Using Genetic Programming to Model Volatility in Financial Time Series

Shu-Heng Chen
AI-ECON Research Group
Department of Economics
National Chengchi University
Taipei, Taiwan 11623
TEL: 886-2-9387308
FAX: 886-2-9390344
E-mail: chchen@cc.nccu.edu.tw

Chia-Hsuan Yeh
AI-ECON Research Group
Department of Economics
National Chengchi University
Taipei, Taiwan 11623
TEL: 886-2-9387308
FAX: 886-2-9390344
E-mail: g3258501@grad.cc.nccu.edu.tw

Abstract

In this paper we propose a *time-variant* and *non-parametric* approach to estimating *volatility*. This approach is based on *recursive genetic programming* (**RGP**). Here, volatility is estimated by a class of non-parametric models which are generated through a *recursive competitive process*. The essential feature of this approach is that it can estimate volatility by simultaneously detecting and adapting to structural changes. Thus, volatility is estimated by taking possible structural changes into account. When **RGP** discovers structural changes, it will quickly suggest a new class of models so that overestimation of volatility due to ignorance of structural changes can be avoided. In this paper, the idea is tested by using Nikkei 225 and S&P 500 as an example.

Key Words: Structural Changes, Model-Specific Structural Changes, Model-Free Structural Changes, Financial Volatility, Recursive Genetic Programming, Improvement Sequence

1 Motivation and Introduction

The paper is motivated by the cross-fertilization of financial engineering and artificial intelligence. From the aspect of financial engineering, there is a tendency to search for a more general or adaptive technique to modeling *volatility*. This tendency is quite important because there is no absolutely objective definition of volatility. In particular, in light of the recent advances in nonlinear models, volatility is a *model-dependent* concept, i.e., different model imply different degrees of volatility. Refenes, Burgess and Bentz (1996) used Monte Carlo simulations to deliver a very interesting message. They showed that when volatility actually follows a nonlinear dynamic stochastic process and can be traced by the nonlinear models such as *artificial neural networks*, then using the *simple* standard deviation, based on the historical data, to estimate volatility can not only mislead investors to overestimate volatility but also generate a lower rate of return due to the resulting non-optimal portfolio. Put in another way, the estimated efficient frontier is located in the interior of the potential efficient frontier.

In financial econometrics, volatility has been modeled with *linear ARCH* or *GARCH* processes. It is not until recently that researchers started to work with *nonlinear GARCH* models. For example, Olmeda and Fernandez (1996) suggests the following general univariate model (*nonlinear ARMA with a nonlinear GARCH process*) to estimate volatility.

$$y_t = f(y_{t-1}, ..., y_{t-p}, \epsilon_{t-1}, ..., \epsilon_{t-q}) + \epsilon_t, \quad (1)$$

$$\epsilon_t = g_t \mu_t, \quad (2)$$

where μ_t is a standard normal random variables and

$$g_t = h(y_{t-1}, ..., y_{t-p}, \epsilon_{t-1}, ..., \epsilon_{t-q}, g_{t-1}, ..., g_{t-r}). \quad (3)$$

While both Refenes et al (1996) and Olmeda and Fernandez (1996) suggest a general class of models to define and estimate volatility, functions f and h assumed in these studies are basically *time-invariant*. In other words, they are *not adaptive*. To consider models in an adaptive fashion, an appropriate technique which can estimate volatility by simultaneously taking structural changes into account is needed.

While detecting structural changes of any observed time series data has its long history in statistics and

econometrics, there are still many issues left to be settled. Recent extensive use of nonlinear and nonstationary time series models seems to make many of these issues even more obscure. For example, it is difficult to distinguish a nonstationary $I(1)$ time series from a stationary time series with structural breaks (Perron, 1989). Also, it is likely to detect a *spurious structural change* for the fractionally integrated processes of order $I(d)$, where $0 < d < 0.5$ is the fractionally differencing parameter (Kuan and Hsu, 1996). Moreover, it should not be surprised by a generalization of these findings, i.e., *detecting structural changes of observed time series is a daunting task when the model class is extended from linear to nonlinear and stationary to nonstationary models.* In this case, a reflection upon *structural changes* seems to be crucial.

Chen and Yeh (1997) proposed a non-parametric approach to the definition and detection of *structural changes*. Their approach is based on *recursive genetic programming* (**RGP**). **RGP** gives us a *model-free notion* of structural changes. This notion, as opposed to the conventional *model-specific* notion of structural changes, has the advantage of being insensitive to the small perturbation of the reference model. As a consequence, it can easily avoid the problem of *spurious structural changes*. In this paper, we shall employ this **RGP** to model financial volatility.

2 Recursive Genetic Programming

The **RGP** used in this paper is an extension of Koza's genetic programming (**BGP**) (Koza, 1992). It is composed of three key parameters, namely, *the size of the major sample* (the width of the sliding window) (n_1), *the size of the marginal sample* (n_2) and *the size of the representative sample* (q). Given the pair (n_1, n_2), we can construct a sequence of samples $S_1, S_2, ...$ described as follows (Figure 1). The first major sample S_1 is composed of the first n_1 observations of a time series, i.e., $\{x_i\}_{i=1}^{n_1} \equiv \{x_1, x_2, ..., x_{n_1}\}$. The second major sample S_2 is the alteration of S_1 by adding the first marginal sample M_1 to S_1 and deleting the first n_2 observations from S_1, i.e., $S_2 = \{x_i\}_{i=n_2+1}^{n_1+n_2}$. Similarly,

$$S_j = \{x_i\}_{i=(j-1)n_2+1}^{n_1+(j-1)n_2}. \tag{4}$$

In other words, $S_j (j = 1, 2, ...)$ is a fixed-size sliding window of the original time series $\{x_i\}$. Given this sequence of the major samples, **BGP** is applied to this sequence of sample in the following manner. Firstly, **BGP** is applied to *learn* the underlying regularity of S_1

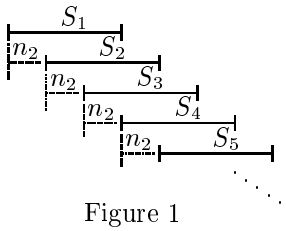


Figure 1

as usual (Chen and Yeh, 1996a,b). When the training is over, the population of the last generation GP_n^1 is kept and is used as the *initial generation* for the second major sample S_2, i.e.,

$$GP_1^2 \equiv GP_n^1. \tag{5}$$

The fitness of GP_1^2 is then evaluated based on the fitness function F. In terms of evaluating learning performance, the fitness function is usually residual-based, i.e., F is a function of the residual (ξ) or $F \equiv F(\xi)$. For example, the function $F(\xi)$ considered in this paper is *the sum of squared errors*. The fitness of each *GP-tree* of the population is then ranked so that $F_1 \leq F_2 \leq ....$ The best q GP trees are then selected as a representative sample for GP_1^2. Call this representative sample Q_1^2. The average fitness of Q_1^2 is then calculated and denoted by

$$\bar{F}_1^2 = \frac{\sum_{i=1}^{q} F_i}{q}, \tag{6}$$

where $F_1 \leq F_2 \leq ... \leq F_q$.

By **BGP**, we then start another n-generation evolution on S_2. The population of the last generation GP_n^2 is kept. With the same procedure described above, we generate Q_n^2 and compute the $\bar{F}_n^2$. The difference D_2

$$D_2 = \overline{F}_1^2 - \overline{F}_n^2 \tag{7}$$

can then be considered an indicator of the improvement after n-periods' training. In this manner, we can generate a sequence D_k, k=1,2,..., and this sequence, called the *improvement sequence*, is an important statistic for us to detect structural changes. Due to the space limit, the interested reader is referred to Chen and Yeh (1996c) for details.

2.1 RGP and Volatility

The relation between **RGP** and volatility can be revealed by the *fitness function* ($F(\xi)$). Since $F(\xi)$ in this paper is chosen to be the sum of squared errors, $\sqrt{\frac{F(\xi)}{n_1}}$ is the volatility. However, since genetic programming is a population-based learning scheme, each GP-tree of the population has its own estimation of

volatility. Therefore, **GP** will not give us a single estimate of volatility $\sqrt{\frac{F(\xi)}{n_1}}$; instead, it will generate a population of it. Based on the ranking described above, they can be arranged in an increasing order, i.e.,

$$\sigma_1 \leq \sigma_2 \leq .... \tag{8}$$

where $\sigma_i = \sqrt{\frac{F_i(\xi)}{n_1}}$ and $F_1(\xi) \leq F_2(\xi) \leq ....$

Similarly, using the representative sample Q_n^j, we can have an estimate of volatility by simply taking the sample average,

$$\sigma_j = \frac{\sum_1^q \sigma_i}{q} \tag{9}$$

where σ_i is the volatility estimate of the ith GP-tree in the representative sample Q_n^j.

The distinctive feature of using **RGP** to estimate volatility is that it will take structural changes into account. Therefore, when the underlying structure experiences a certain change, **RGP** can detect it and, in the mean time, generate a population of volatility estimates under the new structure. As a consequence, we can avoid reliance on out-of-date knowledge and the problem of overestimation of volatility discussed in Refenes et al. (1996) can be avoided. In the next section, we shall exemplify the use of **RGP** with two financial datasets.

3 Data Description

The two financial datasets considered in this paper are taken from Chen and Tan (1996). These datasets are a small subsample of the stock indices S&P 500 and Nikkei 225. This small sample is selected by a *complexity* measure, namely, the *minimum description length principle* proposed by Rissanen (Rissanen, 1989). A brief description of the dataset is given in Table 1 and the time series of these samples are depicted in Figures 2-3.

The data of Nikkei 225 is very interesting. During the sample period, the rate of return R_t of Nikkei wandered over the range [-0.02, 0.02] for the first three quarters and then jumped around in a visually significant larger range in the last quarter (Figure 2). Compared with Nikkei 225, S&P 500 seems to be much more stable. Over the sample period, the R_t only walked randomly between -0.02 and 0.02. So, these two samples together provide us with a great opportunity to test the **RGP-based** notion of structural changes. Visual inspection suspects a structural change for the first sample but not the second one. However, by saying this, one must be wary of the fact that RGP is an adaptive cognitive system, so the seeming structural change might not be a real one for this system if the speed of change is slower than the speed of adjustment.

Table 1: Data Description

Country	Stock Index	Sample Period	Sample Size	MDL
U.S.	S&P 500	1/3/92 - 10/16/92	200	142.472
Japan	Nikkei 225	6/30/89 - 4/24/90	200	142.472

The last column "MDL" refers to the *minimum description length*. The number "142.472" indicated under this column is the maximal value that one can possibly have for a 200-bit string. In other words, the sample selected here is the most *complex* of the whole dataset. For details, see Chen and Tan (1996a).

4 Experimental Results

The design of RGP is given in Table 2. The three major parameters of **RGP** n_1, n_2 and q are set to be 50, 5 and 50. This version of RGP is then applied to the data described in Table 1. Ten simulations are conducted for Nikkei 225 (Simulations 1.1-1.10) and S&P 500 (Simulations 2.1-2.10) respectively. In each simulation, the improvement series $\{D_i\}_{i=1}^{29}$ is recorded. Given the number of observations, n_1 and n_2, it is clear that we have 29 major samples. The summary statistics of the series $\{D_i\}_{i=1}^{29}$ of each simulation are given in Table 3 (Nikkei 225) and Table 4 (S&P 500). These statistics include the mean, standard deviation and maximum value.

The mean of $\{D_i\}_{i=1}^{29}$ of Nikkei 225 ranges from 0.0004 (Simulation 1.3) to 0.0137 (Simulation 1.6). This range is not only higher but also wider than that of S&P 500, which ranges from 0.0001 to 0.0025. This difference also holds for the standard deviation (S.D.) of D_k series. For Nikkei 225, the S.D. ranges from 0.0007 (1.3) to 0.06053 (1.6), while it is only [0.0001, 0.0126] for S&P 500. If we look at the maximum improvement ever made, the range for Nikkei 225 is [0.0030, 0.3278] and that for S&P 500 is only [0.0004, 0.0680]. These results seem to suggest:

- There is more room for improvement with Nikkei 225 than there is with S&P 500 by using **RGP**, which implies that adaptive learning is more important for Nikkei 225 than it is for S&P 500. In other words, it pays more for the Tokyo stock broker to continuously monitor the movement of stock prices than it does for the New York stock broker.

- It is more likely to overestimate the volatility of

Figure 2 : The Time Series of NIKKEI 225

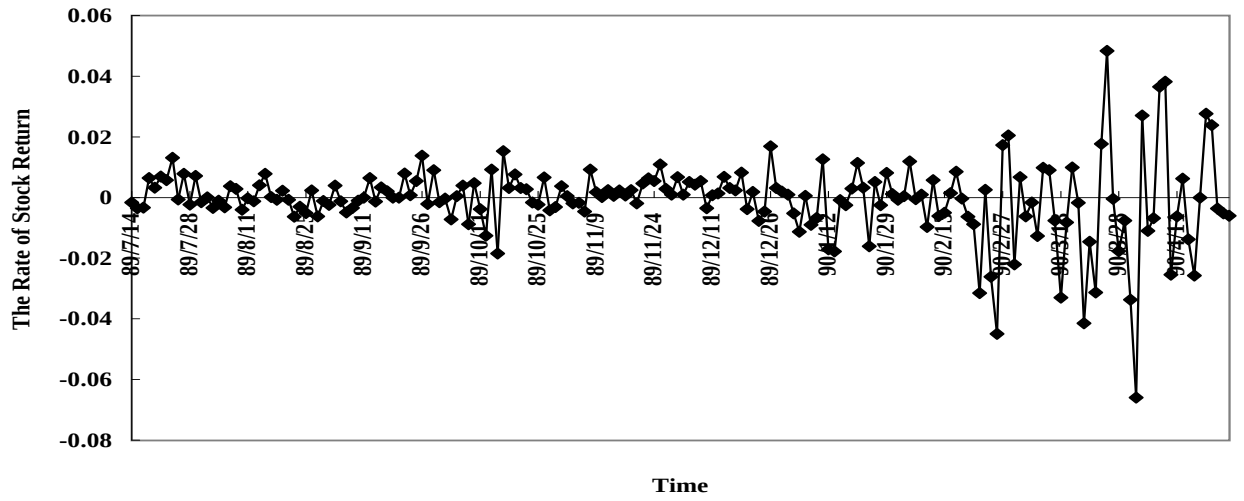


Figure 3 : The Time Series of S&P 500

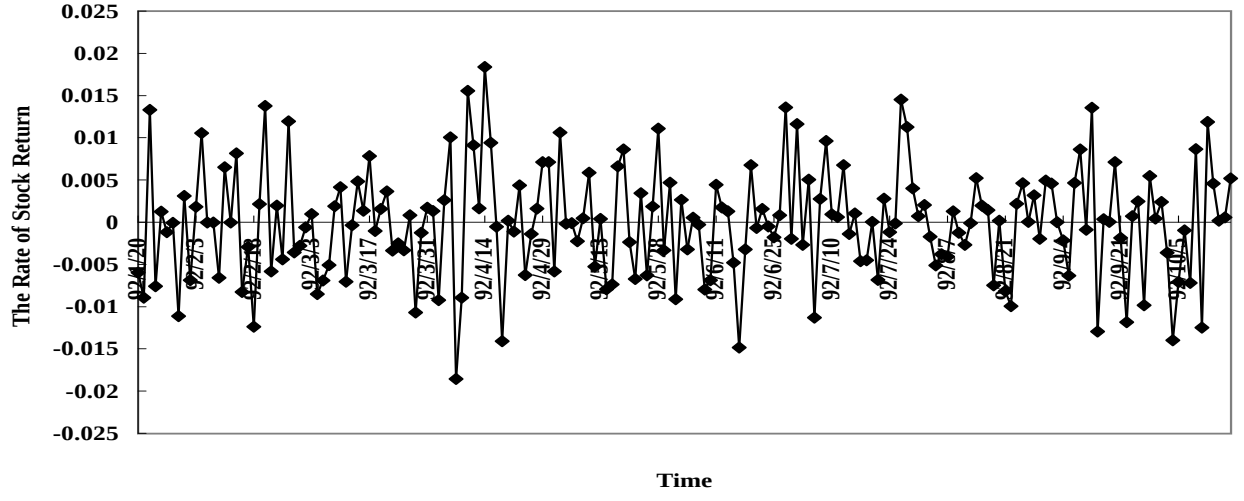


Figure 4.1 : The Improvement Sequence of NIKKEI 225 (Simulation 1.1)

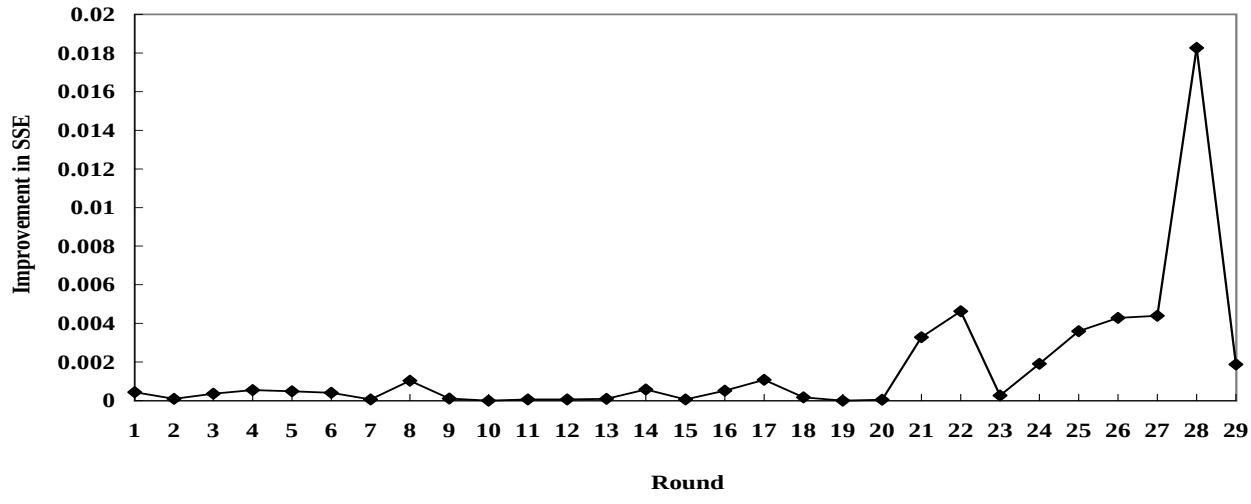


Figure 4.2 : The Improvement Sequence of NIKKEI 225 (Simulation 1.2)

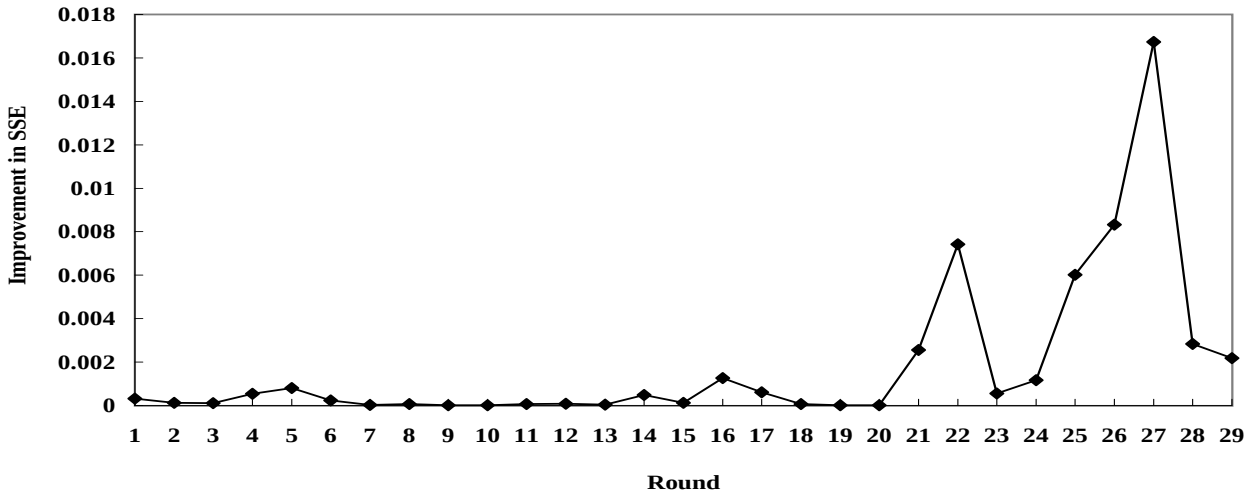


Figure 4.3 : The Improvement Sequence of NIKKEI 225 (Simulation 1.3)

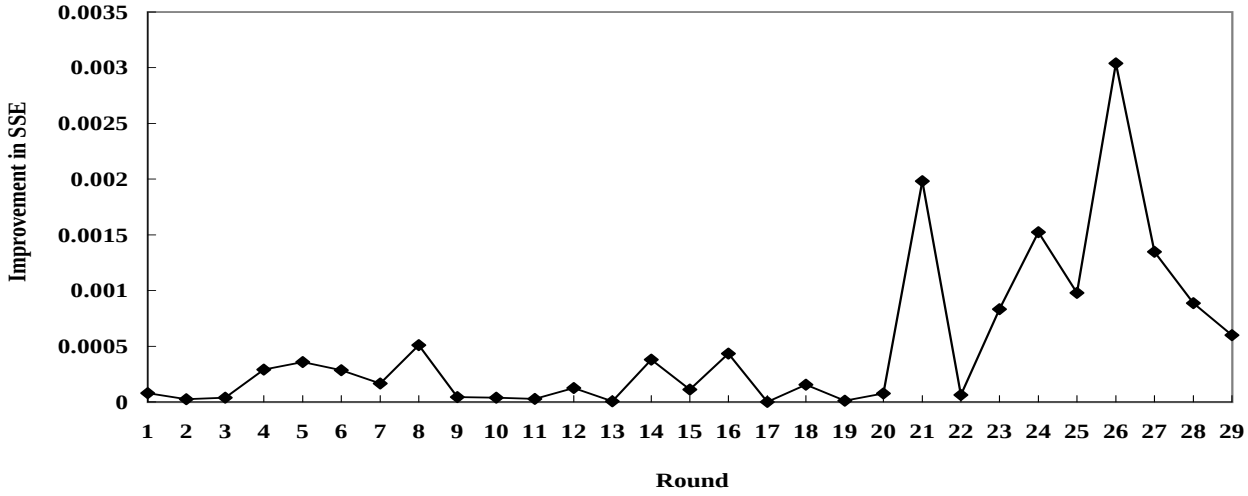


Figure 4.4 : The Improvement Sequence of NIKKEI 225 (Simulation 1.4)

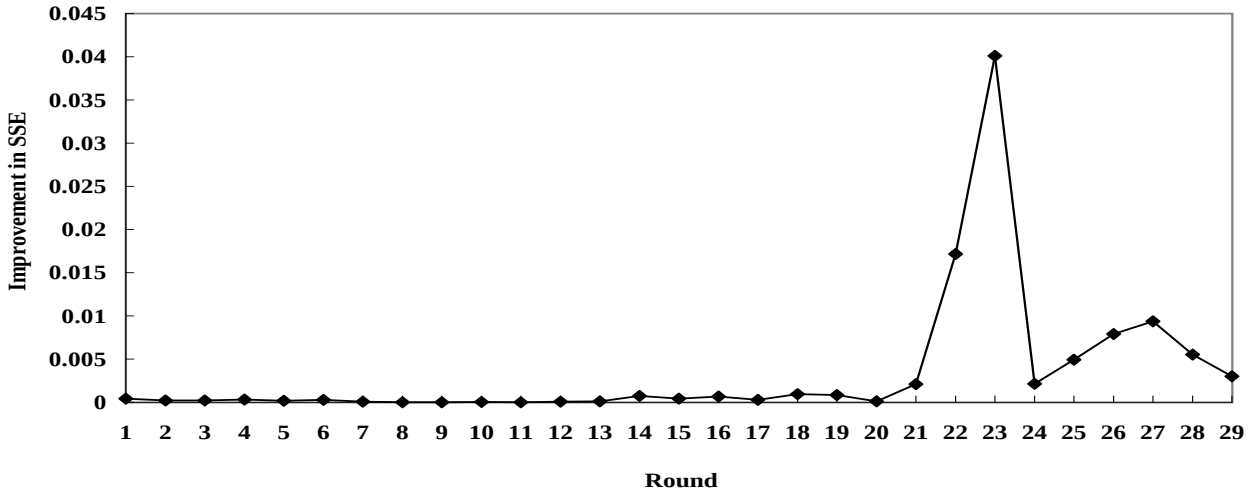


Table 2: Tableau for Recursive Genetic Programming

Population size (N)	500
Number of trees created by complete growth	50
Number of trees created by partial growth	50
Function set	$\{+, -, \times, \%, sin, cos, EXP, RLOG\}$
Terminal set	$\{R_{t-1}, R_{t-2}, \cdots, R_{t-10}, R\}$
Number of trees generated by reproduction	100
Number of trees generated by mutation	100
Probability of mutation	0.0033
Maximum length of the tree	17
Probability of leaf selection under crossover	0.5
Maximum number in the domain of Exp	1700
Criterion of fitness (F)	Sum of Squared Errors
Number of generations (n)	100
Size of the major sample (n_1)	50
Size of the marginal sample (n_2)	5
Memory size (r)	10
Size of the representative sample (q)	50

Nikkei 225 than that to do S&P 500 if we are using a time-invariant (fixed) model to do the estimating.

The performance of **RGP** can be further explored through the improvement sequence D_k. For this purpose, the improvement sequence of ten simulations for Nikkei 225 are also drawn (Figures 4.1-4.10). Some interesting properties are listed below.

- By the visual inspection of Figure 2, we can clearly identify an increase in the volatility of Nikkei 225 from February 1990 to April 1990. Interestingly enough, in all these ten simulations, RGP actually confirms this change. In all ten figures, there is a major hike at the end of the D_k series. The date of this major hike is almost consistently estimated by these ten simulations. It is identified as the 23rd sample (S_{23}) in Simulation 1.4, the 25th (S_{25}) in Simulation 1.9, the 26th (S_{26}) in Simulations 1.3, 1.7, and 1.8, the 27th (S_{27}) in Simulations 1.2, 1.5, the 28th (S_{28}) in Simulation 1.1 and the 29th (S_{29}) in Simulation 1.6.

- While for Nikkei 225 the peak consistently appears in the last one-fourth of the data, the truth does not hold for S&P 500. To see this, the improvement series of S&P 500 of Simulations 2.1, 2.2, and 2.3 are depicted in Figures 5.1-5.3. From these figures, we can see that peaks appear quite inconsistently among these simulations. Furthermore, the heights of most of these peaks are negligible as opposed to those of Nikkei 225.

- Another property of the improvement series D_k is that, after the peak, there is a dramatic fall in D_k. In other words, if S_i is detected as a structural change, then D_{i+1} is much smaller than D_i. This shows that once a structural change is detected, **RGP** will quickly adapt itself to the new environment by generating a new group of models; therefore, in the next period, possible room for improvement shrinks and D_{i+1} returns to the normal.

- The appearance of hikes can be explained by the property that RGP is an adaptive cognitive system. Once a change in volatility is detected, the system quickly adapts to the change and the improvement sequence go back to the normal. This feature makes RGP a very promising tool for the study of structural changes, in particular, the estimation of change points.

Table 3: Summary Statistics of the Improvement Series: Nikkei 225

Simulation	Mean	S.D.	Max
1.1	0.001681921	0.003515061	0.018270935
1.2	0.001820705	0.003628995	0.016730926
1.3	0.000496815	0.000705278	0.003038730
1.4	0.003382256	0.008014909	0.040116956
1.5	0.000832889	0.001763278	0.008795386
1.6	0.013713513	0.060539883	0.327866083
1.7	0.002209311	0.004606436	0.018777644
1.8	0.000766762	0.001359543	0.005966590
1.9	0.000620353	0.000993308	0.004279575
1.10	0.002076856	0.004201890	0.020429868

Table 4: Summary Statistics of the Improvement Series: S&P 500

Simulation	Mean	S.D.	Max
2.1	0.000627237	0.000545084	0.002628560
2.2	0.000151256	0.000136751	0.000490568
2.3	0.000133932	0.000165889	0.000557956
2.4	0.000155246	0.000160896	0.000592844
2.5	0.000271861	0.000203626	0.000666856
2.6	0.002595408	0.012603221	0.068080061
2.7	0.000418231	0.000205636	0.000744960
2.8	0.000355931	0.000333614	0.001237812
2.9	0.000248756	0.000237452	0.000808789
2.10	0.000177195	0.000213592	0.000709973

5 Concluding Remarks

In this paper, a new approach to estimating volatility is proposed. It is exemplified by two samples se-

Figure 4.5 : The Improvement Sequence of NIKKEI 225 (Simulation 1.5)

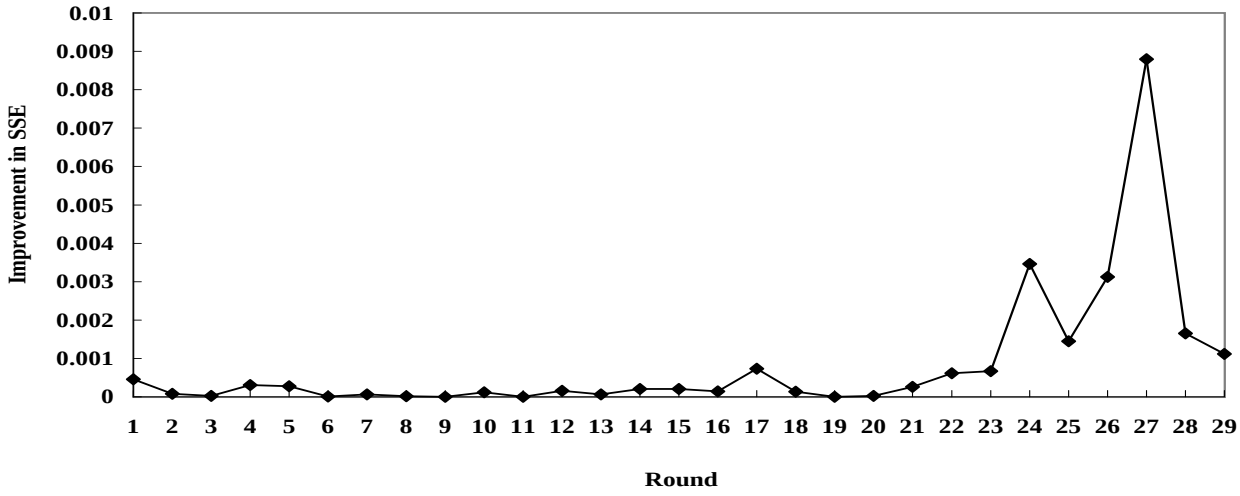


Figure 4.6 : The Improvement Sequence of NIKKEI 225 (Simulation 1.6)

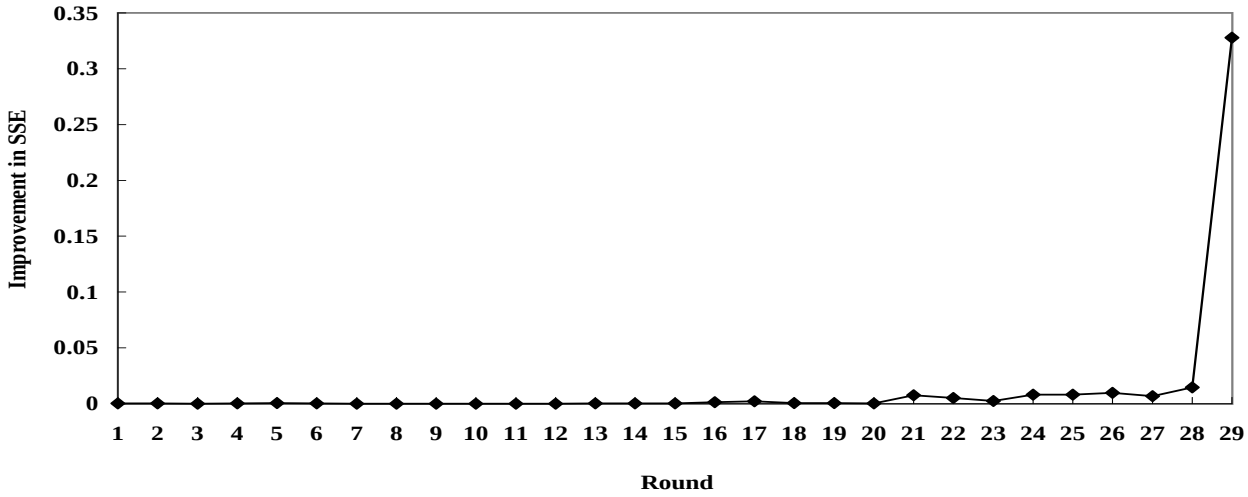


Figure 4.7 : The Improvement Sequence of NIKKEI 225 (Simulation 1.7)

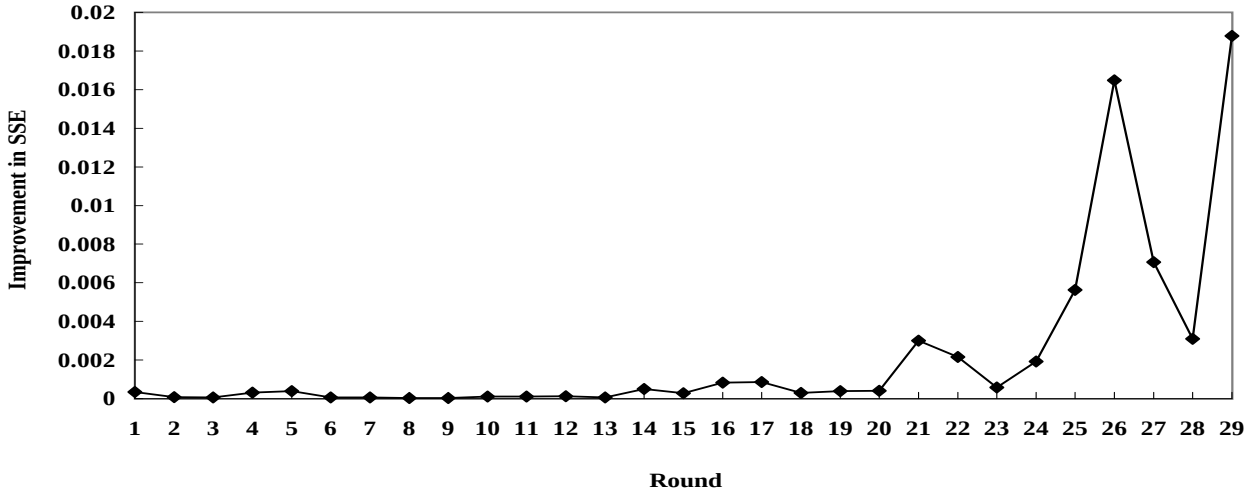


Figure 4.8 : The Improvement Sequence of NIKKEI 225 (Simulation 1.8)

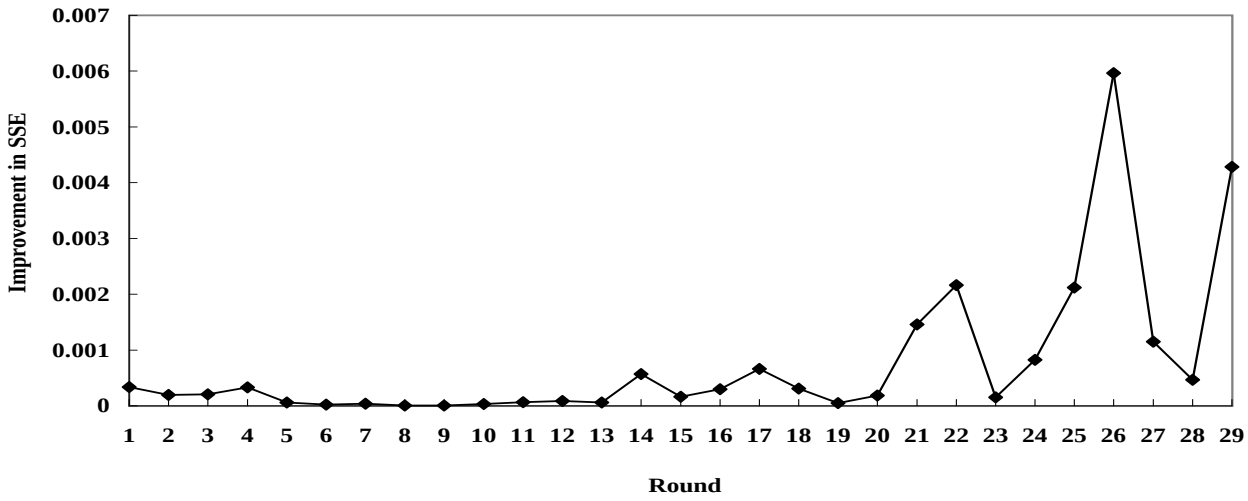


Figure 4.9 : The Improvement Sequence of NIKKEI 225 (Simulation 1.9)

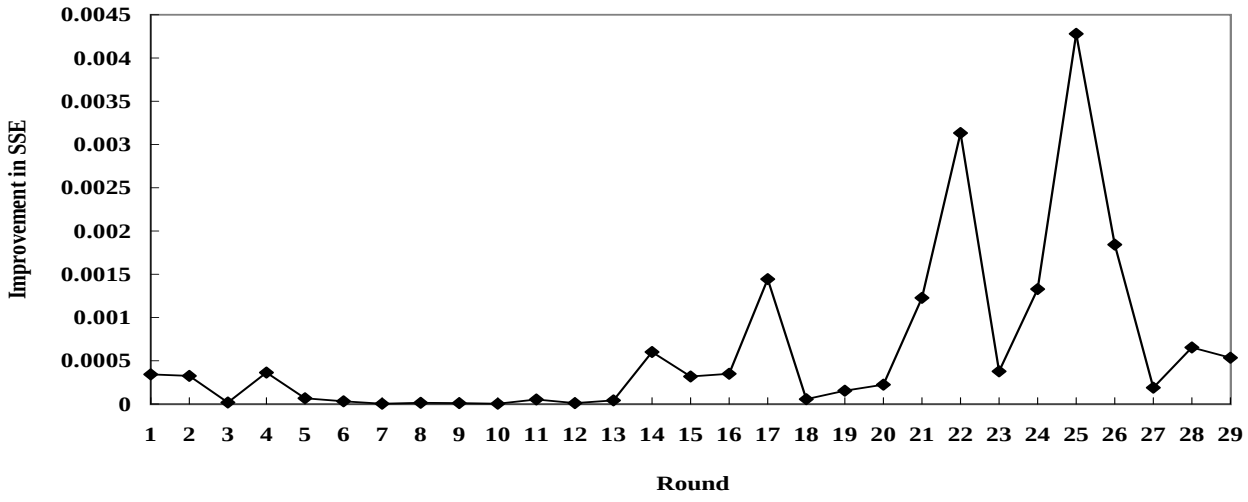


Figure 4.10 : The Improvement Sequence of NIKKEI 225 (Simulation 1.10)

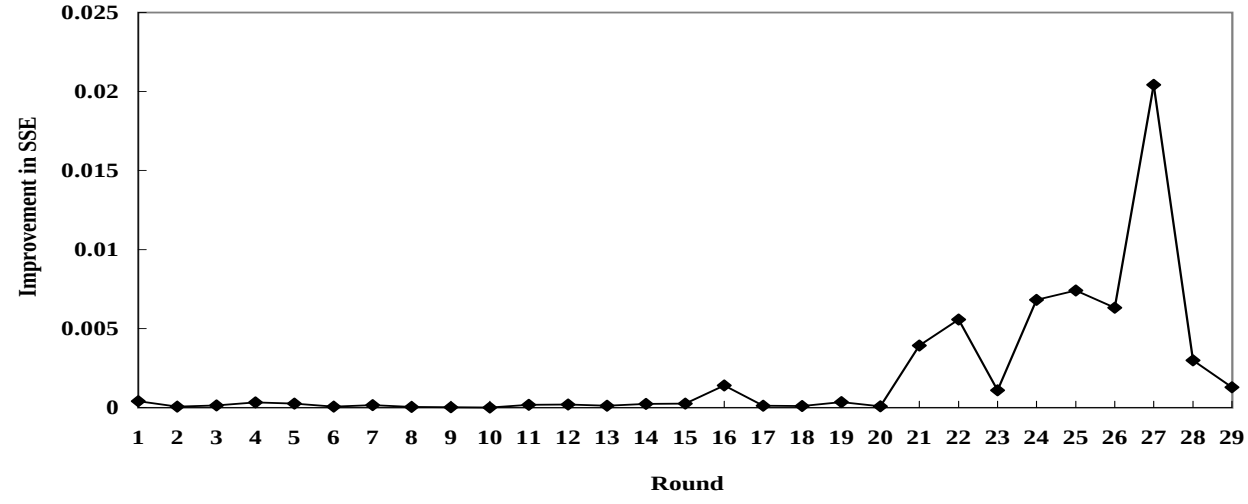


Figure 5.1 : The Improvement Sequence of S&P 500
(Simulation 2.1)

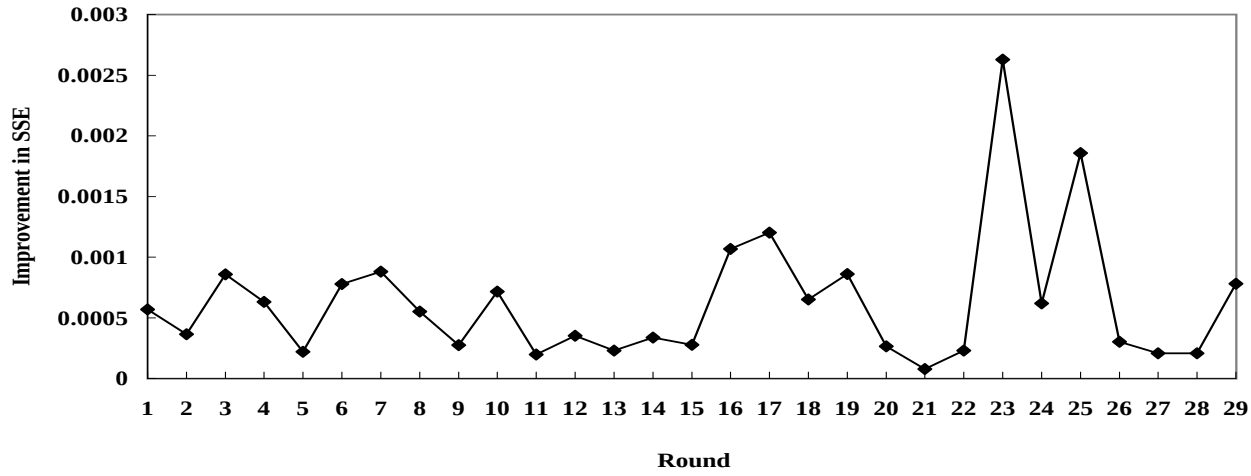


Figure 5.2 : The Improvement Sequence of S&P 500
(Simulation 2.2)

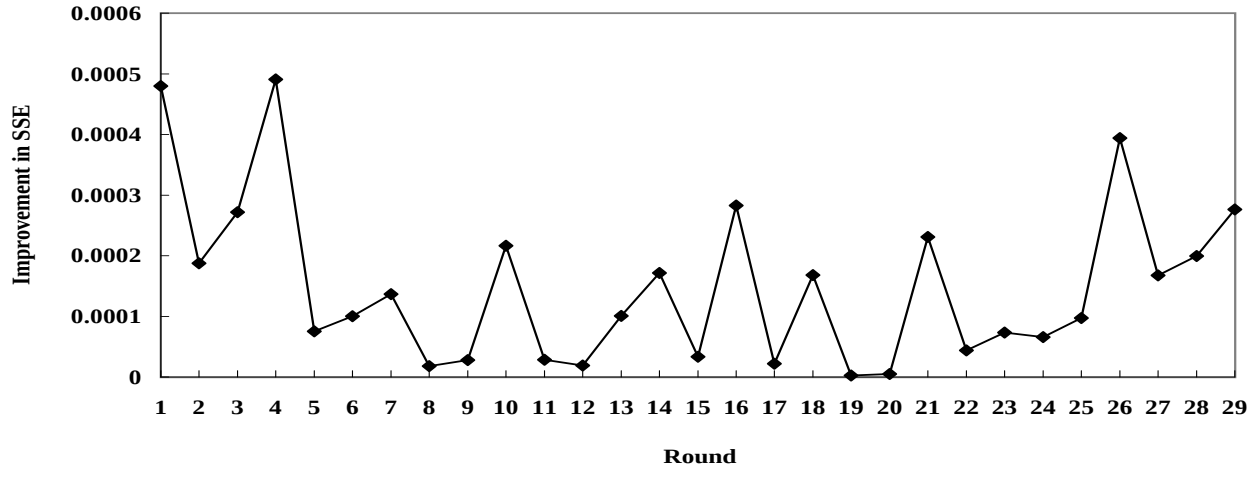
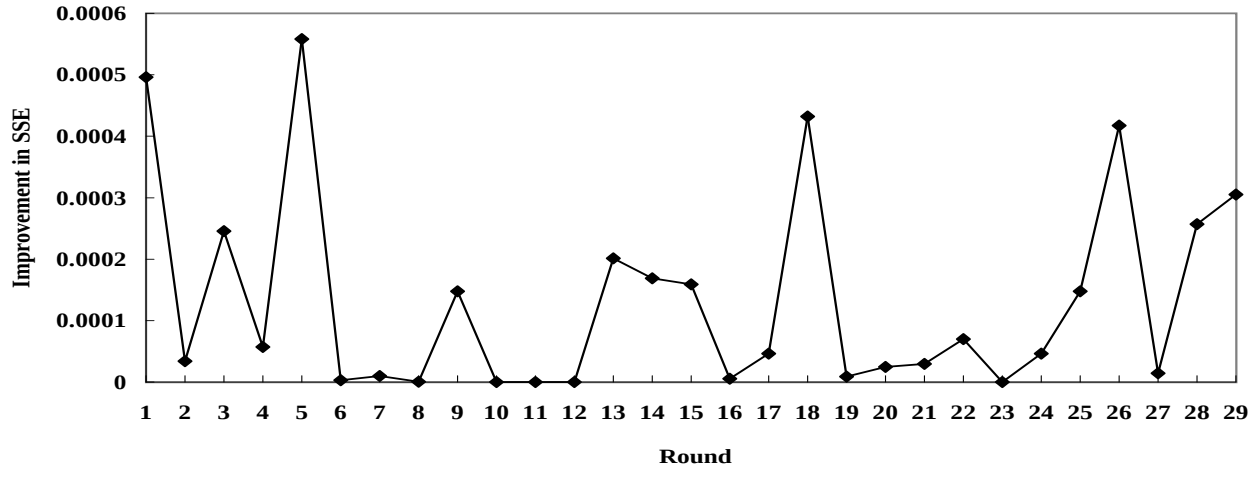


Figure 5.3 : The Improvement Sequence of S&P 500
(Simulation 2.3)



lected from Nikkei 225 and S&P 500. While in this paper we emphasize a model-free notion of structural changes, structural changes were considered a *memory-dependent* concept in Chen and Yeh (1997). The choice of composition of n_1 and n_2, or the ratio $r = \frac{n_1}{n_2}$ indicates the *memory size* upon which structural changes are defined. Using different memory sizes can have different results and implications. For example, with financial data, it is found that it is more difficult to detect structural changes when the ratio r is large. In terms of dynamic landscapes, if the memory size is set to be really large, then it would be possible for the adaptive cognitive system to memorize all the possible patterns. In this case, it is not easy to see any "surprise". However, whether the assumption of an extremely large memory size is practical is another issue yet to be solved.

Acknowledgment

Research support from NSC grant No.84-2415-H-004-001 is gratefully acknowledged. Simulations in this paper were conducted by using the facilities in the **Laboratory for the Advancement of Economics Education** sponsored by the Ministry of Education and National Chengchi University. This paper was revised from its original version by taking the advice and comments of six anonymous referees in the **GP'97** program committee. The authors are very grateful for their painstaking review of this paper. Of course, all remaining errors are the authors' sole responsibility.

References

[1] Chen, S.-H. and C.-W. Tan (1996), "Measuring Randomness by Rissanen's Stochastic Complexity: Applications to the Financial Data," in D. L. Dowe, K. B. Korb and J. J. Oliver (eds.), **ISIS**: *Information, Statistics and Induction in Science*, World Scientific, Singapore. pp.200-211.

[2] Chen, S.-H. and C.-H. Yeh (1997), "Detecting Structural Changes with Recursive Genetic Programming," in *Proceedings of the MLNET Familiarization Workshop: Learning in Dynamically Changing Domains: Theory Revision and Context Dependence Issues*, April 26, 1997, Prague, Czech Republic.

[3] Koza, J. R. (1992), *Genetic Programming: On the Programming of Computers by Means of Natural Selection*, MIT Press.

[4] Kuan, C.-M. and C.-C. Hsu (1996), "Change-Point Estimation of Fractionally Integrated Processes," Working Paper, Department of Economics, National Taiwan University.

[5] Olmeda, I. and E. Fernandez (1996), "Nonparametric Estimation of Fully Nonlinear Models for Assets Returns: Some Results," in D. Borrajo and P. Isasi (eds.), *Proceeding of the First International Workshop on Machine Learning, Forecasting and Optimization* (ISBN:84-89315-04-3), Getafe, Madrid (Spain), July 10-12, 1996, Universidad Carlos III de Madrid, pp. 115-128.

[6] Perron, P. (1989), "The Great Crash, the Oil Price Shock and the Unit Root Hypothesis," *Econometrica* 57, pp.1361-1401.

[7] Refenes, A-P. N., A. N. Burgess and Y. Bentz (1996), "Neural Networks in Financial Engineering," Tutorial Notes, *1996 International Conference on Neural Information Processing* (**ICONIP'96**), Hong Kong Convention and Exhibition Center, Hong Kong, September 24-27, 1996.

[8] Rissanen, J. (1989), *Stochastic Complexity in Statistical Inquiry*, World Scientific, 1989.