2005 International Symposium on Nonlinear Theory and its Applications (NOLTA2005)
Bruges, Belgium, October 18-21, 2005

Estimation of GARCH-Type Models with Markov Switching using Genetic Programming and its Applications

Yoshikazu Ikeda† and Shozo Tokinaga‡

†Faculty of Economics
Shinshu University
3-1-1 Asahi, Matsumoto-shi, Nagano 390-8621 Japan
‡Graduate School of Economics
Kyushu University
6-19-1 Hakozaki, Higashi-ku, Fukuoka 812-8581 Japan
Email: ikeda@econ.shinshu-u.ac.jp, tokinaga@en.kyushu-u.ac.jp

Abstract— In this paper, we propose the method for the estimation of GARCH-type models with Markov switching using Genetic Programming (GP). In the method, we use Genetic Programming to estimate system equations, where the Likelihood is used to evaluate GP individuals. The method is applied to the estimation of GARCH models for known systems including Markov switching, and then applied to real world data.

1. Introduction

Financial time series is usually characterized by its variances (called volatilities), since they affect the evaluation of possessing assets and return on investments. Among various nonlinear model fitting methods , ARCH (Autoregressive Conditional Heteroskedasticity) and GARCH (Generalized ARCH) are developed to describe the model for the volatility. However, conventional GARCH type models usually postulate fixed functional form included in models, and it is not clear the fitted model is the best one.

In this paper, we propose the estimation of GARCH-Type models with Markov switching using Genetic Programming (GP). In the method, we approximate (estimate) system equations to describe the dynamics of GARCH models by using the GP. In the evaluation of GP individuals, we use the Likelihood indicating the degree which observed time series generate from the model which a GP individual expresses.

The method is applied to the estimation of GARCH models for known systems including Markov switching, and then applied to real world data.

2. Basic model

2.1. GARCH-type models

As is known, the ARCH model for time varying volatility is described as follows.

$$y_n = \epsilon_n = g(h_n, w_n) = \sqrt{h_n} w_n,\ w_n \sim N[0, 1.0^2], \quad (1)$$

$$h_t = \alpha_0 + \alpha_1 \epsilon_{t-1}^2 + ... + \alpha_p \epsilon_{t-p}^2. \quad (2)$$

Then, the generalization for the variance included in equations leads us to the GARCH model by changing the portion of $h(t)$ into.

$$h_n = \alpha_0 + \sum_{i=1}^{p} \alpha_i \epsilon_{n-i}^2 + \sum_{j=1}^{q} \gamma_j h_{n-j}. \quad (3)$$

Even more, as recently proposed, by adding a kind of state transition model to h_n process, we have the GARCH with Markov Switching as follows.

$$h_n = \alpha_s + \sum_{i=1}^{p} \alpha_i \epsilon_{n-i}^2 + \sum_{j=1}^{q} \gamma_j h_{n-j}, \quad (4)$$

$$\alpha_s = \alpha_0 + \alpha_n S_n, \quad (5)$$

where, S_n is the state variable obeying to the Markov process having the state transition with probabilities such as.

$$Pr[S_t = 1 | S_{t-1} = 1] = a, Pr[S_t = 0 | S_{t-1} = 0] = b. \quad (6)$$

3. Applying the GP for the approximation of system equations

3.1. Basics of the GP

The prefix representation is equivalent to the tree representation of arithmetic expressions. For example, we have the next prefix representation.

$$(3 \times x_1 - x_2) \times (x_3 - 4) \rightarrow \times - \times 3 x_1 x_2 - x_3 4. \quad (7)$$

To keep the consistency of genetic operations, the so-called stack count (denoted as *StcckCount* is useful. The *StackCount* is the number of arguments it places on minus the number of arguments it takes off the stack. The cumulative *StackCount* never becomes positive until we reach the end at which point the overall sum still needs to be 1. The basic rule is that any two loci on the two parents genomes can serve as crossover points as long as the ongoing *StackCount* just before those points is the same. The crossover operation creates new offsprings by exchanging sub-trees between two parents.

In this report, we calculate the Likelihood by the method shown in the following subsection, and use it as the fitness of individula of GP. The Likelihood indicate the degree which observed time series generate from the model which a GP individual expresses.

We iteratively perform the following steps until the termination criterion has been satisfied.

(Step 1)

Generate an initial population of random composition of the functions and the terminals of the problem (constants and variables).

(Step 2)

Execute each program (evaluation of system equation) in the population and assign it a fitness value using the fitness measure. Then, sort the individuals according to the fitness S_i.

(Step 3)

The operations are applied to the individuals chosen with a probability proportional to the fitness. Create new individuals (offsprings) from two existing ones by genetically recombining randomly chosen parts of two existing individuals using the crossover operation applied at a randomly chosen crossover point.

(Step 4)

If the result designation is obtained by the GP (the maximum value of the fitness become larger than the prescribed value), then terminate the algorithm, otherwise go to Step 2. We apply the mutation operations if necessary.

3.2. caluculation of the Likelyhood

In order to use for evaluation of an individual, we caluculate Likelyhood of individuals of GP according to the procedure shown below.

(1) Generation of initial value

Generate particle F_0 obeying to the initial probability distribution $p(h_0|y_0)$

$$F_0 \sim p(h_0|y_0). \tag{8}$$

h_0 is volatility and y_0 is observed time series at time 0.

(2)one-step ahead prediction

By using particle F_n obtained by equation 8, we calculate a prediction particle P_n of volatility at the next time as

$$P_n = \hat{f}(F_n, \mathbf{h}_{n-1}, \mathbf{y}_{n-1}). \tag{9}$$

Here,$\hat{f}$ is an estimated system function expressed by a individual of GP ,**h** and **y** are vectors of past volatility and time series respectively,$\mathbf{h}_n = (h_n, h_{n-1}, \cdots, h_{n-j})$ and $\mathbf{y}_n = (y_n, y_{n-1}, \cdots, y_{n-k})$

(3) caluculation of the Likelyhood

Then, we calculate the Likelyhood for each time step n as

$$\alpha_n = p(y_n|h_n = P_n) = \gamma(g^{-1}(y_n, P_n)). \tag{10}$$

Here, $\gamma()$ is the density function of a normal distribution $N(0, 1.0^2)$. function $g^{-1}()$ is the inverse function of the observation function.

(4) Modification of a particle

$$F_n = P_n. \tag{11}$$

By using the likelihood α_n of each time step, we can obtain the logarithmic likelihood as follows.

$$l(\theta) = \sum_{n=1}^{N} \log p(y_n|y_0, y_1, \cdots, y_{n-1}) = \sum_{n=1}^{N} \log(\alpha_n). \tag{12}$$

In the functional approximation using the GP, we use the logarithmic likelihood to evaluate the goodness (fitness) of the system approximation.

3.3. Extension to GARCH-type with Markov Switching

Then, we extend the GP procedure to the cases where models includes Markov switching. We assume whole time series is composed of connection of segments of time series which can be modeled by several different GARCH-type models (called category). By learning, for each category $i, i = 1 \sim N$ corresponing to one GARCH-type model is represented by a $Pool_i$ of GP individuals including various functional form to approximate category i. If we have a segment of time series $x(t)$ whose category is unknown, then we calculate the fitness of all of individuals. If the highest fitness is found in the individual belonging to category i, then we decide that the segment is classified to category i.

The overview of the system for feature description and clustering method is given as follows.

(1)Learning data

It is assumed that the time series data is stored and available in the system, each of which is divided into the same length. Moreover, it is assumed that the time series data used for learning process is available, each of which is accompanied with the cluster (category) to which the underlying time series is expected to be classified.

(2)Learning based on the GP

Then, we approximations of functional forms for each cluster corresponding to the individuals in pools. The set of individuals can approximate various generating models for the category. Individuals have relatively higher fitness of approximation are retained in the system, and are used for clustering.

(3)Calculation of fitness of individuals

After applying the Learning Phase, we calculate the fitness of individuals in the Clustering Phase for every individual stored in each pool $i(i = 1 \sim n)$ by adopting (fitting) the observed data $x(t)$ of underlying time series whose cluster is not known. In the Clustering Phase, we calculate the fitness f_i for every individual in every pool by fitting the observation $x(t)$ of time series with known cluster. Then,

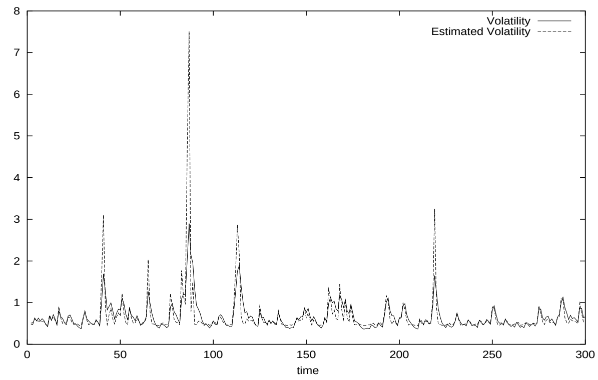

Figure 1: an example of estimated volatility

we estimate (determine) the cluster K of the time series by selecting the highest f_{max} among f_i where the individual i belongs to the K th pool.

(4)Clustering by using sliding window

In a long record of time series, it may happen the cased where the time point of the beginning and ending of the segment are not known beforehand. We assume that clustering system of the paper process the time series of the length T, and we call the length T as the length of window. Then, we move the starting point T_s and the ending point T_e of the window such as $T_s = k, T_e = T_s + T$ where k is integer. Then, the all of the set of windows (called as sliding windows) cover the whole time series by changing the starting point and ending point incrementally.

4. Applications

4.1. Estimating known GARCH models

In this section, we apply estimation method of this report to GARCH time series. The targeted GARCH model to be estimated is given as

$$h_n = 0.2 + 0.2\epsilon_{n-1}^2 + 0.1\epsilon_{n-2}^2 + 0.2h_{n-1} + 0.2h_{n-2}. \quad (13)$$

The parameters for simulation studies are given as follows. Number of individuals:500, Maximum length of array of individuals:50, Operators in individuals:$+, -, \times, abs$, Variables in individuals:$\epsilon_{n-1}, epsilon_{n-2}, h_{n-1}, h_{n-2}$.

After 100 generations of GP procedure, we obtain following estimation for the system equation.

$$h_n = 0.289 + 0.275\epsilon_{n-1}^2 + 0.401h_{n-1} - 0.171h_{n-2}. \quad (14)$$

The estimated volatility are shown in the figure 1.

4.2. Estimating known GARCH models with Markov switching

Then, we apply the estimation method of GARCH-type model where h_n is basically the same model as in previous section, but also includes Markov switching denoted as follows.

$$\alpha_s = \alpha_0 + \alpha_t S_t = 0.1 + 0.1S_t, \quad (15)$$

$$Pr[S_t = 1|S_{t-1} = 1] = a, Pr[S_t = 0|S_{t-1} = 0] = b. \quad (16)$$

Then, we estimate system dynamics of the volatility. The parameters of GP for simulation studies are the same as in previous example.

Since the purpose of this simulation is verifying the classification possibility of the category of time series, it is assumed that time points when the state changes is known and also assumed learning data for GP about $\alpha_s = 0.1 and 0.3$ can be used, respectively.

We assume that Data length is 1000, state changes take place at 300 ($0.1 \to 0.3$) and 600 ($0.3 \to 0.1$), Window size T is 50, starting point T_s is moved 1 time step at 1 time. GP individual pool learning from data with $\alpha = 0.1$ is labeled A, and learning from data with $\alpha = 0.3$ is labeled B.

If the Likelyhood calculated by a individual of pool A is large, it will be estimated that data segment belongs to category A. If Likelyhood of pool B is large, it will be estimated that it belongs to category B.

Classification probability is shown in table 1. 0.1 ,0.2, Mixture in the table are classes of data which are classified. Mixture expresses the data with which $\alpha = 0.1$ and $\alpha = 0.3$ are intermingled. A(0.1) and B(0.3) show the classification result. As a result, the probability to classify the data with $\alpha = 0.1$ with Category A(0.1) is 0.79, the data with $\alpha = 0.3$ with Category B(0.1) is 0.84. These are probabilities classified correctly.

4.3. Estimation of GARCH-type model for real data

We apply prediction method of this report to Nikkei 225 stock prices. In real world data, it is seen that the type

Table 1: Classification probability

	0.1	0.2	Mixture
A(0.1)	0.79	0.16	0.56
B(0.2)	0.21	0.84	0.44

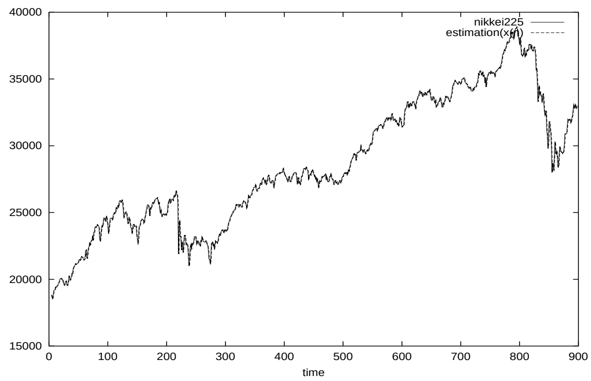

Figure 2: estimated mean value x_n of nikkei225

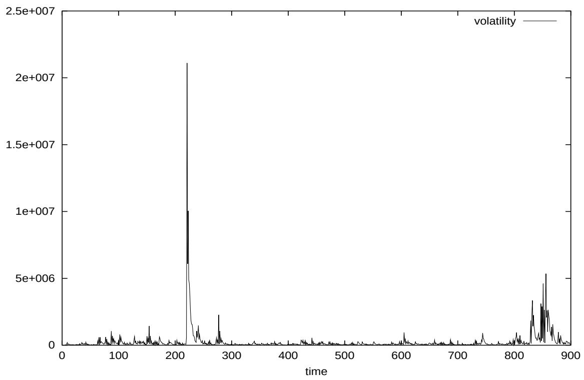

Figure 3: estimated volatility h_n of nikkei225

of model is not restricted to ordinary GARCH model, but the model is allowed to include multiplication of variables and other combinations. The GP procedure is applicable without any change of algorithm for these cases.

The parameters of GP for simulation studies are the same as in previous example.

The system equations estimated by GP are shown as follows.

$$x_n = f1(y_{n-1}, y_{n-2}, y_{n-3}, y_{n-4}), \quad (17)$$

$$h_n = f2(h_{n-1}, h_{n-2}, e_{n-1}^2, e_{n-2}^2). \quad (18)$$

$$y_n = x_n + \sqrt{h_n} w, \ w \sim N(0.0, 1.0^2) \quad (19)$$

Here, x_n is mean value of nikkei225 and caluculated function $f1$ which is estimated by GP, h_n is the volatirity , y_n is price of nikkei225 , and $e_n = x_n - \hat{x}_n$.

The result of estimation x_n and h_n is shown in figures 2 and 3. Volatility is also large when stock price change is large. Estimated functions $f1, f2$ are as follows.

$$x_n = 1.182 + y_{n-1} + 0.1621 y_{n-3} - 0.2041 y_{n-4}, \quad (20)$$

$$h_n = 1.537 + h_{n-2} + 1.4588 e_{n-1}^2. \quad (21)$$

5. Conclusion

In this paper, we showed the method for the estimation of GARCH-type models with Markov switching using the GP. The method was applied to the estimation of GARCH models for known systems including Markov switching, and real world data.

For further works, it is necessary to extend the method to more general type of random noise using the GP.

References

[1] R.F.Engel, "Autoregressive conditional heteroscedasticity with estimates of the variance of UNited KIngdum inflation ," Econometrica, vol.50,pp.987-1008,1982.

[2] T.Bollerslev, "Generalized autoregressive conditional heteroscedasticity," Journal of Econometrica,vol.31,pp.307-327,1986.

[3] G.Kitagawa, "Monte Carlo filter and smoother for non-Gaussian nonlinear state space models", Journal of Computational and Graphical Statistics, vol.5,no.1, pp.1-25, 1996.

[4] G.Kitagawa and H.sato, "Analysis of time series with time varying volatilities based on the generalized state space models," IMES Discussion Paper,No.98-J-22,1998.

[5] Y.Ikeda and S.Tokinaga, "State estimation from non-stationary time series based on functional approximation using the Genetic Programming and Monte-Carlo filters and its applications," Technical Report of IEICE, vol.CAS2004-104, pp.43-48,2005.

[6] Y. Ikeda and S.Tokinaga, "Approximation of chaotic dynamics by using smaller number of data based upon the genetic programming", Trans. IEICE, vol.E83-A,no.8, pp.1599-1607, 2000.

[7] X.Chen and S.Tokinaga, "Synthesis of multi-agent systems based on the co-evolutionary genetic Programming and its applications to the analysis of artificial markets", Trans. IEICE,vol.E86-A,vol.10,pp.1038-1048,2003.

[8] Y.Ikeda and S.Tokinaga, "Chaoticity and fractality analysis of an artificial stock market by the multi-agent systems based on the co-evolutionary Genetic Programming", Trans. IEICE,vol.E87-A, no.9,pp.2387-2394,2004.

[9] J.R.Koza,Genetic Programming,MIT Press, 1992