An algorithm using GARCH process , Monte-Carlo simulation and wavelets analysis for stock prediction

Giovanis, Eleftherios

3 September 2008

An algorithm using GARCH process , Monte-Carlo simulation and wavelets analysis for stock prediction

Abstract

This paper examines and presents a simple algorithm for prediction stock written in MATLAB code. We apply it to thirty stocks of the Athens exchange stock market . We obtain the stock returns and we would like to predict, not the actual price , but the sign of stock returns. The results are very satisfying while we predict the right sign for 25 out of 30 cases or else we have a success of 83.33%. The problem with the algorithm is that we don't have the ability to predict zero returns.

Keywords GARCH, wavelets, forecasting, Monte-Carlo, wavelet discrete transformation

I INTRODUCTION

We use GARCH model because financial time series are characterized by leptokurtosis , clustering volatility and leverage effects. GARCH (1,1) as a symmetry model can capture with success the volatility clustering , but no the leverage effects as EGARCH, GJR and others asymmetries GARCH model do. Wavelet analysis is capable of revealing aspects of data that other signal analysis techniques miss, and aspects like trends, discontinuities in higher derivatives and self-similarity. Furthermore wavelet analysis can often compress or de-noise a signal without appreciable degradation (Misiti et al., 2008).

The period we examine is not the same for all stocks, while some firms gain their entrance to Athens stock market at a later period. The oldest period we examine is September 4th of 1997 and we would like to forecast the stock returns on September 2rd of 2008. The data are daily.

II METHODOLOGY

GARCH

GARCH models characterize the conditional distribution of ε_t by imposing serial dependence on the conditional variance of the innovations.GARCH (1,1) , which process is much resemblance to the general ARMA process but it permits a more parsimonious description in many situations (Bollerslev, 1986). The general GARCH (p,q) model is

$$\sigma_t^{\ 2} = a_0 + \sum_{i=1}^{q} a_i u^2{}_{t-i} + \sum_{j=1}^{p} \beta_j \sigma^2{}_{t-j} \qquad (1)$$

In the algorithm we apply GARCH (1,1), even though we could apply also a apply GARCH (2,2). The GARCH (1,1) process is defined as:

$$\sigma_t^{\ 2} = a_0 + a_1 u^2{}_{t-1} + \beta_1 \sigma^2{}_{t-1} \qquad (2)$$

After GARCH (1,1) estimation we apply a Monte-Carlo simulation. The expected coefficient value can be defined as (Janke, 2002)

$$\overline{X} = \frac{1}{N} \sum_{i=1}^{N} f(X^i) \qquad (3)$$

, where X is the expectation value and the estimator $\overline{X}$ is a random number fluctuating around the theoretical expected value. We apply Monte-Carlo for all coefficients of GARCH process.

Wavelets

A wavelet is a waveform of effectively limited duration that has an average value of zero and it is the breaking up of a signal into shifted and scaled versions of the original wavelet. The filtering process of the wavelet decomposition is to emerge the signal through high and low pass filters, as we can see in the figure 1(Misiti et al., 2008), where we suppose that we have 1000 samples.

The original signal, S, passes through two complementary filters and emerges as two signals . The problem is that signals A and D are interesting but we get 2000 samples (Figure 2.a.) instead of the 2000 we had. For these reason we get cA and cD while we keep only one point out of two in each of the two 2000-length samples as we can see in the figure 2.b and this is the notion of downsampling

Figure 1. Filtering process

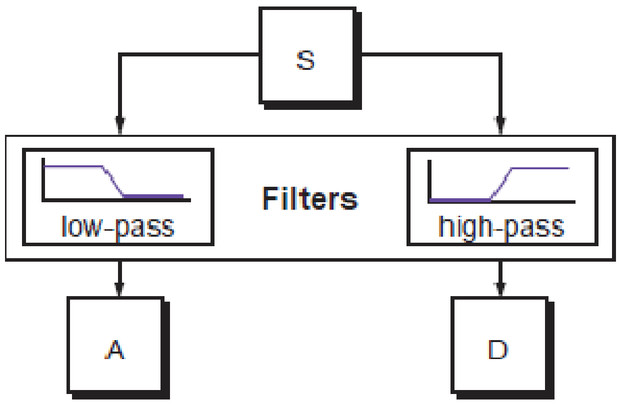
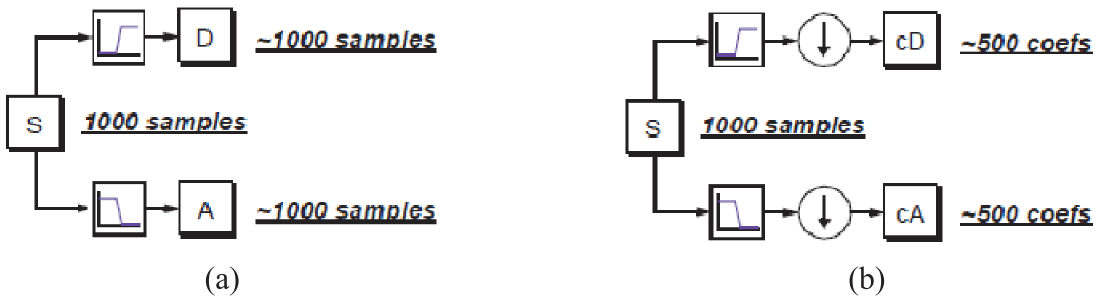


Figure 2. (a) Decomposition without downsampling, (b) Decomposition with downsampling



We apply the single-level decomposition, which is defined in MATLAB as 'dwt' using the wavelet 'db3'. The process in the figure 2.b produces the dwt coefficients. 'db3' wavelet belongs to the Daubechies Wavelets dbN where db is the name of the wavelet and N is the order. DWT is the discrete wavelet transformation for which the wavelets are discretely sampled.

The steps we apply are:

1. First we decompose the first sample of data (half data) with wavelets analysis we refer .
2. Then we estimate the GARCH(1,1) for the decomposing data.
3. The next step is to apply Monte-Carlo simulation for the GARCH estimating coefficients
4. We apply a random permutation and an algorithm to forecast the sign of the stock return in the following day.
5. We repeat steps 1-4 for the second sample of data (the remaining half data).

Finally we apply Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) which are defined (Greene 2002) as :

$$MAE = \frac{1}{T}\sum_{i=1}^{T} | y_i - \hat{y}_i | \quad (4)$$

$$RMSE = \sqrt{\frac{1}{T}\sum_{i=1}^{T} (y_i - \hat{y}_i)^2} \quad (5)$$

, where T is the number of the forecasting stock returns y_i is the actual stock return and $\hat{y}_i$ is the forecasting stock return.

III RESULTS AND FORECASTING PERFORMANCE

Table 1. Forecasting stock returns with the three algorithms for thirty stocks

Firm	Actual stock returns	Forecasting stock returns	Firm	Actual stock returns	Forecasting stock returns
Coca-Cola	-0.004802	-0.004039	Thessaloniki water supply and sewerage	0.019934	0.027205
Folli-Follie	0.005115	0.019343	Electronics ATHENS	0.000000	-0.14571
Microland	-0.220449	-0.044685	Heracles	0.011447	0.0020978
Multirama	0.026956	0.072462	INTRAKOM	0.006116	0.0025032
SATO	0.028171	0.007455	Karelias	0.034665	0.049449
AEGEK	0.000000	-0.004643	METKA	-0.006309	-0.004041
Atlantik	0.028820	0.017835	Minoan Lines	0.027505	0.005998
DIAS	-0.015038	-0.010208	Loulis group	0.017826	0.014485
Eurofarm	0.000000	-0.050065	Mytilineos	0.022757	0.078597
National bank of Greece	0.057454	0.000477	OPAP	-0.001669	-0.012630
EKTER	0.009662	0.040433	OTE	0.005435	0.042003
ELVAL	0.012048	0.037162	DROMEAS	0.057987	0.022979
Hellenic Sugar	0.007463	0.043324	Sanyo Hellas	0.000000	0.000255
Bank of Greece	-0.004823	-0.011358	NAKAS	0.000000	0.020691
Hellenic Oils	0.004662	0.015515	Eurobank	0.031431	0.034841
MAE = 0.02880					
RMSE = 0.04860					

Our purpose is not to forecast the actual price but the sign of the stock return. Some studies suggest that strategies that are followed but this scheme are more more effective, than trying to predict the actual price

As we mentioned previous the estimating period is not the same for all stocks and the forecasting date is 2nd September of 2008. The only stock returns that we couldn’t predict is the zero returns, so this algorithm is not capable of finding this kind of stock returns. We present also the MAE and RMSE , but these measures , even seems to be satisfying, there are not particularly useful, while we don’t compare the forecasting performance of this algorithm , with the performance of other algorithm or method. Also these measures are not increasing significantly when obtaining the zero returns, while the MAE and RMSE of the 25 stocks that we predict their sign correctly are 0.025713 and 0.0432 respectively. So the differences are not significant.

Conclusions

We applied an algorithm for stock prediction with GARCH(1,1) process and wavelets analysis with discrete transformation. We test it in a random sample of 30 stocks in Athens stock market and we predicted the correct sign for the 25 out of 30 stock returns. The problem of this algorithm ,as we stated in the abstract , is the incapability of forecasting zero returns, at least in a sample of five stocks we applied it. This can be harmful, while when for example we predict a negative sign, then we are forced to sell the specific stock, while it wouldn't be necessary if we have zero returns in the following day. The cost is not of loosing money because of the zero returns , but of wasting time and there is also the transaction cost, even this could be low and not significant. Also the day after following stock returns can be positive so we have an opportunity cost.

References

Bollerslev T. 1986. Generalized Autoregressive Conditional Heteroskedasticity. *Journal of Econometrics*. 31: 307-27. Elsevier Science S.A.

Greene H.W. 2002. Econometric Analysis. Fifth edition. Prentice Hall. New Jersey. U.S.A. p.p. 113

Janke W. 2002. Statistical Analysis of Simulations: Data Correlations and Error Estimation. *John von Neumann Institute for Computing.* Julich, NIC Series. Vol. 10. pp. 423-445

Leung M.T. , Daouk H. and Chen A.S. 2000. Forecasting stock indices: a comparison of classification and level estimation models. *International. Journal of Forecasting* 16: 173-190

Misiti M., Misiti Y. Oppenheim G. and Poggi J.M. 2008. Wavelet toolbox 4 User's guide. MATLAB. *The MathWorks* pp. 1.12-1.13

APPENDIX

MATLAB Code: GARCH process with wavelets analysis

```
clc;clear;close all;

% Load input data

load ('c:\file.mat', '-mat');

% T is the number of the days

T=length(data);

% We consider a random permutation of days

ord = randperm(T);

% We consider half-days of our data for first simulation

firstsample=(T/2);

% We consider the remaining days for second simulation

secondsample=T-(firstsample);

% From firstsample days we consider  10 data randomly selected

firstpoint=10;

% From secondsample days we consider  10 data randomly selected

secondpoint=10;

% We consider the simulation parameters

N=32;         % length of input data

M=1;          % length of predicted data

% We decompose our data with function db3

[dD] = dwt(data(1:firstsample)  ,'db3');

% We define GARCH (1,1) process

[Kappa, Alpha, Beta] = ugarch(dD, 1, 1)
```

```
% We set the random number generator seed for reproducability
randn('seed', 10)
NumSamples = firstsample;
% We simulate the process with Monte Carlo
[U , H] = ugarchsim(Kappa, Alpha, Beta, NumSamples);
Y=U(:)
% Length of vector
V=1;
% From current day we extract firstpoint data randomly selected
currentprice = randperm(V+N-M);
currentprice= currentprice+N;
for j=1:firstpoint
        Y1    = currentprice(j);
        Y0    = Y1-N+1;
        p     = Y(Y0:Y1);
        p    = p(:);
        Y(1,:)  = p(1,:);
        pred                    = Y(Y1+1:Y1+M);
       end
end
% We decompose our data with function db3
 [dD1] = dwt(data(firstsample+1):(firstsample+secondsample)  ,'db3');

% We define GARCH (1,1) process
[Kappa, Alpha, Beta] = ugarch(dD1, 1, 1)
% We set the random number generator seed for reproducability
randn('seed', 10)
```

```
NumSamples = seconsample;
% We simulate the process with Monte Carlo
[J , H] = ugarchsim(Kappa, Alpha, Beta, NumSamples);
W=J(:)
% From current day we extract secondpoint data randomly selected
currentprice = randperm(V+N-M);
currentprice= currentprice+N;
for j= firstsample+1: firstsample+secondsample
        W1    = currentprice(j);
        W0    = W1-N+1;
        p     = W(W0:W1);
        p    = p(:);
        W(1,:)  = p(1,:);
        pred                  = W(W1+1:W1+M);
       end
end
```