EECS 545 Final Project

Dapeng Shang dpshang@umich.edu
Feng Wang frwang@umich.edu

Support vector machines approach on forecasting stock volatility with GARCH models

Abstract

In this paper, we use of Support Vector Machines (SVMs) based GARCH is studied in financial forecasting by comparing it with original GARCH model. SVMs forecast better than traditional GARCH based on the criteria of Mean Absolute Error (MAE), Mean Square Error (MSE) and Root Mean Square Error (RMSE). S&P-500 and NASDAQ-100 daily price index is used as the data set. Our empirical results show that the proposed approach improves forecasting performance by 6.72% in RMSE.

1. Introduction

In the financial market, volatility forecasting is essential for risk management and financial derivatives pricing. The estimate volatility could be used as an instrument for assesses future losses by computing Value at Risk (VaR). Option traders also make their own forecasting volatility to evaluate option prices and make decisions. The simplest approach to estimating the stock volatility is to use historical standard deviation. However, when more researches appear on stocks volatility, people find that these volatilities have more property, like autoregressive and heteroscedasticity, that future volatilities have correlation and different variance with historical volatilities. Then the new models ARCH and GARCH have been generated.

A large number of time series based volatility models have been developed since the introduction of ARCH model of Engle (1982). Then, Bollerslev (1986) generalizes it into GARCH model. Typically, researchers use GARCH model to generate volatility forecasts and they use maximum likelihood estimation (rests on the assumption that data (Errors) follow a specific distribution like Gaussian distribution).

Support vector machines are supervised learning tools for linear and nonlinear input-output data classification and regression analysis. Since SVM needs few strong assumptions and should not over-fit the data, it becomes a commonly used forecasting method for financial data compared to other method, like GARCH.

The empirical results show that using machine learning approaches combined with GARCH models yield better results. Chen et al (2008b) applied SVM to model and forecast GARCH(1,1) volatility based on the concept of recurrent SVM, following from the recurrent algorithm of neural network and least square support vector machine of Suykens and Vandewalle (2000). The model was shown to be a dynamic process and capture longer memory of past information than the feed-forward SVM that is just static. Phichhang Ou et al (2010) applied this recurrent SVM into forecasting the China stock market volatility. There result show that the recurrent SVM method generates superior forecasting performance, improving original one by 22.92% in MAD.

In this project, the main idea is to use SVM and Linear regression model to estimate GARCH models parameters. Also, we will test the results from these papers and evaluate them accuracy.

2. Data

We examine of S&P-500 and Nasdaq-100 index in the experiment. The stock index price is collected from Yahoo Finance and is transformed into log return before making analysis. The whole data of size 1510, spanned from 04 Jan. 2010 to 31 Dec. 2015, is used in the experiment to check the predictive capability and reliability of the proposed models. The training data of size 1010, from 04 Jan. 2010 to 8 Jan. 2014, is taken for the estimation and the rest 500 points spanned from 09 Jan. 2014 to 31 Dec. 2015 is reserved for forecasting as testing data.

The graphs show the stock prices and log returns of S&P-500 and Nasdaq-100.

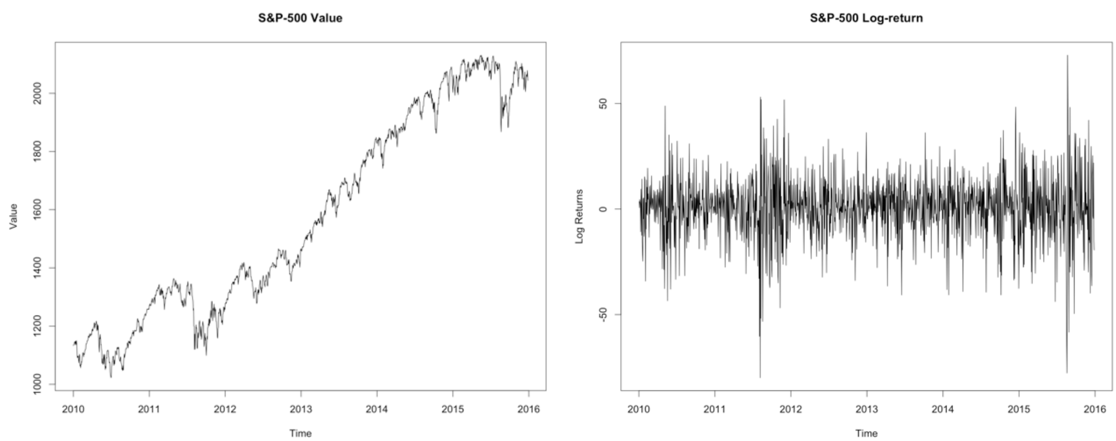


Fig.1: Stock prices and log returns of S&P-500

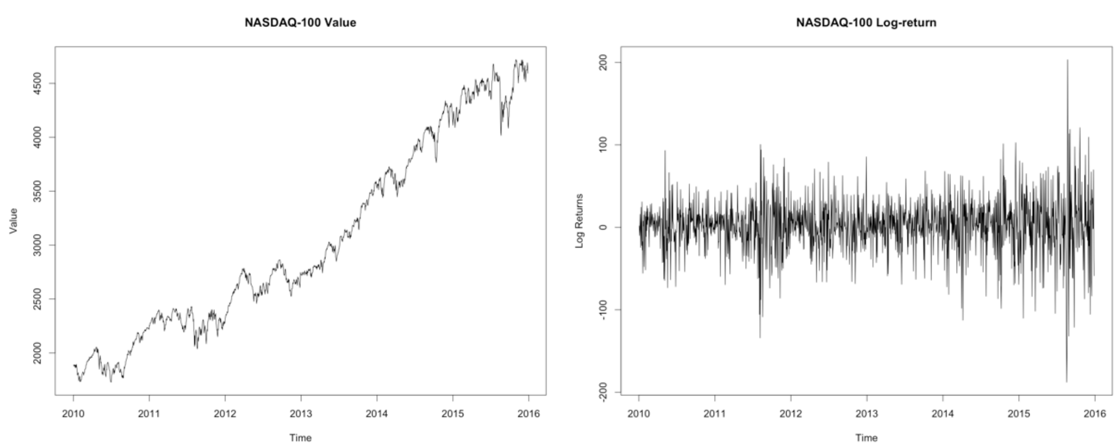


Fig.2: Stock prices and log returns of NASDAQ-1000

3. GARCH and SVR Models

GARCH models are an important time series models in financial data. This is due to its ability to capture many of the empirically stylized facts of financial time series, such as time-varying volatility, persistence and volatility clustering. In this project, we used GARCH (1,1) that provides a simple representation of the main statistical characteristics of a return series for a wide range of assets. The model is as below.

$$y_t = \mu + \varepsilon_t, \varepsilon_t = \sigma_t z_t$$
$$\sigma_t^2 = \omega + \beta\sigma_{t-1}^2 + \alpha\varepsilon_{t-1}^2$$

Here, y_t is the log daily return of stock price at time t; σ_t^2 is variance of stock price at time t; ω, α, β are parameters in the model ensure the positivity of conditional variance. Note that $\sigma_t^2 = E[\varepsilon_t^2|\mathcal{F}_{t-1}] = \hat{\varepsilon}_t$. By Bollerslev (1986), the conditional variance of ε_t^2 is given as $\varepsilon_t^2 = \omega + (\beta + \alpha)\varepsilon_{t-1}^2 + w_t - \beta w_{t-1}$, where $w_t = \varepsilon_t^2 - \hat{\varepsilon}_t^2 = \varepsilon_t^2 - \sigma_t^2$. Then, we can get one step ahead forecast is $\sigma_t^2 = \omega + \alpha + \beta\sigma_t^2 + \varepsilon_t^2$.

From the equations above, the corresponding GARCH model can be formulated as following,

$$y_t = h(y_{t-1}) + \varepsilon_t$$
$$\varepsilon_t^2 = f(\varepsilon_{t-1}^2, w_{t-1}) + w_t$$

For the GARCH model, it is essential to estimate the log return and variance of stock price. Let S_t be stock price at time t. Then $y_t = \ln(S_t/S_{t-1})$ denote the log daily return of stock at time t. Also we need to measure the volatility (variance) of stock price. With the time series being persistent, we decided to measure σ_t^2 as a moving average of the contemporaneous and M lagged squared returns around each time point in the in sample set, that is

$$\sigma_{M,t}^2 = \frac{1}{M}\sum_{j=1}^{M}(y_{j,t} - \bar{y})$$

where $\sigma_{M,t}^2$ is the variance of log return in M-days before time t; $\bar{y}$ is M-days log return average.

This project, we intend use the support vector regression (SVR) instead of MLE for estimating GARCH model. For SVR, it can be written as a convex optimization problem,

$$\min_{\omega,b,\xi_i,\xi_i^*}\left\{\frac{1}{2}\|\omega\|^2 + C\sum_{i=1}^{n}\left(\xi_i + \xi_i^*\right)\right\}$$

Subject to

$$y_i - \omega^T\phi(x_i) - b \le \varepsilon + \xi_i$$

$$\omega^T \phi(x_i) + b - y_i \le \varepsilon + \xi_i^*$$

where x is the vector of features, y is the response and C is regularization parameter. This optimization problem can be solved in a dual formulation and it provides the way for defining nonlinear SVR, using Lagrange Multipliers.

$$L = \frac{1}{2}\|\omega\|^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*) - \sum_{i=1}^{n}(\eta_i \xi_i + \eta_i^* \xi_i^*) \\ - \sum_{i=1}^{n}\alpha_i(\varepsilon + \xi_i - y_i + \omega^T \phi(x_i) + \mathrm{b}) \\ - \sum_{i=1}^{n}\alpha_i^*(\varepsilon + \xi_i^* + y_i - \omega^T \phi(x_i) - \mathrm{b})$$

Subject to

$$\alpha_i, \alpha_i^*, \eta_i, \eta_i^* \ge 0$$

The optimality will achieve when the partial derivatives of the Lagrange function for the primal variables be equal to zero. Then we can express the regression as following,

$$y(x) = \sum_{i=1}^{n}(\alpha_i - \alpha_i^*)\, K(x_i, x) + \mathrm{b}$$

We choose our kernel for the inner product as following table.

Linear:	$K(x_t, x) = x_t^T x$
Radial basis function: (Gaussian)	$K(x_t, x) = \exp(\frac{-\|x - x_t\|^2}{2\sigma^2})$
sigmoid:	$K(x_t, x) = \tanh(a x_t^T x + r)$

Table 1: Kernel function

In this project, we applied recurrent algorithm for SVR. Below is the illustration of recurrent algorithm of SVR for modeling and forecasting GARCH model.

Step 1: Fit the return y_t in the full sample period N, using $y_t = \mu + \varepsilon_t$ to obtain the residual $\varepsilon_1, \varepsilon_2, \ldots\ldots \varepsilon_N$.

Step 2: Recursively run the recurrent SVR for squared residuals $\varepsilon_1^2, \varepsilon_2^2, \ldots\ldots \varepsilon_N^2$, with updating by $\varepsilon_t^2 = f(\varepsilon_{t-1}^2, w_{t-1}) + w_t$, to obtain n one-step-ahead forecasted volatilities:

1^{st} sample: $t = 1, \ldots N_1 \rightarrow \hat{\varepsilon}_{N_1+1}^2$;

2^{nd} sample: $t = 1, \ldots N_1 + 1 \rightarrow \hat{\varepsilon}_{N_1+1+1}^2$;

…………

n^{th} sample: $t = 1, \ldots N_1 + n - 1 \rightarrow \hat{\varepsilon}_{N_1+n}^2$;

For each of n estimations, set the residuals of w_{t-1} to be zero at the first time in the Step 2, and then run the feed-forward SVR to obtain estimated residuals. Using the estimated residuals as new w_{t-1} inputs, this process can be carried out repeatedly until the stopping criterion is satisfied.

4. SVR Model Parameter

In the empirical model, there are three parameters (M, C, γ) that need to be determined. M is smoothing parameter; C is regularization parameter; γ is kernel coefficient parameter. We set $C = 1000$, $\gamma = 10^{-6}$ and have $M = 1 \sim 10$, get following RMSE result in training data.

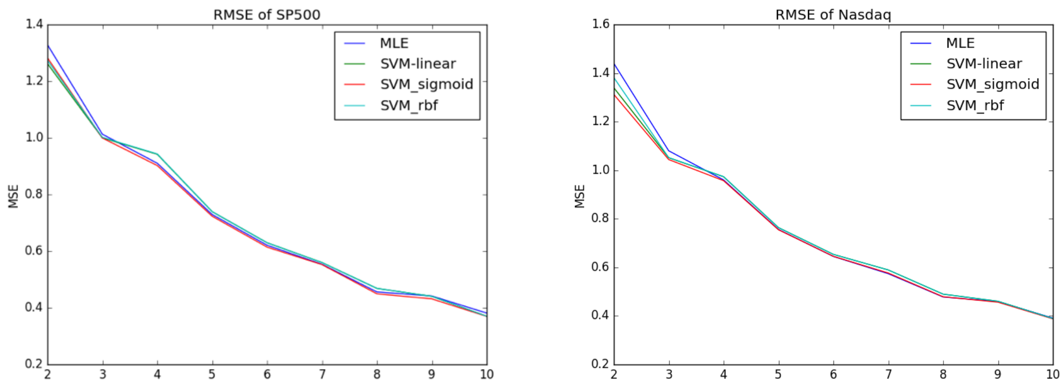


Fig.3: RMSE of S&P-500 and NASDAQ-100 in training data

In these graphs, the root-mean-square error (RMSE) between the forecasted volatility and true volatility decreases as M increases in both S&P 500 and Nasdaq train data. When M increasing, the volatility fluctuation will decrease and the volatility curve would be more smoothed. It's obvious that the larger M would lead to smaller RMSE. However, large M will lose information and short-time compute utility. People usually choose M=5 or 10 for computation. Here we use M=10 for more accuracy. So we set M=10, C=1000 and have $\gamma = 10^{-7} \sim 10^{-2}$. Also we set M=10, $\gamma = 10^{-6}$ and have C=0~10000.

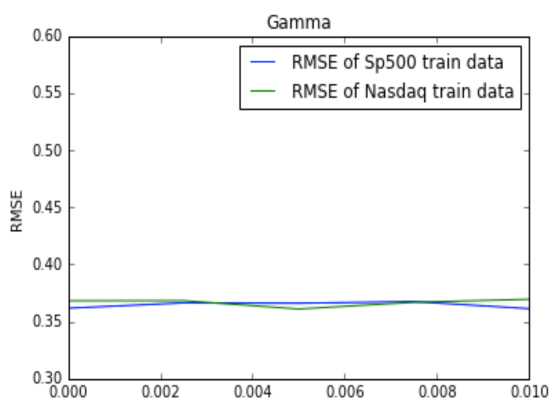


Fig.4: RMSE of S&P-500 and NASDAQ-100 for $\gamma = 10^{-7} \sim 10^{-2}$

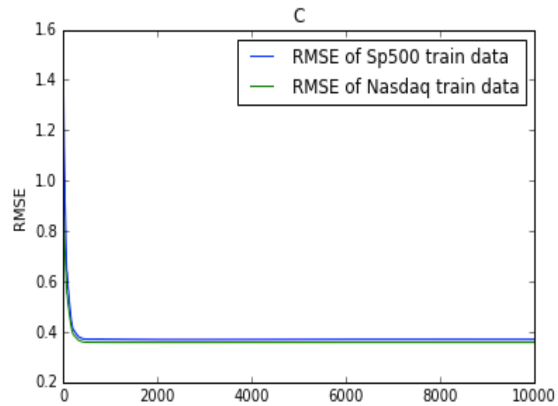


Fig.5: RMSE of S&P-500 and NASDAQ-100 for C=0~10000

In these graphs, as the C increases from zero to 100, the RMSE of train data decreases rapidly, and then convergence and stabilized at a level. However, changing gamma would not have significance effect on finding optimize minimum RMSE. A higher C would also have over fitting problems, and then for a relatively optimize model, we will choose $M = 10, C = 1000, \gamma = 10^{-6}$.

5. Empirical Results

For the original GARCH model, we use Maximum likelihood Estimation dealing with the train data and test data and compute errors for one-step forecasting. For the SVR model approach, we use three different kernel methods, radial basis function (rbf), polynomial and kernel. The MAE, MSE and RMSE of train and test data of S&P-500 and NASDAQ-100 are in the following table.

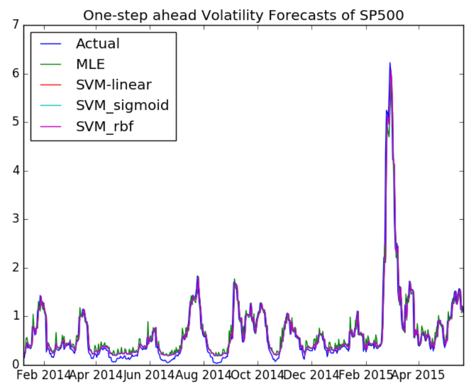
	MLE		SVM-rbf		SVM-linear		SVM-sigmoid	
	train	test	train	test	train	test	train	test
MAE	0.1954	0.1404	0.1651	0.1182	0.1679	0.1212	0.1679	0.1212
MSE	0.1454	0.0472	0.1366	0.0434	0.1370	0.0437	0.1370	0.0438
RMSE	0.3813	0.2174	0.3696	0.2082	0.3701	0.2091	0.3701	0.2092

Table 2: Comparative performance of the algorithms in S&P-500

	MLE		SVM-rbf		SVM-linear		SVM-sigmoid	
	train	test	train	test	train	test	train	test
MAE	0.2060	0.1650	0.1954	0.1618	0.2003	0.1655	0.2002	0.1654
MSE	0.1530	0.0888	0.1505	0.0897	0.1516	0.0900	0.1516	0.0900
RMSE	0.3910	0.2980	0.3879	0.2994	0.3894	0.3001	0.3893	0.3000

Table 3: Comparative performances of the algorithms in NASDQA-100

For S&P-500 index, all three kernels' errors are smaller than MLE method and rbf kernel has the best performance. However, in the NASDAQ-100 index, the SVM does not work well. Though three train errors are smaller than MLE method, all of the test errors are larger than MLE method. Since we use the optimize methods to get best linear regression parameters for the MLE method, the errors from MLE method have already been small enough. Thus, when we choose a relatively optimize parameters for SVM, the train errors and test errors are still not decrease significantly, or even getting increase. Then we can get one-step forecast of S&P-500 and NASDAQ-100 is the following.



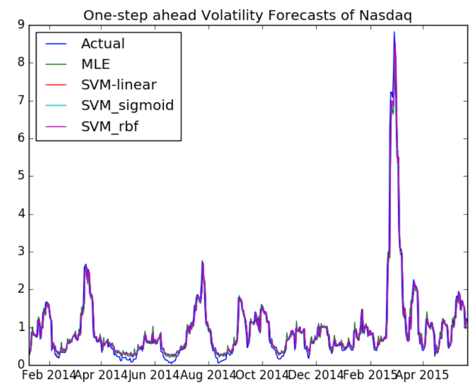


Fig.6: One-step forecast of S&P-500 and NASDAQ-100

In these graphs, the dark blue line is the actual value of the volatility. When the true volatility is small and fluctuate a lot, no matter which kind of methods, the predictions always larger than true value, and the errors are also significant. It also happens when volatility is large. Compared the MLE, three SVM methods' curves are smoother and represents the tendency of true volatility.

Now we compare our results with the result in Wolfgang Härdle's paper.

	MAE	MSE	RMSE
MLE	0.2376	0.1414	0.3760
SVR	0.0781	0.00057	0.0239
Reduced	67.13%	99.60%	93.64%

Table 4: Result from Wolfgang Härdle's paper

	MAE	MSE	RMSE
MLE	0.1404	0.0472	0.2174
SVR	0.1182	0.0434	0.2028
Reduced	15.81%	8.05%	6.72%

Table 5: Result from this project

In Wolfgang's paper, SVR method shows evident deduction of RMSE compared with MLE method. It deduced about 93.64% of RMSE. However, in our project, even we use the best method (SVR with rbf kernel), we only reduce 6.72% of RMSE. We use similar optimal algorithm and method, but get two distinguishable results. There might be two possible reasons that may lead to this difference, mistakes and over fitting.

For SVM method, we have already picked the relatively best parameters for forecasting. Theoretically, our result will be much closer to the ideal test results. In Wolfgang's paper we could not find the value for parameter in predictions, but no matter which value he used, RMSE could not deduce like this. The problem may come from the MLE method to forecast volatility. The MLE method he applied was not sufficient enough. Using a better optimize MLE method algorithm, he may get a smaller error and could be closer to the SVR result. Also, the SVR method he used may be over fitting to the test data. We test two data set S&P-500 and NASDAQ-100, and they get similar results. In Wolfgang's paper, he tested only S&P-100 and the error is so small, which reflects that his model may only work in this data.

After reading other papers that using SVM for GARCH model, we compared the traditional GARCH result and SVM method. SVM approach indeed works better than MLE, but it did not have significant improvements. Volatilities' randomness may lead this problem. SVM could deal with deterministic problem, yet stock future price and volatility are random. Though, by the help of smoothing parameters, the errors could decrease, it will lose some information for short-term volatility.

6. Conclusion

In our project, we use support vector machine based on GARCH to forecast volatility of S&P-500 and NASDAQ-100. The experimental results suggest that the SVM could be an improvement for traditional GARCH prediction, but it may not be efficient as we expected. The long-term smoothing volatility would have less prediction error in SVM, but it will also need more intuitive explanation. There is still much room for improvement SVMs with respect to forecasting financial time series. There will be new algorism and better parameters selection.

Reference

[1] Anzai, Y. (2012). *Pattern Recognition & Machine Learning*. Elsevier.

[2] Chen, S., Jeong, K., & Härdle, W. K. (2008). *Support vector regression based GARCH model with application to forecasting volatility of financial returns* (No. 2008, 014). SFB 649 discussion paper.

[3] Ou, P., & Wang, H. (2010). Predict GARCH Based Volatility of Shanghai Composite Index by Recurrent Relevant Vector Machines and Recurrent Least Square Support Vector Machines. *Journal of Mathematics Research*, *2*(2), 11.

[4] Pérez-Cruz, F., Afonso-Rodriguez, J. A., & Giner, J. (2003). Estimating GARCH models using support vector machines. *Quantitative Finance*, *3*(3), 163-172.

[5] Ruppert, D. (2011). *Statistics and data analysis for financial engineering*. New York, NY: Springer.