparison of support-vector machines and back propagation neural networks in forecasting the six major Asian stock markets", *Int. J. Electronic Finance*, Vol. 1, No. **1**: 49-67.

Chen, S, Jeong, K. and Härdle, W. (2008), "Support Vector Regression Based GARCH Model with Application to Forecasting Volatility of Financial Returns", *SFB 649*

Discussion Paper 2008-014, http://edoc.hu-berlin.de/series/sfb-649-papers/2008-14/PDF/14.pdf

Corsi, F., Mittnik, S., Pigorsch, C. and Pigrosch, U.(2008), 'The Volatility of Realized Volatility', Econometric Reviews **27**: 46 – 78.

Diebold, F.X. and Rudebusch, G.D.(1989), "Long Memory and Persistence in Aggregate Output", *Journal of Monetary Economics*, **24**: 189-209.

Duan, K., Keerthi, S.S., and Poo, A.N. (2003), "Evaluation of simple performance measures for tuning SVM hyperparameters", *Neurocomputing* **51**: 41 – 59.

Granger, C.W.J. and Joyeux, R. (1980), "An Introduction to Long – Range Time Series Models and Fractional Differencing", *Journal of Time Series Analysis*, **1**: 15 – 30.

Härdle, W., N Hautsch and U Pigorsch, "Measuring and Modeling Risk Using High – Frequency Data", SFB 649 Discussion Paper 2008-014, http://sfb649.wiwi.hu-berlin.de/papers/pdf/SFB649DP2008-045.pdf

Heston, S.L. (1993), "A closed solution for options with stochastic volatility, with application to bond and currency options", *Review of Financial Studies*, **6, 2,** 327–343.

Hol, E., Koopman S.J. (2002), "Stock Index Volatility Forecasting with high Frequency Data", Tinbergen Institute Discussion Paper No. 2002-068/4.

Hosking, J.R.M.(1981), "Fractional Differencing", *Biometrika* **68**: 165 – 176.

Kamruzzaman, J. and Sarker, R. (2003), "Forecasting of currency exchange rates using ANN: a case study", *Proc. IEEE Intl. Conf. on Neur. Net. & Sign. Process. (ICNNSP03), China*.

Koopman, S.J., Jungbacker, B. and Hall, E. (2005), "Forecasting Daily Variability of the S&P 100 Stock Index Using Historical, Realized and Implied Volatility Measurements ", *Journal of Empirical Finance* **12, 3**: 445 – 475.

Liebert, W., Pawelzik, K. and Schuster, H.G.(1991), "Optimal embeddings of chaotic attractors from topological considerations", *Europhys. Lett*., **14**: 521 – 526.

Lippmann, R.P.(1987), "An introduction to computing with neural nets", *IEEE ASSP Magazine*, 36-54.

Martens, M. (2001), "Forecasting daily exchange rate volatility using intraday returns", *Journal of International Money and Finance*, **20**, **1:**1-23

Martens, M., Zein, J. (2002), "Predicting Financial Volatility: High-Frequency Time-Series Forecasts Vis-a-Vis Implied Volatility", working paper, Erasmus University

Rotterdam (EUR), Econometric Institute and University of New South Wales, School of Banking and Finance.

Martens, M., van Dijk, D. and dePooter, M. (2004), “Modeling and Forecasting S&P 500 Volatility: Long Memory, Structural Breaks and Nonlinearity”, *Erasmus University Rotterdam, Working Paper*.

Mills, Terence C. (1993), ‘*The econometric modeling of financial time series*’, Cambridge.

Pong, S., Shackleton, M., Taylor, S. J. and Xu, X. (2004), “Forecasting Currency Volatility: A Comparison of Implied Volatilities and AR(FI)MA Models”, *Journal of Banking and Finance*,**28**, **10**: 2541-2563.

Rossi, A.L.D, and Carvalho, A.C.P.L.F. (2008), “Bio-inspired Optimization Techniques for SVM Parameter Tuning*”, 10th Brazilian Symposium on Neural Networks*.

Tay, F.E.H. and Cao, L. (2001), “Application of support vector machines in financial time-series forecasting”, *Omega*, **29**: 309-317.

Tang, H., Chun, K.C. and Xu, Lei(2003), “Finite Mixture of ARMA-GARCH Model for Stock Price Prediction”, *Proc. Of 3rd International Workshop on Computational Intelligence in Economics and Finance* (CIEF2003), North Carolina, USA, 1112-1119.

Thomason, M. (1999a), “The practitioner method and tools: a basic neural network-based trading system project revisited (parts 1 and 2)”, *Journal of Computational Intelligence in Finance*, Vol. 7, No. 3: 36-45.

Thomason, M. (1999b), “The practitioner method and tools: a basic neural network-based trading system project revisited (parts 3 and 4)”, *Journal of Computational Intelligence in Finance*, Vol. 7, No. 3: 35-48.

Vapnik, V. (1995), “*The Nature of Statistical Learning Theory*”, Springer, N.Y.. ISBN 0-387-94559-8.

Vapnik, V. (1998), “*Statistical Learning Theory*”, Springer, N.Y..

Wong, W.C., Yip, F. and Xu, L.(1998), “Financial Prediction by Finite Mixture GARCH Model”, *in Proceeding of Fifth International Conference on Neural Information Processing*, 1351-1354.