

**A STUDY ON THAI EXCHANGE RATE VOLATILITY MODEL COMPARISON:
NONPARAMETRIC APPROACH**

THEEPAKORN JITHITIKULCHAI

**MASTER OF ECONOMICS
(ENGLISH LANGUAGE PROGRAM)
FACULTY OF ECONOMICS, THAMMASAT UNIVERSITY
BANGKOK, THAILAND
MAY 2005**

ABSTRACT

This study compares in- and out-of-sample performance of univariate autoregressive nonparametric and GARCH-classified models for conditional variances, using weekly Baht/U.S. rate of return in 1999-2005. Nonparametric model considers a class of dynamic process in which both the conditional mean and conditional variance (volatility) are unknown functions of the past. Nonparametric estimation deployed local linear estimator and optimal cross-validation bandwidth. GARCH-classified models which are parameterized by constrained maximum likelihood estimation are composed of ARCH, GARCH, TGARCH, EGARCH, I-GARCH, and ARCH-M. For in-sample estimation, there is inclusive evidence to justify the outperformance between nonparametric and parametric approaches. However, predictive efficiency measured by rolling MSPE can find grounds for choosing nonparametric model. An asymmetric U-shaped "smiling face" form of the nonparametric volatility function is found.

ACKNOWLEDGEMENT

My thesis has been accomplished by the gracefulness of many persons. After I finished my thesis, I realize that it will never be finished unless I get supports from each of those who provide valuable contribution and their spare time to help me.

Without the valuable contribution from my advisor and committees, my thesis could not have been fulfilled. I am grateful to my advisor Dr. Sitisak Leelahanon, who always provides supports, suggestion and cheerful conversations, until the last second. His contributions make me conquer all the difficulties. I also so thank both of my thesis committees, Assoc. Prof. Dr. Vimut Vanitkijaroentham and Dr. Orawan Rattanapakorn, who patiently help me work through all of my problems.

I would also like to thanks to all of my friends who always cares, understands and encourages me for the rest of my life. Last but not the least, my deepest appreciation and gratitude go to my beloved family for their love and supports.

I am entirely responsible for all deficiencies, ambiguities and flaws in this study.

Thammasat University
Bangkok, Thailand

Theepakorn Jithitikulchai
May 2005

CONTENTS

ABSTRACT.....	(1)
ACKNOWLEDGEMENT.....	(2)
CONTENTS.....	(3)
LIST OF TABLES.....	(5)
LIST OF FIGURES.....	(6)
 CHAPTER	
1. INTRODUCTION.....	1
1.1 Statement of Problem.....	1
1.2 Objective of the Study.....	3
1.3 Scope of the Study.....	3
1.4 Organization of the Study.....	3
2. INTRODUCTORY NONPARAMETRIC ECONOMETRICS.....	4
2.1 General Concept of Nonparametric Econometrics.....	4
2.2 Kernel Estimator and Asymptotic Properties.....	8
2.3 Local Constant and Local Linear Estimators.....	12
2.4 Cross-Validation and Plug-In Bandwidth.....	16
2.5 Nonparametric Volatility Function.....	21

3. REVIEW OF RELATED LITERATURES.....	24
3.1 Studies on Baht Exchange Rate.....	24
3.2 Studies on Econometrics.....	25
3.2.1 Studies on Conditional Heterogeneous	25
3.2.2 Studies on Nonparametric Econometrics.....	28
3.2.3 Literatures on Models Comparison.....	30
4. THEORETICAL FRAMEWORK AND METHODOLOGY.....	33
4.1 Autoregressive Conditional Heteroskedasticity Processes.....	33
4.2 Nonparametric Volatility Function.....	34
4.3 Data Availability and Description.....	35
4.4 Predictive Performance Comparison.....	39
5. EMPIRICAL RESULTS.....	41
5.1 Introduction to Estimation Results.....	41
5.2 In-Sample Estimation.....	43
5.3 Out-of-Sample Performance.....	63
5.4 Unbiasedness Test.....	65
6. CONCLUSIONS AND LIMITATIONS	68
6.1 Conclusions.....	68
6.2 Limitations.....	70

APPENDICES

BIBLIOGRAPHY

TABLE

Table 4.1 Descriptive Statistics.....	37
Table 4.2 OLS AR(1) and its statistics.....	38
Table 4.3 Unit Root test and ARCH-LM test.....	38
Table 5.1 Estimation output for OLS.....	46
Table 5.2 Estimation output for ARCH.....	47
Table 5.3 Estimation output for GARCH.....	50
Table 5.4 Estimation output for TGARCH.....	53
Table 5.5 Estimation output for EGARCH.....	55
Table 5.6 Estimation output for I-GARCH.....	58
Table 5.7 Estimation output for ARCH-M.....	61
Table 5.8 Compare Rolling MSPE.....	64
Table 5.9 Test for unbiasedness.....	67

FIGURE

Figure 2.1 Different bandwidths for NW estimators.....	6
Figure 2.2 Data Driven Estimator (Local Constant).....	7
Figure 2.3 Data Driven Estimator (Local Linear).....	14
Figure 2.4 Local Constant Estimator.....	19
Figure 2.5 Local Constant Estimator.....	19
Figure 2.6 Local Linear Estimator.....	20
Figure 2.7 Local Linear Estimator.....	20
Figure 4.1 Percentage Weekly Baht/U.S. Return, 2000-2005.....	36
Figure 4.2 ACF and PACF functions.....	37
Figure 5.1 Nonparametric Estimation.....	44
Figure 5.2 OLS Estimation.....	46
Figure 5.3 ARCH Estimation.....	48
Figure 5.4 GARCH Estimation.....	51
Figure 5.5 TGARCH Estimation.....	53
Figure 5.6 EGARCH Estimation.....	56
Figure 5.7 I-GARCH Estimation.....	59
Figure 5.8 ARCH-M Estimation.....	62

CHAPTER 1

INTRODUCTION

1.1 Statement of Problem

Volatility is one of the basic identities of many economic time series because of the fluctuation of fundamental, psychological and behavioral factors. The common stylized- facts related to the properties of economic time series data reveal the variation of conditional mean and conditional variance and most exhibit phases of relative tranquility followed by periods of high volatility. So in applied study, the basic assumption of conditional homoskedasticity is inappropriate because the random disturbances for different times should have diverse variances.¹

In time series regressions, the explanatory variable itself exhibit dynamic behaviors and such specification imply a dynamic structure for the conditional variances. The variation volatility of a time-series process relates to the dynamic pattern of such a process. Engle (1982) modeled the heteroskedasticity by relating the conditional variance of the disturbance term to the linear combination of the squared disturbances in the recent past. This specification is called a simple autoregressive conditional heteroskedasticity (ARCH) because the variance, conditional on prior information, is in an autoregressive form of lagged squared disturbances. This model has been generalized by Bollerslev (1986) with the conditional variance depends on its lagged values as well as squared lagged values of disturbance, which is called generalized autoregressive conditional heteroskedasticity (GARCH) . Because the economic of uncertainty being recognize the importance of econometric approach to estimate the variance of a series, ARCH and

¹ Piyassphan (2003) found that Thailand exchange rate's conditional variance is not constant over time, such as Khanthavit (n.d.) and Phrukpaisal (2003) for Thailand stock market return empirical on variation volatility. Vimolsiri (1995), Khantavit (2001), and Chusil (2003) can distinguished the market regimes of both stock return's conditional means and conditional variances in two and three states.

GARCH models have been extended and implemented in many forms and in various applications.

The volatility of these models depends parametrically on lagged values of the process and lagged values of volatility. One of the standard approaches of fitting these models to data involves nonlinear maximum likelihood estimations (MLE). But these models are sensitive to model misspecification about the features of variables and functional relations among them.

Now the central issues on the stochastic dependency, between regressor and regressand in regression, are the determination of the function. In particular areas of econometrics in any economics applications, a variety of ad hoc transformations have been exploited. The interest in this question of estimating an unknown functional form extends out of range of just estimating a conditional mean to higher order moments. Because there has always been a residual doubt that the functional form might be more complicated than the set of feasible forms allowed for and that the approximation is seriously deficient. In practice, if the functional form is failed to be specified, estimation results based on an incorrect specified parametric model can give the misleading conclusion and thus misdirect policy.

Nonparametric econometrics is one of the solutions in corrective estimation because of the flexibility in unrestricted functional form. The conditional variance, or both conditional mean and conditional variance, could be modeled in a generalized nonparametric regression. The Rosenblatt-Parzen kernel estimator is the well known estimator and has been developed for many years. This approach could explore the unknown functional form in regression function.

This study will focus on several, nonparametric and parametric, conditional volatility models comparison in out-of-sample forecasting performance using Thailand

exchange rate for U.S. Dollar. The focusing issue is the volatility estimation ability in term of econometric contribution.

1.2 Objective of the Study

The study is aimed to analyze the econometrics estimation approaches when the univariate time-series is employed for forecasting purposes.

1. To study the differentiation of parametric and nonparametric estimation on exchange rate conditional volatility.
2. To compare predictability performance among different approached competitive models.

1.3 Scope of the Study

This study will compare the nonparametric and parametric models prediction's performance for the return and volatility of weekly Baht/U.S.\$ rate of return in 1999-2005. This study will use rolling mean squared prediction error (rolling MSPE) as the measure of the predictive ability.

1.4 Organization of the Study

Next chapter will be an expository introduction to some of the basic methods of nonparametric regression. Then, after the illustrated results on Baht/U.S. currency review, the next chapter will also retrospectively survey the studies on the development of related topics of econometrics. Studies of conditional heteroskedasticity will be reviewed compactly before studies on nonparametric volatility function and literatures on model comparison. Chapter four will present the theoretical framework and methodology used in the study. Chapter five will discuss the empirical results and, finally, chapter six concludes.

CHAPTER 2

INTRODUCTORY NONPARAMETRIC ECONOMETRICS

2.1 General Concept of Nonparametric Econometrics

Parametric method is statistically simple and if the assumptions of a parametric model are justified, the regression function can be estimated more efficiently than it can be done by a nonparametric method. But many assumptions are made in coming up with the questions about the functional relations and the distributional features of variables.

In contrast, nonparametric approach is desirable because the minimum of structure imposed on the regression function. It is only necessary that regression function possess some degree of smoothness in order for nonparametric methods to yield reasonable estimates of this function. Typically, continuity of function is enough to ensure the convergence of estimator to the true as the size of data increases. Additional smoothness, such as the existence of derivatives, allows more asymptotic efficient estimation.

Nonparametric estimation can improve an approximation by capture a wide variety of nonlinearity without restrict any particular specification of the nonlinear relation. Nonparametric estimation of $m(x)$, the unknown functional form of conditional mean for any regression function, is employed by the method of density estimation. The Nadaraya-Watson estimator, one of the well-known estimation approach, is a kind of smoothing estimators because the observational errors are reduced by averaging the data.

Let $y_t = m(x_t) + u_t$ $t = 1, \dots, T$ where $m(\cdot)$ is an arbitrary fixed but unknown function of conditional mean, i.e. $m(x) = E(Y | X = x)$ where x is some fixed value of X , $\{u_t\}_{t=1}^T$ is a zero mean and finite constant variance *i.i.d.* process.

The basic idea of estimation is to local averaging all y_t 's whose x_t 's are “close” to the point x at which we want to estimate curve. This is called kernel estimator which use “weighted” average of y_t 's whose x_t 's are close to x . That is by put more weight on y_t 's whose x_t 's are close to x .

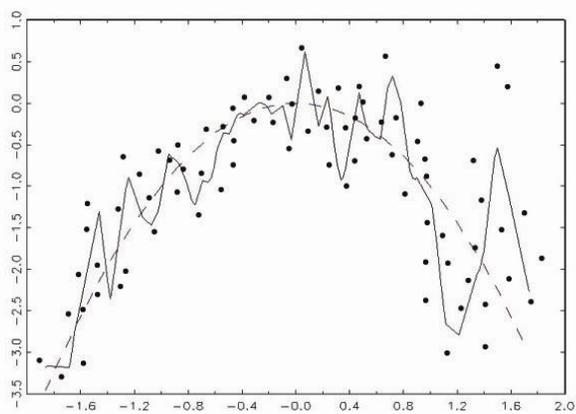
In other words, there's a small neighborhood around x_0 which $m(x_0)$ will be nearly constant and could be estimated by taking the average of the y_t 's that correspond to those x_t 's near x_0 . So the closer of x_t 's are to the value x_0 , the closer an average of corresponding y_t 's to $m(x_0)$.

A feature of Nadaraya-Watson estimator is being a weighted sum at those y_t 's that corresponding to x_t 's in a neighborhood of x . For an arbitrary x , a smoothing estimator of $m(x)$: $\hat{m}(x) = \sum_{t=1}^T w_t(x)y_t$ where $\{w_t(x)\}_{t=1}^T$ are weights which are high for y_t 's paired with x_t 's that are closer to x . To ensure the consistency of $\hat{m}(x)$ it is necessary that $\sum_{t=1}^T w_t(x) = 1$. This condition is guaranteed for each x by the way in which Nadaraya-Watson estimator is constructed. Campbell et al. (1997) illustrate the simulation figures that if one chooses too large a neighborhood around x to compute the average, the weighted average will be too smooth and cannot capture the nonlinearity of $m(\cdot)$. On the other hands, too small neighborhood around x , the weighted average will be too vary, reflecting noise as well as variations in $m(\cdot)$. We have a set of illustrated examples shown in figure 2.1 (a) to (c). The choices in different bandwidths are shown that the weights $\{w_t(x)\}$ need to be optimized balancing on the trade-off between unbiasedness and variation in estimation

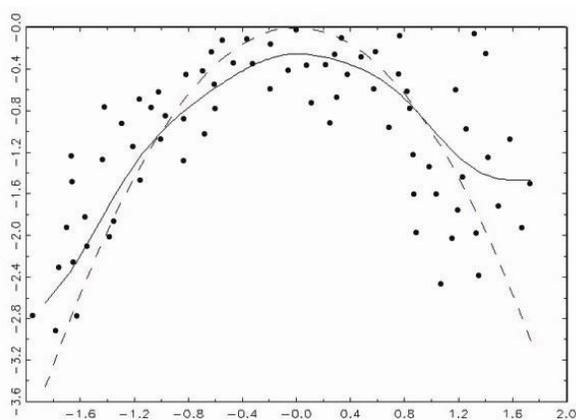
Figure 2.1

Different bandwidths of 0.04, 0.35 and 1 for NW estimators

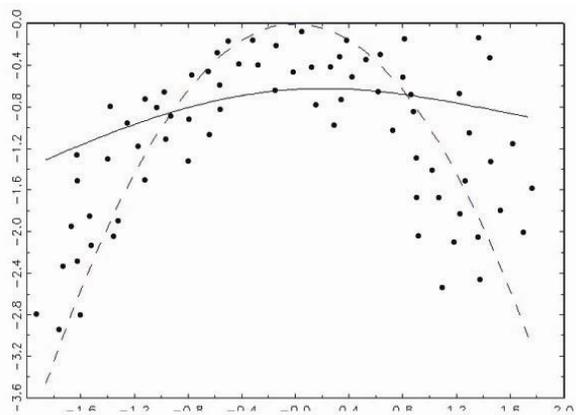
(a)



(b)



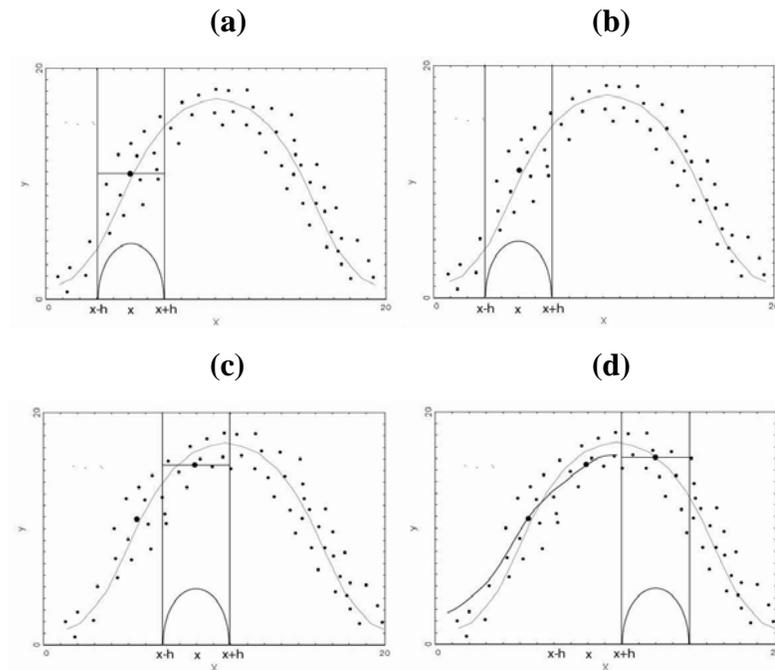
(c)



The weighted average process in the given bandwidth is shown in Figure 2.2, the idea behind this process is to use information in the small neighborhood around point x for the corresponding estimation. Only the data in this small bandwidth matters for estimate function at the given point of explanatory variable. And the observations near this given point will be weighted more than the farther ones in estimation procedure. Figure 2.2 (a) shows that the estimated constant curve that is derived from weighted average all the observations in the bandwidth and give the corresponding estimator: $\hat{m}(x) = \sum_{t=1}^T w_t(x)y_t$ in figure 2.2 (b). Then this data driven process move bandwidth along the smoothing estimation to new supposed x as in figure 2.2 (c) and have the corresponding estimator at this given new point. The different estimators of any given sets of point x are the nonparametric regression function, as some part of curve is show the connected line between estimators as in figure 2.2 (d).

Figure 2.2

Data Driven Estimator (Local Constant)



2.2 Kernel Estimator and Asymptotic Properties

As previous describe that, kernel regression is an important smoothing technique for estimates $m(\cdot)$. From definition of cumulative distribution function (cdf) and indicator function, we can also estimate pdf $f(x)$. If $f(x)$ is smooth in a small neighborhood $[x-h, x+h]$ of x , we can justify the following approximation,

$$2hf(x) = \int_{x-h}^{x+h} f(z)dz = P(X \in [x-h, x+h]), \quad (2.1)$$

by the mean value theorem where h is called the smoothing parameter (or bandwidth).

The right-hand side of (2.1) can be approximated by counting the number of x_t 's in this small interval of length $2h$, and then dividing by T . This is a histogram estimator with bincenter x and binwidth $2h$. For $z = (\frac{X_t - x}{h})$, let $K(z) = \frac{1}{2}I_{[-1,1]}(z) = \frac{1}{2}I(|z| \leq 1)$, where $I(\cdot)$ is the indicator function taking the value 1 when the event is true and zero otherwise, and $K(\cdot)$ is called the kernel function.

$$\begin{aligned} \text{Therefore} \quad K(z) &= \frac{1}{2} && \text{if } z \in [-1, 1] && (2.2) \\ &= 0 && \text{otherwise} \end{aligned}$$

$$\begin{aligned} \text{So} \quad K\left(\frac{X_t - x}{h}\right) &= \frac{1}{2} && \text{if } X_t \in [x-h, x+h] && (2.3) \\ &= 0 && \text{otherwise} \end{aligned}$$

$$\text{Then} \quad \sum_{t=1}^T K\left(\frac{X_t - x}{h}\right) = \frac{\text{number of } X_t \in [x-h, x+h]}{2} \quad (2.4)$$

$$\begin{aligned} \text{From} \quad P(X \in [x-h, x+h]) &= \frac{\text{number of } X_t \in [x-h, x+h]}{T} && (2.5) \\ &= \frac{2 \sum_{t=1}^T K\left(\frac{X_t - x}{h}\right)}{T} \end{aligned}$$

Then the histogram estimator can be written as

$$\hat{f}(x) = \frac{\sum_{t=1}^T K\left(\frac{X_t - x}{h}\right)}{Th} \quad (2.6)$$

In most cases, kernel function is a probability density function which is a piecewise continuous function, unimodal, integrating to one, symmetric about zero, and has first two moments finite. Some kind of kernel weights each observations inside the window equally as the above mathematical derived kernel is the example. However, observations closer to x should possess better information than more distant ones as some another kinds of kernel. As some kinds of kernels have more weights for closer observations such as Gaussian kernel for example.

The examples of the kernel functions are

$$\text{Uniform kernel: } K(z) = \frac{1}{2} I_{[-1,1]}(z) \quad (2.7)$$

$$\text{Standard normal kernel: } K(z) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}z^2}; -\infty < z < \infty \quad (2.8)$$

$$\text{Epanechnikov kernel: } K(z) = \frac{3}{4}(1 - z^2)I_{[-1,1]}(z) \quad (2.9)$$

$$\text{Quartic kernel: } K(z) = \frac{15}{16}(1 - z^2)^2 I_{[-1,1]}(z) \quad (2.10)$$

where $I_{[-1,1]}$ is an indicator function with support $[-1,1]$.

In addition, kernel can be considered as rolling windows to estimate curve at each sample point. This rolling kernel should move along the horizontal axis to estimate the curve at given point x_t 's. The curve of quartic kernel recognizes the unequally weighted local average of corresponding y_t 's in estimation procedure.

From usual literatures on nonparametric function estimation¹, one can show that if x_1, \dots, x_T be *i.i.d.* observations with pdf $f(x)$ which are twice continuously differentiable, also the s^{th} order derivatives of $f(x)$, $f^{(s)}(x)$; $s = 0, 1, 2$, are bounded functions. And the kernel function $K(\cdot)$ satisfies

$$(i) \int K(z)dz = 1 \quad (ii) \int zK(z)dz = 0 \quad (iii) \int z^2K(z)dz = k_2 > 0$$

If as $T \rightarrow \infty, h \rightarrow 0$ and $Th \rightarrow \infty$, where

$$\hat{f}(x) = (Th)^{-1} \sum_{t=1}^T K\left(\frac{X_t - x}{h}\right), \quad (2.11)$$

then $\hat{f}(x) \rightarrow f(x)$ in MSE.

For the general nonparametric regression model ($X_t = (X_{t,1}, X_{t,2}, \dots, X_{t,p}) \in \mathbb{R}^p$);

$$y_t = m(x_t) + u_t = E(y_t | x_t) + u_t \quad (2.12)$$

Given the definition of conditional expectation,

$$E(Y | X = x) = \int y f(y | x) dy = \int y \frac{f(x, y)}{f_X(x)} dy \quad (2.13)$$

The estimator for $\int y f(x, y) dy$ is $\int y \hat{f}(x, y) dy$,

$$\text{where } \hat{f}(x, y) = (Th^{p+1})^{-1} \sum_{t=1}^T K\left(\frac{X_t - x}{h}\right) K\left(\frac{Y_t - y}{h}\right)$$

$$\text{with } K\left(\frac{X_t - x}{h}\right) = k\left(\frac{X_{t,1} - x_1}{h}\right) \dots k\left(\frac{X_{t,p} - x_p}{h}\right) = \prod_{l=1}^p k\left(\frac{X_{t,l} - x_l}{h}\right);$$

where $k\left(\frac{X_{t,l} - x_l}{h}\right)$ is a univariate kernel for l^{th} variable ; $l = 1, \dots, p$

$$\begin{aligned} \text{So } \int y \hat{f}(x, y) dy &= (Th^{p+1})^{-1} \sum_{t=1}^T K\left(\frac{X_t - x}{h}\right) \int y K\left(\frac{Y_t - y}{h}\right) dy \\ &= (Th^{p+1})^{-1} \sum_{t=1}^T K\left(\frac{X_t - x}{h}\right) \int (Y_t - hz) K(z) h dz ; Y_t - y = hz \end{aligned} \quad (2.14)$$

¹ as in Hart (1997), Pagan and Ullah (1999), and Li (2000),

$$\begin{aligned}
&= (Th^{p+1})^{-1} \sum_{t=1}^T K\left(\frac{X_t - x}{h}\right) \left[\int Y_t K(z) h dz - \int h^2 z K(z) dz \right] \\
&= (Th^p)^{-1} \sum_{t=1}^T K_t\left(\frac{X_t - x}{h}\right) Y_t ; \int K(z) dz = 1 \text{ and } \int z K(z) dz = 0
\end{aligned}$$

Thus we will estimate $E(Y | X = x) = m(x)$ by

$$\hat{E}(Y | X = x) = \hat{m}(x) = \frac{(Th^p)^{-1} \sum_{t=1}^T K\left(\frac{X_t - x}{h}\right) Y_t}{(Th^p)^{-1} \sum_{t=1}^T K\left(\frac{X_t - x}{h}\right)} \quad (2.15)$$

The proof of $\hat{m}(x) \xrightarrow{p} m(x)$ is very similar to the proof of $\hat{f}(x) \xrightarrow{p} f(x)$.

For particular exposition below extended from previous theoretical instance should exhibit more intuition about the nonparametric kernel estimator, consider an example for $X_t \in \mathbb{R}^n$ with uniform kernel.

$$\begin{aligned}
\text{Then } \hat{m}(x) &= \frac{\sum_{t=1}^T K\left(\frac{X_t - x}{h}\right) Y_t}{\sum_{t=1}^T K\left(\frac{X_t - x}{h}\right)} = \frac{\sum_{|X_t - x| \leq h} Y_t}{\sum_{|X_t - x| \leq h} 1} = \frac{\sum_{|X_t - x| \leq h} [m(x_t) + u_t]}{\sum_{|X_t - x| \leq h} 1} \quad (2.16) \\
&= \{ \text{average of } m(x_t) \text{ s} + \text{average of } u_t ; |X_t - x| \leq h \} \\
&= \{ \text{conditional mean ; } |X_t - x| \leq h \} + \\
&\quad \{ \text{conditional stochastic disturbance ; } |X_t - x| \leq h \} \\
&\xrightarrow{p} m(x) + 0 = m(x)
\end{aligned}$$

because $|m(X_t) - m(x)| = O_p(h) = o_p(1)$ as $h \rightarrow 0$ for $|X_t - x| \leq h$

and average of $u_t \xrightarrow{p} 0$ by law of large number since $T \rightarrow \infty$ and $E(u) = 0$.

In general, since $\hat{m}(x)$ use local average of Y 's whose X_t is close to x to estimate $m(x)$, hence $\hat{m}(x)$ is also called local constant estimator of $m(x)$. This Nadaraya-Watson estimator satisfies the weighted least square criterion:

$$\hat{m}(x) = \arg \min_{\{m\}} \sum_{t=1}^T \{Y_t - m\}^2 K\left(\frac{X_t - x}{h}\right) \quad (2.17)$$

2.3 Local Constant and Local Linear Estimators

This subsection will introduce the fundamental concept of local constant and local linear (or in general called local polynomial) estimator. The shared common idea of both estimators is to smoothed estimating the unknown form of regression function by remove data variability that has no assignable cause and makes systematic identities of data be apparent. In particular, while local constant estimator weighted averages for constant curve, local linear estimator provides estimator for the unknown curve and its derivative. This part will be focus on local linear estimator. The intuition and theoretical aspect are presented. Matrix notation for the view of GLS estimation also provided.

Local Constant Estimator

Local constant estimator is the one of nonparametric method for estimating the regression function. The local averaging method by weighted averages the data in the given interval bandwidth is the idea of estimating the continuous function. The average is the estimate of the regression function value from moving the window interval along the set of explanatory variable's data to compute the estimate at any point. The closer observations will be systematic weighted more to the computation process, the summation of weights equals to one.

As priori pointed out in previous subsection that the Nadaraya-Watson estimator of $m(x)$ minimizes

$$\sum_{t=1}^T \{Y_t - m\}^2 K\left(\frac{X_t - x}{h}\right) \quad (2.18)$$

and the solution of m is

$$\hat{m}(x) = \left(\sum_{t=1}^T K_t\right)^{-1} \sum_{t=1}^T K_t Y_t \quad (2.19)$$

, the weighted average of y_t values, where $K_t = K\left(\frac{X_t - x}{h}\right)$.

Thus, $\hat{m}(x)$ is the least squared estimator of the parameter in the weighted regression of y_t on unity with weight equal to $K_t^{\frac{1}{2}}$.

In matrix notation, this problem is equivalent to $\min_{\{m\}} (Y - mi)' \Omega (Y - mi)$ (2.20)

where $i = (1 \dots 1)'$, the $T \times 1$ vector of 1, and $\Omega = \text{diag}\left[K\left(\frac{X_t - x}{h}\right)\right]$.

Then $\hat{m} = (i' \Omega i)^{-1} i' \Omega y \equiv \left(\sum_{t=1}^T K_t\right)^{-1} \sum_{t=1}^T K_t Y_t$. (2.21)

Local Linear Estimator

Local linear method will estimate any twice continuous differentiable regression function. This regression function is approximately linear in an interval of given observations of independent variable. The advantage of using a local linear estimator that have degree of polynomial more than zero (local constant estimator) is to reduce the estimation bias. The estimator at any given data is the intercept of its linear estimator.

To obtain local linear estimator, we estimate any sufficiently smoothed function by fitting straight lines locally to the data. Thus, local linear (or local polynomial, in general) estimator of $m(x)$ is the solution of the following minimization problem:

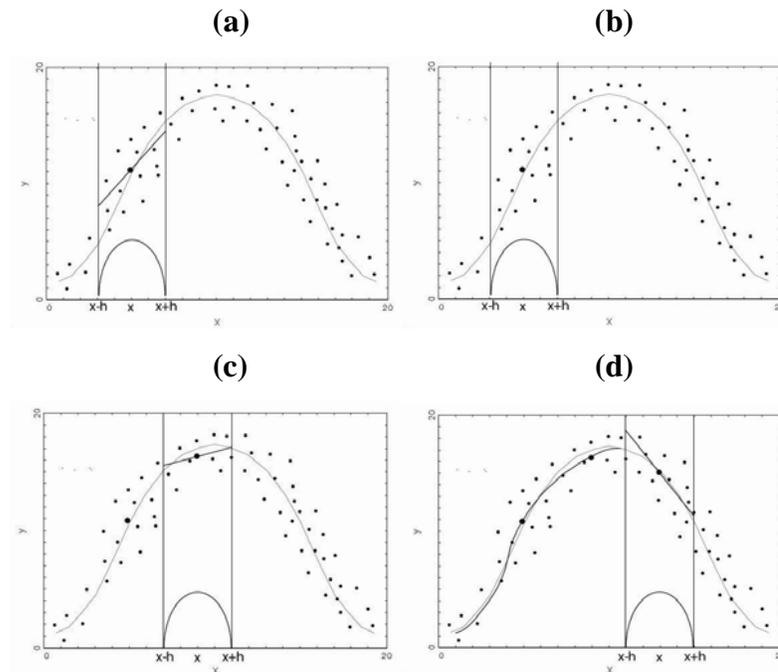
$$\hat{m}(x) = \arg \min_{\{m\}} \sum_{t=1}^T \{Y_t - m - (X_t - x)' \beta\}^2 K\left(\frac{X_t - x}{h}\right) \quad (2.22)$$

We call $\hat{m}(x)$ the local linear estimator of $m(x)$. And slope estimator, $\hat{\beta}(x)$, can be used to estimate the derivative of $m(x)$. This estimate can be found by performing a weighted least squares regression of y_t against $(1 \ (X_t - x)')$ with weights $K_t^{\frac{1}{2}}$. Thus, whereas the Nadaraya-Watson estimator fits a constant to the data close to x , the local linear

approximation fits a straight line. Figure 2.3 give the differences from figure 2.2 in data-driven estimated curve in any given bandwidths. We can see that the least square estimators in the given bandwidths are linear function with slope. The corresponding estimators that occupy on the different estimated local linear in any bandwidths are connected at the corresponding of any explanatory variable points to be the regression function estimator.

Figure 2.3

Data Driven Estimator (Local Linear)



Let \hat{m} and $\hat{\beta}$ be the solutions for this local linear weighted least squared minimization problem. One can show that $\hat{m}(x)$ and $\hat{\beta}(x)$ are consistent estimators for $m(x)$ and $\beta(x) = \partial m(x) / \partial x$, respectively. For matrix notation, denotes

$$\delta = \delta(x) = (m(x), \beta(x))'. \quad (2.24)$$

Let \mathcal{X} be an $T_n (1+ p)$ matrix with t^{th} row being $(1 \quad (X_t - x)')$, i.e.,

$$\mathcal{X} = \begin{pmatrix} 1 & X_{1,1} - x_1 & \dots & X_{1,p} - x_p \\ \mathbf{M} & \mathbf{M} & \mathbf{O} & \mathbf{M} \\ 1 & X_{T,1} - x_1 & \dots & X_{T,p} - x_p \end{pmatrix} \quad (2.25)$$

and let Ω be an $T \times T$ diagonal matrix with t^{th} diagonal element being

$$K\left(\frac{X_t - x}{h}\right) = \prod_{l=1}^p k\left(\frac{X_{t,l} - x_l}{h}\right) \quad (2.26)$$

Thus, local linear estimator's minimization problem can be written as

$$\min_{\{m, \beta\}} (Y - \mathcal{X}\delta)' \Omega (Y - \mathcal{X}\delta) \quad (2.27)$$

which is the standard GLS problem with $\hat{\delta} = (\hat{m} \ \hat{\beta}')'$ be the solution:

$$\begin{aligned} \hat{\delta}(x) &= (\mathcal{X}' \Omega \mathcal{X})^{-1} \mathcal{X}' \Omega y \\ &\equiv \left(\sum_{t=1}^T K\left(\frac{X_t - x}{h}\right) \begin{pmatrix} 1 \\ X_t - x \end{pmatrix} \begin{pmatrix} 1 & (X_t - x)' \end{pmatrix} \right)^{-1} \sum_{t=1}^T K\left(\frac{X_t - x}{h}\right) \begin{pmatrix} 1 \\ X_t - x \end{pmatrix} Y_t \end{aligned} \quad (2.28)$$

Given assumptions

(i) $\{X_t, Y_t\}_{t=1}^T \sim i.i.d.$; $X_t \in \mathbb{R}^p$. Both X_t and $u_t = Y_t - m - (X_t - x)' \beta$ have finite fourth moment

(ii) $m(x)$ is twice differentiable, and its second order derivative is bounded. $\sigma^2(x) = E(u^2 | X = x)$ is continuous in x

and (iii) K is a second order kernel, as $T \rightarrow \infty$, $h \rightarrow 0$, and $Th^{p+2} \rightarrow \infty$. Also assumes that K is a compact supported bounded function, such that $K > 0$ on a set of positive Lebesgue measure,

then one can show that $\hat{\delta}(x) \xrightarrow{P} \delta(x)$. (2.29)

2.4 Cross-Validation and Plug-In Bandwidth

The choice of kernel is not crucial, that is there are a number of ways to eliminate the asymptotic problem; e.g. use the higher order (in bounded derivatives) kernel. The nearly critical is the choice of bandwidth. Generally, the bias of kernel estimator becomes smaller in magnitude as the bandwidth is made smaller. Unfortunately, decreasing the bandwidth also has the effect of increasing the estimator's variance. So there are principles to find a bandwidth that afford a satisfactory compromise. Theoretically, one can show that if $X_i \sim N^p$ and second order kernel is used, $bias(\hat{f}(x)) = c_1 h^2$ and $var(\hat{f}(x)) = c_2 (Th^p)^{-1}$, where c_1 and c_2 are some constants.

Plug-in Rules

From asymptotic property theorem of $\hat{f}(x)$, the estimator of pdf $f(x)$ minimize $MSE(\hat{f}(x))$. The optimal choice of h , bandwidth or smoothing parameter, should minimize the MSE , so

$$h = \arg \min_h MSE = \arg \min_h [c_1 h^4 + c_2 (Th^p)^{-1}] \quad (2.30)$$

$$\text{F.O.C.} \quad 4c_1 h^3 - c_2 (Th^p)^{-1} = 0 \quad \Rightarrow \quad h = cT^{-\frac{1}{4+p}} \quad (2.31)$$

Plug-in Rules exploit an asymptotic approximation to mean average (mean integrated) squared error. But in practice we don't know the constant c , so the sample standard deviation might be used for.

Cross-Validation

The basic idea of cross-validation is that to build a model from one part of the data and then use that model to predict the rest of data. For a given model, one may

compute an average prediction error over all the data points predicted. The model that minimizes the average prediction error is best among a set of models. Define cross-validation criterion as follows:

$$CV(h) = \frac{1}{T} \sum_{t=1}^T (Y_t - \hat{m}_{-t}(X_t, h))^2. \quad (2.32)$$

The cross-validation smoothing parameter is the value h that minimizes $CV(h)$ where \hat{m}_{-t} is computed as the “leave-one-out” estimator deleting the t^{th} observation in the estimation process of this mean average squared error minimization. Without deletion, the optimum of h equates to zero because $\hat{m}(X_t, h) = Y_t$ for $t = 1, \dots, T$ such that $MASE(h) = 0$. The advantages of cross-validation estimator are completely automatic smooth and no estimate of σ^2 is needed.

$$\begin{aligned} \text{Consider } E(CV(h)) &= E \left[\frac{1}{T} \sum_{t=1}^T (Y_t - \hat{m}_{-t}(X_t; h))^2 \right] \\ &= E \left[\frac{1}{T} \sum_{t=1}^T (\hat{m}_{-t}(X_t; h) - Y_t)^2 \right] \\ &= E \left[\frac{1}{T} \sum_{t=1}^T (\hat{m}_{-t}(X_t; h) - m(X_t) + m(X_t) - Y_t)^2 \right] \\ &= \sigma^2 + \frac{1}{T} \sum_{t=1}^T E[\hat{m}_{-t}(X_t; h) - m(X_t)]^2 \\ &= \sigma^2 + MASE(h) \end{aligned} \quad (2.33)$$

where $MASE(h)$ is the mean average squared error.

The second last equality of (2.33) holds because Y_t and $\hat{m}_{-t}(X_t; h)$ are independent. This follows that $CV(h)$ is essentially an unbiased estimator of $\sigma^2 + MASE(h)$, which $CV(h)$ is minimized at the same value of h that could minimized $MASE(h)$.

Figure 2.4 to 2.7 illustrate the example of estimating simulated data from function $y_t = 1.3\sin(17x_t) + u_t$, where $t=1$ to 350 and $u_t \sim N(0,0.25)$. For both local constant and local linear estimators, we can see that cross-validation bandwidth could capture the random generated data better than the plug-in bandwidth. That is the solid line of function estimator could move along the dotted real function line in the way that estimated curve much more capable to fit the real data.

Please note that this is similar to the case local linear estimator prefer to local constant for both plug-in and cross validation bandwidth. The figures of generated data simulation show that local linear estimation using cross-validation bandwidth could yield more efficient estimation than another three cases.

Fig 2.4
Local Constant Estimator with Plug-In Bandwidth=0.2789 :
MSE=1.7591322e-005

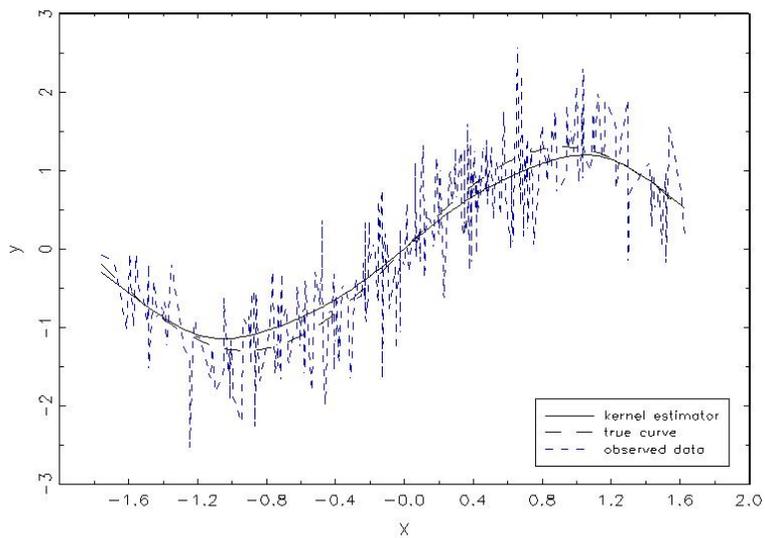


Fig 2.5

Local Constant Estimator with Cross-Validation Bandwidth=0.1402 :

MSE=5.9616176e-006

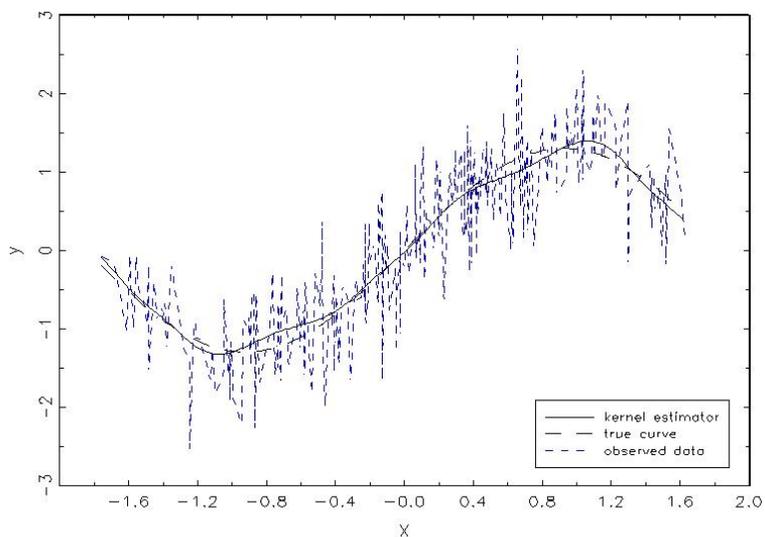
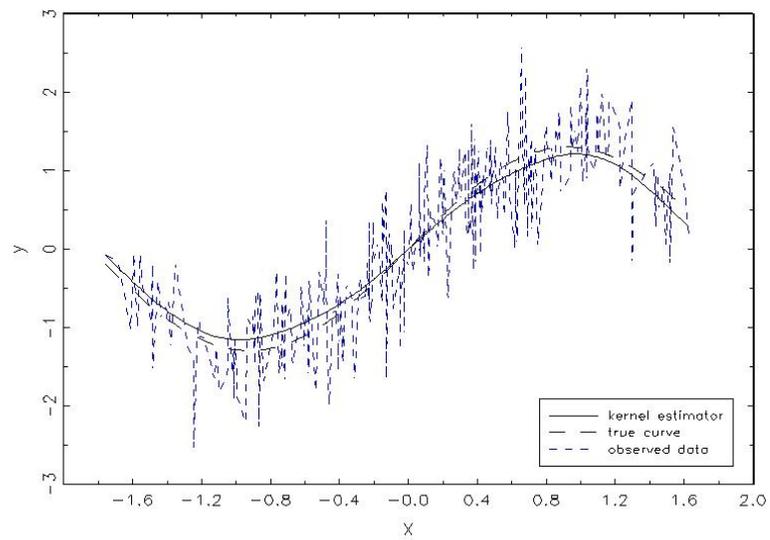


Fig 2.6

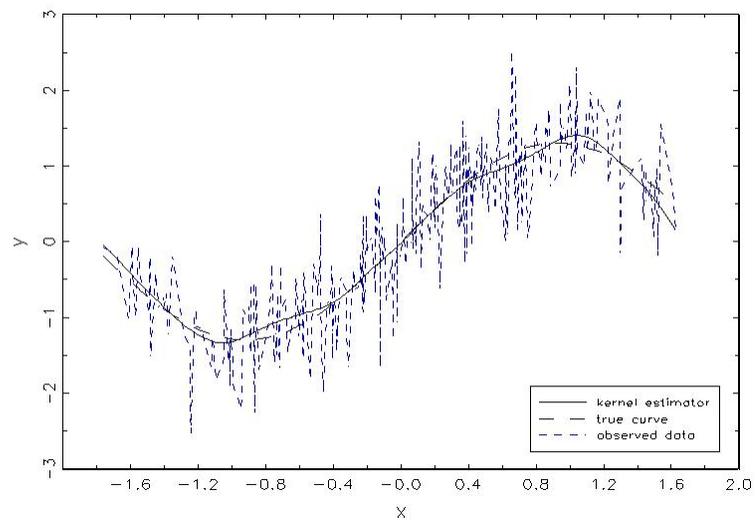
Local Linear Estimator with Plug-In Bandwidth=0.2789 :

MSE=6.8113949e-006

**Fig 2.7**

Local Linear Estimator with Cross-Validation Bandwidth=0.0408 :

MSE=9.4700627e-007



2.5 Nonparametric Volatility Function

There is a generalization for any time series smoothed mean and variance function, m and σ , and estimate both function with nonparametric method. This is called volatility function for

$$y_t = m(y_{t-1}) + \sigma(y_{t-1})u_t, \quad (2.34)$$

the conditional mean and conditional variance are the function of the lag of y_t , where u_t are *i.i.d.* random variables with $E(u_t) = 0$ and $E(u_t^2) = 1$, m and σ are unknown functions on \mathbb{R} , $\sigma(y) > 0$, $y \in \mathbb{R}$ and Y_0 is a random variable independent of $\{u_t\}_{t=1}^T$. The volatility function is then defined by $\sigma^2(x)$.

The objective is to estimate the mean function, $m(x) = E(Y_t | Y_{t-1} = x)$, and the variance function, $\sigma^2(x) = E((Y_t - m(x))^2 | Y_{t-1} = x)$. If $\{y_t\}_{t=1}^T$ is a stationary process, we could have

$$\sigma^2(x) = E(Y_t^2 | Y_{t-1} = x) - (E(Y_t | Y_{t-1} = x))^2. \quad (2.35)$$

To derive the estimator for volatility function, we could show by simple algebra on basic statistical concept to clarify the nonparametric estimation process of conditional volatility.

$$\text{Let } m(x) = E(Y_t | Y_{t-1} = x) \quad (2.36)$$

$$\text{and } g(x) = E(Y_t^2 | Y_{t-1} = x) \quad (2.37)$$

$$\begin{aligned} &= E(m^2(Y_{t-1}) + 2m(Y_{t-1})\sigma(Y_{t-1})u_t + \sigma^2(Y_{t-1})u_t^2 | Y_{t-1} = x) \\ &= m^2(x) + 2m(x)\sigma(x)E(u_t) + \sigma^2(x)E(u_t^2) \\ &= m^2(x) + \sigma^2(x) \end{aligned}$$

Thus, the estimator of volatility function has the form

$$\hat{\sigma}^2(x) = \hat{g}(x) - \hat{m}^2(x) \quad (2.38)$$

Nonparametric volatility function : Local linear estimator with cross-validation bandwidth

In order to define \hat{m} and \hat{g} by the local linear method, consider the following minimization problems with their corresponding estimators:

$$\hat{\beta}_g = \begin{pmatrix} \hat{\beta}_{0g} \\ \hat{\beta}_{1g} \end{pmatrix} \quad (2.39)$$

$$\begin{aligned} &= \arg \min_{(\beta_0, \beta_1)'} \sum_{t=1}^T (Y_t - \beta_0 - (Y_{t-1} - x)\beta_1)^2 K\left(\frac{Y_{t-1} - x}{h}\right) \\ &= \arg \min_{\beta} (Y^2 - \mathcal{X}\beta)' \Omega (Y^2 - \mathcal{X}\beta) \\ &= (\mathcal{X}' \Omega \mathcal{X})^{-1} \mathcal{X}' \Omega Y^2 \end{aligned}$$

$$\hat{\beta}_m = \begin{pmatrix} \hat{\beta}_{0m} \\ \hat{\beta}_{1m} \end{pmatrix} \quad (2.40)$$

$$\begin{aligned} &= \arg \min_{(\beta_0, \beta_1)'} \sum_{t=1}^T (Y_t - \beta_0 - (Y_{t-1} - x)\beta_1)^2 K\left(\frac{Y_{t-1} - x}{h}\right) \\ &= \arg \min_{\beta} (Y - \mathcal{X}\beta)' \Omega (Y - \mathcal{X}\beta) \\ &= (\mathcal{X}' \Omega \mathcal{X})^{-1} \mathcal{X}' \Omega Y \end{aligned}$$

which the last two equalities of both $\hat{\beta}_g$ and $\hat{\beta}_m$ derive from the generalized least squares (GLS) estimator, where

$$\mathcal{X} = T \times 2 \text{ matrix with } t^{\text{th}} \text{ row being } (1 \quad Y_{t-1} - x) \quad (2.41)$$

$$= \begin{pmatrix} 1 & Y_{t-1,1} - x \\ \text{M} & \text{M} \\ 1 & Y_{t-1,T} - x \end{pmatrix}$$

$$\Omega = T \times T \text{ diagonal matrix with } t^{\text{th}} \text{ diagonal element being } K\left(\frac{Y_{t-1} - x}{h}\right)$$

$$Y^2 = \begin{pmatrix} Y_1^2 \\ \text{M} \\ Y_T^2 \end{pmatrix}$$

and $K(\bullet)$ is second order kernel with $\int K(z)dz = 1$, $\int zK(z)dz = 0$ and $\int z^2 K(z)dz = \sigma_k^2$.

Under some assumptions, Härdle and Tsybakov (1997) proved that

$$\hat{\beta}_m(x) \xrightarrow{p} \beta_m(x), \quad (2.42)$$

$$\hat{\beta}_g(x) \xrightarrow{p} \beta_g(x), \quad (2.43)$$

and
$$\hat{\sigma}(x) \xrightarrow{p} \sigma(x), \quad (2.44)$$

where $\hat{\sigma}(x) = \hat{g}(x) - \hat{m}^2(x) = \hat{\beta}_{0g}(x) - \left(\hat{\beta}_{0m}(x)\right)^2$ and $\sigma(x) = g(x) - m^2(x)$.

CHAPTER 3

REVIEW OF RELATED LITERATURES

3.1 Studies on Baht Exchange Rate

Since the currency system was changed from basket to managed-floated system in 1997, the price of Baht currency as the value of exchange rate could be much more respond to the market mechanism. The more flexible system means the exchange rate has allowed to have higher volatility at the acceptable rate of changes. Thitinantapong (2002) also found that floating system is statistically significantly related to the higher volatility in the movement of Baht.

Characteristics of exchange rate

The closer linkage between global financial markets causes the loss of independence in monetary authority's policy and creates a possible contagion effect that quickly infected the rest of Asia. Thus the exchange rate have move more sensitively to the changes of factors. Pimsaen (2000) used Akaike information criteria (AIC) for both forward and spot rates, of rolling periods on 1994-1997 and 1997-1999, to verify the estimation model with one lag for all cases. Augmented Dickey-Fuller (ADF) test can detect the stationarity of both forward and spot rates for the set of samples for Baht against U.S. Dollar, Japan Yen, and Germany Deutschmark. He also found the existence of speculation and suggested that the investors need some risk premium to compensate their risk. Chinprateep (1998) used the monthly data on March 1995 to March 1998 for vector autoregressive model (VAR) to show that the unpredictable monetary news has strongest effect among other economic news and some non-monetary, e.g. cycles, trends, deficit, and inflation, also has statistically significant effects on exchange rate movement.

For application of cointegration and error correction techniques in Thailand macroeconomic modeling, Hataiseree (1995) modeled Purchasing Power Parity (PPP). In his analysis of stationarity test on Augmented Dickey Fuller (ADF) revealed that movements in Baht and main currencies had no long-run equilibrium relationship with the respective differences in inflation. And he also found that Baht/U.S. currency was I(1) which has high volatility and persistent shock. Another critical feature of I(1) process is that its variance is not constant. Khanthavit (2004) studied on Baht/U.S. exchange rate between periods September 6, 2000 to August 6, 2002 with number of observations equals to 500. He found very small conditional mean and not different significantly from zero. The distribution is right skewed (coefficient equates 0.7373) and has fat tail. Compare to zero Kurtosis coefficient of normal distribution, the studying rate of changes of Baht/U.S. exchange rate has coefficient of its distribution very high to 20.9527. Null hypothesis of normality test has rejected at every high-confidence level. Autoregressive degree has found to be one.

3.2 Studies on Econometrics

The estimation of regression function is a pervasive statistical problem in scientific economics. The basic purpose of regression analysis is to study how a variable respond to changes in other variables. This is the intrinsic core of the answer to the question arises in the surprising phenomena of the explosion in the research of econometric subject. The next subsections will introduce the conditional heteroskedasticity process. Then the nonparametric econometric concept studies are also provided. The last subsection discuss on the comparison of econometric models literatures which related to our studies.

3.2.1 Studies on Development of Conditional Heterogeneous

The Costs of Volatility: "Capital-market liberalization is inevitably accompanied by huge volatility, and this volatility impedes growth and increases poverty. It increases the risks of investing in the

country, and thus investors demand a risk premium in the form of higher-than-normal profits. Not only is growth not enhanced but poverty is increased through several channels. The high volatility increases the likelihood of recessions--and the poor always bear the brunt of such downturns.”¹

Sometimes economists might be interest in forecasting not only the level of series but also its variance. Macroeconomics time-series such as foreign exchange rates, inflation rates and stock returns may exhibit volatility which varies over time. This suggests that the variances of these time series may be heteroskedastic. Engle (1982) simultaneously modeled the mean and conditional variance due to the heteroskedasticity by relating the conditional variance of the disturbance term at present to the size of the squared disturbance terms in the past. This function is called an autoregressive conditional heteroskedasticity (ARCH) because the variance is conditional on prior information in an autoregressive form of lagged squared disturbances. The variance of u_t conditional on the information set prior to period t is an autoregressive function of order q in squared lagged values of u_t .

This model is given by

$$\sigma_t^2 = E(u_t^2 | \Omega_t) = \zeta + \alpha_1 u_{t-1}^2 + \alpha_2 u_{t-2}^2 + \dots + \alpha_q u_{t-q}^2 \quad (3.1)$$

$$u_t = \sigma_t \varepsilon_t \quad ; \varepsilon_t \sim iid(0,1)$$

where Ω_t denotes the information set on which σ_t^2 , the variance of u_t , is conditioned. This information set consists of all information prior to recent dated.

The equation represents ARCH (q) process with covariance-stationary condition:

$$\alpha_1 + \alpha_2 + \dots + \alpha_q < 1, \quad (3.2)$$

in which the unconditional variance equals $\text{var}(u_t) = \sigma^2 = \zeta / (1 - \alpha_1 - \alpha_2 - \dots - \alpha_q)$. For the conditional variance to be positive, the parameters must be satisfy $\zeta > 0$ and $\alpha_1 \geq 0, \dots, \alpha_q \geq 0$. Engle (1982) also showed that the simple test for heteroskedasticity can

¹ Stiglitz (2002)

be based on an ordinary F-test which regress the squared OLS residuals with their lagged values and a constant.

Many extensions of the ARCH model have been proposed in the past few decades. The generalized ARCH or GARCH model was suggested by Bollerslev (1986). The GARCH (p, q) model can be written as

$$\begin{aligned}\sigma_t^2 &= E(u_t^2 | \Omega_t) \\ &= \kappa + \delta_1 \sigma_{t-1}^2 + \delta_2 \sigma_{t-2}^2 + \dots + \delta_p \sigma_{t-p}^2 + \alpha_1 u_{t-1}^2 + \alpha_2 u_{t-2}^2 + \dots + \alpha_q u_{t-q}^2\end{aligned}\quad (3.3)$$

with GARCH covariance-stationary condition:

$$\delta_1 + \delta_2 + \dots + \delta_p + \alpha_1 + \alpha_2 + \dots + \alpha_q < 1 \quad (3.4)$$

Covariance stationarity of model is justified if and only if all the roots of $\alpha(L) + \delta(L) = 1$ in (3.4) lie outside the unit circle, in which $\alpha(L)$ and $\delta(L)$ are lag polynomials. The conditional variance in GARCH (p, q) model's positivity constraint is satisfied if and only if all the coefficients in the infinite power series expansion for $\alpha(L)/(1 - \delta(L))$ are nonnegative.

In many financial researches with GARCH model the estimate for $\alpha(L) + \delta(L)$ turns out to be very close to unity. This provided Engle and Bollerslev(1986) extension of GARCH to the class of integrated GARCH (IGARCH) that have the restrictions

$$\delta_1 + \delta_2 + \dots + \delta_p + \alpha_1 + \alpha_2 + \dots + \alpha_q = 1, \quad (3.5)$$

so the autoregressive polynomial has a unit root, and consequently the shock of conditional variance is persistent because stationarity of GARCH process is violated.

In particular for negative stock returns lead to larger stock volatility than equivalent positive returns, Nelson (1991) proposed the Exponential GARCH (EGARCH) for the implication of exponential leverage effect rather than quadratic with guarantee of unconstrained nonnegative conditional variance. The model is

$$\ln \sigma_t^2 = \omega + \delta \ln \sigma_{t-1}^2 + \alpha \left| \frac{u_{t-1}}{\sigma_{t-1}} \right| + \gamma \frac{u_{t-1}}{\sigma_{t-1}}. \quad (3.6)$$

The leverage effect refers to the tendency for changes in returns to be negatively correlated with changes in volatility.

Another route for applying asymmetric effects is to set conditional variance by

$$\sigma_t^2 = \kappa + \alpha u_{t-1}^2 + \alpha^- u_{t-1}^2 d_{t-1} + \delta \sigma_{t-1}^2 \quad (3.7)$$

where d_t is the indicator function which equals to one for negative u_t and zero otherwise.

This is called Threshold GARCH (TGARCH) for allows lopsided response of volatility to factors with different coefficients for good and bad news.

An explicit tradeoff between conditional variance and expected returns was also designed to capture such relationship. Engle et al. (1987) introduced the ARCH-M model allowing the explicit influence of conditional variance term in the conditional mean equation, $y_t = f(\sigma_t, \Omega_t)$ where the derivative of the $f(.,.)$ function with respect to the first element is nonzero. We can measure the effect for perceived volatility of u_t has on the level of y_t .

All of the estimators in any GARCH models can be obtained by nonlinear estimation methods such as MLE, QMLE, or GMM which Hamilton (1994) had reviewed these applications and illustrated by using the numerical methods.

3.2.2 Studies on Nonparametric Econometrics

For consider a class of dynamic models in which both the conditional mean and the conditional variance are unknown functions of the past. Härdle and Tsybakov (1997) derived probabilistic condition and construct the nonparametric estimators based on local polynomial fitting of $y_t = m(x_t) + \sigma(x_t)u_t$, where $u_t \sim i.i.d.(0,1)$. The advantage of such

local polynomial estimators is that they approximated the volatility function better when it is smoother in the nonparametric situation where the exact parametric form of $m(\cdot)$ and $\sigma(\cdot)$ is not predefined. By considering the joint local polynomial estimation of conditional mean and volatility function with derived ergodicity, geometric ergodicity, and mixing properties of a Markov chain process $\{y_t\}$, and examined the rates of convergence of these estimators, they gave the main result of research on joint asymptotic normality by proving that the local linear estimators of volatility function are consistent. Inspection of their proofs shows that the result also holds for the nonparametric regression model with heteroskedasticity disturbances.

Härdle and Tsybakov (1997) also compared the efficiency of local linear estimator between using cross-validation and plug-in bandwidth by employing the prediction mean square error (PMSE) as a criterion for measuring efficiency. The kernel chosen was the quartic one,

$$K(z) = \frac{15}{16}(1-z^2)^2 I_{[-1,1]}(z) \quad (3.8)$$

The conclusion of their finance applications was that the cross-validation bandwidth method has more efficient than the plug-in bandwidth method.

Local polynomial estimators of regular stationary process have joint asymptotic consistency and normality (converge in probability and distribution respectively) and local linear estimator is a kind of local polynomial estimator for polynomial order one. Bossaerts et al. (1995) chose linear estimator in favor of the Nadaraya-Watson (NW) or Gasser-Muller (GM) estimator and applied cross validation bandwidth. They suggested that GM is preferable to NW because it's smaller bias, under fixed design. But variance of GM is worse, under random design. However, local linear estimator combines the advantage of GM and NW by having the same bias as GM and the same variance as NW asymptotically. Hafner (1998) extended the reasoning at this point that local linear estimator corresponds to a local least-square problem for which easy and fast efficient

algorithms are available in practice. And not only the intercept of polynomial estimators be estimated for the regression function, but also its derivatives up to order of polynomial are estimated simultaneously. Bossaerts et al. (1995)'s results showed for all three exchange rates mean reversion and conditional heteroskedasticity.

Hafner (1998) applied a nonparametric ARCH model of order one for volatility function to high-frequency foreign exchange rate (HFFX). Local linear estimation technique was applied with cross validation bandwidth. The results showed significant asymmetry of the volatility function. That is the news impact curves had different shapes for different lags and tend to increase slower at the boundaries. This is like another researches done by Bossaerts et al. (1995) and Härdle and Tsybakov (1997), they found an asymmetry U-shaped “smiling” form of the volatility function, for various applications on exchange rates, because the abnormal observations from conditional mean are highly correlated with the volatility. This is called “reverted leverage effect” meaning that the conditional variance is higher for positive lagged returns than for negative ones of the same size.

In practical computation, Buhlmann and Mcneil (2002) proposed the theoretical justification for the iterative algorithm and examples of its application for nonparametric first-order GARCH modeling. This paper considered the model that is very general for any ARCH-series models. An estimation algorithm step-by-step was described and examples were provided in both simulation and empirical examples.

3.2.3 Literatures on Models Comparison

For compare the models in term of in-sample and out-of-sample test to study both the parametric and nonparametric GARCH models of financial application, there are some literatures find out the conclusion of these results. Pagan and Schwert (1990) compared several statistical models, both parametric and nonparametric, for U.S. stock

return volatility. With covariance stationary did not rejected at small significant levels, monthly stock returns from 1834-1925 was used to concentrate on models comparison. Parametric models (two-step conditional variance, GARCH(1,2), EGARCH(1,2), Markov switching-regime) tended to give an inferior explanation of the squared returns than any nonparametric models (Gaussian kernel and flexible Fourier form(FFF)). Both GARCH and Markov switching-model produced weak explanations. EGARCH had explanatory power close to the nonparametric models, because its parameter allowed the effect of stock's asymmetric behavior between past returns and volatility. Although the superior of nonparametric models in the in-sample-test, they was worse than the parametric models for the out-of-sample prediction of conditional variances, because of too much variability in the estimates of σ_t^2 . Pagan and Schwert (1990) suggested the results implied that standard parametric models are not sufficient and augmenting these models with nonparametric methods could yield significant increase in explanatory power.

West and Cho (1995) compared the predictive ability of six models which they used squared return as a proxy for a (population) variance conditional on information generated by past return, using U.S. Dollar versus currencies of Canada, France, Germany, Japan, and the United Kingdom's weekly data of 1973-1989. The models include homoskedastic, GARCH(1,1), IGARCH(1,1), AR(12) in u_t^2 , AR(12) in $|u_t|$, and nonparametric Gaussian kernel to compare the out-of-sample realization of the square of the weekly change in exchange rates for horizon of one, twelve, and twenty-four weeks.

These models of conditional variance also have been formally tested for equality of mean squared prediction error (MSPE) across models, by chi-squared asymptotic inferences on three hypotheses: (i) MSPEs for all models are equal, (ii) MSPEs for the best model and homoskedastic model are equal, (iii) MSPEs for the homoskedastic, GARCH(1,1), and two autoregressive are equal. For twelve- and twenty-four-week-ahead forecasts of the squared weekly change were difficult to choose any model over another. But at one-week horizon, GARCH's MSPE tend to have more predictive power.

However, statistical tests cannot reject at conventional significant levels for the null that GARCH's MSPE is equal to other models.

Please note that these two comparative models literatures of Pagan and Schwert (1990) and West and Cho (1995) selected the local constant estimator with the plug-in bandwidth, for Gaussian kernel,

$$h = \hat{\sigma}(R - j)^{\frac{1}{5}}, \quad (3.9)$$

where $\hat{\sigma}$ = sample standard deviation of u_t ; $t = 1, \dots, R - j$

R = endpoint of first sample used in estimation of regression parameters

$j = 1, 12, 24$.

For make amendments to field of conditional heteroskedasticity process study, Hafner (1998)'s comparison of the conditional variances estimation focused on the asymmetry and persistence issues. Since his estimation results for parametric models confirmed standard results for high-frequency foreign exchange rate, namely no significance of the asymmetry coefficient in an EGARCH model and high persistence in an I-GARCH model. To find out whether these outcomes are robust against alternative specification of nonparametric model, local linear estimation was applied with bandwidth chosen by cross-validation criteria. Here, a bounded, symmetric quartic kernel function was used. The results from nonparametric ARCH model of order one showed significant asymmetry of the volatility function. In according to the EGARCH specification, the news impact curves have different shapes for different lags and tend to increase slower at the boundaries. That is, the variance function they estimated was skewed and thus revealed asymmetry. Volatility increased more for a large increase of rat than for a large decrease of the same size. This is similar to Thailand currency, according to Khanthavit (2004), he concluded that asymmetric autoregressive conditional heteroskedasticity models should be the most appreciated description of exchange rate volatility features.

CHAPTER 4

THEORETICAL FRAMEWORK AND METHODOLOGY

4.1 Autoregressive Conditional Heteroskedasticity Processes

The general autoregressive process of $\{Y_t\}_{t=1}^T$ with Gaussian white noise sequence, $\{u_t\}_{t=1}^T$; $u_t \sim N(0, \sigma_t^2)$, is used in the analytical and numerical study.

The parametric regression model $Y_t = Y_{t-1}'\beta + u_t$ will be the initial model to study. We could have $\{u_t\}_{t=1}^T$ with some predetermined forms of any conditional heteroskedasticity functions, i.e.

$$\text{ARCH: } \sigma_t^2 = \zeta + \alpha u_{t-1}^2 \quad (4.1)$$

$$\text{GARCH: } \sigma_t^2 = \kappa + \delta \sigma_{t-1}^2 + \alpha u_{t-1}^2 \quad (4.2)$$

$$\text{I-GARCH: } \sigma_t^2 = \kappa + \delta \sigma_{t-1}^2 + \alpha u_{t-1}^2 \text{ with } \delta + \alpha = 1 \quad (4.3)$$

$$\text{EGARCH: } \ln \sigma_t^2 = \omega + \delta \ln \sigma_{t-1}^2 + \alpha |u_{t-1} / \sigma_{t-1}| + \gamma (u_{t-1} / \sigma_{t-1}) \quad (4.4)$$

$$\text{TGARCH: } \sigma_t^2 = \kappa + \alpha u_{t-1}^2 + \alpha^- u_{t-1}^2 d_{t-1} + \delta \sigma_{t-1}^2; d_t = 1 (u_t < 0) \text{ or } 0 \text{ (otherwise)} \quad (4.5)$$

$$\text{ARCH-M: } \sigma_t^2 = \zeta + \alpha u_{t-1}^2 \text{ with } Y_t = Y_{t-1}'\beta + \xi \sigma_t^2 + u_t \quad (4.6)$$

Assume the traditional assumption about the serial dependence of u_t , $u_t = \sigma_t v_t$; $v_t \sim i.i.d. N(0,1)$ and v_t is independent of Y_t, Y_{t-1} , then the conditional distribution of Y_t is Gaussian with mean $Y_{t-1}'\beta$ and variance σ_t^2 ;

$$f(Y_t | Y_{t-1}, Y_{t-2}, \dots, Y_1) = \frac{1}{\sqrt{2\pi\sigma_t^2}} \exp\left(\frac{-(Y_t - Y_{t-1}'\beta)^2}{2\sigma_t^2}\right) \quad (4.7)$$

Thus, any parametric functions of conditional heteroskedastic volatility could be estimated numerically by using MLE methods to obtain the unknown parameters. And

these parametric estimates from this maximization problem will be used to forecast $\hat{Y}_{T+j}, \hat{\sigma}_{T+j}^2 | Y_t, \dots, Y_T$ for $t = 1, \dots, 50; T = 214, \dots, 263; j = 1, \dots, 50$.

4.2 Nonparametric Volatility Function:

Local Linear Estimator with Cross-Validation Bandwidth

The nonparametric form of volatility function for this study is $y_t = m(y_{t-1}) + \sigma(y_{t-1})u_t$, with volatility function estimator: $\hat{\sigma}^2(x) = \hat{g}(x) - \hat{m}^2(x)$ where $m(x) = E(Y_t | Y_{t-1} = x)$ and $g(x) = E(Y_t^2 | Y_{t-1} = x)$.

Calculate optimal cross-validation bandwidth

As previous discussed in chapter two that the cross-validation bandwidth minimizes the cross-validation criterion: $CV(h) = \frac{1}{T} \sum_{t=1}^T (Y_t - \hat{m}_{-t}(y_{t-1}; h))^2$ where \hat{m}_{-t} is a computed kernel estimate from the leave-one-out data set: $(Y_1, Y_2), \dots, (Y_{t-2}, Y_{t-1}), (Y_t, Y_{t+1}), \dots, (Y_{T-1}, Y_T)$, and \hat{m}_{-t} is the best mean squared error predictor of Y_t given the leave-one-out data set.

So we would expect the optimal smoothing parameter, h^* , for this best predictor in the class $\{\hat{m}_{-t}(y_{t-1}; h^* : h^* > 0)\}$ to be the one that's close to the conditional mean of Y_t , $m(Y_{t-1})$.

So in practice, numerical optimization by Newton algorithm in CML module in GAUSS program could yield the computed sum of squared residuals such that

$RSS(h^*) = \sum_{t=1}^T (Y_t - \hat{m}_{-t}(y_{t-1}; h^*))^2$ is minimized. The bandwidth, h^* , that derived from this

optimization is completely automatic.

Local linear estimation in the nonparametric volatility function.

The cross-validation bandwidth, h^* , derived from *RSS* minimization will be used for the local linear estimators of nonparametric volatility function. The recursive iteration method search for

$$\hat{\beta}_g = \begin{pmatrix} \hat{\beta}_{0g} \\ \hat{\beta}_{1g} \end{pmatrix} \equiv \left(\sum_{t=1}^T K\left(\frac{Y_{t-1} - y_{t-1}}{h^*}\right) \begin{pmatrix} 1 \\ Y_{t-1} - y_{t-1} \end{pmatrix} (1 \quad (Y_{t-1} - y_{t-1})') \right)^{-1} \sum_{t=1}^T K\left(\frac{Y_{t-1} - y_{t-1}}{h^*}\right) \begin{pmatrix} 1 \\ Y_{t-1} - y_{t-1} \end{pmatrix} Y_t^2 \quad (4.8)$$

$$\hat{\beta}_m = \begin{pmatrix} \hat{\beta}_{0m} \\ \hat{\beta}_{1m} \end{pmatrix} \equiv \left(\sum_{t=1}^T K\left(\frac{Y_{t-1} - y_{t-1}}{h^*}\right) \begin{pmatrix} 1 \\ Y_{t-1} - y_{t-1} \end{pmatrix} (1 \quad (Y_{t-1} - y_{t-1})') \right)^{-1} \sum_{t=1}^T K\left(\frac{Y_{t-1} - y_{t-1}}{h^*}\right) \begin{pmatrix} 1 \\ Y_{t-1} - y_{t-1} \end{pmatrix} Y_t \quad (4.9)$$

Thus, from the recurrence of iterated algorithm, we could find the forecast of $\sigma^2(\bullet)$ based on Y_t equivalent to

$$\hat{\sigma}^2(y_{t-1}) = \hat{g}(y_{t-1}) - \hat{m}^2(y_{t-1}) = \hat{\beta}_{0g}(y_{t-1}) - \left(\hat{\beta}_{0m}(y_{t-1}) \right)^2. \quad (4.10)$$

4.3 Data Availability and Description

This study will employ weekly data of Baht/U.S.\$ rate in 2000-2005. Since there is no historical weekly data publicly announcement available, so the weekly data could be obtained by taking average the daily spot rate among the opening days for each week. This is the same as the Bank of Thailand's calculation procedure for monthly exchange rate by average over all opening days in that month. For the period between January, 2000 to January, 2005 of 1,325 daily spot rates data available was downloaded from the Federal Reserve Bank of New York. Because this released series provides zero for closing dates so it's much easier to written the weekly average calculation program. This customized program could yields output of 264 observations of percentage change in the level of Baht/U.S. exchange rate, that is $100 * (\ln(ER_t) - \ln(ER_{t-1}))$ where ER_t is the

exchange rate at time t . So the series for variables in studying models are $Y_t = 100 * (\ln(ER_t) - \ln(ER_{t-1}))$ and $X_t = Y_{t-1}$.

Descriptive Statistics

The scope of our study is to compare the predictive performance between nonparametric and parametric models by comparing both for the level and the volatility of weekly percentage Baht/U.S. rate of return in 2000-2005 and, also, to incorporate with in-sample properties. Figure 4.1 represents the series which we used in study. The investigation on characteristics of data in this section will explore for next chapter of both parametric and nonparametric approaches studies.

Fig 4.1
Percentage Weekly Baht/U.S. Return, 2000-2005, with 264 observations

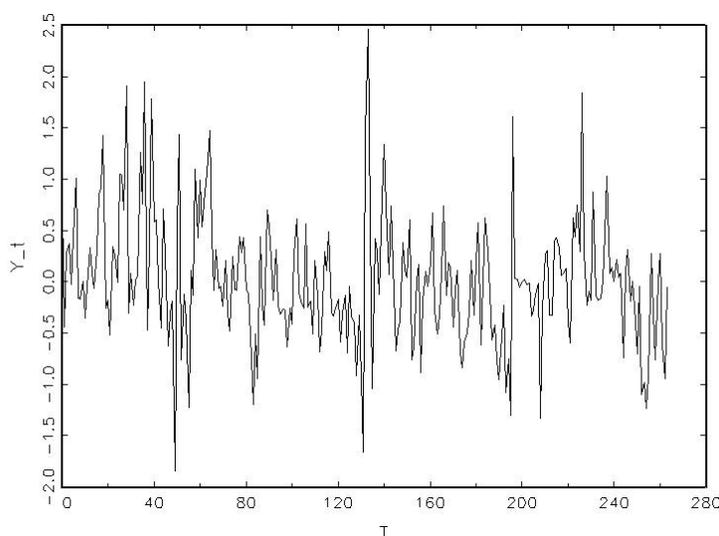


Table 4.1 represents the descriptive statistics for the distribution of percentage return of Baht/U.S. rate. We can see that the expectation mean of series is very small (0.126E-04) and right skewed bias because positive coefficient of skewness (0.561577).

These imply the asymmetric probability of distribution. The measure of peakness,

coefficient of kurtosis, which is 4.478065 (higher than 3.00 in normal distribution case) imply more peaked probability distribution function or leptokurtic. The Jarque-Bera statistic decisively rejects the hypothesis of normal distribution.

Table 4.1

Descriptive Statistics

Mean	0.012550
Median	-0.044315
Maximum	2.463100
Minimum	-1.845640
Std. Dev.	0.624152
Skewness	0.561577
Kurtosis	4.478065
Jarque-Bera	37.90766
Probability	0.000000

Number of observations = 265

Figure 4.2

ACF and PACF functions

Aurocorrelation	Partial Correlation	AC	PAC	Q-Stat	Prob	
		1	0.310	0.310	25.687	0.000
		2	0.094	-0.003	28.037	0.000
		3	0.146	0.130	33.785	0.000
		4	0.095	0.014	36.208	0.000
		5	0.023	-0.018	36.353	0.000
		6	0.057	0.044	37.246	0.000
		7	0.061	0.021	38.260	0.000
		8	0.050	0.026	38.945	0.000
		9	0.002	-0.034	38.946	0.000
		10	0.054	0.056	39.750	0.000
		11	0.113	0.082	43.323	0.000
		12	0.058	0.000	44.265	0.000
		13	-0.024	-0.061	44.431	0.000
		14	0.002	-0.004	44.433	0.000
		15	0.012	0.000	44.471	0.000
		16	-0.002	0.003	44.473	0.000
		17	0.068	0.076	45.787	0.000
		18	0.058	0.004	46.734	0.000
		19	0.001	-0.024	46.735	0.000
		20	-0.033	-0.042	47.047	0.001
		21	-0.070	-0.074	48.469	0.001
		22	0.040	0.084	48.927	0.001
		23	0.034	0.006	49.261	0.001
		24	-0.007	0.002	49.274	0.002

Model selection criteria by autocorrelation and partial autocorrelation functions could verify that the lag of return series has statistical relation on current return series at autoregressive degree one. As it is show in figure 4.2. However, the full observations in-

sample-test of all possible models, i.e. AR(1) with or without constant, show that the best model is AR(1) without constant. Because the constant in AR(1) mean equation is not significant at all level of standard test level (constant estimator's p-Statistic = 0.8721). So the percentage rate of Baht/U.S. return series used in study will be based on using first-order autoregressive without constant as the main function, $Y_t = \beta Y_{t-1} + u_t$.

Table 4.2
OLS AR(1) and its statistics

$Y_t = 0.3104Y_{t-1}$		
s.d. = (0.078640)		
p-value = [0.000051]		
$Y_t = 0.310179 + 0.000059Y_{t-1}$		
s.d.= (0.000366) (0.058688)		
p-value = [0.8721] [0.000000]		
Note: (.) = standard deviation and [.] = p-value		

Although the null hypothesis of having the unit root has rejected with very small p-value of Augmented Dickey-Fuller test with deploying of constant and linear trend as exogenous variables (1.82e-21 or shown as 0.0000 in table 4.3).

Table 4.3
Unit Root test and ARCH-LM test

Unit Root Test	Statistic	Probability
Augmented Dickey-Fuller t-test	-12.14690	0.000000
Test Critical Values :		
1% level	-3.993335	
5% level	-3.427004	
ARCH Test	Statistic	Probability
T*R-squared	14.56590	0.000135
F-statistic	15.30563	0.000117

LM-test hypothesis is the test of none heteroskedasticity, which the test is being asymptotical chi-squared test. The corresponding result on test of the relationship between residuals from autoregressive degree one's OLS squared residual and its lags show that there exists the heteroskedasticity at very high significant level (p-statistic is 0.000136) which shown in table 4.3. The regression's F-statistic and its p-statistic are also has the results according to LM-test.

In conclusion, the weekly percentage rate of return in study is not distributed as normal with right skewed and leptokurtic. The autocorrelation function and OLS statistics suggest that the base model for study is the AR(1) without constant model. And the last important feature of this series is the heteroskedasticity relationship. So these facts are the reasoning to the next chapter of conditional heteroskedasticity models study.

4.4 Predictive Performance Comparison

The objectives of this study are to examine the differences between nonparametric and parametric volatility models estimation and also to compare predictability performance. The series used in prediction from various models are composed of estimated return and estimated conditional heteroskedasticity series, so that we need to find the true basis for compare with the estimated series. The return is exquisitely since it's already available by obtaining and manipulating data as previously described to use for this study. However, conditional variance is not such be as this case. There're many literatures such as Pagan and Schwert (1990), West and Cho (1994), and Khanthavit (1997) use the unconditional sample standard variance as the base to compare with their estimated volatility. But the comparison between conditional and unconditional may not valid in theory. So this study proposes another base for compare with the estimated conditional heteroskedasticity from any models as the alternative way.

The proposed base is the squared return with theoretical reasoning behind. Because from $y_t = m(y_{t-1}) + \sigma_t u_t$, where $m(y_{t-1})$ and σ_t^2 are any unknown functional form with $u_t \sim iid(0,1)$, one can show that

$$var(y_t | y_{t-1}) = var(m(y_{t-1}) + \sigma_t u_t | y_{t-1}) = var(\sigma_t u_t | y_{t-1}) = \sigma_t^2 \quad (4.11)$$

So we use (population) variance of y_t , $E_t y_t^2$, conditional on information generated by past return as an estimator of $var(y_t | y_{t-1})$. West and Cho (1995) was a paper that came to similar conclusion to employed squared return. We therefore will limit ourselves to models in which the conditional mean of y_t is zero.

However, the traditional unconditional variance derived from sample standard variance, as in Pagan and Schwert (1990) and Khanthavit (1997), will not be neglected to use as a base case. The computed sample standard variance is

$$\sum_{i=1}^{50} (\{y_{t+i}^T\} - E\{y_{t+i}^T\})^2 \frac{1}{213} \quad (4.12)$$

for $t = 2, \dots, 51; T = 215, \dots, 264; i = 1, \dots, 50$ and $\{y_t^T\} = y_t, \dots, y_T$. This is just a simple rolling sample standard deviation for different 50 information sets.

So there are three types of forecast comparison by rolling mean square prediction error, that are (1) compare between true and estimated return, (2.1) compare between unconditional and estimated conditional variance, and (2.2) compare between estimated conditional variance and the squared return. The results will be presented in next chapter.

CHAPTER 5

EMPIRICAL RESULTS

The empirical results of comparisons between nonparametric and parametric econometric models as the objectives of study are reported in this chapter. They are partitioned into three parts of in-sample estimation, out-of-sample performance and test of unbiasedness respectively. All are analyzed in comparison between the previous proposed competitive models.

5.1 Introduction to Estimation Results

The nonparametric model is the volatility function estimated by local linear estimator with cross-validation bandwidth, which is the debut of nonparametric econometrics approach in empirical literature on Thailand exchange rate. The parametric models used in this study include Autoregressive Conditional Heteroskedasticity (ARCH), Generalized Autoregressive Conditional Heteroskedasticity (GARCH), Threshold GARCH (TGARCH), Integrated GARCH (I-GARCH), Exponential GARCH (EGARCH), and ARCH in Mean (ARCH-M). ARCH and GARCH are the common models in time series econometrics. While TGARCH and EGARCH are the asymmetric autoregressive conditional heteroskedasticity models that could describe the skewed distribution of volatility. I-GARCH model is for the volatility persistent feature of series. ARCH-M can capture the trade-off relationship between the expected returns and conditional variance. All the parametric models will have the various predetermined structural form of conditional variance equation such that we have the disturbance error equal to $u_t = \sigma_t \varepsilon_t ; \varepsilon_t \sim iid(0,1)$.

We will estimate the unknown parameters of variance equations in case of parametric models. These parametric series will be used for weighted the observations of regression to obtain the generalized least squared (GLS) estimator of autoregressive

degree one function with out constant. On the other way, we'll estimate the unknown functional form of regression function by nonparametric approach to derive the conditional heteroskedastic series.

An important objective of study is to be search the best predictive performance model among all the competitive models. Then, for out-of-sample criteria, we will use rolling mean squared prediction error (rolling MSPE) as the measurement of predictive performance. Rolling MSPE means that each of the squared prediction errors computation will have equal number of observations usages in predicting of the (t+1)th observation.

For yearly prediction error calculation of full-sample 264 observations in AR(1), this iterated recurrences will be done for 50 times to calculate rolling MSPE, so we will use 1st to 213th observation for the first round and rolling by one observation for each additional round of iteration until the last round use 51st to 263rd observation, i.e. $\Psi_1 = (y_t = Y_i, x_t = Y_j; i = 2, \dots, 214, j = 1, \dots, 213)$, $\Psi_2 = (y_t = Y_i, x_t = Y_j; i = 3, \dots, 215, j = 2, \dots, 214)$, ..., $\Psi_{50} = (y_t = Y_i, x_t = Y_j; i = 51, \dots, 263, j = 50, \dots, 262)$ where Y_t is the weekly percentage rate of return and Ψ_t is the information set.

However, the estimations of these models have the results as shown below use the full sample observations. These tables are parametric volatility estimations of percentage weekly Baht/U.S. return in 2000-2005 with all-263 observations. Each set of tables for each model will compose of the regression coefficients and statistics for conditional mean equation, conditional variance equation, and the traditional unbiasedness test equation. To test the unbiasedness for each model, the regression $\hat{u}_t^2 = \alpha + \beta \hat{\sigma}_t^2 + \xi_t$ will test (i) $\alpha = 0$ and (ii) $\beta = 1$. Because the conditional expectation of squared estimated residual should be equal to the conditional variance or $E_t \hat{u}_t^2 = \hat{\sigma}_t^2$. However, the simple OLS

model of AR(1) without constant will haven't the result of unbiasedness test since its conditional variance is given to be constant by a classical regression assumption.

For each model, the log-likelihood was maximized numerically using the optimization program CML (Constrained Maximum Likelihood) from the GAUSS programming language. The constraints will be imposed to restrict the stationarity and positivity condition for all parametric models. And only positivity constraint is restricted for optimal cross-validation bandwidth of nonparametric model as previously discussed in chapter 3. For estimation and prediction purposes, there are different programs to achieve the purposes. These customized programs are tested by the simulated data generation with the predetermined structure of equations. The program testing are satisfied if the estimators converge to the predetermined values of coefficients.

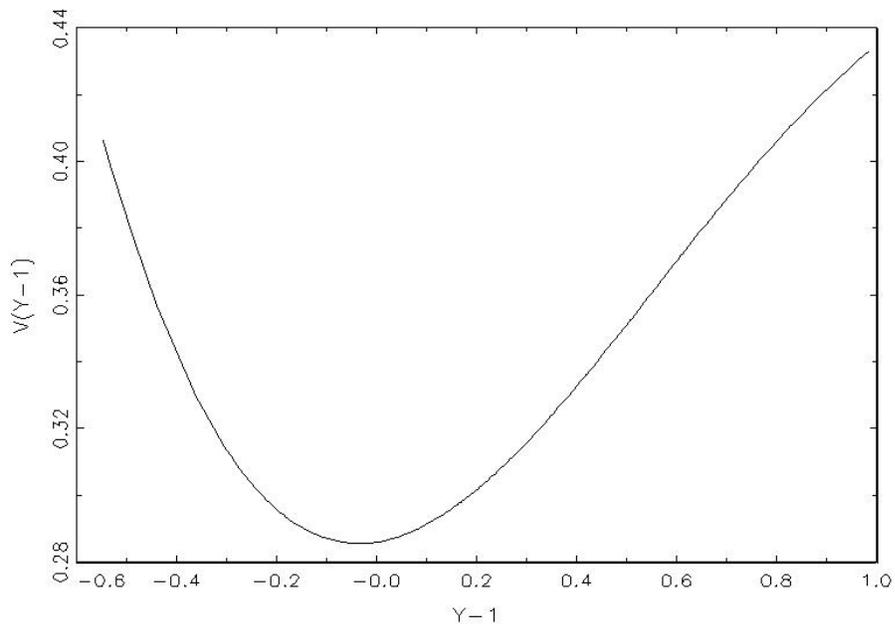
5.2 In-Sample Estimation

The in-sample evidence in table 5.2 to 5.7 are the parametric estimators and statistics of each parametric heteroskedasticity models. These are the results of full sample estimation with 263 observations for AR(1) mean equation and various parametric conditional variance equations. The out-of-sample illustrations in figure (b) and (c) are also represented accordingly with the in-sample tables for clear inspection of the estimation results. Please note that for the variance - covariance matrix correction of estimator inefficiency, we use White's Heteroskedasticity Consistent Covariance for appropriate robust inferences.

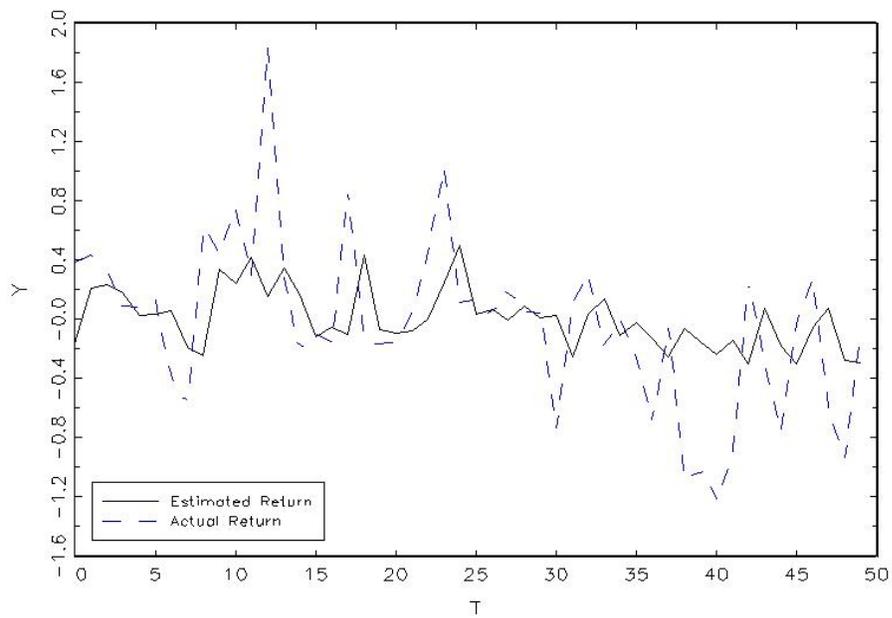
Altogether, the descriptive statistics in previous chapter and the time-varying variances illustration in Figure 5.1 (a) show that nonparametric volatility function models could capture the features of percentage weekly Baht/U.S. return in 2000-2005 which its scattered distribution is asymmetric and there exists heteroskedasticity.

Fig 5.1

(a) Nonparametric estimated volatility



(b) Nonparametric estimated and true return



(c) Nonparametric estimated volatility and squared return

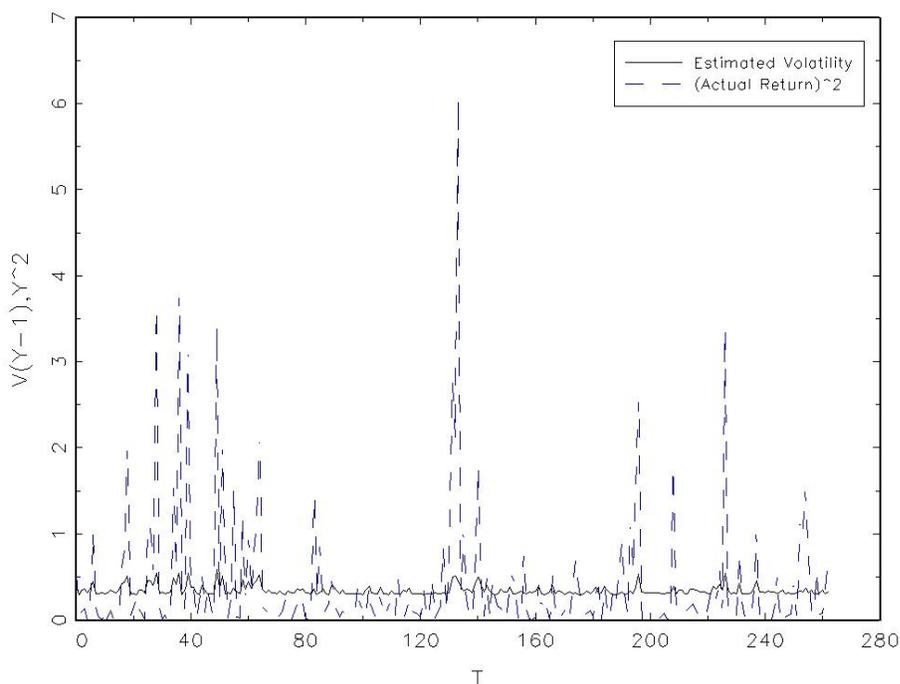


Figure 5.1 (a) show that the nonparametric estimated volatility function is skewed and thus reveals asymmetry as display a U-shaped structure, as Härdle and Tsybakov (1997) and Bossaerts et al. (1995) called a “smiling face”. The fluctuation of percentage return will be higher when it is not around its mean. We can observe that downward movements in the weekly Baht/U.S. percentage return are followed by higher volatilities than upward movements of the same size of changes. That is the abnormal return will have more volatility especially for the negative abnormal. And figure 5.1 (a) also illustrates that the risks of returns are much higher for extreme values taken on the past day. This asymmetric volatility function is smoothed and estimated by cross-validation criteria with local linear estimator. h^* for $\hat{g}(y_{t-1})$ and $\hat{m}(y_{t-1})$ in equation (4.10) are 0.5735 and 0.6967 respectively.

Table 5.1 reports the result for OLS estimation of model AR(1) without constant. The estimated coefficient is 0.3104 with very high significant. However, we are already discussing about the existence of the heteroskedasticity feature of this model in chapter 4

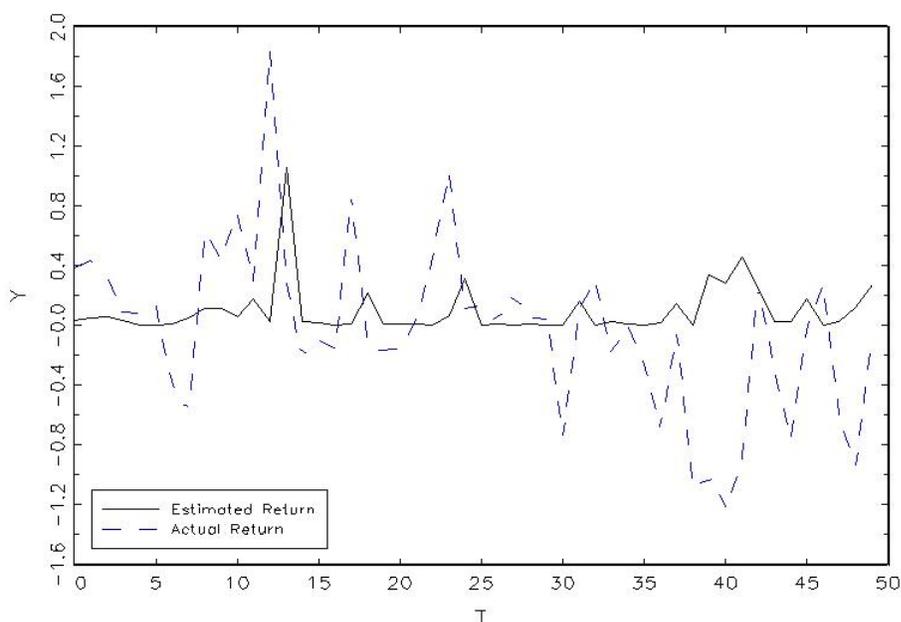
of descriptive statistics table. Therefore this ordinary least square estimator is not an efficient one. In figure 5.1 of OLS estimated and true percentage rate of Baht/U.S. return, we can see that the normal OLS predictors are quite different from the actual values. Although the least square estimator is asymptotically unbiased, but in this empirical of small sample size we can find inefficient of prediction.

Table 5.1
Estimation output for OLS

$Y_t = 0.3104Y_{t-1}$			
s.d. = (0.0786)			
p-value = [0.0000]			
R-squared	0.0965	Mean dependent variable	0.0098
Adjusted R-squared	0.0965	S.D. dependent variable	0.6238
S.E. of regression	0.5929	Akaike info criterion	0.0082
Sum squared residual	92.1100	Schwarz criterion	0.0098
Log likelihood	-0.0828	F-statistic	27.9000

Note: (.) = standard deviation and [.] = p-value

Fig 5.2
OLS estimated and true return



The next tables until the last one are the results of nonparametric conditional heteroskedasticity models which the unknown parameters are estimated from the maximum likelihood estimation method. Starting the initial value for the mean equation from the simple OLS estimator and for the variance equation start from AR(1) estimator of the squared residuals process. We also impose covariance stationarity and positivity constraints as the previous discussion in chapter 3. The experiences from trials and error learning procedure show that without these constraints, the estimations may be diverge or possibly to obtain the irrational results.

ARCH model given in table 5.2 show that all estimated parameters are significant at very high level with the coefficient of mean equation is very close to of simple OLS model. Its variance equation has significant constant and slope coefficient equate 0.278 and 0.196 respectively.

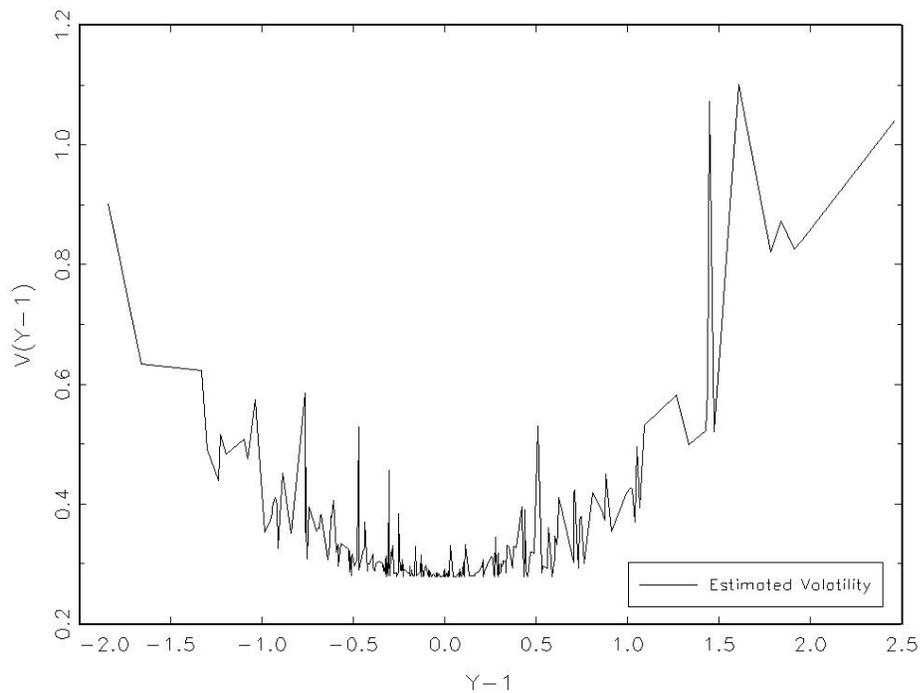
Table 5.2
Estimation output for ARCH

$Y_t = 0.3386Y_{t-1}$			
s.d. = (0.0671)			
p-value = [0.0000]			
$\sigma_t^2 = 0.2780 + 0.1961u_{t-1}^2$			
s.d. = (0.0300) (0.0786)			
p-value = [0.0000] [0.0132]			
R-squared	0.1054	Mean dependent variable	0.0098
Adjusted R-squared	0.1054	S.D. dependent variable	0.6238
S.E. of regression	0.5932	Akaike info criterion	0.0142
Sum squared residual	92.1900	Schwarz criterion	0.0158
Log likelihood	-0.8685	F-statistic	30.7521

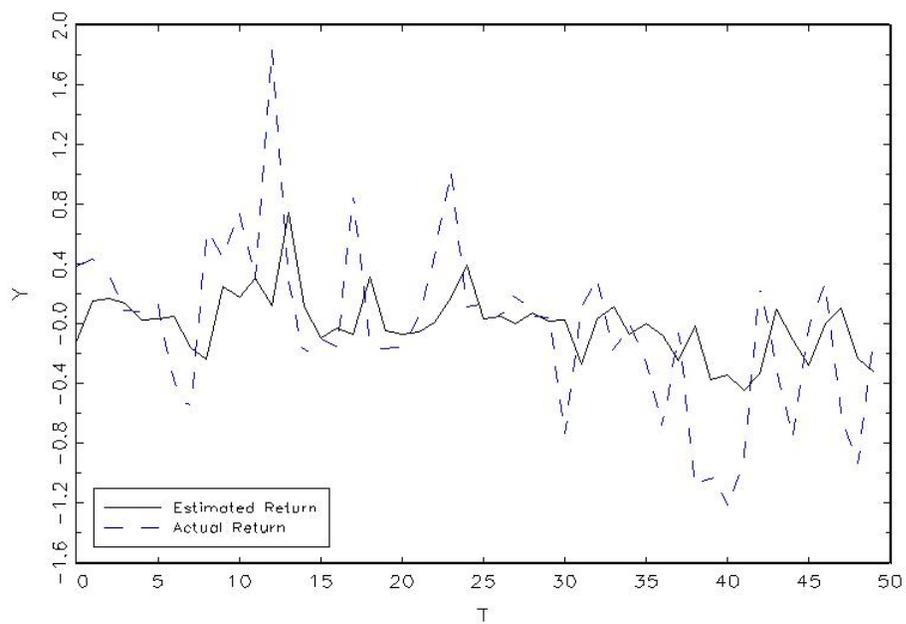
Note: (.) = standard deviation and [.] = p-value

Fig 5.3

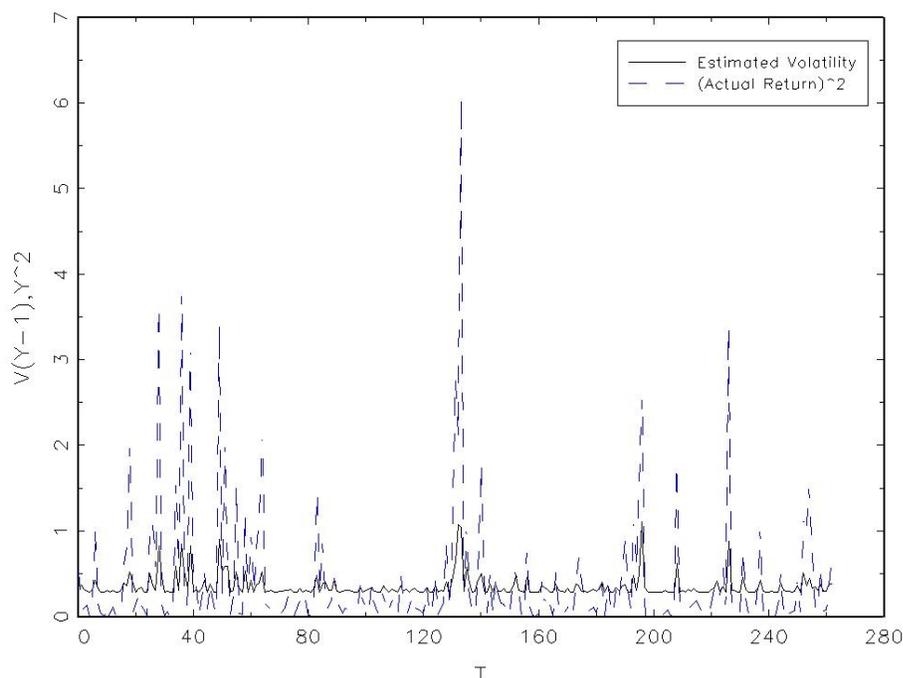
(a) ARCH estimated volatility



(b) ARCH estimated and true return



(c) ARCH estimated volatility and squared return



The prediction of percentage return with ARCH model as illustrated in figure 5.3 (b) seems to be much more fitted to the actual values than previous figure of simple OLS model prediction. ARCH estimated volatility function in figure 5.3 (a) show the rough curve which the value of percentage return around its mean will not have large vary in its corresponding variance. In this figure, we can easily seen that the variance of variance is very small around its return's mean. Figure 5.3 (c) depicts the estimated volatility compared with squared return. The movements are reflex in the same motion but the squared return has greater versatility than ARCH volatility.

Table 5.3 is the GARCH result that the coefficients of lagged conditional variance and lagged squared residual estimation yield summation turns out to be close to unity implies the volatility persistence. This is consistent with the common results from another studies of financial applications. Since this summation between the coefficients of lagged conditional variance and lagged squared residual is the measure for volatility persistence. Persistence increases as this summation approaches one.

However, the summation equates to 0.795 which seems not too high as in some applications which are higher than 0.9 i.e. Pagan and Schwert (1990), West and Cho (1995) and Engle (2002). All parameters of GARCH model estimation are statistically significant, except the constant of variance equation has p-value equates 0.06689.

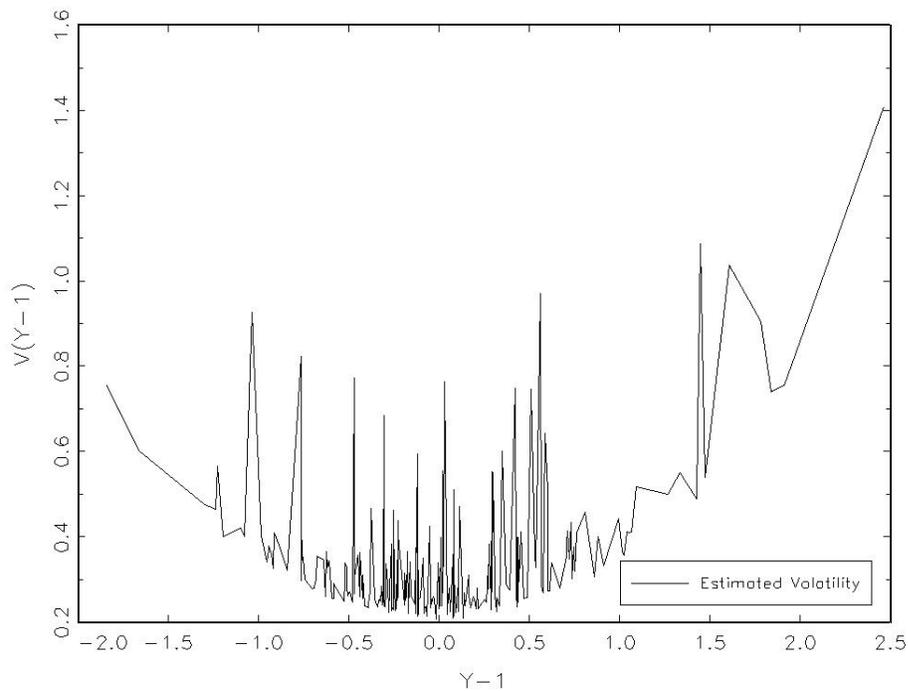
The prediction performance illustration of GARCH model in figure 5.4 (b) is also similar to of ARCH model in the sense that it could captures the true percentage return much better than the simple OLS model. The similarity between ARCH and GARCH also occurred to the conditional heteroskedasticity prediction in which the plotted volatility function reveals the asymmetric curve with rough movement and tend to have higher value for the return which more deviated from its mean. This is resulted in figure 5.4 (a). The squared return which used as proxy of true variance is also has the same motion of movement as the GARCH predicted conditional volatility as in figure 5.4 (c).

Table 5.3
Estimation output for GARCH

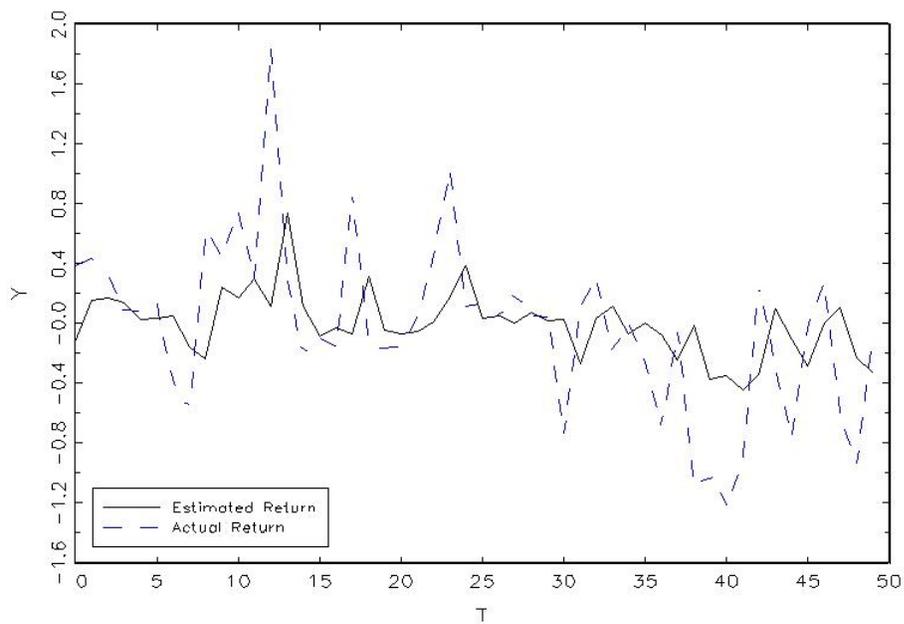
$Y_t = 0.3191Y_{t-1}$			
s.d. = (0.0624)			
p-value = [0.0000]			
$\sigma_t^2 = 0.0710 + 0.1621u_{t-1}^2 + 0.6333\sigma_{t-1}^2$			
s.d. = (0.0472) (0.0643) (0.1733)			
p-value= [0.1337] [0.0123] [0.0003]			
R-squared	0.0993	Mean dependent variable	0.0098
Adjusted R-squared	0.0993	S.D. dependent variable	0.6238
S.E. of regression	0.5930	Akaike info criterion	0.0141
Sum squared residual	92.1200	Schwarz criterion	0.0157
Log likelihood	-0.8556	F-statistic	28.7745

Note: (.) = standard deviation and [.] = p-value

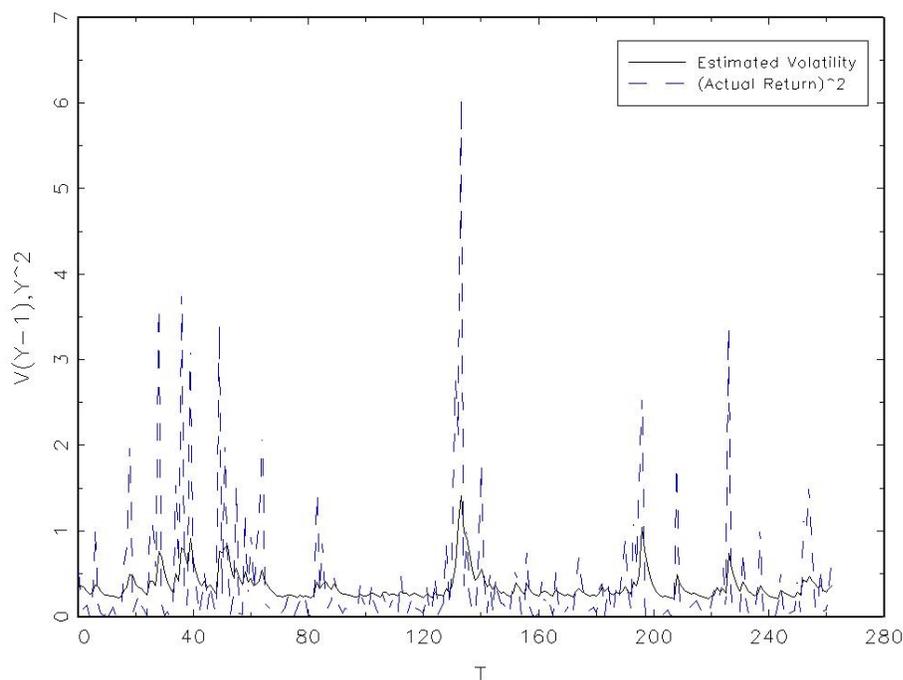
Fig 5.4
(a) GARCH estimated volatility



(b) GARCH estimated and true return



(c) GARCH estimated volatility and squared return



Because it is often observed that the realized return below the mean of the market are followed by higher volatility than realized return above the mean for the same magnitude of dispersion. To account for this phenomenon, we propose asymmetric GARCH models that describe the asymmetric response to good and bad news. These models are Threshold GARCH (TGARCH) and Exponential GARCH (EGARCH)

TGARCH (table 5.4) and EGARCH (table 5.5) results are co-consistent altogether in the sense that the negative response of bad news of both models are not substantial. Although their size and direction are rational but their statistics are not significant. The coefficient of negative residuals in TGARCH model equates to 0.06755 which is very small. However, its p-statistic could infer that bad news has the insignificant effect to the conditional volatility.

Table 5.4
Estimation output for TGARCH

$Y_t = 0.3227Y_{t-1}$

s.d. = (0.0614)

p-value = [0.0000]

$\sigma_t^2 = 0.0636 + 0.1181u_{t-1}^2 + 0.0675u_{t-1}^2d_{t-1} + 0.6647\sigma_{t-1}^2$

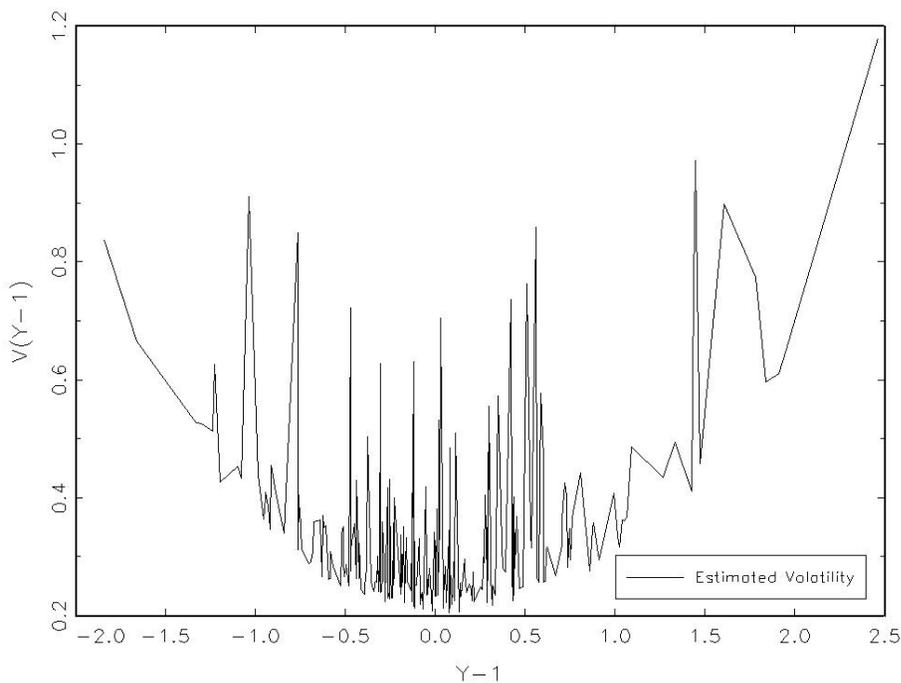
s.d. = (0.0403) (0.0665) (0.0766) (0.1513)

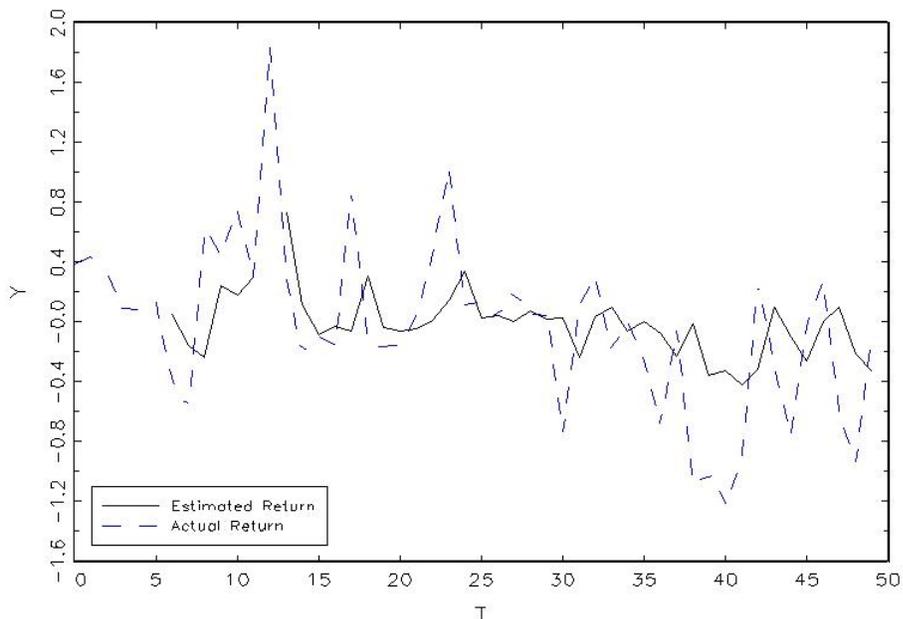
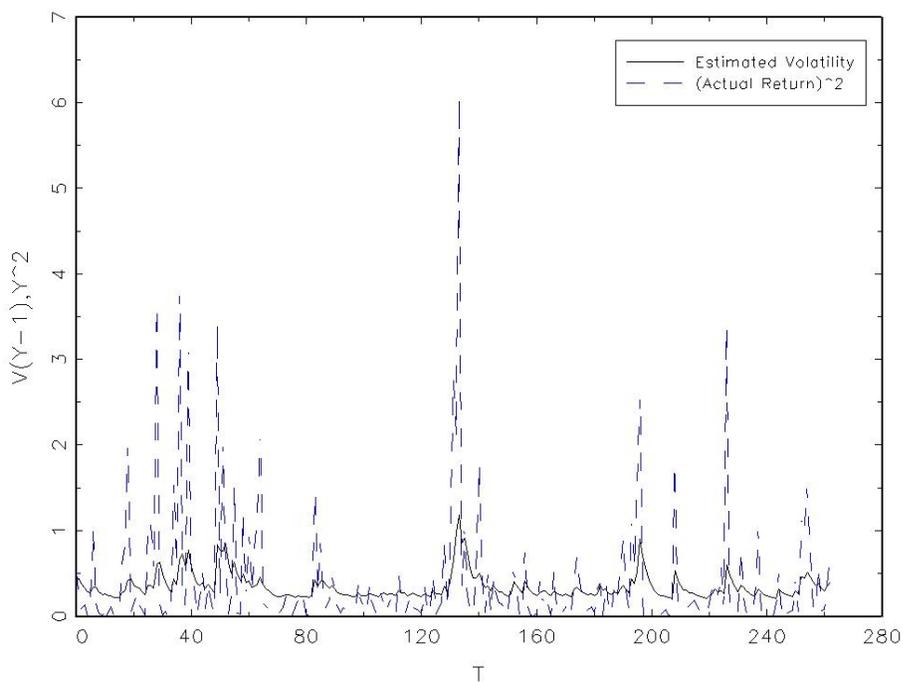
p-value = [0.0582] [0.0385] [0.1895] [0.0000]

R-squared	0.1004	Mean dependent variable	0.0098
Adjusted R-squared	0.1004	S.D. dependent variable	0.6238
S.E. of regression	0.5930	Akaike info criterion	0.0141
Sum squared residual	92.1200	Schwarz criterion	0.0157
Log likelihood	-0.8542	F-statistic	29.1400

Note: (.) = standard deviation and [.] = p-value

Fig 5.5
(a) TGARCH estimated volatility



(b) TGARCH estimated and true return**(c) TGARCH estimated volatility and squared return**

This is similar to the log of conditional variance equation in EGARCH model, that the log of conditional variance is not responsive to negative lagged residual because

in table 5.5 reveals that -0.05869 estimator is very small number in absolute term and extremely not significant. So we can conclude that the asymmetric response is not found in these models.

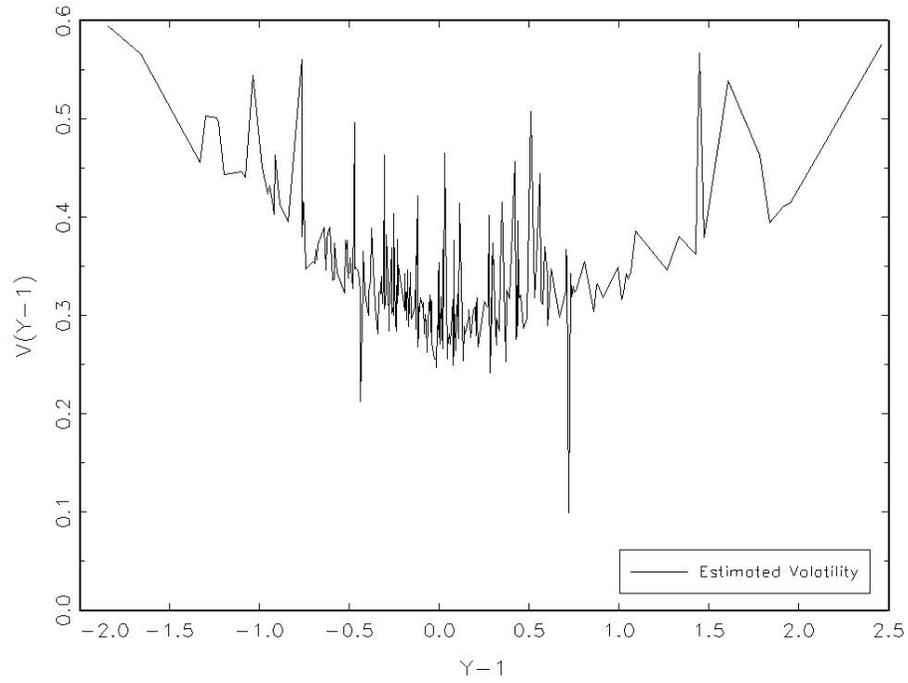
Table 5.5
Estimation output for EGARCH

$Y_t = 0.2917Y_{t-1}$			
s.d. = (0.0620)			
p-value = [0.0000]			
$\ln \sigma_t^2 = -0.4926 + 0.6680 \ln \sigma_{t-1}^2 + 0.1774 u_{t-1} / \sigma_{t-1} - 0.0586(u_{t-1} / \sigma_{t-1})$			
s.d. = (0.3143) (0.2330) (0.0866) (0.0540)			
p-value = [0.1182] [0.0044] [0.0416] [0.2785]			
R-squared	0.0907	Mean dependent variable	0.0098
Adjusted R-squared	0.0907	S.D. dependent variable	0.6238
S.E. of regression	0.5930	Akaike info criterion	0.0142
Sum squared residual	92.1500	Schwarz criterion	0.0158
Log likelihood	-0.8708	F-statistic	26.0500
Note: (.) = standard deviation and [.] = p-value			

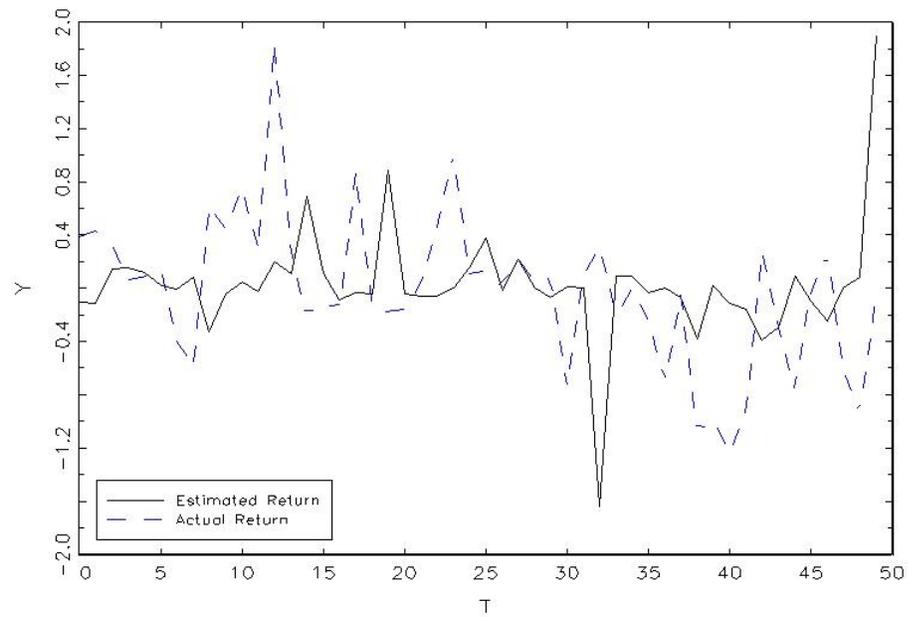
For TGARCH predicted conditional heteroskedasticity series, their visual representations are very similar to the ARCH and GARCH cases in both values and structures. But for EGARCH estimation, there's a lower outlier on the right-hand side of the minimum and the asymmetry can be revealed clearly. These can be seen in figure 5.6 (a). And EGARCH predicted conditional heteroskedasticity has lower motility compared than ARCH, GARCH, and TGARCH. It's clearly shown in figure 5.6 (c).

Fig 5.6

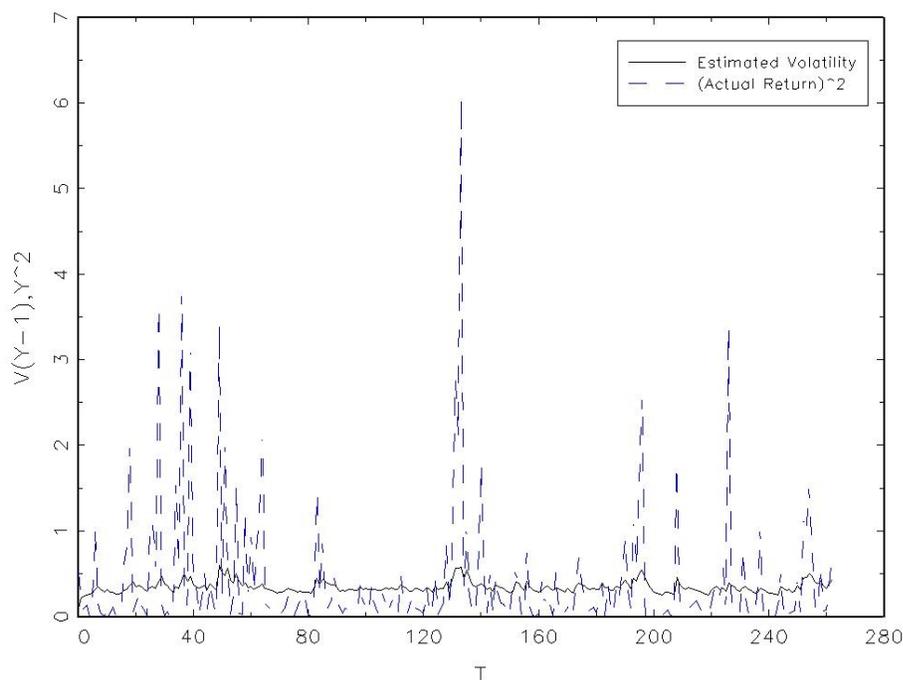
(a) EGARCH estimated volatility



(b) EGARCH estimated and true return



(c) EGARCH estimated volatility and squared return



Covariance nonstationary process of I-GARCH estimation is reported in table 5.6 in which the results show that all the estimators are significant in both mean and conditional variance equations. The unity of summation in lagged ARCH and lagged GRACH effects was restricted to capture the persistence of conditional variance.

The results in table 5.6 have the significant estimators of lagged ARCH and lagged GARCH effects equal to 0.1815 and 0.8185 respectively, given the volatility persistence restriction is imposed in equality constraint of maximum likelihood estimation process. However, note that Thai Baht/U.S. percentage rate of return follow GARCH process which is stationary. Thus, the interpretation of this volatility persistence phenomenon is the result from a model misspecification since another tables of estimation results suggest that this series has a finite variance (as the numerical optimization approaches of another different models are converge at sufficiently fast rate,

and the optimum at boundary of constraints is infrequently occurs from the inappropriate starting initial values).

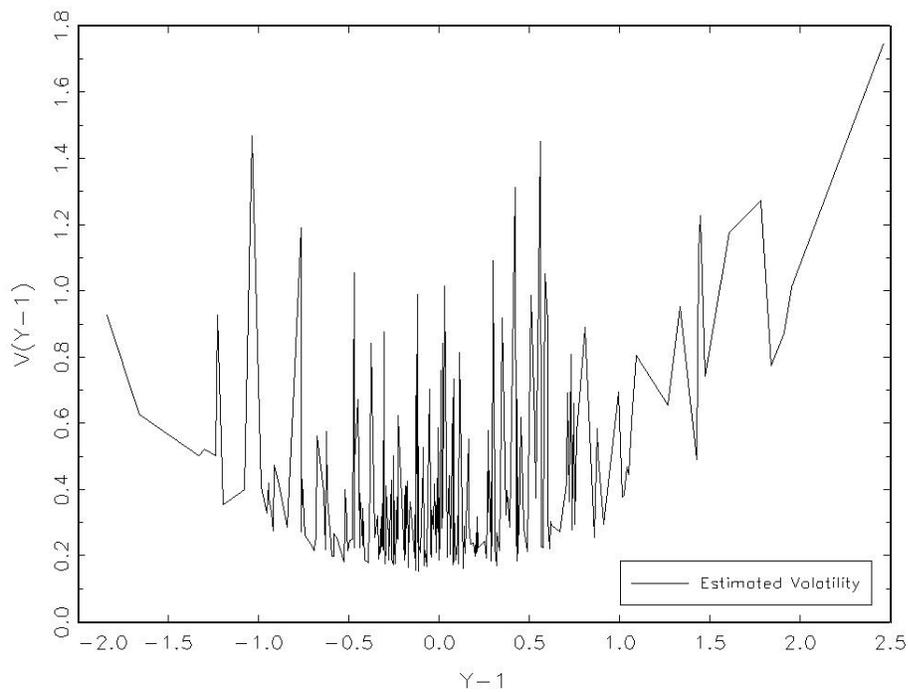
Figure 5.7 (a) and (c) represent the quite differences of I-GARCH predicted conditional heteroskedasticity from another previous conditional volatility models. Since I-GARCH predicted variance is very volatile, although around its return's zero mean. We can see from figure 5.7 (a) that, although it has roughed u-shape function, the movements are vary too much. And, surprisingly, this volatility function could capture squared return most efficient compared to another models in which its move is closer to squared return.

Table 5.6
Estimation output for I-GARCH

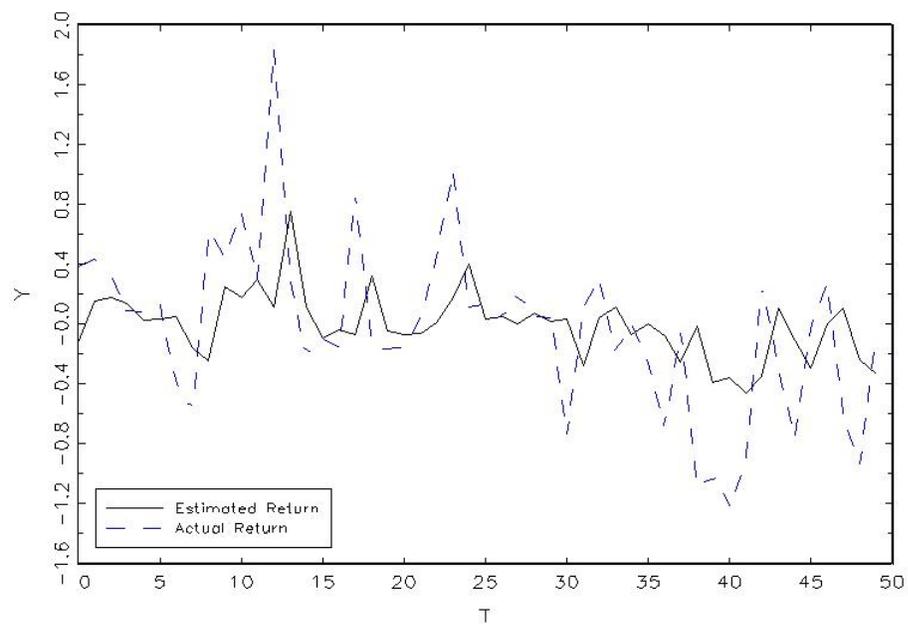
$Y_t = 0.3136Y_{t-1}$			
s.d. = (0.0600)			
p-value = [0.0000]			
$\sigma_t^2 = 0.0118 + 0.1815u_{t-1}^2 + 0.8185\sigma_{t-1}^2$			
s.d. = (0.0064) (0.0482) (0.0482)			
p-value = [0.0344] [0.0001] [0.0000]			
R-squared	0.0975	Mean dependent variable	0.0098
Adjusted R-squared	0.0975	S.D. dependent variable	0.6238
S.E. of regression	0.5929	Akaike info criterion	0.0142
Sum squared residual	92.1100	Schwarz criterion	0.0158
Log likelihood	-0.8723	F-statistic	28.2200
Note: (.) = standard deviation and [.] = p-value			

Fig 5.7

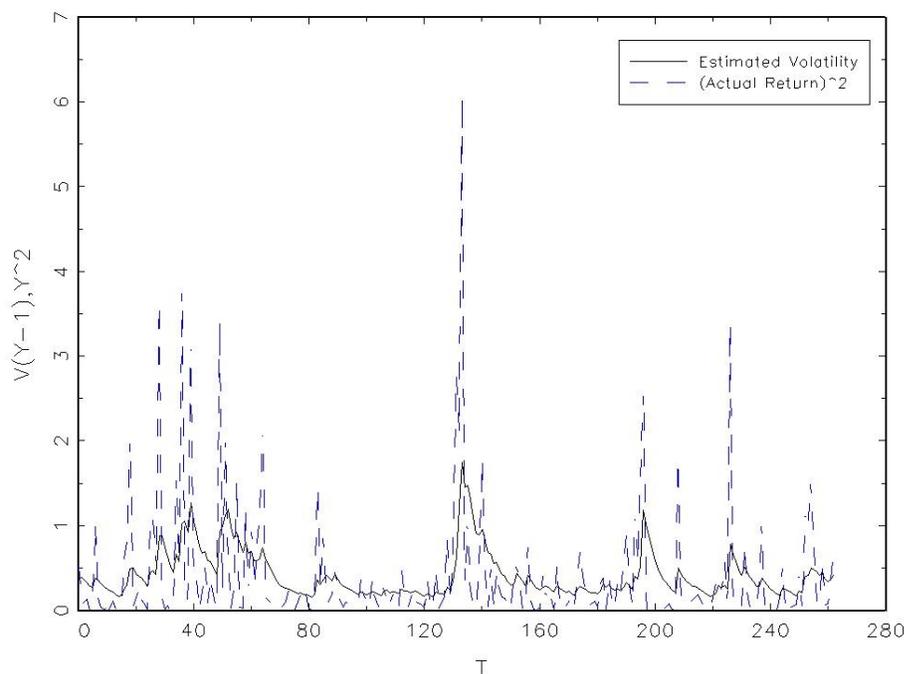
(a) I-GARCH estimated volatility



(b) I-GARCH estimated and true return



(c) IGARCH estimated volatility and squared return



According to capital asset pricing model (CAPM), the expected excess returns is proportional to its own conditional variance. Engle et al. (1987)'s ARCH-M which introduced to capture such relationships is estimated and its results is reported in table 5.7. The model, which is extended to allow the conditional variance to be a determinant of the mean, have the finding results that the estimators of variance equation are very close to ARCH model in table 5.2. Such that the estimators of constant and lagged square residual are 0.27 and 0.19 respectively, and they are significant.

However, the extension of including the lagged of ARCH effect in the conditional mean equation seems to be insignificant at all standard level of inference test with p-value equals to 0.5894. Thus, there is no tradeoff between the expected return of weekly Thai baht/U.S. percentage rate of return and its conditional variance. So the small negative effect, -0.0125, of the incremental in current conditional variance has not effect to the changes on the percentage return in values. In ARCH-M model, if we wipe out the

conditional variance effect in mean equation which is not significant and the remaining significant estimators of mean and variance equations are seems to be duplicated from ARCH model. And this duplication is not only for in-sample estimation but also out-of-sample prediction that ARCH-M has the similar features of predicted conditional heteroskedasticity. Figure 5.3 (a) and (b) of ARCH model are almost the same as figure 5.8 (a) and (b) of ARCH-M, respectively. Therefore, the conclusion about conditional volatility function drawn from ARCH-M model estimation is the same as ARCH model.

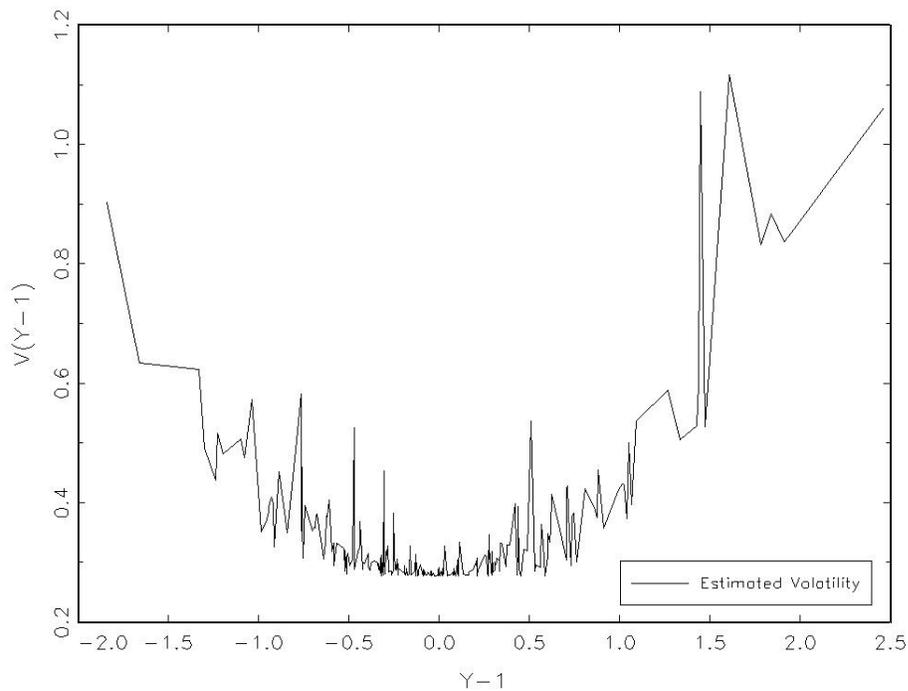
Table 5.7
Estimation output for ARCH-M

$Y_t = 0.3372Y_{t-1} - 0.0125\sigma_t^2$			
s.d. = (0.0677) (0.0555)			
p-value = [0.0000] [0.5894]			
$\sigma_t^2 = 0.2774 + 0.1985u_{t-1}^2$			
s.d. = (0.0301) (0.0800)			
p-value = [0.0000] [0.0068]			
R-squared	0.1045	Mean dependent variable	0.0098
Adjusted R-squared	0.1010	S.D. dependent variable	0.6238
S.E. of regression	0.5945	Akaike info criterion	0.0218
Sum squared residual	92.2500	Schwarz criterion	0.0250
Log likelihood	-0.8685	F-statistic	15.1000

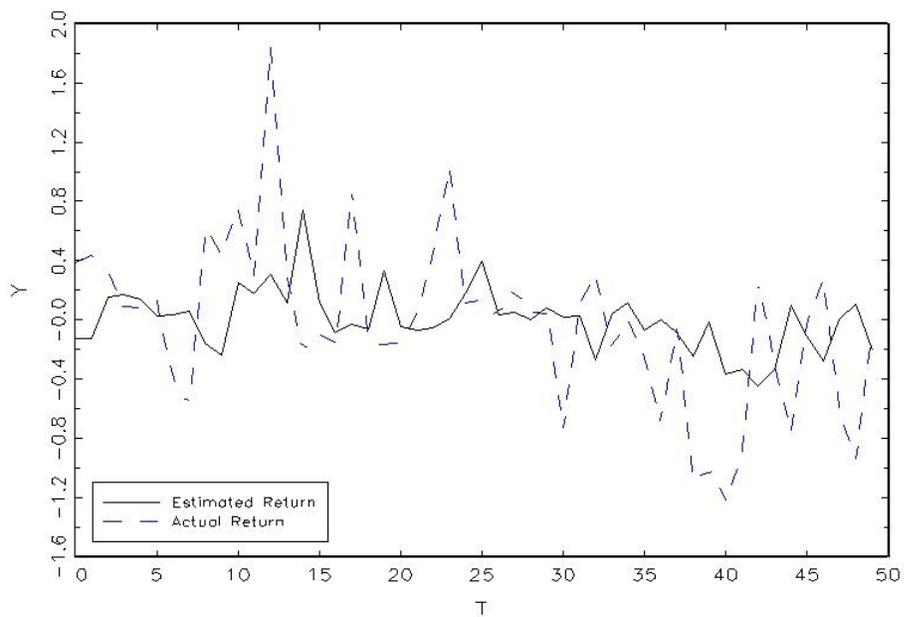
Note: (.) = standard deviation and [.] = p-value

Fig 5.8

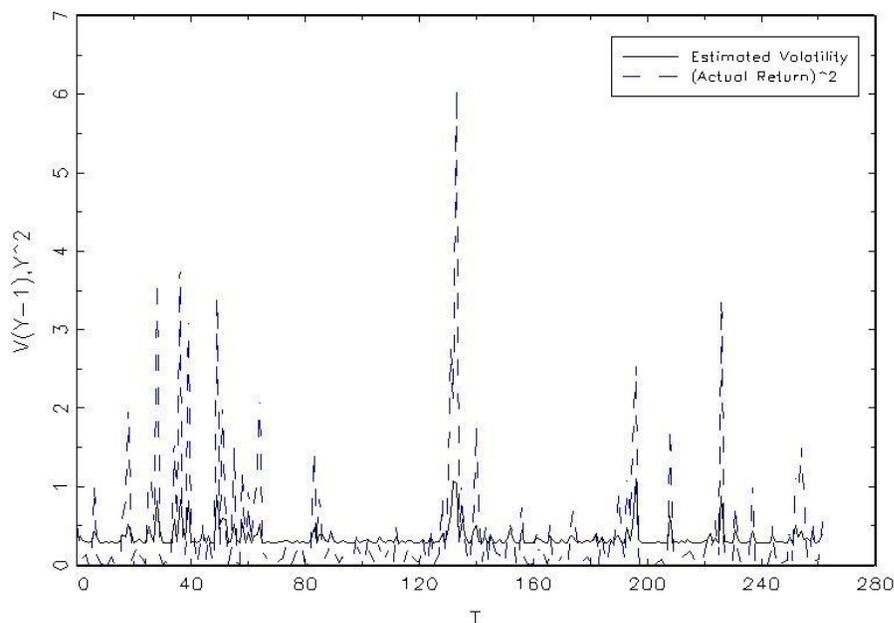
(a) ARCH-M estimated volatility



(b) ARCH-M estimated and true return



(c) ARCH-M estimated volatility and squared return



5.3 Out-of-Sample Performance

The process and sets of data employed in rolling MSPE process described previously have the outcomes of endeavor in table 5.8 We compare between the forecasted return with the true data in the sense of square in their differences for the sake of return forecasting performance comparison among competitive models. For variance comparison of conditional heteroskedasticity equations, we have two types of comparison between forecasted variance of various models with (1) the estimated nonparametric conditional variance or (2) the unconditional variance. The best forecasting performance model pay smallest rolling mean squared prediction error in term of smallest sizes in mean of the squared in differences between the forecast and the true value. Note that simple OLS with constant variance model could not have comparison contribution.

Table 5.8
Compare Rolling MSPE (# of rolling = 50)

1) True and Estimated Percentage Return	
MSPE of Nonparametric	0.2777
MSPE of OLS	0.3795
MSPE of ARCH	0.2724
MSPE of GARCH	0.2722
MSPE of TGARCH	0.2719
MSPE of I-GARCH	0.2722
MSPE of EGARCH	0.5159
MSPE of ARCH-M	0.2721
2.1) Estimated and Unconditional Variance	
MSPE of Nonparametric	0.0688
MSPE of ARCH	0.0960
MSPE of GARCH	0.1096
MSPE of TGARCH	0.1230
MSPE of I-GARCH	0.1055
MSPE of EGARCH	0.1096
MSPE of ARCH-M	0.0975
2.2) Estimated Variance and Squared Return	
MSPE of Nonparametric	0.3237
MSPE of ARCH	0.3408
MSPE of GARCH	0.3400
MSPE of TGARCH	0.3321
MSPE of I-GARCH	0.3558
MSPE of EGARCH	0.3387
MSPE of ARCH-M	0.3461

* Included 263 Rolling Observations for each 50 iterations

Table 5.8 are the results of the comparison of the rolling MSPE of percentage return and variance. For comparing the out-of-sample performance of percentage rate of return prediction, TGARCH yields the smallest rolling MSPE where most of the model have approximately close to the smallest one as nonparametric, ARCH, GARCH, I-GARCH and ARCH-M. By overview in contra to the comparison aspect, the results are quite satisfaction since the series level is range on -1.8456 to 2.4631. EGARCH and OLS have highest rolling MSPE equal 0.5159 and 0.3795 respectively.

To compare between estimated conditional heteroskedasticity with the unconditional variance as in Pagan and Schwert (1990), West and Cho (1994), and Khanthavit (1997). In which the comparison examination by Pagan and Schwert (1990)

and Khanthavit (1997) was employed with the increasing number of observations, appending, MSPE rather than rolling MSPE as did by West and Cho (1994). The comparison results suggest that nonparametric model and ARCH have the smallest rolling MSPE equal to 0.0688 and 0.0960 respectively. Note that OLS is not included because its constant variance is trivial to perform this.

For exploring the best model that can predict the conditional heteroskedasticity's nonequivalence from the squared return, nonparametric volatility model use local linear estimator with cross-validation bandwidth can perform the smallest rolling MSPE compare to the any other parametric models.

For this particular conclusion, nonparametric model can predict the Baht/U.S. percentage rate of weekly return inferior to many parametric model. This is similar to the out-of-sample performance deteriorated of nonparametric models as in Pagan and Schwert (1990). And the results from West and Cho (1994) and Khanthavit (1997) were not conclusive on which nonparametric or parametric models could be outperformed. However, our proposed squared return from theoretical reasoning to use it as a base to verify the error from conditional heteroskedasticity prediction can show that the nonparametric model can surpasses that of any parametric conditional volatility.

5.4 Unbiasedness Test

Table 5.9 reports the results of traditional examination for unbiasedness of all models deployed in the study. This test emerged in various financial applications of conditional heteroskedasticity works i.e. Pagan and Schwert (1990), West and Cho (1994), Khanthavit (1997), and Hafner (1998). They consider unbiasedness of the conditional variances estimation as an alternative measure of model performance. The last two columns in table 5.9 illustrate the chi-squared test of unbiasedness hypothesis. The last column contributes to White's heteroskedasticity consistent covariance matrix

while the previous use simple OLS covariance. The fallacy of inference may occurred if we use simple OLS covariance as the choice of covariance matrix because only GARCH model is fail to the unbiasedness test while another remaining models are significant at all standard level of statistical test. However, if using corrected White's instead, only nonparametric model endure to indicate the unbiasedness. Thus, there is the evidence indicate that I-GARCH model is prefer to another models for the in-sample test.

Table 5.9
Test for unbiasedness with model $\hat{u}_t^2 = \alpha + \beta\hat{\sigma}_t^2 + \xi_t$

Included Observations: 263

	Log-likelihood	α	β	R^2	F-statistic	$\chi^2_{(2)}^a$	$\chi^2_{(2)}^b$
NP	-1.4515	-1.3231 (1.053) [0.2101]	5.3588 (3.156) [0.0907]	0.0372	4.9664 [0.0261]	10.1898 [0.0061]	3.53076 [0.1711]
ARCH	-0.0940	-0.0844 (0.2019) [0.6761]	1.2536 (0.6194) [0.0439]	0.0604	8.3270	15.8441 [0.7062]	0.1749 [0.9162]
GARCH	-0.0940	0.0099 (0.1351) [0.9416]	0.9753 (0.4290) [0.0238]	0.0627	8.6620	0.0122 [0.9938]	0.0081 [0.9959]
TGARCH	-0.0918	-0.0569 (0.1471) [0.6505]	1.1840 (0.4745) [0.0066]	0.0744	10.4100	8.3928 [0.0150]	0.1523 [0.9267]
I-GARCH	-0.0946	0.1445 (0.0715) [0.0443]	0.4951 (0.1988) [0.0133]	0.0441	5.9329	15.2970 [0.0004]	6.6556 [0.0359]
EGARCH	-0.0935	-0.5239 (0.3514) [0.1372]	2.5736 (1.0860) [0.0184]	0.0695	9.6058	7.3698 [0.0250]	2.3514 [0.3086]
ARCH-M	-0.0942	-0.0785 (0.2015) [0.6513]	1.2360 (0.6183) [0.0233]	0.0602	8.2890	13.5555 [0.0011]	0.1517 [0.9270]

Standard deviations using White's heteroskedasticity correction are in parentheses under the coefficient estimates. The p-values for t distribution also provided in blankets. R-squared is the coefficient of determination. F-statistic reported is from the test of the hypothesis that the slope coefficients (excluding the intercept) in a regression is zero. Under the null hypothesis of linear restriction $\alpha = 0$ and $\beta = 1$, the Wald statistic which has an asymptotic chi-squared distribution will be the justification for null hypothesis testing with the respective p-value for chi-squared distribution in the blankets below. Last two columns based on different choices of covariance matrix, those are (a) simple regression covariance, and (b) White's heteroskedasticity consistent covariance.

CHAPTER 6

CONCLUSIONS AND LIMITATIONS

6.1 Conclusions

This thesis studies the volatility models in both parametric and nonparametric approaches. The data used in this study is the weekly Baht/U.S.\$ rate of return in 1999-2005 to investigate the best prediction model for the percentage return and conditional volatility. The descriptive statistics for the distribution reveals that the expectation mean of series is very small and right skewed and leptokurtic. The standard tests reject the hypothesis of normal distribution and none heteroskedasticity. And autocorrelation function and OLS statistics explore that the model for study is the AR(1) without constant model.

The empirical results of comparisons between nonparametric and parametric econometric models are rationed into three sections of in- and out-of-sample estimation and unbiasedness test. The competitive models of conditional heteroskedasticity functions are nonparametric volatility function estimated by local linear estimator with cross-validation criteria and several parametric models.

- Nonparametric volatility function estimates the smoothed and asymmetric U-smile shape of conditional heteroskedasticity. The fluctuation of percentage return will be higher when it is not around its mean especially if the past returns were below zero mean. This is called asymmetric smiling face. Other models, estimated by parametric nonlinear estimation, also reveal the smiling face and almost of them have the same structure and values. However, only ARCH and ARCH-M are rational in both theoretical and empirical senses for their low volatility around zero mean. Surprisingly, asymmetric coefficients of EGARCH and TGARCH are not significant while their conditional volatility functions figures reveal that their asymmetry is not skewed as other models.

- (1) The best model for return out-of-sample performance is TGARCH, although it has no clear comparative advantage in return prediction. Its rolling MSPE of predicted return is not higher from other models significantly. (2) For the conditional variance prediction, nonparametric model are the best for out-of-sample criteria both deployed unconditional variance and squared return as the basis.

- Test of unbiasedness consider chi-squared hypothesis to investigate model performance. The results of using classical OLS variance covariance matrix are significance in almost every model of both parametric and nonparametric approaches, except for GARCH model. However, if we employ White's heteroskedasticity consistent covariance matrix correction instead, only I-GARCH model is significant for indicate the unbiasedness of conditional heteroskedasticity estimation.

Since there is no exact model which could be the best model in all criteria for both estimation and prediction views. So there are inconclusive evidences for the different approaches of nonparametric and parametric that which could yield the most efficient results. Although nonparametric model could be the best model for the conditional volatility prediction, as this is our main objective of this study. So the researcher and applied econometrician should deploy different models for the different purposes, which is to predict or to estimate. Thus, in our recommendation for the generality, the combinations of nonparametric and parametric models, i.e. semiparametric, could yield the improvement in balancing between prediction and estimation purposes. Its hybrid feature is an interesting point for further applied research in econometrics field. And there are many financial researches employ our proposed volatility function for estimate the conditional variance of high frequency foreign exchange rate (HFFX) and have the clear result of efficiency. Thus, the nonparametric may could be employed to the high frequency series.

From this empirical econometrics study, nonparametric volatility model can perform the volatility forecast. So the contributions from the study are the rectification in applied financial applications such as risk management, asset pricing, and investment analysis. Because these works are require variance to use as a measure of risk for making decision, i.e. value at risk (VaR), or capital asset pricing model (CAPM) for examples. And there are many grounds for more advanced research.

6.2 Limitations

1) The lack of experiences and limited time of the researcher are due to the result that many new advanced ARCH models, as the advancements in this field was summarized in Engle (2002), such as (Markov) Switching ARCH, option pricing models, or large scale ARCH models for examples, cannot covered in this study. This is the same as nonparametric approaches. There are many estimation methods such as spline estimators, series estimators, or k-NN estimators, etc. While there are many choices of data driven bandwidths could selected for estimation, for examples, generalized cross-validation or finite prediction error. See Härdle and Linton (1994).

2) More lags can't be included in the mean function as well as the variance function for nonparametric model. This would bring up the "curse of dimensionality" one usually has in nonparametric estimation. As the possible solution could be the additive model class.

3) The prediction criteria using rolling MSPE is implicitly allowed for a failure of stationarity by using rolling samples.

4) Goodness of fit tests and robustness tests of these models should be concentrated. Because these are the limitations in models estimation which the extensions on these tests could increase the confidence for applied these conditional volatility models.

BIBLIOGRAPHY

Baltagi, Badi H. 2002 "Econometrics." Springer, 3th edition.

Bollerslev, T. 1986. "Generalized Autoregressive Conditional Heteroskedasticity."
Journal of Econometrics 31, 307-327.

Bollerslev, T. 1987. "A Conditional Heteroskedastic Time Series Model for
Speculative Prices and Rates of Return." Review of Economics and Statistics
Vol. 69 Issue 3, 542-547.

Bollerslev, T., Engle, Robert E., and Nelson, Daniel B. 1994. "ARCH Models."
Handbook of Econometrics, Volume IV. Elsevier Science B.V.

Bossaerts, Peter, Wolfgang Härdle, and Christian Hafner. "A New Method for
Volatility Estimation with Application to Foreign Exchange Rate Series." 1995.
Unpublished Manuscript.

Buhlmann, Peter and Alexander J. McNeil. 2002. "An Algorithm for Nonparametric
GARCH Modelling." Computational Statistics & Data Analysis 40, 665-683.

Chinprateep, Apirada. 1998. "How does Economic News affect Exchange and Interest
Rate: the Case of Thailand". Master of Economics Thesis. Thammasat
University. (in Thai).

Chusil, Supoj. 2003. "Investigation on Regime Switching on Stock Market: Case
Studies of Thailand and U.S." Independent Study, Master of Science Program in
Finance. Thammasat University.

Davidson, Russell and James G. Mackinnon. 1993. "Estimation and Inference in
Econometrics." Oxford University Press.

Defusco, Richard A., Mcleavey, Dennis W., Pinto Jerald E., and Runkle, David E..
2001. "Quantitative Methods for Investment Analysis." Association for
Investment Management and Research.

- Enders, Walter. 1995. "Applied Econometric Time Series." John Wiley & Sons.
- Engle, Robert F. 1982. "Autoregressive Conditional Heteroskedasticity with estimates of the variances of United Kingdom inflation." *Econometrica* 50, 987-1006.
- Engle, Robert F. 2002. "New Frontiers for ARCH Models." Modeling and Forecasting Financial Volatility Conference, Perth, Australia. This draft June 6, 2002.
- Engle, Robert F. and Tim Bollerslev. 1986. "Modeling the persistence of conditional variances." *Econometric Reviews* 5, 1-50.
- Engle, Robert F., D. Lilien, and R. Robins. 1987. "Estimating Time Varying Risk Premia in the Term Structure: The ARCH-M Model." *Econometrica*. 55,391-407.
- Franke, Jurgen, Michael H. Neumann, and Jean-Pierre Stockis. 2004. "Bootstrapping Nonparametric Estimators of the Volatility Function." *Journal of Econometrics* 118, 189-218.
- Ghysels, Eric, Santa-Clara, Pedro, and Valkanov, Rossen. 2004. "On the Relation between the Expected Value and the Volatility of Nominal Excess Return on Stocks." NBER working paper 10914.
- Glosten, Lawrence R., Jagannathan, Ravi, and Runkle, David E. 1993. "On the Relation between the Expected Value and the Volatility of Nominal Excess Return on Stocks." *Journal of Finance*, Vol. 48 Issue 5, 1779-1801.
- Greene, William H. 2000. "Econometric Analysis." Prentice Hall, 4th edition.
- Hafner, Christian. 1998. "Estimation High-Frequency Foreign Exchange Rate Volatility with Nonparametric ARCH Models." *Journal of Statistical Planning and Inference* 68, 247-269.
- Hausmann, Ricardo, Panizza, Ugo, and Rigobon, Roberto. 2004. "The Long Run Volatility Puzzle of the Real Exchange Rate." NBER working paper 10751.

- Hamilton, James D. 1994. "Time Series Analysis." Princeton University Press.
- Härdle, Wolfgang and Linton, Oliver. 1994. "Applied Nonparametric Methods." Handbook of Econometrics, Volume IV. Elsevier Science B.V.
- Härdle, Wolfgang and Tsybakov A. 1997. "Local Linear Estimator of the Volatility Function in Nonparametric Autoregression." Journal of Econometrics 81, 223-242.
- Hart, Jeffrey D. 1997. "Nonparametric Smoothing and Lack-of-Fit Tests." Springer.
- Hart, Jeffrey D. n.d. "Lecture Notes on Nonparametric Function Estimation." STAT627. Department of Statistics, Texas A&M University.
- Jintakan ,Jaruwan. 1998. "Service and Income Balance and Change in Exchange Rate System". Master of Economics Thesis. Thammasat University. (in Thai).
- Kaenmanee, Sumeth. 1997. "Lecture Note on Research Methodology." Department of Economics, Faculty of Management Science. Khonkaen University.
- Khanthavit, Anya. 1997. "Alternative Models for Conditional Volatility: Some Evidence from Thailand's Stock Market." Thammasat University.
- Khanthavit, Anya. 2001. "A Markov-Switching Model for Mutual Fund Performance Evaluation." Thammasat University.
- Khanthavit, Anya. 2004. "Risk Analysis in Securities Investment." Stock Exchange of Thailand. (in Thai)
- Leelahanon, Sittisak. 2000. "Local Linear Estimation of the Volatility Function in Nonparametric Autoregression." Econometrics IV Term Paper, Econometrics678, Spring. Texas A&M University.
- Li, Qi. 2000. "Lecture Notes on Econometric IV." Econometrics678, Spring. Department of Economics, Texas A&M University.

- Mills, Terence C. 1999. "The Econometric Modeling of Financial Time Series." Cambridge University Press.
- Nelson, Daniel B. 1991. "Conditional Heteroskedasticity in Asset Returns: A New Approach." *Econometrica* 59, 347-370.
- Pagan, Adrian R. and Aman Ullah. 1999. "Nonparametric Econometrics." Cambridge University Press.
- Pagan, Adrian R. and G. William Schwert. 1990. "Alternatives Models for Conditional Stock Volatility." *Journal of Econometrics* 45, 267-290.
- Phrukpaisal, Pongsak. 2003. "Return-Volatility Relationship in Thai Market." Independent Study, Master of Science Program in Finance. Thammasat University.
- Pimsaen, Chaiyan. 2000. "Economics News Effect on Exchange Rate". Master of Economics Thesis. Thammasat University. (in Thai).
- Rosenberg, Michael Roy. 2003. "Exchange-Rate Determination: Models and Strategies for Exchange-Rate Forecasting." McGraw-Hill.
- Stiglitz, Joseph. 2002. "Globalism's Discontents." *American Prospect*. January 14, 2002.
- Thitinantapong, Narongchai. 2002. "Exchange Rate Volatility Effects on Output Level: Banking Sector Model". Master of Economics Thesis. Thammasat University. (in Thai).
- West, Kenneth D. and Dongchul Cho. 1995. "The Predictive Ability of Several Models of Exchange Rate Volatility." *Journal of Econometrics* 69, 367-391.
- Xekalaki, Evdokia and Degiannakis, Stavros. 2004. "Evaluating Volatility Forecasts in Option Pricing in the Context of a Simulated Options Market." *Computational Statistics and Data Analysis*, © 2004 Elsevier B.V. All rights reserved.

- Bollerslev, T. 1986. "Generalized Autoregressive Conditional Heteroskedasticity." *Journal of Econometrics* 31, 307-327.
- Chen, Gongmeng, Choi, Yoon K., and Zhou, Yong. 2004. "Nonparametric Estimation of Structural Change Points in Volatility Models for Time Series." *Journal of Econometrics*.
- Franke, Jürgen, Neumann, Michael H., and Stockis, Jean-Pierre. 2004. "Bootstrapping nonparametric estimators of the volatility function." *Journal of Econometrics* 118, 189-218.
- Comte, F. and Rozenholc, Y. 2002. "Adaptive estimation of mean and volatility function in (auto-)regressive models." *Stochastic Processes and their Applications* 97, 111-145.
- Laïb, Naâmane. 2004. "Kernel Estimates of the Mean and the Volatility functions in a nonlinear autoregressive model with ARCH errors." *Journal of Statistical Planning and Inference*.

APPENDICES

```

/* er.g is the program for transform daily exchange rate to weekly      */
/* percentage rate of return which weekly average by working day      */

new ; cls ;
format 4,6 ;
output file = retw.rst on ;
load d[1325,1] = er.txt ; /* Daily exchange rate data */
w = zeros(rows(d)/5,1) ;
sw = zeros(rows(d)/5,2) ;
i = 1 ; j = 1 ; do while i <= 1325 ;
    a = zeros(5,1) ; /* Vector of days in a week */
    a = d[i:i+4] ;
    a2 = selif(a,(a[.,.] .ne 0)) ; /* Vector of active days in a week */
    w[j] = meanc(a2) ; /* WEEKLY OBSERVATIONS */
    sw[j,.] = sumc(a2)~rows(a2) ; /* Information for verification */
i = i + 5 ; j = j + 1 ; endo ;

w[2:rows(w)]~sw[2:rows(w),.]~100*(ln(w[2:rows(w)])-ln(w[1:rows(w)-1])) ;

```

```

/*      garchall.g is the estimation program for autoregressive          */
/*      conditional heteroscedasticity series to compute full sample   */
/*      estimators and statistics. The end of program also test the    */
/*      unbiasedness of estimated conditional variance.                 */

new ; cls ; format 2,6 ;
tstrt = timestr(0) ; dstrt = date ;

library CML, pgraph ;
_cml_DirTol = 1e-6 ;
_cml_MaxIters = 9e+2 ;
_cml_Algorithm = 3 ;
_cml_LineSearch = 1 ;
_cml_CovPar = 1 ;
output file = thesall.rst ; output reset on ;

n_obs = 264 ;
load data[n_obs,1] = retw.txt ;          /* retw = ln(ER_t) - ln(ER_t-1) */
data = data*100 ;                       /* percentage rate of return */

case = 0 ; do while case <= 6 ;

    y = data[2:rows(data),.] ; n = rows(y) ;
    x = data[1:rows(data)-1,.] ;
    b0 = inv(x'x)*x'y ;
    u = y - x*b0 ;
    u2 = u^2 ;
    xu_2 = ones(rows(u2)-1,1)~u2[1:rows(u2)-1,.] ;
    a = inv(xu_2'xu_2)*xu_2'u2[2:rows(u2),.] ; /* Initial parameters (1) */
    sy = zeros(n,1) ; sx = zeros(n,1) ;
    is = 1 ; do while is <= n ;
        sy[is,] = stdc(y[1:is,1]) ; sx[is,] = stdc(x[1:is,1]) ;
    is = is + 1 ; endo ;
    bs = inv(sx'sx)*sx'sy ;                /* Initial parameters (2) */
    e0_2 = sumc(u2)/n ; h0_2 = e0_2 ;      /* possible disturbance at time 0 */
    h0 = zeros(n,1) ;
    L = h0 ; u_hat = h0 ;

output off ;

if case == 0 ;
    " @----- Nonparametric & OLS -----@ " ;
    _cml_DirTol = 1e-6 ;
    _cml_Algorithm = 3 ;
    _cml_LineSearch = 1 ;
    _cml_Bounds = { 0.00015 1e0 } ;
    h_np = h0 ;
    b_ini = stdc(x)*(n^(-1/5)) ;
    m_hat = LocalLin(y, x, b_ini) ;
    g_hat = LocalLin(y^2, x, b_ini) ;
    v_hat = g_hat - (m_hat^2) ;

```

```

        h_np = v_hat ;
        u_hat_np = (y./sqrt(v_hat)) - (m_hat./sqrt(v_hat)) ;
        /*[ y = m(y-1) + v(y-1)*u ]*/
        bols = b0 ;                               /* OLS Estimator */

elseif case == 1 ;
    "@----- ARCH -----@" ;
    _cml_DirTol = 1e-2 ;
    _cml_Algorithm = 3 ;
    _cml_LineSearch = 1 ;
    _cml_ParNames = "b0"|"ba0"|"ba1" ;
    _cml_A = zeros(1,3) ;                          /* _cml_A * p = _cml_B */
    _cml_B = 0 ;                                   /* (Equality Constraint) */
    _cml_C = zeros(1,2)~-ones(1,1) ;              /* _cml_C * p >= _cml_D */
    _cml_D = -1 ;                                  /* (Inequality Constraint) */
    _cml_Bounds = { -1e0 1e0 , -1e0 1e0 , -1e0 1e0 } ; /* Parameter Boundaries */
    h_a = h0 ;
    ba0 = a[1,1]/10 ;
    ba1 = a[2,1] ;
    p_a = b0|ba0|ba1 ;
    {p_a, f_a, g_a, cov_a, rcode_a} = cmlPRT(cml(y~x, 0, &arch, p_a)) ;
    "maximized function valued of ARCH : " ftos(f_a,"%#*.IG",15,6); "" ;
    "Variance-Covariance Matrix of ARCH Parameters" ;
    cov_a ; "" ;
    u_hat_a = (y - x*p_a[1,1])^2 ;

elseif case == 2 ;
    "@----- GARCH -----@" ;
    _cml_Algorithm = 3 ;
    _cml_LineSearch = 1 ;
    _cml_ParNames = "b0"|"bg0"|"ag1"|"bg1" ;
    _cml_A = zeros(1,4) ;
    _cml_B = 0 ;
    _cml_C = zeros(1,2)~-ones(1,2) ;
    _cml_D = -1 ;
    _cml_Bounds = { -1e0 1e0 , -1e0 1.5e0 , -1e0 1e0 , -1e0 1e0 } ;
    h_g = h0 ;
    bg0 = a[1,1]/100 ;
    ag1 = a[2,1] ;
    bg1 = bs/2 ;
    p_g = b0|bg0|ag1|bg1 ;
    {p_g, f_g, g_g, cov_g, rcode_g} = cmlPRT(cml(y~x, 0, &garch, p_g)) ;
    "maximized function valued of GARCH : " ftos(f_g,"%#*.IG",15,6); "" ;
    "Variance-Covariance Matrix of GARCH Parameters" ;
    cov_g ; "" ;
    u_hat_g = (y - x*p_g[1,1])^2 ;

elseif case == 3 ;
    "@----- TGARCH -----@" ;
    _cml_Algorithm = 1 ;

```

```

_cml_LineSearch = 2 ;
_cml_MaxIters = 2e+2 ;
_cml_ParNames = "b0"|"bt0"|"bt1"|"bt_n"|"bt2" ;
_cml_A = zeros(1,5) ;
_cml_B = 0 ;
_cml_C = zeros(1,2)~ones(1,3) ;
_cml_D = -1 ;
_cml_Bounds = { -1e0 1e0 , -.1e0 1e0 , 1e-10 1e0 , 0e0 1e0 , -1e0 1e0 } ;
    h_t = h0 ;
    bt0 = a[1,1]/100 ;
    bt1 = a[2,1] ;
    bt_n = a[2,1]/10 ;
    bt2 = bs/1.5 ;
    p_t = b0|bt0|bt1|bt_n|bt2 ;
    {p_t, f_t, g_t, cov_t, rcode_t} = cmlPRT(cml(y~x, 0, &tgarch, p_t)) ;
    "maximized function valued of TGARCH : " ftos(f_t,"%#*. *IG",15,6); "" ;
    "Variance-Covariance Matrix of TGARCH Parameters" ;
    cov_t ; "" ;
    u_hat_t = (y - x*p_t[1,1])^2 ;

```

```

elseif case == 4 ;

```

```

" @----- I-GARCH -----@ " ;
_cml_Algorithm = 3 ;
_cml_LineSearch = 1 ;
_cml_MaxIters = 4e+1 ;
_cml_ParNames = "b0"|"bi0"|"ai1"|"bi1" ;
_cml_A = zeros(1,2)~ones(1,2) ;
_cml_B = 1 ;
_cml_C = zeros(1,4) ;
_cml_D = 0 ;
_cml_Bounds = { -1e0 1e0 , -1e0 1e0 , -1e0 1e0 , -1e0 1e0 } ;
    h_i = h0 ;
    bi0 = a[1,1]/100 ;
    ai1 = a[2,1] ;
    bi1 = 1 - ai1 ;
    p_i = b0|bi0|ai1|bi1 ;
    {p_i, f_i, g_i, cov_i, rcode_i} = cmlPRT(cml(y~x, 0, &igarch, p_i)) ;
    "maximized function valued of I-GARCH : " ftos(f_i,"%#*. *IG",15,6); "" ;
    "Variance-Covariance Matrix of I-GARCH Parameters" ;
    cov_i ; "" ;
    u_hat_i = (y - x*p_i[1,1])^2 ;

```

```

elseif case == 5 ;

```

```

" @----- EGARCH -----@ " ;
_cml_Algorithm = 3 ;
_cml_LineSearch = 1 ;
_cml_MaxIters = 4e+2 ;
_cml_ParNames = "b0"|"be0"|"be1"|"be2"|"be3" ;
_cml_A = zeros(1,5) ;
_cml_B = 0 ;

```



```

" @----- OLS -----@ " ;
betaols(y, x) ; "" ;

" @----- ARCH -----@ " ;
_cml_ParNames = "b0"|"ba0"|"ba1" ;
{p_a, f_a, g_a, cov_a, rcode_a} = cmlPRT(p_a, f_a, g_a, cov_a, rcode_a) ;
"maximized function valued of ARCH : " f_a ; "" ;
stat(p_a, f_a, cov_a) "" ;
"" ;

" @----- GARCH -----@ " ;
_cml_ParNames = "b0"|"bg0"|"ag1"|"bg1" ;
{p_g, f_g, g_g, cov_g, rcode_g} = cmlPRT(p_g, f_g, g_g, cov_g, rcode_g) ;
"maximized function valued of GARCH : " f_g ; "" ;
stat(p_g, f_g, cov_g) ; "" ;
"" ;

" @----- TGARCH -----@ " ;
_cml_ParNames = "b0"|"bt0"|"bt1"|"bt_n"|"bt2" ;
{p_t, f_t, g_t, cov_t, rcode_t} = cmlPRT(p_t, f_t, g_t, cov_t, rcode_t) ;
"maximized function valued of TGARCH : " f_t ; "" ;
stat(p_t, f_t, cov_t) ; "" ;
"" ;

" @----- I-GARCH -----@ " ;
_cml_ParNames = "b0"|"bi0"|"ai1"|"bi1" ;
{p_i, f_i, g_i, cov_i, rcode_i} = cmlPRT(p_i, f_i, g_i, cov_i, rcode_i) ;
"maximized function valued of I-GARCH : " f_i ; "" ;
stat(p_i, f_i, cov_i) ; "" ;
"" ;

" @----- EGARCH -----@ " ;
_cml_ParNames = "b0"|"be0"|"be1"|"be2"|"be3" ;
{p_e, f_e, g_e, cov_e, rcode_e} = cmlPRT(p_e, f_e, g_e, cov_e, rcode_e) ;
"maximized function valued of EGARCH : " f_e ; "" ;
stat(p_e, f_e, real(cov_e)) ; "" ;
"" ;

" @----- ARCH-M -----@ " ;
_cml_ParNames = "b0"|"b0-h"|"bm0"|"am1" ;
{p_m, f_m, g_m, cov_m, rcode_m} = cmlPRT(p_m, f_m, g_m, cov_m, rcode_m) ;
"maximized function valued of ARCH-M : " f_m ; "" ;
x = x~sqrt(h_m) ; /* Independent Variables for Mean Equation */
stat(p_m, f_m, real(cov_m)) ; "" ;
"" ;
"" ;

"Test for Unbiasedness " ;
"e^2 = alpha0 + beta0*h_hat + error " ; "" ;
"Included Observations: " n ; "" ;

```

```

"NP" ; "" ; forecast(u_hat_np, h_np) ; "" ;
"ARCH" ; "" ; forecast(u_hat_a, h_a) ; "" ;
"GARCH" ; "" ; forecast(u_hat_g, h_g) ; "" ;
"TGARCH" ; "" ; forecast(u_hat_t, h_t) ; "" ;
"I-GARCH" ; "" ; forecast(u_hat_i, h_i) ; "" ;
"EGARCH" ; "" ; forecast(u_hat_e, exp(ln_h_e)) ; "" ;
"ARCH-M" ; "" ; forecast(u_hat_m, h_m) ; "" ;

```

```

tfini = timestr(0) ; dfini = date ;
"" ; "" ; "start at " tstrt ; dstrt[1:3,.]' ;
"" ; "" ; "finish at " tfini ; dfini[1:3,.]' ;

```

```

proc(1) = LocalLin(y,x,b_ini) ;
local kernel, r, i, xx, xk, xi, beta, b, func, grad, varc, retc ;
kernel = zeros(n,1) ;
r = zeros(n,1) ;
{b, func, grad, varc, retc} = cmlPRT(cml(y~x, 0, &cvll, b_ini)) ;
i = 1 ; do while i <= n ;
    xi = x[i,.] * ones(n,1) ;
    kernel = (1/sqrt(2*pi)) * exp(-0.5 * ((x - xi)/b)^2) ;
    xx = ones(n,1) ~ (x-xi) ;
    xk = xx * kernel ;
    beta = inv(xk'xx) * xk'y ;
    r[i,.] = beta[1,.] ;
i = i + 1 ; endo ;
retp(r) ;
endp ;

```

```

proc(1) = cvll(th,z) ;
local b, i, xi, sr, r, kernel, xx, xk, beta ;
b = th[1,1] ; y = z[.,1] ; x = z[.,2:cols(z)] ;
sr = zeros(n,1) ;
i = 1 ; do while i <= n ;
    xi = x[i,.] * ones(n,1) ;
    kernel = (1/sqrt(2*pi)) * exp(-0.5 * ((x - xi)/b)^2) ;
    kernel[i,.] = 0 ;
    xx = ones(n,1) ~ (x-xi) ;
    xk = xx * kernel ;
    beta = inv(xk'xx) * xk'y ;
    sr[i,.] = (y[i,.] - beta[1,.] )^2 ;
i = i + 1 ; endo ;
retp(-sumc(sr)) ;
endp ;

```

```

proc(1) = arch(th,z) ;

```

```

local ig, u_hat, p_f, f_f, g_f, cov_f, rcode_f ;
b0 = th[1:cols(x),1] ;
ba0 = th[cols(x)+1,1] ;
ba1 = th[cols(x)+2,1] ;
y = z[.,1] ; x = z[.,2:cols(z)] ;
ig = 1 ; do while ig <= n ;
    if ig == 1 ;
        h_a[ig,.] = ba0/(1-ba1) ;
    else ;
        h_a[ig,.] = ba0 + ba1*(y[ig-1,.]-x[ig-1,.]*b0)^2 ;
    endif ;
l[ig,.] = - 0.5*ln(2*pi*h_a[ig,.]) - 0.5*( (y[ig,.] - x[ig,.]*b0)^2 )/h_a[ig,.] ;
ig = ig + 1 ; endo ;
retp(sumc(l)) ;
endp ;

```

```

proc(1) = garch(th,z) ;
local ig ;
b0 = th[1:cols(x),1] ;
bg0 = th[cols(x)+1,1] ;
ag1 = th[cols(x)+2,1] ;
bg1 = th[cols(x)+3,1] ;
y = z[.,1] ; x = z[.,2:cols(z)] ;
ig = 1 ; do while ig <= n ;
    if ig == 1 ;
        h_g[ig,.] = bg0/(1-bg1-ag1) ;
    else ;
        h_g[ig,.] = bg0 + ag1*((y[ig-1,.] - x[ig-1,.]*b0)^2) + bg1*h_g[ig-1,.] ;
    endif ;
l[ig,.] = - 0.5*ln(2*pi) - 0.5*ln(h_g[ig,.]) - 0.5*( (y[ig,.] - x[ig,.]*b0)^2 )/h_g[ig,.] ;
ig = ig + 1 ; endo ;
retp(sumc(l)) ;
endp ;

```

```

proc(1) = tgarch(th,z) ;
local ig ;
b0 = th[1:cols(x),1] ;
bt0 = th[cols(x)+1,1] ;
bt1 = th[cols(x)+2,1] ;
bt_n = th[cols(x)+3,1] ;
bt2 = th[cols(x)+4,1] ;
y = z[.,1] ; x = z[.,2:cols(z)] ;
ig = 1 ; do while ig <= n ;
    if ig == 1 ;
        h_t[ig,.] = bt0 / (1-bt1-bt_n-bt2) ;
    else ;
        if (y[ig-1,.]-x[ig-1,.]*b0) < 0 ;
            h_t[ig,.] = bt0 + (bt1*(y[ig-1,.]-x[ig-1,.]*b0)^2)

```

```

                bt_n*((y[ig-1,]-x[ig-1,]*b0)^2) + bt2*h_t[ig-1,] ;
            else ;
                h_t[ig,] = bt0 + (bt1*(y[ig-1,]-x[ig-1,]*b0)^2) + bt2*h_t[ig-1,] ;
            endif ;
        endif ;
l[ig,] = - 0.5*ln(2*pi) - 0.5*ln(h_t[ig,]) - 0.5*( (y[ig,] - x[ig,]*b0)^2 )/h_t[ig,] ;
ig = ig + 1 ; endo ;
retp(sumc(l)) ;
endp ;

```

```

proc(1) = igarch(th,z) ;
local ig ;
b0 = th[1:cols(x),1] ;
bi0 = th[cols(x)+1,1] ;
ai1 = th[cols(x)+2,1] ;
bi1 = th[cols(x)+3,1] ;
y = z[.,1] ; x = z[.,2:cols(z)] ;
ig = 1 ; do while ig <= n ;
    if ig == 1 ;
        h_i[ig,] = bi0 + ai1*e0_2 + bi1*h0_2 ;
    else ;
        h_i[ig,] = bi0 + ai1*(y[ig-1,]-x[ig-1,]*b0)^2 + bi1*h_i[ig-1,] ;
    endif ;
l[ig,] = - 0.5*ln(2*pi) - 0.5*ln(h_i[ig,]) - 0.5*( (y[ig,] - x[ig,]*b0)^2 )/h_i[ig,] ;
ig = ig + 1 ; endo ;
retp(sumc(l)) ;
endp ;

```

```

proc(1) = egarch(th,z) ;
local ig ;
b0 = th[1:cols(x),1] ;
be0 = th[cols(x)+1,1] ;
be1 = th[cols(x)+2,1] ;
be2 = th[cols(x)+3,1] ;
be3 = th[cols(x)+4,1] ;
y = z[.,1] ; x = z[.,2:cols(z)] ;
ig = 1 ; do while ig <= n ;
    if ig == 1 ;
        ln_h_e[ig,] = be0 / (1 - be1 - be2 - be3) ;
    else ;
        ln_h_e[ig,] = be0 + be1*ln_h_e[ig-1,]
            + be2*abs( (y[ig-1,]-x[ig-1,]*b0) / sqrt(exp(ln_h_e[ig-1,])) )
            + be3*(y[ig-1,]-x[ig-1,]*b0)/sqrt(exp(ln_h_e[ig-1,])) ;
    endif ;
l[ig,] = - 0.5*ln(2*pi) - 0.5*ln(exp(ln_h_e[ig,]))
    - 0.5*( (y[ig,] - x[ig,]*b0)^2 )/(exp(ln_h_e[ig,])) ;
ig = ig + 1 ; endo ;
retp(real(l)) ;

```

```
endp ;
```

```
proc(1) = archm(th,z) ;  
local ig ;  
b0 = th[1:cols(x),1] ;  
bh = th[cols(x)+1,1] ;  
bm0 = th[cols(x)+2,1] ;  
am1 = th[cols(x)+3,1] ;  
y = z[.,1] ; x = z[.,2:cols(z)] ;  
ig = 1 ; do while ig <= n ;  
    if ig == 1 ;  
        h_m[ig,.] = bm0/(1-am1) ;  
    else ;  
        h_m[ig,.] = bm0 + am1*( (y[ig-1,.] - x[ig-1,.]*b0  
            - sqrt(h_m[ig-1,.]*bh)^2 ) ;  
    endif ;  
l[ig,.] = - 0.5*ln(2*pi) - 0.5*ln(h_m[ig,.])  
    - 0.5*( (y[ig,.] - x[ig,.]*b0 - sqrt(h_m[ig,.]*bh)^2 )/h_m[ig,.] ;  
ig = ig + 1 ; endo ;  
retp(sumc(l)) ;  
endp ;
```

```
/* Procedure for estimated statistics for parametric volatility models */
```

```
proc(1) = stat(bp, f, varcov) ;  
local k, u, sse, wvarcov, sd, n, tstat, p, ybar2, b, r2 ;  
if cols(x) == 1 ;  
    b = bp[1,1] ;  
else ;  
    b = bp[1:2,.] ;  
endif ;  
k = cols(x) ; n = rows(x) ;  
u = y - x*b ;  
sse = u'u/(n-k) ;  
sd = sqrt(diag(varcov)) ;  
tstat = bp./sd ;  
p = cdfc(tstat,n-k) ;  
ybar2 = (sumc(y)/n)^2 ;  
r2 = (b'x'y - n*ybar2)/(y'y - n*ybar2) ;  
"Included Observations: " n ; "" ;  
"Coefficient Std. Error t-Statistic Prob." ;  
bp~sd~tstat~p ;  
"" ;  
"R-squared = " r2 ;  
"Mean dependent var = " meanc(y) ;  
"Adjusted R-squared = " 1-(1-r2)*((n-1)/(n-k)) ;  
"S.D. dependent var = " sqrt( sumc((y-meanc(y))^2) / (n-1) ) ;  
"S.E. of regression = " sqrt(sse) ;  
"AIC = " -2*(f/n)+2*(k/n) ;
```

```

"Sum squared resid = " u'u ;
"SC = " -2*(f/n)+k*log(n)/n ;
"Log likelihood = " f ;
"F-statistic = " (r2/(1-r2))*((n-k)/k-1) ;
"" ;
"Inverse of the Hessian Variance-Covariance Matrix" ;
retp(varcov) ;
endp ;

/* Procedure for test of unbiasedness in conditional variance estimation */
proc(1) = forecast(u_hat, h_hat) ;
local yz, n, xz, kz, b, u, sse, wvarcov, co, sd, tstat, p, ybar2, r2, theta, fun, gra, cov, ret, theta0,
sigma, aic, sc, pf, wa, wb, q, g, pc2a, pc2b ;
yz = u_hat ; n = rows(yz) ;
xz = ones(n,1)~h_hat ; kz = cols(xz) ;
b = inv(xz'xz)*xz'yz ;
u = yz - xz*b ;
sse = u'u/(n-kz) ;
theta0 = b|sqrt(sse) ;
_cml_A = zeros(1,3) ;
_cml_B = 0 ;
_cml_C = zeros(1,3) ;
_cml_D = 0 ;
_cml_Bounds = { -1.5e0 1.5e0 , -1.5e0 1.5e0 , -1.5e0 1.5e0 } ;
_cml_ParNames = "alpha0"|"beta0"|"sigma" ;
output off ; {theta, fun, gra, cov, ret} = cmlprt(cml(yz~xz, 0, &mle, theta0)) ;
output on ;
"" ;
/* White's Covariance Matrix */
wvarcov = (n/(n-kz))*inv(xz'xz)*(xz'*diagrv(eye(n),u^2)*xz)*inv(xz'xz) ;

sd = sqrt(diag(wvarcov)) ;
tstat = b./sd ;
p = cdfc(tstat,n-kz) ;
ybar2 = (sumc(yz)/n)^2 ;
r2 = (b'xz'yz - n*ybar2)/(yz'yz - n*ybar2) ;
aic = -2*(fun/n)+2*(kz/n) ;
sc = -2*(fun/n)+kz*log(n)/n ;
pf = cdfc((r2/(1-r2))*((n-kz)/kz-1),kz-1,n-kz) ;
g = eye(2) ;
q = 0|1 ;
/* Wald's Test with OLS variance-covariance */
wa = (g*b - q)'inv(g*sse*inv(x'x)*g)*(g*b - q) ;
pc2a = cdfchic(wa,2) ;
/* Wald's Test with White's variance-covariance */
wb = (g*b - q)'inv(g*wvarcov*g)*(g*b - q) ;
pc2b = cdfchic(wb,2) ;

"Log-likelihood alpha beta R-squared F-statistic Chi-squared(a) Chi-squared(b) " ; "" ;

```

```

        fun~b'~r2~(r2/(1-r2))*((n-kz)/kz-1)~wa~wb ;
        0~sd'~0~pf~pc2a~pc2b ;
        0~tstat' ;
        0~p' ;
/* retp(wvarcov) ; */
retp("");
endp ;

proc mle(theta, z) ;
local yz, xz, kz, b, sigma ;
yz = z[.,1] ; xz = z[.,2:cols(z)] ; kz = cols(xz) ;
b = theta[1:kz] ;
sigma = theta[kz+1] ;
        retp(-0.5*ln(2*pi*sigma^2)-0.5*(yz-xz*b)'(yz-xz*b)/sigma^2) ;
endp ;

proc(1) = betaols(y, x) ;
local n, k, b, u, sse, wvarcov, co, sd, tstat, p, ybar2, r2, ar2, theta, funct, gradi, covar, retco,
theta0, sigma, aic ;
        n = rows(y) ; k = cols(x) ;
        b = inv(x'x)*x'y ;
        u = y - x*b ;
        sse = u'u/(n-k) ;
        theta0 = b|sqrt(sse) ;
        _cml_A = zeros(1,2) ;
        _cml_B = 0 ;
        _cml_C = zeros(1,2) ;
        _cml_D = 0 ;
        _cml_Bounds = { -1.5e0 1.5e0 , -1.5e0 1.5e0 } ;
        _cml_ParNames = "beta"|"sigma" ;
        {theta, funct, gradi, covar, retco} = cmlprt(cml(y~x, 0, &mle, theta0)) ;
        "" ;
        wvarcov = (n/(n-k))*inv(x'x)*(x'*diagrv(eye(n),u^2)*x)*inv(x'x) ;
        sd = sqrt(diag(wvarcov)) ;
        tstat = b./sd ;
        p = cdfrc(tstat,n-k) ;
        ybar2 = (sumc(y)/n)^2 ;
        r2 = (b'x'y - n*ybar2)/(y'y - n*ybar2) ;
        "Ordinary Least Square " ;
        "Included Observations: " n ; "" ;
        "Coefficient Std. Error t-Statistic Prob." ;
        b~sd~tstat~p ; "" ;
        "R-squared = " r2 ;
        "Mean dependent var = " meanc(y) ;
        "Adjusted R-squared = " 1-(1-r2)*((n-1)/(n-k)) ;
        "S.D. dependent var = " sqrt( sumc((y-meanc(y))^2) / (n-1) ) ;
        "S.E. of regression = " sqrt(sse) ;
        "AIC = " -2*(funct/n)+2*(k/n) ;

```

```

"Sum squared resid = " u'u ;
"SC = " -2*(funct/n)+k*log(n)/n ;
"Log likelihood = " funct ;
"F-statistic = " (r2/(1-r2))*((n-k)/k-1) ;
" " ;
r2~meanc(y) ;
1-(1-r2)*((n-1)/(n-k))~sqrt( sumc((y-meanc(y))^2) / (n-1) ) ;
sqrt(sse)~-2*(funct/n)+2*(k/n) ;
u'u~-2*(funct/n)+k*log(n)/n ;
funct~(r2/(1-r2))*((n-k)/k-1) ;
"" ;
"White's Variance-Covariance Matrix" ;
retp(wvarcov) ;
endp ;

```

```

/*      garch.g is the program for compare out-of-sample performances      */
/*      between nonparametric and parametric approaches of conditional      */
/*      heteroscedasticity models by rolling MSPE                          */

new ; cls ; format 2,6 ;
tstrt = timestr(0) ; dstrt = date ;

library CML, pgraph ;
_cml_DirTol = 1e-6 ;
_cml_MaxIters = 9e+1 ;
_cml_Algorithm = 3 ;
_cml_LineSearch = 1 ;
_cml_CovPar = 1 ;

output file = thesis.rst ; output reset on ;

n_obs = 264 ;
load data[n_obs,1] = retw.txt ;          /* retw = ln(ER_t) - ln(ER_t-1) */
data = data*100 ;

i_ini = 214 ;                            /* start first set of data, t = 1,..,214 */

spe = zeros(n_obs-i_ini,1) ;            /* zeros vector for squared prediction*/
                                        /* error (SPE) with size = 50      */
vyhat = spe ;                            /* OLS yhat vector */
spe_ols = spe ;                          /* OLS SPE */

/* vyhat_X = Predicted return vector from model Xth                        */
/* spe_X = SPE for Xth model's predicted return                          */
/* speh1_X = SPE of predicted variance compared with h_np                */
/* speh2_X = SPE compared with h_unc (type I) or y^2 (type II)          */
/* vh_X = predicted conditional variance from model Xth                  */
vyhat_a = spe ; spe_a = spe ; speh1_a = spe ; speh2_a = spe ; vh_a = spe ;
vyhat_g = spe ; spe_g = spe ; speh1_g = spe ; speh2_g = spe ; vh_g = spe ;
vyhat_t = spe ; spe_t = spe ; speh1_t = spe ; speh2_t = spe ; vh_t = spe ;
vyhat_i = spe ; spe_i = spe ; speh1_i = spe ; speh2_i = spe ; vh_i = spe ;
vyhat_e = spe ; spe_e = spe ; speh1_e = spe ; speh2_e = spe ; vh_e = spe ;
vyhat_m = spe ; spe_m = spe ; speh1_m = spe ; speh2_m = spe ; vh_m = spe ;
vyhat_np = spe ; spe_np = spe ;
sfeh_np = spe ;                          /* SPE of predicted NP variance */
vh_np = spe ;                            /* Predicted nonparametric variance */
h_np = spe ;                             /* estimated nonparametric variance */
h_unc = spe ;                            /* estimated unconditional variance */

case = 0 ; do while case <= 6 ;
i = 1 ; do while i <= n_obs - i_ini ;
    y = data[1+i:i_ini+i-1,.] ; n = rows(y) ;
    x = data[i:i_ini+i-2,.] ;
    b0 = inv(x'x)*x'y ;
    u = y - x*b0 ;

```

```

u2 = u^2 ;
xu_2 = ones(rows(u2)-1,1)~u2[1:rows(u2)-1,.] ;
a = inv(xu_2'xu_2)*xu_2'u2[2:rows(u2),.] ; /* Initial parameters (1) */
sy = zeros(n,1) ; sx = zeros(n,1) ;
is = 1 ; do while is <= n ;
    sy[is,.] = stdc(y[1:is,1]) ; sx[is,.] = stdc(x[1:is,1]) ;
is = is + 1 ; endo ;
bs = inv(sx'sx)*sx'sy ; /* Initial parameters (2) */
h0_2 = sumc(u2)/n ; e0_2 = sumc(u2)/n ; /* possible disturbance at time 0 */
h0 = zeros(n,1) ; L = h0 ;
output off ;

if case == 0 ;
    "@----- Nonparametric & OLS -----@" ;
    _cml_DirTol = 1e-6 ;
    _cml_Algorithm = 3 ;
    _cml_LineSearch = 1 ;
    _cml_Bounds = { 0.00015 1e0 } ;
    h_np = h0 ;
    b_ini = stdc(x)*(n^(-1/5)) ;
    m_hat = LocalLin(x,y,b_ini) ;
    g_hat = LocalLin(x,y^2,b_ini) ;
    v_hat = g_hat - (m_hat^2) ;
    h_np[i,.] = v_hat[rows(v_hat),.] ;
    /* h_unc[i,.] = stdc(data[i+1:i_ini+i,]) ; */ /* unc. variance type I */
    h_unc[i,.] = data[214+i,.]^2 ; /* unc. variance type II */
    speh_np[i,.] = (h_unc[i,.] - h_np[i,.] )^2 ;
    vh_np[i,.] = v_hat ;
    vyhat_np[i,.] = m_hat ;

    bols = b0 ; /* OLS Estimator */
    vyhat[i,.] = bols*y[n,.] ;
    spe_ols[i,.] = (data[i+i_ini,.] - vyhat[i,.] )^2 ;

elseif case == 1 ;
    "@----- ARCH -----@" ;
    _cml_DirTol = 1e-2 ;
    _cml_Algorithm = 3 ;
    _cml_LineSearch = 1 ;
    _cml_ParNames = "b0"|"ba0"|"ba1" ;
    _cml_A = zeros(1,3) ; /* _cml_A * p = _cml_B */
    _cml_B = 0 ; /* (Equality Constraint) */
    _cml_C = zeros(1,2)~-ones(1,1) ; /* _cml_C * p >= _cml_D */
    _cml_D = -1 ; /* (Inequality Constraint) */
    _cml_Bounds = { -1e0 1e0 , -1e0 1e0 , -1e0 1e0 } ; /* Parameter Boundaries */
    if i == 1 ;
        h_a = h0 ;
        ba0 = a[1,1]/10 ;
        ba1 = a[2,1] ;
        p_a = b0|ba0|ba1 ;

```

```

else ;
    load p_a ;
endif ;
if i /= n_obs - 1 - i_ini ;
    {p_a, f_a, g_a, cov_a, rcode_a} = cml(y~x, 0, &arch, p_a) ; save p_a ;
else ;
    {p_a, f_a, g_a, cov_a, rcode_a} = cmlPRT(cml(y~x, 0, &arch, p_a)) ;
    "maximized function valued of ARCH : " ftos(f_a,"%#*.IG",15,6); "" ;
    "Variance-Covariance Matrix of ARCH Parameters" ;
    cov_a ; "" ;
endif ;
y_a = y./sqrt(h_a) ; x_a = x./sqrt(h_a) ;
beta_a = inv(x_a'x_a)*x_a'y_a ; /* GLS estimator */
yhat_a = y[n,]*beta_a ;
vyhat_a[i,] = yhat_a ;
spe_a[i,] = (data[i+i_ini,] - yhat_a)^2 ;
/* note :h_a[ig,] = ba0 + ba1*(y[ig-1,]-x[ig-1,]*b0)^2 ; */
h_a_hat = p_a[2,1] + p_a[3,1]*(y[n,1]-x[n,1]*p_a[1,1])^2 ;
vh_a[i,] = h_a_hat ;
speh1_a[i,] = (h_np[i,] - h_a_hat)^2 ;
speh2_a[i,] = (h_unc[i,] - h_a_hat)^2 ;

elseif case == 2 ;
    "@----- GARCH -----@" ;
    _cml_Algorithm = 3 ;
    _cml_LineSearch = 1 ;
    _cml_ParNames = "b0|"bg0|"ag1|"bg1" ;
    _cml_A = zeros(1,4) ;
    _cml_B = 0 ;
    _cml_C = zeros(1,2)~ones(1,2) ;
    _cml_D = -1 ;
    _cml_Bounds = { -1e0 1e0 , -1e0 1.5e0 , -1e0 1e0 , -1e0 1e0 } ;
    if i == 1 ;
        h_g = h0 ;
        bg0 = a[1,1]/100 ;
        ag1 = a[2,1] ;
        bg1 = bs/2 ;
        p_g = b0|bg0|ag1|bg1 ;
    else ;
        load p_g ;
    endif ;
    if i /= n_obs - 1 - i_ini ;
        {p_g, f_g, g_g, cov_g, rcode_g} = cml(y~x, 0, &garch, p_g) ; save p_g ;
    else ;
        {p_g, f_g, g_g, cov_g, rcode_g} = cmlPRT(cml(y~x, 0, &garch, p_g)) ;
        "maximized function valued of GARCH : " ftos(f_g,"%#*.IG",15,6); "" ;
        "Variance-Covariance Matrix of GARCH Parameters" ;
        cov_g ; "" ;
    endif ;
    y_g = y./sqrt(h_g) ; x_g = x./sqrt(h_g) ;

```

```

beta_g = inv(x_g'x_g)*x_g'y_g ;
yhat_g = y[n,]*beta_g ;
vyhat_g[i,] = yhat_g ;
spe_g[i,] = (data[i+i_ini,] - yhat_g)^2 ;
/* note :h_g[ig,] = bg0 + ag1*((y[ig-1,] - x[ig-1,]*b0)^2) + bg1*h_g[ig-1,] ;*/
h_g_hat = p_g[2,1] + p_g[3,1]*((y[n,] - x[n,]*p_g[1,1])^2) + p_g[4,1]*h_g[n,] ;
vh_g[i,] = h_g_hat ;
speh1_g[i,] = (h_np[i,] - h_g_hat)^2 ;
speh2_g[i,] = (h_unc[i,] - h_g_hat)^2 ;

elseif case == 3 ;
" @----- TGARCH -----@ " ;
_cml_Algorithm = 3 ;
_cml_LineSearch = 1 ;
_cml_MaxIters = 2e+2 ;
_cml_ParNames = "b0"|"bt0"|"bt1"|"bt_n"|"bt2" ;
_cml_A = zeros(1,5) ;
_cml_B = 0 ;
_cml_C = zeros(1,2)~-ones(1,3) ;
_cml_D = -1 ;
_cml_Bounds = { -1e0 1e0 , -.1e0 1e0 , 1e-10 1e0 , 0e0 1e0 , -1e0 1e0 } ;
if i == 1 ;
    h_t = h0 ;
    bt0 = a[1,1]/100 ;
    bt1 = a[2,1] ;
    bt_n = a[2,1]/10 ;
    bt2 = bs/2 ;
    p_t = b0|bt0|bt1|bt_n|bt2 ;
else ;
    load p_t ;
endif ;
if i /= n_obs - 1 - i_ini ;
    {p_t, f_t, g_t, cov_t, rcode_t} = cml(y~x, 0, &tgarch, p_t) ; save p_t ;
else ;
    {p_t, f_t, g_t, cov_t, rcode_t} = cmlPRT(cml(y~x, 0, &tgarch, p_t)) ;
    "maximized function valued of TGARCH : " ftos(f_t,"%#*. *IG",15,6); "" ;
    "Variance-Covariance Matrix of TGARCH Parameters" ;
    cov_t ; "" ;
endif ;
y_t = y./sqrt(h_t) ; x_t = x./sqrt(h_t) ;
beta_t = inv(x_t'x_t)*x_t'y_t ;
yhat_t = y[n,]*beta_t ;
vyhat_t[i,] = yhat_t ;
spe_t[i,] = (data[i+i_ini,] - yhat_t)^2 ;
/* note :if (y[ig-1,]-x[ig-1,]*b0) < 0 ;
    h_t[ig,] = bt0 + (bt1*(y[ig-1,]-x[ig-1,]*b0)^2)
                +bt_n*((y[ig-1,]-x[ig-1,]*b0)^2) + bt2*h_t[ig-1,] ;
else ;
    h_t[ig,] = bt0 + (bt1*(y[ig-1,]-x[ig-1,]*b0)^2) + bt2*h_t[ig-1,] ;
endif ; */

```

```

if (y[n,..]-x[n,..]*p_t[1,1]) < 0 ;
    h_t_hat = p_t[2,1] + (p_t[3,1]*(y[n,..]-x[n,..]*p_t[1,1])^2)
                + p_t[4,1]*((y[n,..]-x[n,..]*p_t[1,1])^2) + p_t[5,1]*h_t[n,..] ;

else ;
    h_t_hat = p_t[2,1] + (p_t[3,1]*(y[n,..]-x[n,..]*p_t[1,1])^2) + p_t[5,1]*h_t[n,..] ;
endif ;
vh_t[i,..] = h_t_hat ;
speh1_t[i,..] = (h_np[i,..] - h_t_hat)^2 ;
speh2_t[i,..] = (h_unc[i,..] - h_t_hat)^2 ;

elseif case == 4 ;
    " @----- I-GARCH -----@ " ;
    _cml_Algorithm = 3 ;
    _cml_LineSearch = 1 ;
    _cml_MaxIters = 4e+1 ;
    _cml_ParNames = "b0"|"bi0"|"ai1"|"bi1" ;
    _cml_A = zeros(1,2)~ones(1,2) ;
    _cml_B = 1 ;
    _cml_C = zeros(1,4) ;
    _cml_D = 0 ;
    _cml_Bounds = { -1e0 1e0 , -1e0 1e0 , -1e0 1e0 , -1e0 1e0 } ;
    if i == 1 ;
        h_i = h0 ;
        bi0 = a[1,1]/100 ;
        ai1 = a[2,1] ;
        bi1 = 1 - ai1 ;
        p_i = b0|bi0|ai1|bi1 ;
        /* Restricted to Unt Root */
    else ;
        load p_i ;
    endif ;
    if i /= n_obs - 1 - i_ini ;
        {p_i, f_i, g_i, cov_i, rcode_i} = cml(y~x, 0, &igarch, p_i) ; save p_i ;
    else ;
        {p_i, f_i, g_i, cov_i, rcode_i} = cmlPRT(cml(y~x, 0, &igarch, p_i)) ;
        "maximized function valued of I-GARCH : " ftos(f_i,"%#.*IG",15,6); "" ;
        "Variance-Covariance Matrix of I-GARCH Parameters" ;
        cov_i ; "" ;
    endif ;
    y_i = y./sqrt(h_i) ; x_i = x./sqrt(h_i) ;
    beta_i = inv(x_i'*x_i)*x_i'*y_i ;
    yhat_i = y[n,..]*beta_i ;
    vyhat_i[i,..] = yhat_i ;
    spe_i[i,..] = (data[i+i_ini,..] - yhat_i)^2 ;
    _cml_ParNames = "b0"|"bi0"|"ai1"|"bi1" ;
    /* note :h_i[ig,..] = bi0 + ai1*(y[ig-1,..]-x[ig-1,..]*b0)^2 + bi1*h_i[ig-1,..] ; */
    h_i_hat = p_i[2,1] + p_i[3,1]*(y[n,..]-x[n,..]*p_i[1,1])^2 + p_i[4,1]*h_i[n,..] ;
    vh_i[i,..] = h_i_hat ;
    speh1_i[i,..] = (h_np[i,..] - h_i_hat)^2 ;
    speh2_i[i,..] = (h_unc[i,..] - h_i_hat)^2 ;

```

```

elseif case == 5 ;
    "@----- EGARCH -----@" ;
    _cml_Algorithm = 4 ;
    _cml_LineSearch = 1 ;
    _cml_MaxIters = 4e+2 ;
    _cml_ParNames = "b0"|"be0"|"be1"|"be2"|"be3" ;
    _cml_A = zeros(1,5) ;
    _cml_B = 0 ;
    _cml_C = zeros(1,5) ;
    _cml_D = 0 ;
    _cml_Bounds = { -10e0 10e0 , -15e0 15e0 , -10e0 10e0 , -10e0 10e0 , -10e0 10e0 } ;
    if i == 1 ;
        ln_h_e = h0 ;
        be0 = -a[1,1]*10 ;
        be1 = a[2,1]*2 ;
        be2 = a[2,1] ;
        be3 = -a[2,1]/30 ;
        p_e = b0|be0|.001|be2|.001 ;
    else ;
        load p_e ;
    endif ;
    if i /= n_obs - 1 - i_ini ;
        {p_e, f_e, g_e, cov_e, rcode_e} = cml(y~x, 0, &egarch, p_e) ;
        "Variance-Covariance Matrix of EGARCH Parameters" ;
        cov_e ; "" ; save p_e ;
    else ;
        {p_e, f_e, g_e, cov_e, rcode_e} = cmlPRT(cml(y~x, 0, &egarch, p_e)) ;
        "maximized function valued of EGARCH : " ftos(f_e,"%#*.1G",15,6) ; "" ;
        "Variance-Covariance Matrix of EGARCH Parameters" ;
        cov_e ; "" ;
    endif ;
    y_e = y./sqrt(exp(ln_h_e)) ; x_e = x./sqrt(exp(ln_h_e)) ;
    beta_e = inv(x_e'x_e)*x_e'y_e ;
    yhat_e = y[n,]*beta_e ;
    vyhat_e[i,] = yhat_e ;
    spe_e[i,] = (data[i+i_ini,] - yhat_e)^2 ;
    _cml_ParNames = "b0"|"be0"|"be1"|"be2"|"be3" ;
    /* note :ln_h_e[ig,] = be0 + be1*ln_h_e[ig-1,]
                + be2*abs( (y[ig-1,]-x[ig-1,]*b0) / sqrt(exp(ln_h_e[ig-1,])) )
                + be3*(y[ig-1,]-x[ig-1,]*b0)/sqrt(exp(ln_h_e[ig-1,])) ; */
    lnhe_hat = p_e[2,1] + p_e[3,1]*ln_h_e[n,]
                + p_e[4,1]*abs( (y[n,]-x[n,]*p_e[1,1])/ sqrt(exp(ln_h_e[n,])) )
                + p_e[5,1]*(y[n,]-x[n,]*p_e[1,1])/sqrt(exp(ln_h_e[n,])) ;
    vh_e[i,] = exp(lnhe_hat) ;
    speh1_e[i,] = (h_np[i,] - exp(lnhe_hat))^2 ;
    speh2_e[i,] = (h_unc[i,] - exp(lnhe_hat))^2 ;

elseif case == 6 ;
    "@----- ARCH-M -----@" ; i ;

```

```

_cml_Algorithm = 3 ;
_cml_LineSearch = 1 ;
_cml_ParNames = "b0"|"bh"|"bm0"|"am1" ;
_cml_A = zeros(1,4) ;
_cml_B = 0 ;
_cml_C = zeros(1,4) ;
_cml_D = 0 ;
_cml_Bounds = { -1.5e0 1.5e0 , -5e0 5e0 , -1e0 1e0 , -1e0 1e0 } ;
if i == 1 ;
    h_m = h0 ;
    bh = bs*1.2 ;
    bh = -bs ;
    bm0 = a[1,1]/10 ;
    am1 = a[2,1] ;
    p_m = b0|bh|bm0|am1 ;
else ;
    load p_m ;
endif ;
if i /= n_obs - 1 - i_ini ;
    {p_m, f_m, g_m, cov_m, rcode_m} = cml(y~x, 0, &archm, p_m) ; save p_m ;
else ;
    {p_m, f_m, g_m, cov_m, rcode_m} = cmlPRT(cml(y~x, 0, &archm, p_m)) ;
    "maximized function valued of ARCH-M : " ftos(f_m,"%#*.*IG",15,6); "" ;
    "Variance-Covariance Matrix of ARCH-M Parameters" ;
    cov_m ; "" ;
endif ;
yhat_m = (y[n,.] - h_m[n,])*p_m[1:2,.] ;
vyhat_m[i,.] = yhat_m ;
spe_m[i,.] = (data[i+i_ini,.] - yhat_m)^2 ;
/* note :h_m[ig,.] = bm0 + am1*( (y[ig-1,.] - x[ig-1,.]*b0 - h_m[ig-1,.]*bh)^2 ) ;*/
h_m_hat = p_m[3,1] + p_m[4,1]*( (y[n,.] - x[n,.]*p_m[1,1] - h_m[n,.]*p_m[2,1])^2 ) ;
vh_m[i,.] = h_m_hat ;
speh1_m[i,.] = (h_np[i,.] - h_m_hat)^2 ;
speh2_m[i,.] = (h_unc[i,.] - h_m_hat)^2 ;
endif ;

i = i + 1 ; endo ;
case = case + 1 ; endo ;

output on ;

i = i - 1 ; /* back to last round of iteration */

tfini = timestr(0) ; dfini = date ;
"" ; "" ; "start at " tstrt ; dstrt[1:3,.]' ;
"" ; "" ; "finish at " tfini ; dfini[1:3,.]' ;

graphset ;
_pdate = "" ;
_pcolor = {0,1,4,4} ;

```

```

_pmcolor = zeros(8,1)|15 ;
_pcolor = {0,1} ;
_pltype = {6,1} ;
_pegctl = { 2 4 1 1 } ;

/* ..... */
/* Nonparametric */
_ptek = "npV-Y_1.tkf" ;
xlabel ("Y-1") ; ylabel ("V(Y-1)") ;
m = data[i_ini-1:i_ini+i-2,.]~vh_np ;
m = sortc(m,1) ;
xy(m[:,1],m[:,2]) ;
_ptek = "npVY2-T.tkf" ;
xlabel ("T") ; ylabel ("V(Y-1),Y^2") ;
_pegstr = "Estimated Volatility\000(Actual Return)^2" ;
xy( seqa(0,1,n_obs- i_ini) ,vh_np~data[i_ini+1:i_ini+i,.] ) ;

/* ARCH */
_ptek = "aV-Y_1.tkf" ;
xlabel ("Y-1") ; ylabel ("V(Y-1)") ;
m = data[i_ini-1:i_ini+i-2,.]~vh_a ;
m = sortc(m,1) ;
xy(m[:,1],m[:,2]) ;
_ptek = "aVY2-T.tkf" ;
xlabel ("T") ; ylabel ("V(Y-1),Y^2") ;
_pegstr = "Estimated Volatility\000(Actual Return)^2" ;
xy( seqa(0,1,n_obs- i_ini) ,vh_a ~data[i_ini+1:i_ini+i,.] ) ;

/* GARCH */
_ptek = "gV-Y_1.tkf" ;
xlabel ("Y-1") ; ylabel ("V(Y-1)") ;
m = data[i_ini-1:i_ini+i-2,.]~vh_g ;
m = sortc(m,1) ;
xy(m[:,1],m[:,2]) ;
_ptek = "gVY2-T.tkf" ;
xlabel ("T") ; ylabel ("V(Y-1),Y^2") ;
_pegstr = "Estimated Volatility\000(Actual Return)^2" ;
xy( seqa(0,1,n_obs- i_ini) ,vh_g ~data[i_ini+1:i_ini+i,.] ) ;

/* TGARCH */
_ptek = "tV-Y_1.tkf" ;
xlabel ("Y-1") ; ylabel ("V(Y-1)") ;
m = data[i_ini-1:i_ini+i-2,.]~vh_t ;
m = sortc(m,1) ;
xy(m[:,1],m[:,2]) ;
_ptek = "tVY2-T.tkf" ;
xlabel ("T") ; ylabel ("V(Y-1),Y^2") ;
_pegstr = "Estimated Volatility\000(Actual Return)^2" ;
xy( seqa(0,1,n_obs- i_ini) ,vh_t ~data[i_ini+1:i_ini+i,.] ) ;

```



```

_cml_ParNames = "b0"|"ba0"|"ba1" ;
{p_a, f_a, g_a, cov_a, rcode_a} = cmlPRT(p_a, f_a, g_a, cov_a, rcode_a) ;
"maximized function valued of ARCH : " f_a ; "" ;
stat(p_a, cov_a) "" ;
"MSPE of ARCH = " meanc(spe_a) ; "" ;
"MSPE b/w ARCH's and nonparametric's variance = " meanc(speh1_a) ;
"MSPE b/w ARCH's and unconditional variance = " meanc(speh2_a) ;
"" ;
_ptek = "ARCH.tkf" ;
xy( seqa(0,1,n_obs - i_ini) ,vyhat_a~data[i_ini+1:i_ini+i,.] ) ;

" @----- GARCH -----@ " ;
_cml_ParNames = "b0"|"bg0"|"ag1"|"bg1" ;
{p_g, f_g, g_g, cov_g, rcode_g} = cmlPRT(p_g, f_g, g_g, cov_g, rcode_g) ;
stat(p_g, cov_g) ; "" ;
"maximized function valued of GARCH : " f_g ; "" ;
"MSPE of GARCH = " meanc(real(spe_g)) ; "" ;
"MSPE b/w GARCH's and nonparametric's variance = " meanc(speh1_g) ;
"MSPE b/w GARCH's and unconditional variance = " meanc(speh2_g) ;
"" ;
_ptek = "GARCH.tkf" ;
xy( seqa(0,1,n_obs - i_ini) ,vyhat_g~data[i_ini+1:i_ini+i,.] ) ;

" @----- TGARCH -----@ " ;
_cml_ParNames = "b0"|"bt0"|"bt1"|"bt_n"|"bt2" ;
{p_t, f_t, g_t, cov_t, rcode_t} = cmlPRT(p_t, f_t, g_t, cov_t, rcode_t) ;
"maximized function valued of TGARCH : " f_t ; "" ;
stat(p_t, cov_t) ; "" ;
"MSPE of TGARCH = " meanc(real(spe_t)) ; "" ;
"MSPE b/w TGARCH's and nonparametric's variance = " meanc(real(speh1_t)) ;
"MSPE b/w TGARCH's and unconditional variance = " meanc(real(speh2_t)) ;
"" ;
_ptek = "TGARCH.tkf" ;
xy( seqa(0,1,n_obs - i_ini) ,vyhat_t~data[i_ini+1:i_ini+i,.] ) ;

" @----- I-GARCH -----@ " ;
_cml_ParNames = "b0"|"bi0"|"ai1"|"bi1" ;
{p_i, f_i, g_i, cov_i, rcode_i} = cmlPRT(p_i, f_i, g_i, cov_i, rcode_i) ;
"maximized function valued of I-GARCH : " f_i ; "" ;
stat(p_i, cov_i) ; "" ;
"MSPE of I-GARCH = " meanc(spe_i) ; "" ;
"MSPE b/w I-GARCH's and nonparametric's variance = " meanc(real(speh1_i)) ;
"MSPE b/w I-GARCH's and unconditional variance = " meanc(real(speh2_i)) ;
"" ;
_ptek = "I-GARCH.tkf" ;
xy( seqa(0,1,n_obs - i_ini) ,vyhat_i~data[i_ini+1:i_ini+i,.] ) ;

" @----- EGARCH -----@ " ;
_cml_ParNames = "b0"|"be0"|"be1"|"be2"|"be3" ;
{p_e, f_e, g_e, cov_e, rcode_e} = cmlPRT(p_e, f_e, g_e, cov_e, rcode_e) ;

```

```

"maximized function valued of EGARCH : " f_e ; "" ;
stat(p_e, real(cov_e)) ; "" ;
"MSPE of EGARCH = " meanc(spe_e) ; "" ;
"MSPE b/w EGARCH's and nonparametric's variance = " meanc(real(speh1_e)) ;
"MSPE b/w EGARCH's and unconditional variance = " meanc(real(speh2_e)) ;
"" ;
_ptek = "EGARCH.tkf" ;
xy( seqa(0,1,n_obs - i_ini) ,vyhat_e~data[i_ini+1:i_ini+i,.] ) ;

" @----- ARCH-M -----@ " ;
_cml_ParNames = "b0"|"b0-h"|"bm0"|"am1" ;
{p_m, f_m, g_m, cov_m, rcode_m} = cmlPRT(p_m, f_m, g_m, cov_m, rcode_m) ;
"maximized function valued of ARCH-M : " f_m ; "" ;
stat(p_m, real(cov_m)) ; "" ;
"MSPE of ARCH-M = " meanc(spe_m) ; "" ;
"MSPE b/w ARCH-M's and nonparametric's variance = " meanc(speh1_m) ;
"MSPE b/w ARCH-M's and unconditional variance = " meanc(speh2_m) ;
"" ;
_ptek = "ARCH-M.tkf" ;
xy( seqa(0,1,n_obs - i_ini) ,vyhat_m~data[i_ini+1:i_ini+i,.] ) ;

proc(1) = LocalLin(x,y,b_ini) ;
local b, f, g, cov, retcode, kernel, m_hat, i, xx, xk, xi, beta ;
{b, f, g, cov, retcode}= cml(y~x, 0, &cvll, b_ini) ;
kernel = zeros(n,1) ;
xi = y[rows(y),.]*ones(n,1) ;
kernel = (1/sqrt(2*pi))*exp(-0.5*((x - xi)/b)^2) ;
xx = ones(n,1)~(x-xi) ;
xk = xx.*kernel ;
beta = inv(xk'xx)*xk'y ;
m_hat = beta[1,.] ;
retp(m_hat) ;
endp ;

proc(1) = cvll(th,z) ;
local b, i, xi, sr, r, kernel, xx, xk, beta ;
b = th[1,1] ; y = z[.,1] ; x = z[.,2:cols(z)] ;
sr = zeros(n,1) ;
i = 1 ; do while i <= n ;
xi = x[i,.]*ones(n,1) ;
kernel = (1/sqrt(2*pi))*exp(-0.5*((x - xi)/b)^2) ;
kernel[i,.] = 0 ;
xx = ones(n,1)~(x-xi) ;
xk = xx.*kernel ;
beta = inv(xk'xx)*xk'y ;
sr[i,.] = ( y[i,.] - beta[1,.] )^2 ;
i = i + 1 ; endo ;
retp(-meanc(sr)) ;

```

```

endp ;

proc(1) = arch(th,z) ;
local ig ;
b0 = th[1:cols(x),1] ;
ba0 = th[cols(x)+1,1] ;
ba1 = th[cols(x)+2,1] ;
y = z[.,1] ; x = z[.,2:cols(z)] ;
ig = 1 ; do while ig <= n ;
    if ig == 1 ;
        h_a[ig,.] = ba0/(1-ba1) ;
    else ;
        h_a[ig,.] = ba0 + ba1*(y[ig-1,.]-x[ig-1,.]*b0)^2 ;
    endif ;
    l[ig,.] = - 0.5*ln(2*pi*h_a[ig,.]) - 0.5*( (y[ig,.] - x[ig,.]*b0) ^2 )/h_a[ig,.] ;
ig = ig + 1 ; endo ;
retp(sumc(l)) ;
endp ;

proc(1) = garch(th,z) ;
local ig ;
b0 = th[1:cols(x),1] ;
bg0 = th[cols(x)+1,1] ;
ag1 = th[cols(x)+2,1] ;
bg1 = th[cols(x)+3,1] ;
y = z[.,1] ; x = z[.,2:cols(z)] ;
ig = 1 ; do while ig <= n ;
    if ig == 1 ;
        h_g[ig,.] = bg0/(1-bg1-ag1) ;
    else ;
        h_g[ig,.] = bg0 + ag1*((y[ig-1,.] - x[ig-1,.]*b0)^2) + bg1*h_g[ig-1,.] ;
    endif ;
    l[ig,.] = - 0.5*ln(2*pi) - 0.5*ln(h_g[ig,.]) - 0.5*( (y[ig,.] - x[ig,.]*b0)^2 )/h_g[ig,.] ;
ig = ig + 1 ; endo ;
retp(sumc(l)) ;
endp ;

proc(1) = tgarch(th,z) ;
local ig ;
b0 = th[1:cols(x),1] ;
bt0 = th[cols(x)+1,1] ;
bt1 = th[cols(x)+2,1] ;
bt_n = th[cols(x)+3,1] ;
bt2 = th[cols(x)+4,1] ;
y = z[.,1] ; x = z[.,2:cols(z)] ;
ig = 1 ; do while ig <= n ;
    if ig == 1 ;
        h_t[ig,.] = bt0 / (1-bt1-bt_n-bt2) ;
    else ;
        if (y[ig-1,.]-x[ig-1,.]*b0) < 0 ;

```

```

        h_t[ig,.] = bt0 + (bt1*(y[ig-1,.] - x[ig-1,.] * b0)^2)
                    + bt_n*((y[ig-1,.] - x[ig-1,.] * b0)^2) + bt2*h_t[ig-1,.] ;
        else ;
        h_t[ig,.] = bt0 + (bt1*(y[ig-1,.] - x[ig-1,.] * b0)^2) + bt2*h_t[ig-1,.] ;
        endif ;
    endif ;
    l[ig,.] = - 0.5*ln(2*pi) - 0.5*ln(h_t[ig,]) - 0.5*( (y[ig,.] - x[ig,.] * b0)^2 )/h_t[ig,.] ;
ig = ig + 1 ; endo ;
retp(sumc(l)) ;
endp ;

```

```

proc(1) = igarch(th,z) ;
local ig ;
b0 = th[1:cols(x),1] ;
bi0 = th[cols(x)+1,1] ;
ai1 = th[cols(x)+2,1] ;
bi1 = th[cols(x)+3,1] ;
y = z[.,1] ; x = z[.,2:cols(z)] ;
ig = 1 ; do while ig <= n ;
    if ig == 1 ;
        h_i[ig,.] = bi0 + ai1*e0_2 + bi1*h0_2 ;
    else ;
        h_i[ig,.] = bi0 + ai1*(y[ig-1,.] - x[ig-1,.] * b0)^2 + bi1*h_i[ig-1,.] ;
    endif ;
    l[ig,.] = - 0.5*ln(2*pi) - 0.5*ln(h_i[ig,]) - 0.5*( (y[ig,.] - x[ig,.] * b0)^2 )/h_i[ig,.] ;
ig = ig + 1 ; endo ;

```

```

retp(sumc(l)) ;
endp ;

```

```

proc(1) = egarch(th,z) ;
local ig ;
b0 = th[1:cols(x),1] ;
be0 = th[cols(x)+1,1] ;
be1 = th[cols(x)+2,1] ;
be2 = th[cols(x)+3,1] ;
be3 = th[cols(x)+4,1] ;
y = z[.,1] ; x = z[.,2:cols(z)] ;
ig = 1 ; do while ig <= n ;
    if ig == 1 ;
        ln_h_e[ig,.] = be0 / (1 - be1 - be2 - be3) ;
    else ;
        ln_h_e[ig,.] = be0 + be1*ln_h_e[ig-1,.]
                    + be2*abs((y[ig-1,.] - x[ig-1,.] * b0)/sqrt(exp(ln_h_e[ig-1,.])))
                    + be3*(y[ig-1,.] - x[ig-1,.] * b0)/sqrt(exp(ln_h_e[ig-1,.])) ;
    endif ;
    l[ig,.] = - 0.5*ln(2*pi) - 0.5*ln(exp(ln_h_e[ig,.]))
            - 0.5*( (y[ig,.] - x[ig,.] * b0)^2 )/(exp(ln_h_e[ig,.])) ;
ig = ig + 1 ; endo ;
retp(sumc(l)) ;

```

```

endp ;

proc(1) = archm(th,z) ;
local ig ;
b0 = th[1:cols(x),1] ;
bh = th[cols(x)+1,1] ;
bm0 = th[cols(x)+2,1] ;
am1 = th[cols(x)+3,1] ;
y = z[:,1] ; x = z[:,2:cols(z)] ;
ig = 1 ; do while ig <= n ;
    if ig == 1 ;
        h_m[ig,.] = bm0/(1-am1) ;
    else ;
        h_m[ig,.] = bm0 + am1*( (y[ig-1,]-x[ig-1,]*b0
                                - h_m[ig-1,]*bh)^2 ) ;
    endif ;
    l[ig,.] = - 0.5*ln(2*pi) - 0.5*ln(h_m[ig,])
              - 0.5*((y[ig,]-x[ig,]*b0-h_m[ig,]*bh)^2)/h_m[ig,] ;
ig = ig + 1 ; endo ;
retp(sumc(l)) ;
endp ;

proc(1) = stat(bp,varcov) ;
local sd, k, tstat, p ;
sd = real(sqrt(diag(varcov))) ; k = rows(sd) ;
tstat = bp./sd ;
p = cdfrc(tstat,n-k) ;
" beta-hat sd t-stat p-value " ;
bp~sd~tstat~p ; "" ;
"Variance-Covariance Matrix" ;
retp(varcov) ;
endp ;

```