VOLATILITY FORECASTING WITH FUZZY METHODS

ZIKAI WEI

M.Phil

The Hong Kong Polytechnic University

2017

The Hong Kong Polytechnic University

Department of Applied Mathematics

Volatility Forecasting with Fuzzy Methods

Zikai Wei

A thesis submitted in partial fulfilment of the requirements

for the degree of Master of Philosophy

January 2017

Certificate of Originality

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgment has been made in the text.

_____(Signed)

WEI Zikai (Name of student)

Dedicated to my family, to my mentors and to all my best friends.

Abstract

The thesis will present the volatility forecasting using some fuzzy methods. Three topics are considered:

1. The proposed volatility modeling technique based on fuzzy method is used to replace the model averaging technique in multivariate volatility forecasting.

2. Use the Hidden Markov Model (HMM) for the volatility forecasting of the univariate and multivariate time series.

3. Application of this proposed volatility forecasting technique with fuzzy methods.

For topic 1, a fuzzy-method-based multivariate volatility model is intended to improve the model averaging technique for multivariate volatility forecasting. Volatility modeling is crucial for risk management and asset allocation. This is an influential area in financial econometrics. The central requirement of volatility modeling is to be able to forecast volatility accurately. The literature review of volatility modeling shows that the approaches of model averaging estimation are commonly used to reduce model uncertainty to achieve a satisfactory forecasting reliability. However, those methods attempt to produce a more reliable forecast by confirming all forecasting outcomes equally from several volatility models. Forecasting patterns generated by each model may be similar. Using all forecasting results may cause redundant computations without improving prediction reliability. The proposed mul-

tivariate volatility modeling method which is called the Fuzzy-method-based Multivariate Volatility Model (abbreviated as FMVM) classifies the individual models into smaller scale clusters and selects the most representative model in each group. Hence, repetitive but unnecessary computational burden can be reduced, and forecasting patterns from representative models can be integrated. The proposed FMVM is benchmarked against existing multivariate volatility models on forecasting volatilities of Hong Kong Hang Seng Index (HSI) constituent stocks. Numerical results show that it can obtain relatively smaller forecasting errors with less model complexity.

For topic 2, HMM-based Multivariate Volatility Model is proposed to present another orientation for multivariate volatility forecasting. The foundation of this method is retrieved from technical analysis, which believes historical data will influence the future performance. Recognition of distinct patterns and search similar pattern is crucial for this approach. The proposed HMM-based volatility forecasting algorithm can obtain a volatility matrix from the past distinct patterns, which is found to generate accurate volatility forecast. We use the K-means clustering algorithm to classify the latest attributes into predetermined clusters and label each attributes vector with the index of its cluster. Then, each attributes vector labeled with a distinct index, and those indexes can form a new pattern with a window of a predetermined size. Subsequently, compute the likelihood values of each pattern using trained HMM and define the similarities by using these likelihood values. The closest value of past likelihood value to the likelihood value of current pattern means the corresponding pattern with this closest value share the same similarity with the current pattern. As the foundation of this method implies, we believe the behavior of the most similar pattern will reoccur. The HMM-based multivariate volatility model is also compared to the FMVM and the volatility averaging model with the same data used in topic 1. Numerical results show that HMM-based multivariate model can obtain a better forecasting accuracy than that of the multivariate volatility model

with model averaging technique.

For topic 3, A case of portfolio management of a hedge fund is taken as one application of the proposed volatility forecasting algorithms. A framework of a quantitative trading hedge fund is designed, and a changing allocation problem is considered. The application of both our multivariate volatility model and classical portfolio management models is considered at the same time. Regarding the application of these proposed models, a full flow line of product development with quantitative research is shown in this topic.

Acknowledgements

The endeavor of carrying out research is a fascinatingly non-isolated activity. I am grateful to the several individuals who have supported me in various ways during the M.Phil program and would like to acknowledge their assistance hereby.

First and foremost, I wish to express my deepest thanks to my supervisor, Prof. Wong Heung, for his enlightening guidance, valuable discussions, and insightful ideas throughout the years.

Furthermore, at the forefront of my M.Phil experience, there was the guidance and the kindness of my co-supervisor, Prof. Cedric Kai-Fai Yiu who has been a constant source of inspiration and mentorship.

Especially, I would like to express my heartfelt appreciation to Prof. Zhang Kai, for his strong encouragement and patience during my M.Phil program. What I have benefited most from him is the rigorous and diligent attitude to scientific research.

Finally, I would like to express my special thanks to my parents and my friends for their love, encouragement, and support.

Contents

Certificate of Originality **iii**

Abstract **vii**

Acknowledgements **xi**

List of Figures **xvii**

List of Tables **xix**

List of Notations **xxi**

1 Introduction **1**

1.1 Background 1

1.1.1 Single Factor Model 3

1.1.2 Multi-Factor Model 4

1.1.3 Generalized Auto Regressive Conditional Heteroskedasticity (GARCH) Models 6

1.1.4 Realized Volatility Model 8

1.2 Literature Review 9

1.2.1 Volatility Models 9

1.2.2 Hidden Markov Models and Fuzzy Technique 10

1.2.3 Copulas and Trading Strategies 12

1.3 Summary of Contributions of the Thesis 14

1.4 Organization of the Thesis 15

2 A Novel Multivariate Volatility Modeling **17**

2.1 Univariate Volatility Model 18

2.2 Multivariate Volatility Model 19

2.2.1 General Form 19

2.2.2 Constant Conditional Correlation Model 22

2.2.3 Dynamic Conditional Correlation Model 22

2.2.4 Asymmetric Dynamic Conditional Correlation Model 23

2.2.5 Orthogonal GARCH Model 23

2.2.6 BEKK GARCH Model (BEKK) 24

2.3 The Framework of Fuzzy Multivariate Volatility Modeling 25

2.4 Methodology 27

2.4.1 Training Phase 27

2.4.2 Operation Phase 33

2.5 Empirical Results 36

2.5.1 The Low-Dimensional Case 38

2.5.2 The High-Dimensional Case 39

2.6 Conclusion 40

3 The Hidden Markov Model Based Forecasting Algorithm **41**

3.1 Hidden Markov Models 41

3.1.1 Discrete Markov Processes 42

3.1.2 The HMM 45

3.1.3 An Example 47

3.2 Clustering Multivariate Data Using a Single HMM 53

3.2.1 Generating Likelihood Values Using HMM 54

3.3 HMM-Based Financial Time Series Forecasting 59

3.3.1 Methodology of HMM Forecasting 60
3.3.2 Methodology of HMM Multivariate Volatility Forecasting . . . 63
3.4 Empirical Results 66
3.4.1 Hang Seng Index Forecasting: A Univariate Forecasting Example 66
3.4.2 Volatility Forecasting 69
3.4.3 Univariate Forecasting 72
3.4.4 Multivariate Forecasting 79
3.4.5 Conclusion 87
4 The Application of Fuzzy Volatility Model 89
4.1 Background: Source of Inspiration 89
4.1.1 Trading Strategy Using Copulas Method 91
4.1.2 Tail-Dependence Method with Copulas 96
4.1.3 Relative Method with Copulas 100
4.2 The Quantitative Trading Product Design 101
4.2.1 The Fundamentals of the Product and its Design Idea 101
4.2.2 Copulas-Based Two-Sided-Long-Short Strategy 102
4.2.3 Copulas-Based One-Sided-Long Strategy 105
4.3 The Application of the FMVM to Portfolio Management 106
4.4 The Empirical Result: Trading Strategies 110
4.4.1 Copulas-Based Two-Sided-Long-Short Strategy 110
4.4.2 Copulas-Based One-Sided-Long Strategy 113
4.5 The Empirical Result: the Application of the Proposed Volatility Model 114
5 Conclusions and Future Work 121
5.1 Conclusions 121
5.2 Future Work 122

Bibliography **127**

List of Figures

2.1 The mechanism of the simple model averaging technique 25

2.2 The mechanism of the FMVM 27

2.3 The brief mechanism of the training phase 28

2.4 The mechanism of the training phase 29

3.1 Markov process example 43

3.2 Hidden Markov model example 44

3.3 The state digram of three volatility states 48

3.4 The HMM forecasting result of HSI 69

3.5 Difference between the realized and the forecasted HSI 70

3.6 The absolute difference between the realized and the forecasted HSI . 70

3.7 Bias ratio of the realized results 71

3.8 Bias ratio of the realized results with boundaries 71

3.9 Date item with log-likelihood values (RWS: 20, N: 4) 73

3.10 Log-likelihood values in ascending order (RWS: 20, N: 4) 74

3.11 Difference between the realized and the forecasted results (RWS:20, N:4, P1) 75

3.12 Difference between the realized and the forecasted results (RWS: 20, N: 4, P2) 75

3.13 Difference between the realized and the forecasted results (RWS: 20, N: 4, P3) 76

3.14 Difference between the realized and the forecasted results (RWS: 20, N: 4, P4) 76

3.15 Difference between the realized and the forecasted results (RWS: 20, N: 4, P5) 77

3.16 Date item with log-likelihood values (RWS: 120, N: 15) 77

3.17 Log-likelihood values in ascending order (RWS: 120, N: 15) 78

3.18 Difference between the realized and the forecasted results (RWS: 120, N: 15, P1) 79

3.19 Difference between the realized and the forecasted results (RWS: 120, N: 15, P2) 79

3.20 Difference between the realized and the forecasted results (RWS: 120, N: 15, P3) 80

3.21 Difference between the realized and the forecasted results (RWS: 120, N: 15, P4) 80

3.22 Difference between the realized and the forecasted results (RWS: 120, N: 15, P5) 81

4.1 The diagram of product design 102

4.2 The application of the FMVM in portfolio management 107

4.3 The net values of the CMB pair and the BOC pair 114

4.4 The net values of the BOCs pair and the ABC pair 115

4.5 The net values of the ICBC pair and the CCB pair 115

4.6 The net values of the MB pair and the CITIC pair 116

5.1 The future work of the first orientation 123

5.2 The mechanism of the second orientation 124

5.3 The mechanism of the third orientation 124

List of Tables

2.1 Selected stocks and models 37

2.2 Tracing errors in the training and the operation phase (low-dim) . . . 38

2.3 Tracing errors in the training and the operation phase (high-dim) . . 39

3.1 The last 100 observations 50

3.2 The last 100 observations: identification 50

3.3 The last 100 observations: the same sequence with the same probability 51

3.4 The probability of the occurrence of a given observation sequence . . 52

3.5 The probability of the occurrence of a given observations sequence . . 53

3.6 Selected stocks in each case 72

3.7 Simulation results of forecasting errors with different rolling windows size (low-dim) 82

3.8 Simulation results in two digits (low-dim, two decimal) 83

3.9 Tracking errors of the FMVM, the AMVM and the HMM-based algorithm (low-dim) 83

3.10 Tracking errors of the FMVM, the AMVM and the HMM-based algorithm (low-dim, all attributes) 84

3.11 Simulation results of forecasting errors with different rolling windows size (high-dim) 85

3.12 Simulation results of forecasting errors with different rolling windows size (high-dim, two decimal) 85

3.13 Tracking errors of the FMVM, the AMVM and the HMM-based algorithm (high-dim) 86

3.14 Tracking errors of the FMVM, the AMVM and the HMM-based algorithm (high-dim, all attributes) 86

4.1 Copulas coefficients and tail dependence coefficients 112

4.2 Copulas-based two-sided-long-short strategy in the training Phase . . 112

4.3 Copulas-based two-sided-long-short strategy in the validation Phase . 113

4.4 Copulas-based single-long strategy empirical results 114

4.5 A fusion of the minimum-variance portfolio and the FMVM 116

4.6 The realized results of the minimum-variance portfolio 117

4.7 The comparison of the beforehand adjustment to the benchmarks under the minimum-variance portfolio 117

4.8 A fusion of the maximum-Sharpe-ratio portfolio and the FMVM . . . 118

4.9 The realized results of the maximum-Sharpe-ratio portfolio 118

4.10 The comparison of the beforehand adjustment to the benchmarks under the maximum-Sharpe-ratio portfolio 118

List of Notations

$\mathbb{R}^{m \times n}$	set of $m \times n$ real matrices
$\mathbf{H}_t \in \mathbb{R}^{n \times n}$	conditional volatility matrix at time t, with dimension $n \times n$
$\mathbf{H}_t(k) \in \mathbb{R}^{n,(k \times n)}$	conditional volatility matrix with k-length rolling window size at time t, with dimension $n \times (k \times n)$
$E_t(k) \in \mathbb{R}^{n,(k \times n)}$	residual matrix with k-length rolling window size at time t, with dimension $n \times (k \times n)$
$\mathbf{A} \in \mathbb{R}^{n \times n}$	transmission matrix, with dimension $n \times n$
$\hat{\mathbf{A}} \in \mathbb{R}^{n \times n}$	estimated transmission matrix, with dimension $n \times n$
$\mathbf{B} \in \mathbb{R}^{n \times n}$	emission matrix, with dimension $n \times n$
$\hat{\mathbf{B}} \in \mathbb{R}^{n \times n}$	estimated emission matrix, with dimension $n \times n$
x^T	transpose of matrix/vector x

Chapter 1

Introduction

1.1 Background

In the finance industry, the volatility is maintained to characterize the risk of a financial instrument. Volatility, always denoted by σ, is attributed by the standard deviation of returns, given the trading price series. Volatility is a human-made notion. In some statistical models, the dynamic progress of the volatility is usually presented by the variance of the returns series, the square of the volatility, denoted by σ^2. Therefore the methods to indicate volatility become crucial. Historical volatility and implied volatility are two common ways to measure volatility. Historical volatility is derived from the historical returns series (originating from trading prices series). The reported frequency of returns series can be by ticks, seconds, minutes, hours, days, even months. Volatility is annualized to fill the gaps generated by different report intervals (or different minimum time intervals). Another measurement of volatility is implied volatility. As the name of implied volatility is, volatility is deducted from the trading data of options and Black-Scholes Option Pricing Model (BS formula).

A common measurement to describe the risk of a financial asset is historical volatility. The term and report frequency need to be specified when calculating historical volatility. This period is usually denoted as a 30-day run, 90-day term or 120-day. After defining the term and report frequency, the current realized volatility

is usually defined as the historical volatility based on the latest prices or returns series. More description of different measurements of volatility is shown below:

- Realized volatility is the current or latest historical volatility which is calculated from the returns series with a specified term. Realized volatility is the square root of the realized variance. The annualized variance is realized variance in a specified term multiplies the ratio how many terms can form one trading year. Annualized volatility is the square root of annualized variance.
- Implied volatility is deducted from BS formula with observed price and other information mentioned in BS formula.

Investors care about volatility for many reasons, including but not limited to:

- The change of volatility is a threshold switching between tendency strategy or mean reversion strategy.
- Volatility presents the emotion of investors.
- Volatility is one measurement concerning market uncertainty.
- Volatility matrix, collaborating volatilities and covariances of particular assets in a portfolio, can suggest an optimal position for each asset, given a portfolio management model.
- Change of volatility affects the prices of options. Projection of volatility can instruct a trader's behavior on trading options.
- Upward volatility reading implies further sharply changing of the market.
- Raising of volatility may foretell the beginning of a further market tendency.

- Volatility is a crucial component in some mean reversion trading strategies. For instance, volatility is one of the most important parameters in Bollinger Band means reversion strategy.

- Volatility trading can be fulfilled by trading options, swap, or VIX directly.

Volatility is a traditional risk measurement in quantitative investment; also, volatility plays a paramount role in quantitative investment. There are several classes of volatility models giving different estimation methods of volatility. These categories of volatility models can be grouped as 1) obtaining an estimation of second-moment measurement based on the first-moment measurement; 2) modeling the movement of volatility; 3) modeling volatility by using intra-day data; 4) inferring the volatility from the market expectation.

1.1.1 Single Factor Model

The single factor model can be presented as

$$R_i = \alpha_i + \beta_i F + \varepsilon_i,$$

where R_i is the return of the financial asset, F is a random variable, α is a constant, and β is the coefficient of random variable F. Once the estimations of α and β are known, the expectation of the return and variance are

$$E(R_i) = \alpha_i + \beta_i E(F),$$

and

$$\sigma_i^2 = {\beta_i}^2 {\sigma_F}^2 + \sigma_{\varepsilon_i}^2,$$

where σ_F^2 is variance of the common factor of F, and $\sigma_{\varepsilon_i}^2$ is the variance of the residual term ε_i.

Assume that the factorial term is uncorrelated with the residual term and that different securities share the same common factor. Thus, the covariance with any two securities can be described as

$$\sigma_{ij} = \beta_i \beta_j {\sigma_F}^2.$$

Suppose that any security follows the single factor model, then the return and the variance of a portfolio consisting of n securities can be described as

$$E(R_p) = \sum_{i=1}^{n} \alpha_i \omega_i + \sum_{i=1}^{n} \beta_i \omega_i E(F),$$

and

$$\sigma_p^2 = \sum_{i=1}^{n} {\beta_i}^2 {\omega_i}^2 {\sigma_F}^2 + \sum_{i=1}^{n} \sigma_{\varepsilon_i}^2 {\omega_i}^2.$$

If α_i, β_i, $\sigma_{\varepsilon_i}^2$, the mean and variance of common factor F, $E(F)$, σ_F^2 are known, the mean and variance of the security can be solved directly by using those information.

1.1.2 Multi-Factor Model

Assume that the return of the security i can be described by k factors f_j, $j = 1, 2, \ldots, k$. The multi-factor model can be illustrated as

$$R_i = \alpha_i + \beta_{i1} f_1 + \cdots + \beta_{ik} f_k + \varepsilon_i,$$

where R_i is the return of the security i, α is a constant, β_{ik} is factor exposure coefficient of the k^{th} factor, f_k is a random variable called the k^{th} factor, and ε_i is the error term. The expectation of the return of the security is

$$E(R_i) = \alpha_i + \beta_{i1} f_1 + \cdots + \beta_{ik} f_k.$$

The risk of the security can be described as

$$\begin{aligned} V(R_i) &= V(\alpha_i + \beta_{i1} f_1 + \cdots + \beta_{ik} f_k) + V(\varepsilon_i) \\ &= \beta_i' V(f) \beta_i + V(\varepsilon_i), \end{aligned}$$

where $\beta_i = (\alpha_i, \beta_{i1}, \ldots, \beta_{ik})'$ and $V(f)$ is the $(k+1) \times (k+1)$ covariance matrix. Similarly, the risk of security, under the scheme of the multi-factor model, consists of $\beta_i' V(f) \beta_i$ and $V(\varepsilon)$.

The calculation of the volatility is decided by the covariance matrix $V(f)$ and the variance of the error term $V(\varepsilon_i)$. The $V(\varepsilon_i)$ can be obtained by

$$\hat{\sigma} = \frac{1}{T} \sum_{t=1}^{T} (r_{it} - B_{it}' \hat{f})^2.$$

By deduction, the factor premium can be described by

$$V(f) = (\sum_{t=1}^{T} \sum_{i=1}^{N} B_{it} B_{it}')^{-1} (\sum_{t=1}^{T} \sum_{i=1}^{N} \sigma_i^2 B_{it} B_{it}') (\sum_{t=1}^{T} \sum_{i=1}^{N} B_{it} B_{it}')^{-1},$$

where σ_i^2 is the variance of ε_{it}. Thus, the estimation of $V(f)$ can be described as

$$\hat{V}(\hat{f}) = (\sum_{t=1}^{T} \sum_{i=1}^{N} B_{it} B_{it}')^{-1} (\sum_{t=1}^{T} \sum_{i=1}^{N} \hat{\sigma}_i^2 B_{it} B_{it}') (\sum_{t=1}^{T} \sum_{i=1}^{N} B_{it} B_{it}')^{-1}.$$

Thus, the risk of a portfolio is

$$\begin{aligned} V(R_i) &= V(\alpha_i + f_1 \beta_{i1} + \cdots + f_k \beta_{ik}) + V(\varepsilon_i) \\ &= \beta_i' V(f) \beta_i + V(\varepsilon) \\ &= \beta_i' (\sum_{t=1}^{T} \sum_{i=1}^{N} B_{it} B_{it}')^{-1} (\sum_{t=1}^{T} \sum_{i=1}^{N} \hat{\sigma}_i^2 B_{it} B_{it}') (\sum_{t=1}^{T} \sum_{i=1}^{N} B_{it} B_{it}')^{-1} \beta_i + \hat{\sigma}_i^2. \end{aligned}$$

Assume that there are two securities i and j, their return series are r_{it} and r_{jt} respectively. Their covariance is

$$C(r_{it}, r_{jt}) = B_{it}' V(f) B_{it} + C(\varepsilon_i, \varepsilon_j),$$

where $C(\varepsilon_i, \varepsilon_j) = 0$.

1.1.3 Generalized Auto Regressive Conditional Heteroskedasticity (GARCH) Models

The models from the GARCH class consist of the conditional mean equation and conditional variance equation, where the conditional mean equation represents the first-moment measurement (return) and the conditional variance equation represents the second-moment measurement (variance). Some conventional models are ARCH model, GRACH model, Integrated GARCH model, Asymmetric GARCH model, and Exponential GARCH model.

ARCH Model

Engle founded ARCH(p) model [18], which is described as

$$r_t = \mu + \varepsilon_t,$$
$$\sigma_t^2 = \alpha_0 + \alpha_1 \varepsilon_{t-1}^2 + \cdots + \alpha_p \varepsilon_{t-p}^2,$$

where ε_t is error term, $\alpha_0 > 0$, and $\alpha_1, \ldots, \alpha_p \geq 0$. Due to the continuity of volatility, ARCH model requires a large p, even let $p \to \infty$ (GARCH(∞)).

GARCH Model

Due to the lag of ARCH model, Bollerslev adds the autoregressive terms to ARCH model [7], which is named as GARCH(p,q) model:

$$\sigma_t^2 = \omega + \alpha_1 \varepsilon_{t-1}^2 + \cdots + \alpha_p \varepsilon_{t-p}^2 + \beta_1 \sigma_{t-1}^2 + \cdots + \beta_q \sigma_{t-q}^2,$$

where $\omega > 0$, $\alpha_1, \ldots, \alpha_p \geq 0$, and $\beta_1, \ldots, \beta_p \geq 0$. Comparing to ARCH model, the number of estimation of the parameter in GARCH model much fewer. In the practice, GARCH model with the parameters p and q, no larger than two, can describe the volatility well enough. One of commonly used GARCH models is GARCH(1,1) model:

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

where $\omega > 0$, and $\alpha, \beta \geq 0$. The unconditional variance is

$$\sigma^2 = \frac{\omega}{1 - \alpha - \beta}. \tag{1.1}$$

Integrated GARCH Model

As shown in Equation (1.1): while $\alpha + \beta = 1$, the unconditional covariance is ∞. This is integrated GARCH (IGARCH) model:

$$\sigma_t^2 = \omega + (1 - \lambda)\varepsilon_{t-1}^2 + \lambda\sigma_{t-1}^2.$$

Asymmetric GARCH Model

The conditional variances of the individual returns are given by [22]

$$\sigma_t^2 = c + \sum_{k=1}^{p} \alpha_k \varepsilon_{t-k}^2 + \sum_{k=1}^{p} \vartheta_k I(\varepsilon_{t-k} < 0)\varepsilon_{t-k}^2,$$

$$+ \sum_{j=1}^{q} \beta_j \sigma_{t-j}^2$$

where $I(\varepsilon_{t-k} < 0)$ denotes the indicator function which takes the value of unity if $\varepsilon_{t-k} < 0$, and zero otherwise.

Exponential GARCH model

All the parameters in GARCH model are larger than or equal to zero, i.e., $\omega > 0$, $\alpha_1, \ldots, \alpha_p \geq 0$, $\beta_1, \ldots, \beta_p \geq 0$. Nelson released this constraint [34]

$$log(\sigma_t^2) = \alpha + g(z_{t-1}) + \beta log(\sigma_{t-1}^2),$$

$$g(z_t) = \omega z_t + \lambda(|z_t| - \sqrt{2/\pi}),$$

$$z_t = \varepsilon_t/\sigma \sim N(0, 1).$$

1.1.4 Realized Volatility Model

GARCH is widely used to describe the volatility in low-frequency area. However, the information in the high-frequency world is far more complex and perhaps meaningful than that in a low-frequency world. Further, volatility information retrieved from high-frequency data can be much more useful than that from low-frequency data. Realized volatility, proposed by Anderson and Bollerslev [2], is a risk measurement built on the intraday return series. Because this method does not depend on a particular model, the calculation of realized volatility is simply and directly. Thus, realized volatility is widely used in high-frequency trading. The realized volatility is the square root of the realized variance, where the realized variance is defined as the follows. Divide the trading day k into L intervals. Assume that $P_{t,i}$ is closing price of the i^{th} interval, $i = 1, \ldots, L$, and the log returns in this interval is $r_{i,t} = ln(P_{t,i}) - ln(P_{t,i-1})$; then the realized volatility of the day t can be described as

$$RV = \sum_{i=1}^{L} r_{t,i}^2.$$

If the intraday log return time series are uncorrelated, then Merton uses the quadratic variation approach [32] to prove that

$$\begin{aligned} RV &= \sum_{i=1}^{L} r_{t,i}^2 \\ &= \lim_{L \to \infty} \sum_{i=1}^{L} r_{t,i}^2. \end{aligned}$$

As sampling frequency goes up, the realized volatility will be closer and closer to the integrated volatility. Also, as the sampling frequency becomes bigger, the estimate of realized volatility will be more and more accurate. However, the liquidity problem, noise, and spread usually existed in the market microstructure. Thus the length of

each interval of high-frequency data shall not be too small. More research needs to be done on the optimal length of the interval.

1.2 Literature Review

1.2.1 Volatility Models

The global financial crisis of 2008 has led investors to reassess the forecasting adequacy of financial models against soaring volatilities. The use of volatility models in quantitative risk management has gained increasing importance among academics and practitioners concerned with measuring and managing financial risks [11, 49, 37, 50]. There are also important applications in insurance [31, 43] and supply chain management [48, 44]. These models can be used to forecast volatility in order to assist investors in making financial decisions. These models can be classified into univariate volatility models and multivariate ones. The most widely used univariate model, named Generalized Autoregressive Conditional Heteroscedastic (GARCH) model, was developed in [7].

The multivariate volatility models attempt to specify the dynamic process of the diagonal elements of volatility matrix or variance-covariance matrix. The distinction of each multivariate volatility model derives from the differences between their specifications of the conditional correlation processes. In the literature, various multivariate GARCH models have been developed, including BEKK GARCH model [20], the constant conditional correlation (CCC) model [8], the orthogonal GARCH (OGARCH) model [1], the dynamic conditional correlation model (DCC) [17], and the asymmetric dynamic conditional correlation model (ADCC) [9].

To improve the forecasting accuracy and reduce model uncertainty in multivariate volatility models, the model averaging strategy deploying multi-models, namely the average multivariate volatility model (AMVM) [36] has been developed. The AMVM

generates a multivariate volatility model by integrating the forecasting outcomes of all models equally. However, using forecasting results from all selected multivariate models may not be the most effective, as some models may generate similar prediction patterns with the others. Repetitive but unnecessary computation is used to analyze similar forecasting patterns. The approach increases computational burden of the multivariate volatility models; however, the approach does not improve forecasting accuracy.

To avoid the redundant computation, multivariate volatility models with similar forecasting patterns can be clustered in a group and only the most significant model in each cluster is integrated into the final volatility model. Optimal weights for all the groups are determined in order to reduce forecasting errors. Fuzzy classification operations can perform these processes. Indeed, fuzzy classification operations have been developed successfully in many practical classification systems [3, 4]. They are also widely used in finance [40, 10].

1.2.2 Hidden Markov Models and Fuzzy Technique

Hidden Markov modeling is a probabilistic technique for the study of observed items arranged in a discrete-time series [38]. These observed items can be countably or continuously distributed; they can be scalars or vectors [38]. The hidden Markov model (HMM) is a parametric probability model can generate and analyze a discrete–time series. HMM has two components:

- a finite state Markov chain,
- a finite set of output probability distributions.

In a generating perspective, the Markov chain synthesizes a states sequence and the corresponding output distribution and then converts this sequence to times series. Conversely, in the view of analysis, an observed time series give evidence

about the hidden sequence and the HMM parameters. The Baum-Welch algorithm is one widely used method for maximum likelihood estimation of a HMM. HMMs are widely applied for two reasons: rich mathematical structure and high effectiveness in applications[39]. The HMM technique can be applied to advanced speech recognition [39], natural language processing [6], standard profile analysis [16], time series prediction [24, 25, 26, 23]. These applications have one commonality: they are applied to the pattern recognition field. Financial time series can be viewed as an observed pattern. Therefore, HMM is a useful tool for pattern recognition in the finance industry. Due to the proven suitability for dynamic system modeling, a new method is proposed to forecast stock price using HMM [24]. Later on, Hassan synthesizes the HMM, Artificial Neural Networks and Genetic Algorithms to identify financial market behavior in the historical data [25, 26]. Furthermore, a fuzzy model along with the HMM is synthesized for stock market data forecasting [23]. To reduce the number of fuzzy rules in the previous method, a multi-objective evolutionary algorithm is implemented to make a trade-off between the number of fuzzy rules and the prediction accuracy [26]. However, the details of how HMM can be used to figure out the similar pattern is hidden in the literature. Those HMM forecasting algorithms merely concern the prices data. The other attributes, such as trading volumes, linear correlations, and volatilities, do not involved in these models.

The fuzzy c-means (FCM) clustering algorithm is applicable to a wide variety of numerical data, and it is also accessible to generate fuzzy partitions for any set of numerical data; also, the FCM clustering introduces functions namely membership functions to describe the similarity a data point shares with each cluster [5]. An integrated approach that uses the FCM, mixture models, and the collaborative clustering framework to classify the mixed data which contains both numerical and categorical attributes. This novel clustering framework not only uses the FCM as a component of it but also highlights that the FCM is a very effective and popular

algorithm for numerical data [35]. A new segmentation approach named I-Ching spatial shadowed FCM is very efficient not only in tackling the overlapping segments but also in suppressing the noise in images. This method is also founded on the FCM with some reversion of the FCM: the fuzzy set in the FCM is replaced by the shadowed set and I-Ching operators are used to find the optimal cluster centers of the shadowed FCM [53]. Fuzzy methods are not only widely used in clustering but also in solving programming and transportation problem. An interactive fuzzy goal programming algorithm is proposed to solve a decentralized bi-level multiobjective fractional programming problem, and it can make a fuzzy decision by taking most satisfactory solution for all decision makers at the both levels [13]. In solving the transportation problem, an interval programming model using the nearest interval approximation of trapezoidal fuzzy numbers is developed to obtain the optimal solution of the multi-objective multi-item solid transportation problem under uncertainty [15, 14]. The fuzzy methods are widely applied to different areas. In numerical data clustering, the FCM is very effective and common algorithm. Thus, our framework handling redundant computation is inspired by the FCM clustering initially.

1.2.3 Copulas and Trading Strategies

In probability theory and statistics, a copula is a multivariate probability distribution for which the marginal probability distribution of each variable is uniform. Sklar's Theorem states that any multivariate joint distribution can be written in terms of univariate marginal distribution functions and a copula which describes the dependence structure between the variables [42]. Copulas are popular in high-dimensional statistical applications as they provide an opportunity to easily model and estimate the distribution of random vectors by estimating marginals and copulas separately. There are numerous parametric copula families available, which usually have parameters that control the strength of dependence. Some popular parametric copula

models are outlined below.

Pairs trading is a famous trading strategy invented by Wall Street quantitative analyst Nunzio Tartaglia and his fellow in 1980s and this strategy has been conducted for nearly thirty years. The intrinsic idea of pair trading is to identify a pair of financial assets those have same or similar risk exposure and price movement in the past. Regarding this trading idea, distance method and co-integration method, namely the traditional methods here, are widely conducted. Distance method uses the distance between the standardized prices of two financial assets as the mispricing measurement and conducts the trading instruction when the measurement satisfies the trigger conditions. Similarly, the co-integration method enters into long/short position when the difference of those returns is relatively large and closes its position when the difference is reverting to its mean level. More details can be found in the literature by Engle [19].

Nowadays, people were debating about the usefulness of the traditional methods. While some asserted that one could not receive an abnormal return, some still earned nearly 10% annual return by using this strategy. The traditional methods are based on the assumptions: mean-reverting property and jointly normal distributed property of the returns. Once the latter property is assumed, the linear correlation can thoroughly describe dependency. The distribution of spread/difference is stationary under different circumstances of prices and returns [47]. However, returns in financial markets cannot always follow those properties, which may cause few trading opportunities and low return consequently [28, 12]. Considering this drawback, a new technique should be considered in pairs trading. An alternative way to solve this problem is to relax the assumption of linear dependence structure. Copulas theory is considerable in this case because copulas can describe the nonlinear dependence structure of two random variables with more robust and accurate description. This method can find joint distribution of two marginal distributions of two random vari-

ables and it was originally founded in the book by Sklar [42]. As one knows the historical returns of two stocks, she or he can estimate the marginal distribution of each stock with the data of historical returns and then estimate a particular kind of copula of this two marginal distributions.

1.3 Summary of Contributions of the Thesis

The original contributions of this thesis are as follows:

- We propose a novel multivariate volatility model namely the Fuzzy-method-involving Multivariate Volatility Model (FMVM) to cluster multivariate volatility models with similar forecasting patterns. Here we propose the FMVM as this is effective in gathering similar patterns for forecasting stock exchanges which are involved with uncertainty [10]. As only significant forecasting patterns are used by the proposed multivariate volatility model, repetitive and unnecessary computational burden can be reduced in the final multivariate volatility model. This algorithm contributes on the following research issues: (1) this algorithm is the vanguard which combines the fuzzy clustering technique and multivariate volatility models in order to perform forecasting; (2) this research proposes a weighting mechanism to select models from different clusters and aggregate forecasting power of those selected models to perform forecasting; (3) our proposed FMVM overcomes the limitation of the existing AMVM which requires excessive utilization and computation burden, and existing AMVM is likely to generate unnecessary forecasting errors; (4) the proposed FMVM can achieve better prediction accuracy with smaller computational efforts.

- The Hidden Markov Model based (HMM) forecasting algorithm is proposed to increase the forecasting accuracy, which can be used in both univariate

and multivariate volatility forecasting. Any attributes can be generated into recognition patterns. Based on the similarity of recognition pattern, the Hidden Markov Model based (HMM) forecasting algorithm can give forecasting results based on more information than other methods. This algorithm contributes on the following research issues: (1) this algorithm implements the foundation of the technical analysis, "the history implies the future"; (2) As most volatility matrix forecasting methods are restricted to the dynamic processes of volatiles and covariances, the HMM forecasting algorithm can extend the attributes, for instance, more on trading volume, prices (open, high, low, close), some macroeconomic factors than volatiles and covariances.

- General volatility forecasting research only seldom show their applications to a real application. The application of our proposed multivariate volatility forecasting algorithm is presented in a full chain of a financial product design: the formation of a financial product along with a trading strategy; the volatility matrix prediction of a portfolio consisting of self-designed financial products; the adjustment of the allocation of the portfolio based on the portfolio management model and the forecasted volatility matrix.

1.4 Organization of the Thesis

The thesis is structured as it follows.

- Chapter 2 focuses on a proposed fuzzy-involving forecasting algorithm. The proposed multivariate volatility modeling method which is called the Fuzzy-method-involving Multivariate Volatility Model (abbreviated as FMVM) classifies the individual models into smaller scale clusters and selects the most representative model in each cluster. Hence, repetitive but unnecessary computational burden can be reduced and forecasting patterns from representative

models can be integrated. The proposed FMVM is benchmarked against existing multivariate volatility models on forecasting volatilities of Hong Kong Hang Seng Index (HSI) constituent stocks. Numerical results show that it can obtain relatively lower forecasting errors with less model complexity.

- Chapter 3 proposes a HMM-based forecasting algorithm. Technical analysis believe that the historical data will interpret the future performance; technical analysis needs the statistics and machine learning tools to recognize distinct patterns. Thus the data patterns can be used to trained the HMM and later use the trained HMM to calculate the log-likelihood values of the new pattern; then, find the date that shares the most close value with the log-likelihood values of the present pattern and use the change between this data point and the following data point to do forecasting of the present data point.

- Chapter 4 is devoted on the application of proposed volatility forecating models. To start with, we present a quantitative trading product design. Then we integrate the application of the fuzzy volatility model to the product design. As the framework of the product is assumed, the risk management of the product should be taken in to consideration. Usually, the practitioner need to manage a portfolio after constructing different financial products in a period. Then, the portfolio management is take into consideration afterwards. Therefore, the volatility matrix, a crucial part in portfolio risk management, should be accurately forecasted. To tackle the difficulty of forecasting volatility matrix, the methods as the fuzzy volatility modeling technique and the fuzzy HMM technique can be applied to the product design. Finally, detailed implementation procedures are also presented by means of the case analysis.

- Chapter 5 concludes the whole thesis and plans for the future work.

Chapter 2

A Novel Multivariate Volatility Modeling

In this chapter, we propose a novel multivariate volatility models namely the Fuzzy-method-involving Multivariate Volatility Model (FMVM) in order to cluster multivariate volatility models with similar forecasting patterns. Here we propose the FMVM as this is effective on clustering similar patterns for forecasting stock exchanges which are involved with uncertainty [10]. In order to evaluate the performance of the forecasted volatility matrix, the Frobenius norm is used to evaluate forecasting errors of different multivariate volatility models relative to the realized covariance matrix [2]. First the FMVM classifies all the individual models into smaller scale clusters by using the fuzzy C-means clustering algorithm where the optimal number of clusters is given. Hence, models which generate similar forecasting patterns can be grouped into a cluster. It then selects the model with the lowest tracking error from each cluster. Subsequently, it determines the optimal weight for each selected model. As only significant forecasting patterns are used by the proposed multivariate volatility model, repetitive and unnecessary computational burden can be reduced in the final multivariate volatility model. The effectiveness of the proposed FMVM is evaluated based on two cases with either 4 or 15 HSI constituent stocks. The empirical result shows that the FMVM is able to improve the forecasting

accuracy compared with the average multivariate volatility model (AMVM). It also shows that the computational cost required by the FMVM is less than that required by the AMVM.

The proposed FMVM is applied to risk management and portfolio management. As an example, a fund manager needs to determine the predetermined weights of the assets from the same portfolio. When the other constraints are unchanged, the change of the volatility matrix affects the optimal weights of all the assets in the same portfolio. To address the uncertainty, one needs to process risk management in order to adjust portfolio in advance. However, some widely used optimal portfolio models are very sensitive to the volatility matrix. Therefore, a better forecasting method is essential.

The rest of this chapter is organized as follows: Section 2.1 and 2.2 give a brief introduction of some widely used univariate and multivariate volatility models; Section 2.3 and 2.4 discuss the particularization of the proposed FMVM; Section 2.5 presents the empirical results and analysis. Several well-known and widely used volatility models were applied to 15 highly weighted HSI constituent stocks from November 2010 to October 2014. The effectiveness of the FMVM is compared with those tested volatility models.

2.1 Univariate Volatility Model

This section first introduces the univariate generalized autoregressive conditional heteroscedastic (GARCH) model and it then shows a general form of the multivariate GARCH models.

Bollerslev proposed a univariate volatility model namely GARCH model [7]. For a log return time series r_t, let $\varepsilon_t = r_t - \mu_t$ be innovation at time t. Denote the

information set available at time t by F_t.

$$\mu_t = E(r_t|F_t)$$

is the conditional mean of r_t and

$$\begin{aligned}\sigma_t^2 &= Var(r_t|F_t) \\ &= E[(r_t - \mu_t)^2|F_t]\end{aligned}$$

is conditional variance of r_t. ε_t is formulated in the model GARCH(p, q) as

$$r_t = \mu_t + \varepsilon_t,\ \varepsilon_t = \sigma_t z_t,$$

$$\sigma_t^2 = c + \sum_{i=1}^{p} \alpha_i \varepsilon_{t-i}^2 + \sum_{j=1}^{q} \beta_j \sigma_{t-j}^2,$$

where $\{z_t\}$ is a sequence of independent and identically distributed $(i.i.d.)$ random variables with mean equal to 0 and variance equal to 1, $c > 0$, $\alpha_i \geq 0$, $\beta_j \geq 0$, and $\sum_{i=1}^{max(p,q)}(\alpha_i + \beta_i) < 1$. For $i > p$ and $j > q$, a_i and b_j are given as $\alpha_i = 0$ and $\beta_j = 0$ respectively. Given the GARCH(1,1) model and assume that the forecast origin is t, we define one-step-ahead forecast form can be defined as

$$\sigma_{t+1}^2 = c + \alpha_1 \varepsilon_t^2 + \beta_1 \sigma_t^2.$$

2.2 Multivariate Volatility Model

In this section, we turn to the general form of the multivariate volatility models and the widely-used multivariate volatility models.

2.2.1 General Form

Almost all the multivariate volatility models can be represented as the decomposition of the conditional volatility matrix $\mathbf{H}_t$ [8]:

$$\mathbf{H}_t = \mathbf{D}_t \mathbf{R}_t \mathbf{D}_t \tag{2.1}$$

where $\mathbf{R}_t$ is the one-step-ahead conditional correlation matrix with its (i,j)th entry denoted by $\rho_{ij,t}$, conditional linear correlation of $r_{i,t}$ and $r_{j,t}$, and $\mathbf{D}_t$ is a diagonal matrix with $\sqrt{\sigma_{ii,t}}$ on the (i,i)th entry. Equation (2.1) is a convenient decomposition, which allows separate specification of the conditional volatilities and conditional cross-asset returns correlations. The specification of $\sqrt{\sigma_{ii,t}}$ and $\rho_{ij,t}$ is varied among those multivariate GARCH models which is given in the following part. Given that, we have n financial assets and $\varepsilon_t = r_t - \mu_t$, is the $n \times 1$ vector of residuals from the ordinary least square regressions (OLS) of the predictor variables. For n financial assets, D_t can be given as

$$\mathbf{D}_t = \begin{bmatrix} \sigma_{1,t} & & & \\ & \sigma_{2,t} & & \\ & & \ddots & \\ & & & \sigma_{n,t} \end{bmatrix}$$

and $\mathbf{R}_t$ is given as

$$\mathbf{R}_t = \begin{bmatrix} 1 & \rho_{12,t} & \cdots & \rho_{1n,t} \\ \rho_{21,t} & 1 & \cdots & \rho_{2n,t} \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{n1,t} & \rho_{n2,t} & \cdots & 1 \end{bmatrix}$$

where $\rho_{ij,t} = \rho_{ji,t}, i \neq j$. $\mathbf{H}_t$ in Equation (2.1) can be elaborated as

$$
\begin{aligned}
\mathbf{H}_t &= \mathbf{D}_t \mathbf{R}_t \mathbf{D}_t \\
&= (\mathbf{D}_t \mathbf{R}_t) \mathbf{D}_t \\
&= \begin{bmatrix} \sigma_{1,t} & \sigma_{1,t}\rho_{12,t} & \cdots & \sigma_{1,t}\rho_{1n,t} \\ \sigma_{2,t}\rho_{21,t} & \sigma_{2,t} & \cdots & \sigma_{2,t}\rho_{2n,t} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{n,t}\rho_{n1,t} & \sigma_{n,t}\rho_{n2,t} & \cdots & \sigma_{n,t} \end{bmatrix} \mathbf{D}_t \\
&= \begin{bmatrix} \sigma_{1,t}\sigma_{1,t} & \sigma_{1,t}\rho_{12,t}\sigma_{2,t} & \cdots & \sigma_{1,t}\rho_{1n,t}\sigma_{n,t} \\ \sigma_{2,t}\rho_{21,t}\sigma_{1,t} & \sigma_{2,t}\sigma_{2,t} & \cdots & \sigma_{2,t}\rho_{2n,t}\sigma_{n,t} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{n,t}\rho_{n1,t}\sigma_{1,t} & \sigma_{n,t}\rho_{n2,t}\sigma_{2,t} & \cdots & \sigma_{n,t}\sigma_{n,t} \end{bmatrix} \\
&= \begin{bmatrix} \sigma_{11,t} & \sigma_{12,t} & \cdots & \sigma_{1n,t} \\ \sigma_{21,t} & \sigma_{22,t} & \cdots & \sigma_{2n,t} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{n1,t} & \sigma_{n2,t} & \cdots & \sigma_{nn,t} \end{bmatrix}
\end{aligned}
$$

where $\sigma_{ij,t} = \sigma_{ji,t}, i \neq j$. Based on the one-step-ahead forecast, Equation (2.1) are given as $\hat{\mathbf{H}}_{t+1} = \hat{\mathbf{D}}_{t+1} \hat{\mathbf{R}}_{t+1} \hat{\mathbf{D}}_{t+1}$.

Given the different specifications of $\mathbf{D}_t$ or/and $\mathbf{R}_t$, the general form can be represented as a different type of multivariate volatility models:

1) When $\mathbf{R}_t$ is simply assumed as the constant conditional correlation matrix without varying with time, $\mathbf{R}$, $\mathbf{R}_t = \mathbf{R}$. The general form can be written into the Constant Conditional Correlation Model (CCC) which is detailed in [8].

2) When $\mathbf{R}_t$ is a time varying matrix, the CCC can be reformulated as the Dynamic Conditional Correlation Model (DCC) which can be referred to [17].

3) When the possibility of asymmetric effects can be allowed on conditional variance and correlations, the DCC can be relaxed into the the Asymmetric Dynamic Conditional Correlation Model (ADCC) which can be referred to [9].

4) When the standardized return is defined and the static principle component decompositions of standardized residuals are used, the OGARCH can be formulated, where the Orthogonal GARCH model (OGARCH) can be referred to [1].

5) When $\mathbf{H}_t$ is assumed as the positive definite, the BEKK GARCH model (BEKK) is somehow a quartic form of the general which can be referred to [20].

2.2.2 Constant Conditional Correlation Model

Bollerslev developed a multivariate GARCH model which assumes the (constant) conditional correlations of the $\varepsilon_{it}, t = 1, 2, \ldots, n$ constituting the conditional correlation matrix, $\mathbf{R}_{1t}$ [8], ahead conditional correlations are constant. This developed model is namely CCC is given by $\mathbf{H}_{1t} = \mathbf{D}_{1t}\mathbf{R}_{1t}\mathbf{D}_{1t}$, where $\mathbf{D}_{1t}$ is a diagonal matrix with σ_{it} [7]

$$\sigma_{i,t}^2 = c_i + \sum_{k=1}^{p} \alpha_{ik}\varepsilon_{i,t-k}^2 + \sum_{j=1}^{q} \beta_{ij}\sigma_{i,t-j}^2,$$

where $c_i, \alpha_{i1}, \ldots, \alpha_{ip}, \beta_{i1}, \ldots,$ and β_{iq} are all constants.

2.2.3 Dynamic Conditional Correlation Model

Engle developed time-varying conditional correlation matrix [17]. Let $\mathbf{D}_{2t}$ equal to $\mathbf{D}_{1t}$. Given the standardized residuals, $\tilde{\varepsilon}_{2t} = (\mathbf{D}_{2t})^{-1}\varepsilon_t$, the DCC model assumes that the (i, j)th element of the conditional covariance matrix of $\tilde{\varepsilon}_{2t}$, namely $\mathbf{R}_{2t}$, is given by $q_{ijt}/\sqrt{q_{iit}q_{jjt}}$, where q_{ijt} is the (i, j)th element of matrix of $\mathbf{Q}_t$,

$$\mathbf{Q}_t = \overline{\mathbf{Q}}(1 - \gamma_2 - \delta_2) + \gamma_2\tilde{\varepsilon}_{2,t-1}\tilde{\varepsilon}_{2,t-1}^{'} + \delta_2\mathbf{Q}_{t-1},$$

for a fixed positive definite matrix $\overline{\mathbf{Q}}$, and positive parameters satisfying $\gamma_2 + \delta_2 < 1$. Finally, $\mathbf{H}_{2t}$ is obtained by re-combining $\mathbf{D}_{2t}$ and $\mathbf{R}_{2t}$ based on Equation (2.1). The details can be found in [17].

2.2.4 Asymmetric Dynamic Conditional Correlation Model

Cappiello, et al. generalized the DCC allowing for the possibility of asymmetric effects on conditional variances and correlations [9]. The conditional variances of the individual returns are given by [22]

$$\sigma_{i,t}^2 = c_i + \sum_{k=1}^{p} \alpha_{ik}\varepsilon_{t-k}^2 + \sum_{k=1}^{p} \vartheta_{ik} I(\varepsilon_{i,t-k} < 0)\varepsilon_{i,t-k}^2$$

$$+ \sum_{j=1}^{q} \beta_{ij}\sigma_{i,t-j}^2,$$

where $I(\varepsilon_{i,t-k} < 0)$ denotes the indicator function which takes the value of unity if $\varepsilon_{i,t-k} < 0$, and zero otherwise. Let $\tilde{\varepsilon}_{3t} = (\mathbf{D}_{3t})^{-1}\varepsilon_t$, where $\mathbf{D}_{3t}$ is the diagonal matrix formed with the square roots of $\sigma_{i,t}^2$. The ADCC model assumes that the (h,j)th entry of the conditional covariance matrix of $\sigma_{i,t}^2$, namely $\mathbf{R}_{3t}$, is given by $q_{ijt}/\sqrt{q_{iit}q_{jjt}}$, where q_{ijt} is the (i,j)th element of matrix $\mathbf{Q}_t$ defined by

$$\mathbf{Q}_t = \overline{\mathbf{Q}}(1 - \gamma_3 - \delta_3 - \vartheta_3) + \gamma_3\tilde{\varepsilon}_{3,t-1}\tilde{\varepsilon}_{3,t-1}^{'} + \delta_3\mathbf{Q}_{t-1}$$

$$+\vartheta_3\underline{\varepsilon}_{3,t-1}\underline{\varepsilon}_{3,t-1}^{'},$$

where $\underline{\varepsilon}_{3,t} = \tilde{\varepsilon}_{3,t} \odot I(\varepsilon_{3,t} < 0)$ (here $\odot$ denotes the Hadamard product), $\overline{\mathbf{Q}}$ is a fixed positive definite matrix, and γ_3, δ_3 and ϑ_3 are positive parameters satisfying $\gamma_3 + \delta_3 + \vartheta_3 < 1$. Finally, $\mathbf{H}_{3t}$ is constructed using $\mathbf{D}_{3t}$ and $\mathbf{R}_{3t}$ as in Equation (2.1).

2.2.5 Orthogonal GARCH Model

The orthogonal GARCH, abbreviated as O-GARCH, is developed in [1] and uses a static principle component decomposition of standardized residuals defined by

$$\tilde{\varepsilon}_{it} = \frac{\varepsilon_{it} - \bar{\varepsilon}_{iT}}{s_{iT}}, t = 1, 2, \ldots, T,$$

where $\bar{\varepsilon}_{iT}$ and s_{iT} are the sample mean and the standard deviation of the returns. Denote the sample covariance matrix of the standardized return by

$$\tilde{\mathbf{S}}_T = \frac{\sum_{t=1}^{T} \tilde{\varepsilon}_t \tilde{\varepsilon}_t{}'}{T}, \tilde{\varepsilon}_t = (\tilde{\varepsilon}_{1t}, \ldots, \tilde{\varepsilon}_{nt}).$$

Then

$$\tilde{\mathbf{S}}_T \mathbf{W}_T = \Lambda_T \mathbf{W}_T,$$

where $\mathbf{W}_T$ and Λ_T are the corresponding $n \times n$ matrices of eigenvectors and eigenvalues, respectively. Referring to [1], set

$$\mathbf{H}_{4t}(u) = \mathbf{V}\mathbf{W}(u)\Gamma_t(u)\mathbf{W}(u)'\mathbf{V},$$

where $\mathbf{W}(u) = (\mathbf{w}_1, \ldots, \mathbf{w}_u)$ denotes the $n \times u$ matrix of eigenvectors corresponding to the first largest u eigenvalues, $\mathbf{V}$ is a diagonal matrix with the sample standard deviation of r_{it}, illustrated in Section 2.1, on the (i,i)th entry and $\Gamma_t(u)$ is a $u \times u$ diagonal matrix whose (j,j)th entry, $\gamma_{jt}, j = 1, \ldots, u$, is assumed to satisfy the following univariate GARCH(p,q) specification

$$\gamma_{jt} = c_j + \sum_{k=1}^{p} \alpha_{jk} s_{jt-k}^2 + \sum_{l=1}^{q} \beta_{jl} \gamma_{jt-l}, j = 1, \ldots, u,$$

where $s_j = (\epsilon_1, \ldots, \epsilon_T)'\mathbf{w}_j, j = 1, \ldots, N$. The details refer to [1].

2.2.6 BEKK GARCH Model (BEKK)

It is developed in [20] a multivariate volatility model known as BEKK GARCH model (BEKK). The specification of BEKK is given by

$$\mathbf{H}_t = \mathbf{C}_0\mathbf{C}_0' + \sum_{k=1}^{n}\sum_{i=1}^{p} \mathbf{A}_{ki}' \mathbf{E}_{t-i}\mathbf{E}_{t-i}\mathbf{A}_{ki}$$

$$+ \sum_{k=1}^{n}\sum_{j=1}^{q} \mathbf{B}_{kj}' \mathbf{H}_{t-j}\mathbf{B}_{kj},$$

where $\mathbf{C}_0$ is a lower triangle matrix, $\mathbf{A}_{ki}$ and $\mathbf{B}_{ki}$ are $n \times n$ parameter matrices, and $\mathbf{E}_{t-i} = (\varepsilon_{1,t-i}, \ldots, \varepsilon_{n,t-i})'$.

2.3 The Framework of Fuzzy Multivariate Volatility Modeling

In order to moderate the model uncertainty and to improve the forecasting accuracy simultaneously, Pesaran et al. developed the average volatility models by applying model average techniques [36]. The average multivariate volatility model (AMVM) is given as

$$\mathbf{H}_t^{amvm} = \frac{1}{m} \sum_{i=1}^{m} \mathbf{H}_{i,t},$$

where $\mathbf{H}_{i,t}$ is the forecasted conditional volatility matrix in the multivariate volatility model and m is the total number of the multivariate volatility models. Figure 2.1 illustrates the mechanism of the model averaging technique which generates an identical weight to m multivariate volatility models to forecast the one-day ahead volatility matrix. However, this method has several limitations:

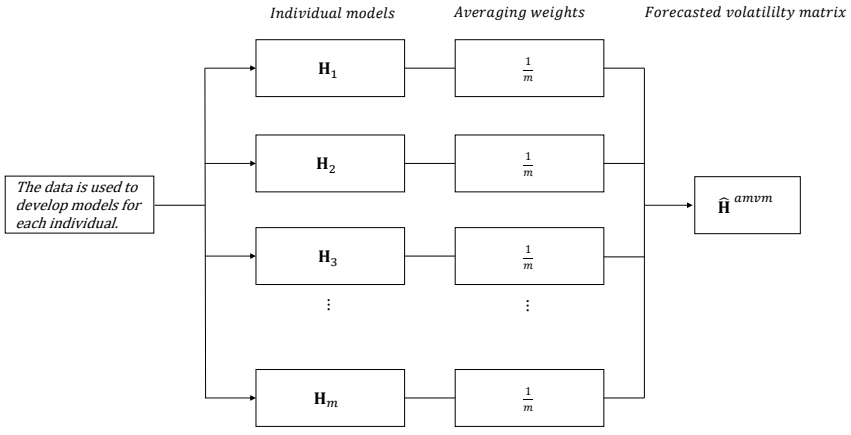


Figure 2.1: The mechanism of the simple model averaging technique

1) The excessive utilization ascribes to the similarity among different multivariate volatility models. These models sharing the similarity are classified into the same cluster. One forecasted volatility matrix generated by a volatility model may have linear correlation with another forecasted matrix generated by another similar model. The model averaging technique does not take this phenomenon into consideration.

2) The computation burden derived from the utilization of large scale multimodels. Although this large amount of computation helps to adapt more situation and it performs indeed better than a single model, it occupies a lot of computation resources. Also, they may share similar properties or attributes with some other models, which can be grouped together to release the computation burden.

3) There exist some individual models which may involve higher tracking errors comparing with other considered models. Although it is a forthright way to weight each model equally, these models with higher tracking errors still affect the averaging tracking errors by model averaging. A more proper way is to give smaller weights to those models with higher tracking errors, and to give larger weights to those models with higher forecasting accuracy.

To tackle the limitations of the existing model averaging technique, we propose the FMVM which attempts to give an alternative way to improve the forecasting accuracy. As illustrated in Figure 2.2, the mechanism of the FMVM is divided by training and operation phases. In the training phase, the FMVM inputs the data and then applies the data to all individual models; it groups such individual models into different clusters using fuzzy C-means (FCM) clustering algorithm [10], when the number of the clusters is given; it selects the model with minimum tracking errors from each cluster; it determines the optimal weights to those selected models

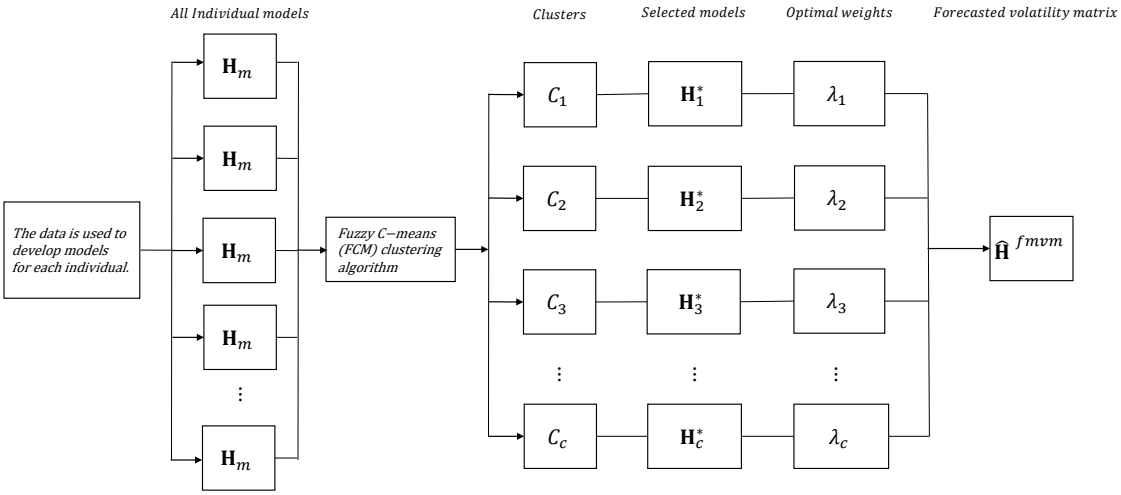


Figure 2.2: The mechanism of the FMVM

such that the mixed model has the minimum tracking errors. The FMVM decides the optimal number of the clusters, c, and selects the model with minimum tracking errors in each cluster. Subsequently, in the operation phase, the FMVM performs an one-day ahead forecasting based on c and the resolved optimal weights of the selected models. It uses the selected individual models with their optimal weights to forecast the volatility matrix.

2.4 Methodology

2.4.1 Training Phase

The mechanism of the training phase is shown in the Figure 2.3 and Figure 2.4. In the training phase, the FMVM attempts to solve model-averaging problem based on fuzzy clustering and quadratic programming. We assume that m multivariate volatility models are used and the data length is $3k - 1$. In both training and

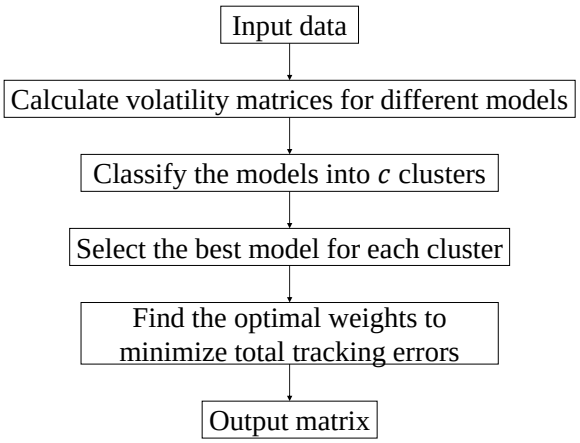


Figure 2.3: The brief mechanism of the training phase

operation phases, $2k-1$ data are involved. Define

$$\mathbf{H}_t^{real}(k) = (\mathbf{H}_{t-k+1}^{real}, \ldots, \mathbf{H}_t^{real}),$$
$$\mathbf{H}_{i,t}(k) = (\mathbf{H}_{i,t-k+1}, \ldots, \mathbf{H}_{i,t}), i = 1, \ldots, m,$$
$$L_{i,t}(k) = \|\mathbf{H}_{i,t}(k) - \mathbf{H}_t^{real}(k)\|_F,\ i = 1, \ldots, m,$$

where $\mathbf{H}_t^{real}(k)$, $\mathbf{H}_{i,t}(k)$, and $L_{i,t}(k)$ are the $n \times (k \times n)$ realized volatility matrix, m input time-varying $n \times (k \times n)$ volatility matrix for the ith model, and the tracking error for the ith model, respectively. Assume that there is a data set with m input time-varying $(k \times n) \times n$ volatility matrices $\mathbf{H}_{i,t}(k)$ $(i = 1, \ldots, m)$ and one output time-varying matrix $\mathbf{H}_t^{fmvm}$. The training phase performs three tasks: (1) classify the input models with the fuzzy C-means classification and select the model with the least tracking errors from each cluster, (2) determine the optimal weight for each selected model, (3) determine the optimal number of clusters, c.

Model Selection

The FMVM first transforms the input matrices into the vectors using the half-vectorization. The half-vectorization, $vech(\mathbf{H}_{j,t})$, processes the symmetric $n \times n$

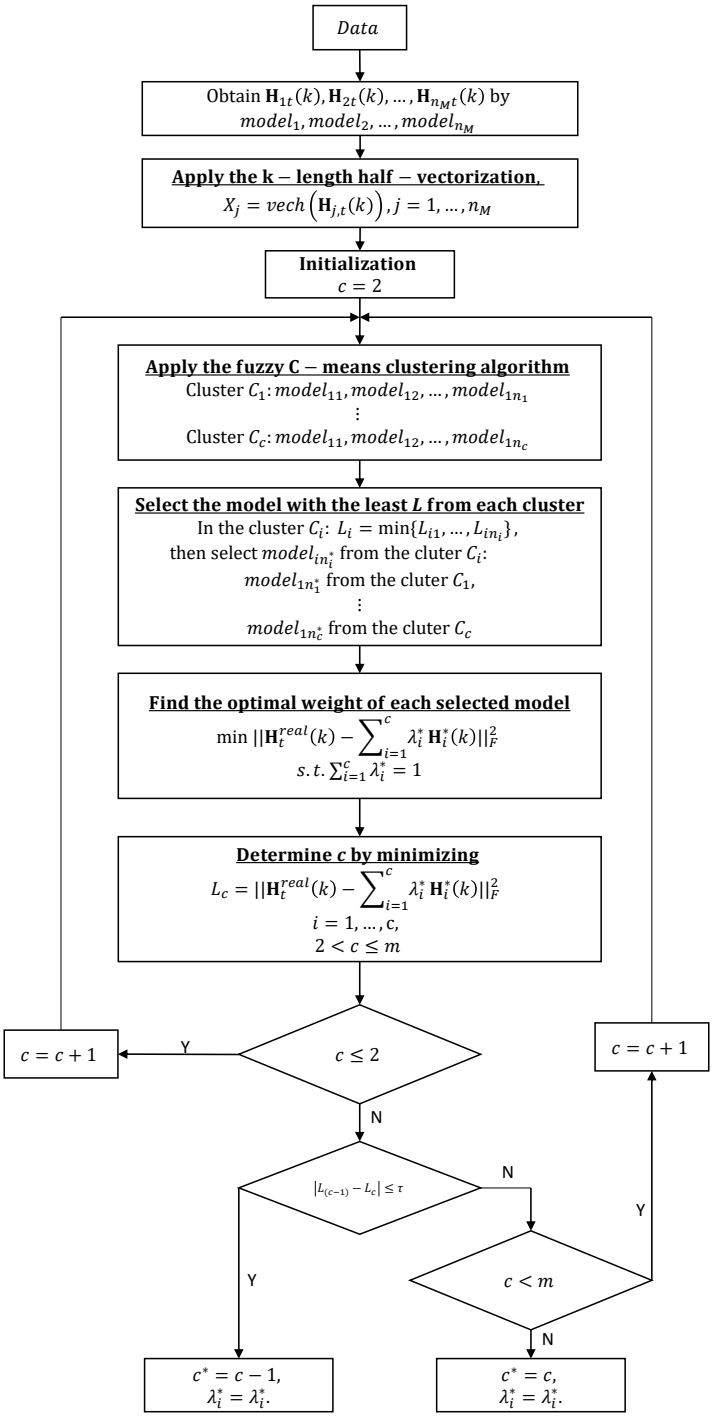


Figure 2.4: The mechanism of the training phase

matrix $\mathbf{H}_{j,t}$ of the $n(n+1)/2 \times 1$ columns $vech(\mathbf{H}_{j,t})$ only processes the lower triangular part of $\mathbf{H}_{j,t}$, where $vech(\mathbf{H}_{j,t})$ is given by:

$$\begin{aligned} vech(\mathbf{H}_{j,t}) = [&H_{j,t}(1,1), \ldots, \mathbf{H}_{j,t}(n,1), \mathbf{H}_{j,t}(2,2), ..., \\ &\mathbf{H}_{j,t}(n,2), ..., \mathbf{H}_{j,t}(n-1,n-1), \mathbf{H}_{j,t}(n,n-1), \mathbf{H}_{j,t}(n,n)], \ j = 1, \ldots, m \end{aligned} \tag{2.2}$$

Given the volatility matrices $\mathbf{H}_{j,t}(k)$ $(j = 1, \ldots, m)$, the k-length half-vectorization, $vech(\mathbf{H}_{j,t}(k))$, is defined as

$$vech(\mathbf{H}_{j,t}(k)) = [(vech(\mathbf{H}_{j,t-k+1})), \ldots, (vech(\mathbf{H}_{j,t}))], \ (j = 1, \ldots, m). \tag{2.3}$$

Let $\mathbf{X}_j = vech(\mathbf{H}_{j,t}(K)), j = 1, \ldots, m$. Define the number of clusters as c. The fuzzy C-means (FCM) clustering algorithm [10] is used to partition m training samples, $\mathbf{X}_j = vech(\mathbf{H}_{j,t}(k))$, into clusters C_1, C_2, ..., and C_c, where (5) computes the cluster center $\mathbf{V}_i$ of the cluster C_i and the membership grade u_{ij} of the training sample $\mathbf{X}_j$ belonging to C_i, with $1 \leq i \leq c$ and $1 \leq j \leq m$. We used the FCM clustering algorithm as this is widely used in pattern recognition [10]. In FCM, it denotes a fussiness index as s, $1 \leq s < \infty$. This fuzziness index s [5] is usually chosen to be 2. The FCM clustering algorithm merges m matrices $\mathbf{X}_j$) into clusters C_i $(i = 1, 2, \ldots, c; \ 2 \leq c \leq m)$ by solving the following minimization problem [5]:

$$J_s = \sum_{i=1}^{c} \sum_{j=1}^{m} (u_{ij})^s ||\mathbf{V}_i - \mathbf{X}_j||_F^2, \tag{2.4}$$

where $||\mathbf{V}_i - \mathbf{X}_j||_F$ is the Frobenius norm between matrix $\mathbf{X}_j$ and the cluster center $\mathbf{V}_i$ $(i = 1, 2, \ldots, c)$; u_{ij}^s is the membership grade of $\mathbf{X}_j$ belonging to cluster C_i, in which s is the weighting exponent controlling the relative weights placed on each of the $||\mathbf{V}_i - \mathbf{X}_j||_F^2$, $1 \leq s < \infty$. The FCM clustering algorithm is summarized as following steps:

1: Initialize $c = 2$.

2: Generate the random number as the membership grade $u_{ij}^{(0)}$ of the sample $\mathbf{X}_j$ to the cluster C_i, where $0 \leq u_{ij}^{(0)} \leq 1$, $\sum_{i=1}^{c} u_{ij}^{(0)} = 1$, $1 \leq i \leq c$ and $1 \leq j \leq m$. Set $k = 1$.

3: Compute the cluster center $\mathbf{V}_i$ of C_i based on

$$\mathbf{V}_i^{(k)} = \frac{\sum_{j=1}^{n} (u_{ij}^{(k-1)})^s \mathbf{X}_j}{\sum_{j=1}^{n} (u_{ij}^{(k-1)})^s}, \tag{2.5}$$

where $1 \leq i \leq c$.

4: Update u_{ij} based on:

$$u_{ij}^{(k)} = \frac{1}{\sum_{d=1}^{c} (\frac{||\mathbf{V}_i^{(k)} - \mathbf{X}_j||}{||\mathbf{V}_d^{(k)} - \mathbf{X}_j||})^{\frac{2}{s-1}}}, \tag{2.6}$$

where $1 \leq i \leq c$, $1 \leq j \leq m$, and $\sum_{i=1}^{c} u_{ij} = 1$.

5: Set $k = k + 1$. Repeat Step 3 and Step 4 until J_s is no longer decreasing [10].

6: Let $c = c + 1$. Go to Step 1 until $c \geq m + 1$.

Weight Assignment

The optimal weights of the selected models can be obtained by solving the following minimization problem:

$$\begin{aligned} &\min \|\mathbf{H}_t^{real}(k) - \sum_{i=1}^{c} \lambda_i^* \mathbf{H}_{i,t}^*(k)\|_F^2, \\ &s.\ t.\ \ \sum_{i=1}^{c} \lambda_i^* = 1, \end{aligned} \tag{2.7}$$

where $\mathbf{H}_{i,t}^*(k)$ is the best prediction generated by the optimal model from C_i. (2.7) can be reformulated as a quadratic form as

$$\begin{aligned} &\min \quad \boldsymbol{\lambda}^T \mathbf{A} \boldsymbol{\lambda} + \boldsymbol{\lambda}^T B + C, \\ &s.\ t. \quad \sum_{i=1}^{c} \lambda_i = 1, \end{aligned} \tag{2.8}$$

where

$$\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \ldots, \lambda_c)^T,$$

$$\mathbf{A} = \begin{pmatrix} \sum_{i=1}^{n} \sum_{j=1}^{nk} x_{ij}^{(1)} x_{ij}^{(1)} & \cdots & \sum_{i=1}^{n} \sum_{j=1}^{nk} x_{ij}^{(1)} x_{ij}^{(c)} \\ \vdots & \ddots & \vdots \\ \sum_{i=1}^{n} \sum_{j=1}^{nk} x_{ij}^{(c)} x_{ij}^{(1)} & \cdots & \sum_{i=1}^{n} \sum_{j=1}^{nk} x_{ij}^{(c)} x_{ij}^{(c)} \end{pmatrix},$$

$$B = -2((x_{ij}^{(1)} y_{ij}), (x_{ij}^{(2)} y_{ij}), \ldots, (x_{ij}^{(c)} y_{ij}))^T,$$

$$C = \sum_{i=1}^{n} \sum_{j=1}^{nk} y_{ij}^2 = \|\mathbf{H}_t^{real}(k)\|_F,$$

$x_{ij}^{(c)}$ is the entry in the i^{th} row and j^{th} column of $\mathbf{H}_{i,t}^{*}(k)$, and y_{ij} is the element in the i^{th} row and j^{th} column of $\mathbf{H}_t^{real}(k)$.

Based on the FMVM, the forecasted volatility matrix $\mathbf{H}_t^{fmvm}$ can be developed as

$$\begin{aligned} &\mathbf{H}_t^{fmvm} = \sum_{i=1}^{c} \lambda_i^* \mathbf{H}_{i,t}^{*}, \\ &i = 1, \ldots, c, \\ &2 \leq c \leq m. \end{aligned} \tag{2.9}$$

where n_M is the amount of the one-day-ahead volatility matrices determined by each individual multivariate volatility model; $\mathbf{H}_{i,t}^{*}$ is the one-day-ahead volatility matrix of the selected model i^*; and λ_i^* is the fuzzy optimal weight the selected model i^*.

Determination of the Number of Clusters

Given the predetermined number of clusters, c, the optimal number of clusters, namely c^*, can be determined as follows:

$$\min_{c} \quad L_c = \|\mathbf{H}_t^{fmvm}(k,c) - \mathbf{H}_t^{real}(k)\|_F. \tag{2.10}$$

2.4.2 Operation Phase

Given c^*, and the optimal weights of the i^*th selected model from the ith cluster which are determined in the training phase, the forecasted volatility matrix, namely $\mathbf{H}_t^{fmvm}$, can be determined as:

$$\mathbf{H}_t^{fmvm} = \sum_{i=1}^{c^*} \lambda_i^* \mathbf{H}_{i,t}^*, \tag{2.11}$$

where $\mathbf{H}_{i,t}^*$ is the one-day-ahead volatility matrix of the selected model i, and λ_i^* is the optimal weight for the ith cluster.

Assume that we have n financial assets and the rolling windows size of realized covariance matrix is k. For the i^{th} asset, $\varepsilon_{i,t} = r_{i,t} - \mu_{i,t}$, is the $n \times 1$ vector of residuals from the ordinary least square (OLS) regressions. Given a $k \times n$ residual matrix

$$\mathbf{E}_t^{real} = \begin{bmatrix} \varepsilon_{1,t-k+1} & \varepsilon_{2,t-k+1} & \cdots & \varepsilon_{n,t-k+1} \\ \varepsilon_{1,t-k+2} & \varepsilon_{2,t-k+2} & \cdots & \varepsilon_{n,t-k+2} \\ \vdots & \vdots & \ddots & \vdots \\ \varepsilon_{1,t} & \varepsilon_{2,t} & \cdots & \varepsilon_{n,t} \end{bmatrix}.$$

The realized covariance matrix with k time-series samples is defined as

$$\mathbf{H}_t^{real} = \mathbf{E}_t^{real^T} \mathbf{E}_t^{real} / k.$$

$\mathbf{H}_t^{real}$ can be elaborated as

$$\mathbf{H}_t^{real} = \begin{bmatrix} \frac{1}{k}\sum_{i=T_0}^{t} \varepsilon_{1,i}\varepsilon_{1,i} & \frac{1}{k}\sum_{i=T_0}^{t} \varepsilon_{1,i}\varepsilon_{2,i} & \cdots & \frac{1}{k}\sum_{i=T_0}^{t} \varepsilon_{1,i}\varepsilon_{n,i} \\ \frac{1}{k}\sum_{i=T_0}^{t} \varepsilon_{2,i}\varepsilon_{1,i} & \frac{1}{k}\sum_{i=T_0}^{t} \varepsilon_{2,i}\varepsilon_{2,i} & \cdots & \frac{1}{k}\sum_{i=T_0}^{t} \varepsilon_{2,i}\varepsilon_{n,i} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{k}\sum_{i=T_0}^{t} \varepsilon_{n,i}\varepsilon_{1,i} & \frac{1}{k}\sum_{i=T_0}^{t} \varepsilon_{n,i}\varepsilon_{2,i} & \cdots & \frac{1}{k}\sum_{i=T_0}^{t} \varepsilon_{n,i}\varepsilon_{n,i} \end{bmatrix}, \tag{2.12}$$

where $T_0 = t - k + 1$. $\mathbf{H}_t^{real}$ is the benchmark for each $\mathbf{H}_{i,t}$, $i = 1, \ldots, m$.

The proposed FMVM and the AMVM generate the tracking errors L_t^{fmvm} and L_t^{amvm} at t which are denoted respectively as

$$L_t^{fmvm} = \|\mathbf{H}_t^{real} - \mathbf{H}_t^{fmvm}\|_F,$$

$$L_t^{amvm} = \|\mathbf{H}_t^{real} - \mathbf{H}_t^{amvm}\|_F,$$

where $\|.\|_F$ is the Frobenius norm. The covariance matrices generated by the FMVM and AMVM using k time-series samples can be written respectively as

$$\mathbf{H}_t^{fmvm}(k) = [\mathbf{H}_{t-k+1}^{fmvm}, \ldots, \mathbf{H}_t^{fmvm}],$$

$$\mathbf{H}_t^{amvm}(k) = [\mathbf{H}_{t-k+1}^{amvm}, \ldots, \mathbf{H}_t^{amvm}],$$

and, the proposed FMVM and the AMVM generate the tracking errors L_t^{fmvm} and L_t^{amvm} in k-length time period at t which are denoted respectively as

$$L_t^{fmvm}(k) = \|\mathbf{H}_t^{real}(k) - \mathbf{H}_t^{fmvm}(k)\|_F,$$

$$L_t^{amvm}(k) = \|\mathbf{H}_t^{real}(k) - \mathbf{H}_t^{amvm}(k)\|_F.$$

Consequently, we denote Δ_t as the evaluation indicator of the FMVM which is relative to the AMVM at time index t, and $\Delta_t(k)$ is denoted as those with a k-length-time-period data. Based on the tracking errors of the FMVM and AMVM, Δ_t and $\Delta_t(k)$ can be elaborated as

$$\Delta_t = Lt^{fmvm} - L_t^{amvm},$$

$$\Delta_t(k) = L_t^{fmvm}(k) - L_t^{amvm}(k).$$

The algorithm of the operation phase is summarized as follows:

1: If $\Delta_t > 0$, the FMVM lags behind the AMVM with time t, else the FMVM leads the AMVM with time t.

2: If $\Delta_t(k) > 0$, the FMVM lags behind the AMVM with time t with a k-length-time-period data, else the FMVM leads the AMVM with time t. If $\Delta_t(k) = 0$ or $\Delta_t = 0$, the performance of the FMVM is equivalent to that of the AMVM.

3: Prepare the time series samples, based on a particular decision period, of n financial assets. Determine the types and number of the individual multivariate volatility models, m, and the rolling window size k;

4: Determine the parameters of multivariate volatility models based on the time series samples of the log-return of assets. Generate the $n \times n$ covariance matrix $\mathbf{H}_{i,t}$ with a k-length rolling window size at each time point of the index t. Generate the realized volatility based on another k-length period samples.

5: Generate a multivariate volatility model, $\mathbf{H}_{i,t}(k)$, by plugging $\mathbf{H}_{t-k+1}, \mathbf{H}_{t-k+2}, \ldots, \mathbf{H}_t$, in chronological order. Hence, the realized form $\mathbf{H}_t^{real}(k)$ can be developed.

6: For $i = 1, 2, \ldots, m$, the matrix A is recombined by the elements of those $\mathbf{H}_{i,t}(k)$ using the optimization algorithm; determine matrix B, based on the elements of $\mathbf{H}_{i,t}(k)$ and $\mathbf{H}_t^{real}(k)$; determine C based on the Frobenius norm of $\mathbf{H}_t^{real}(k)$; based on A, B and C, $\Lambda = (\lambda_1, \lambda_2, \ldots, \lambda_m)^T$ in Equation (2.8) can be determined by harnessing the quadratic programming.

7: Obtain the L_t^{fmvm} and L_t^{amvm} and make decision based on FMVM using Δ_t if this is necessary.

2.5 Empirical Results

In this section, we demonstrate with examples how the proposed FMVM overcomes the limitations of the average multivariate volatility model (AMVM), namely that: (1) the FMVM selects less models without sacrificing forecasting powers; (2) the FMVM can achieve more accurate forecasting results than those achieved by AMVM. Indeed, the proposed FMVM can achieve better forecasting accuracy with smaller computational efforts.

The proposed FMVM is first evaluated based on 15 highly weighted HSI constituent stocks. Table 2.1 shows the selected samples in both the low-dimensional cases and the high-dimensional cases of the volatility matrix forecasting, and 17 different widely-used multivariate volatility models. In the low-dimensional cases, top 4 weighted HSI constituent stocks are chosen and those stocks are '5.HK', '939.HK', '1299.HK', and '9413.HK'. In the high-dimensional cases, top 15 weighted HSI constituent stocks, including the 4 identical stocks in the low-dimensional cases, are selected. Another 11 stocks are '1398.HK', '3988.HK', '883.HK', '857.HK', '386.HK', '2628.HK', '3.HK', '151.HK', '762.HK', '494.HK', and '3328.HK'. Table 2.1 specifies 17 most widely used multivariate volatility models. The models from the same class with different p or q is distinct from other models in the same class. For example, $CCC(1,1)$ and $CCC(2,1)$ come from the same family; however, $CCC(1,1)$ is a distinctive model and $CCC(1,1)$ is a independent model from $CCC(2,1)$. These daily trading samples, from 29 October 2010 to 7 October 2014, are obtained from the Bloomberg database which is collected from 29 October 2010 to 7 October 2014. The forecasting results of the proposed FMVM are compared with those obtained by the 17 commonly used forecasting models which are listed in listed in Table 2.1. All the models are used to forecast the one-day-ahead volatility matrices based on different rolling window size with $k = 20, 30, \ldots, 120$. Also we compared the results

obtained by the proposed FMVM with the existing multivariate volatility approach, the AMVM [36].

The CCC, DCC, ADCC, BEEK, OGARCH, AMVM models and the proposed FMVM were implemented using matlab with toolboxes including the Oxford MFE Matlab toolbox [41]. The one-day-ahead volatility matrices for AMVM is given as:

Table 2.1: Selected stocks and models

The Low-Dim Case	The High-Dim Case	Multivariate Volatility Models
'5.HK'	'5.HK'	$\underline{CCC(p,q)}$
'939.HK'	'939.HK'	$\overline{CCC(1,1)}$, $CCC(1,2)$
'1299.HK'	'1299.HK'	$CCC(2,1)$, $CCC(2,2)$
'941.HK'	'941.HK'	
	'1398.HK'	$\underline{DCC(p,q)}$
	'3988.HK'	$\overline{DCC(1,1)}$, $DCC(1,2)$
	'883.HK'	$DCC(2,1)$, $DCC(2,2)$
	'857.HK'	
	'386.HK'	$\underline{ADCC(p,q)}$
	'2628.HK'	$\overline{ADCC(1,1)}$, $ADCC(1,2)$
	'3.HK'	$ADCC(2,1)$, $ADCC(2,2)$
	'151.HK'	
	'762.HK'	$\underline{OGARCH(p,q)}$
	'494.HK'	$\overline{OGARCH(1,1)}$, $OGARCH(1,2)$
	'3328.HK'	$OGARCH(2,1)$, $OGARCH(2,2)$
		$BEKK$
4 assets	15 assets	17 models

$$\mathbf{H}_t^{amvm} = \frac{1}{17}\sum_{i=1}^{17}\mathbf{H}_{i,t} \tag{2.13}$$

where $\mathbf{H}_{1,t}, \mathbf{H}_{2,t}, \ldots, \mathbf{H}_{17,t}$ are the conditional volatility matrix of each model listed in Table 2.1.

We evaluate the models with different rolling window sizes. The window size is varied from $k = 20$ to 120 in a step of 10. The date, 7 October 2014, is used as the end window. We used the last k trading days as the operation period and

the preceding k trading days as the training period. Therefore each model can be evaluated $2k$ times. For instance, in the training phase, the forecasted results are $\hat{Y}_t, \ldots, \hat{Y}_{t+k}$, where $\hat{Y}_t$ is a forecasted value at time t based on the information the time series samples of $t-k+1, \ldots, t-1$. Similarly, in the operation phase, the forecasted results are $\hat{Y}_{t+k+1}, \ldots, \hat{Y}_{t+2k}$, where $\hat{Y}_{t+k+1}$ is forecasted based on the time series samples of $t+1, \ldots, t+k$. In training phase, we try to use k forecasted results $\hat{Y}_t, \ldots, \hat{Y}_{t+k}$ to train the FMVM model (where the optimal number of clusters and the optimal weights of the selected models are predefined); in operation phase, we use the trained FMVM to forecast k forecasted results $\hat{Y}_{t+k+1}, \ldots, \hat{Y}_{t+2k}$. We need the operation phase to validate the performance of the proposed FMVM. Both low-dimensional and high-dimensional circumstances are considered.

2.5.1 The Low-Dimensional Case

Here, we consider a simple case with 4 stocks which are shown in the first column of Table 2.1. The tracking errors $\Delta_I(k)$ and the differences $\Delta_{II}(k)$ are shown in Table 2.2. $L_I^F(k)$ and $L_I^F(A)$ are the total tracking errors for the training phase obtained by the FMVM and the AMVM, respectively. $L_{II}^F(k)$ and $L_{II}^A(k)$ are the total tracking errors for the operation phase obtained by the FMVM and the AMVM, respectively.

Table 2.2: Tracing errors in the training and the operation phase (low-dim)

RWS k	clusters	$L_I^F(k)$	$L_I^A(k)$	$\Delta_I(k)$	$L_{II}^F(k)$	$L_{II}^A(k)$	$\Delta_{II}(k)$
20	3	3.1162	7.7491	-4.6329	4.8583	7.9454	-3.0870
30	3	7.5967	15.0727	-7.4761	4.4505	7.3815	-2.9310
40	5	5.0178	16.6071	-11.5893	6.6014	9.6715	-3.0701
50	4	7.3173	18.0725	-10.7552	6.7466	10.8176	-4.0710
60	7	8.9751	18.9535	-9.9784	6.8483	10.3731	-3.5247
70	4	8.0716	19.3060	-11.2343	8.0451	12.0133	-3.9682
80	6	7.0084	16.1564	-9.1481	7.9594	13.0397	-5.0804
90	6	7.1209	13.4263	-6.3055	10.2467	14.1335	-3.8868
100	4	9.9625	12.9350	-2.9725	10.5589	14.4315	-3.8726
110	7	9.5720	14.0492	-4.4772	10.1283	14.7852	-4.6569
120	7	8.7855	15.5307	-6.7453	9.5383	13.6743	-4.1360

This results for both training and operation phases show that the tracking errors of the FMVM are smaller than those of the AMVM as the differences are negative in all cases. Hence, the FMVM can surpass the AMVM not only in the training but also in the operation phase. Table 2.2 shows that the number of clusters with respect to different rolling windows. Table 2.2 shows that the FMVM required less computational effort, as three to seven models are only involved of the FMVM. As the number of clusters is identical to the number of the individual models, the FMVM does not need to involve with all 17 models. Therefore, this low-dimensional case study shows that the proposed FMVM can achieve better forecasting accuracy with smaller computational efforts comparing with the AMVM.

2.5.2 The High-Dimensional Case

In the high dimensional case, we consider 15 stocks from the Hang Seng Index, as shown in the second column of Table 2.1. Table 2.3 shows the tracking errors and the differences between the FMVM and the AMVM.

Table 2.3: Tracing errors in the training and the operation phase (high-dim)

RWS k	clusters	$L_I^F(k)$	$L_I^A(k)$	$\Delta_I(k)$	$L_{II}^F(k)$	$L_{II}^A(k)$	$\Delta_{II}(k)$
20	5	15.6529	49.2121	-33.5592	18.4214	33.4936	-15.0722
30	3	26.8251	50.9704	-24.1454	17.1039	33.4648	-16.3608
40	4	34.6771	85.3520	-50.6749	20.5884	40.1193	-19.5309
50	4	25.2633	97.2940	-72.0308	31.0473	48.5467	-17.4994
60	5	27.5873	94.5644	-66.9771	29.7626	51.8209	-22.0584
70	4	25.1795	93.0677	-67.8883	30.0937	57.6992	-27.6054
80	3	23.2929	87.4429	-64.1500	26.4884	58.3126	-31.8242
90	5	18.1365	84.6518	-66.5153	29.4302	64.1372	-34.7070
100	3	21.0107	82.8058	-61.7951	22.8918	68.6020	-45.7102
110	3	19.7757	78.7352	-58.9595	21.6531	69.0370	-47.3840
120	4	20.1628	76.0412	-55.8784	21.2291	70.0627	-48.8337

The result shows that the tracking errors of the FMVM are smaller than those of the AMVM. It also shows that the FMVM can surpass the AMVM not only in the training phase but also in the operation phase. Also smaller computational effort is

required by the FMVM as three to five models are involved among all the seventeen models. Therefore, the empirical results show that the FMVM can obtain better forecasting accuracy and the FMVM requires much less computational effort than that required by the AMVM.

2.6 Conclusion

In this thesis, a novel multivariate volatility model FMVM is proposed to improve forecasting accuracy. The proposed FMVM overcomes the disadvantages of incurring redundant computation when all multivariate volatility models are employed in an averaging manner. The proposed FMVM classifies individual volatility model into smaller scale clusters by using fuzzy C-means clustering algorithm and selects the most representative model with the lowest tracking error from each cluster. Optimal weight for each selected model can then be sought via training. The effectiveness of the proposed FMVM is evaluated based on 15 stocks from the Hang Seng Index. Empirical results have shown that the proposed FMVM can obtain better accuracy in volatility forecasting with less number of volatility models. Hence, less computational effort is required by the proposed FMVM than that required by the AMVM. For the future work,we will apply the proposed FMVM to other financial products such as the commodity market.

Chapter 3

The Hidden Markov Model Based Forecasting Algorithm

The primary purpose of this chapter is to present a novel volatility forecasting technique by using Hidden Markov Model. The opening, high, low, closing prices, trading volume, turnovers and similar patterns are also crucial factors in volatility matrix forecasting. However, current methods only discuss the relationship between the returns series and volatility matrix. Lacking information may lead to forecasting errors. The HMM-based algorithm can process more attributes than prices and returns. This algorithm can also define the similar pattern of current evidence in the historical data.

3.1 Hidden Markov Models

The Hidden Markov Model (HMM) is proposed as a stochastic signal model with a wide range of application, such as speech recognition, multiple sequence alignment, protein homolog recognition, signal processing and other pattern recognitions. [39, 16, 6]. The HMM is kind of extension of discrete Markov chains.

3.1.1 Discrete Markov Processes

Markov processes describe such a system described as follows. The system is in one of a set of N distinct states, $S_1, S_2, \ldots, S_N$. The system undergoes a change or a invariability unchanged of probabilities associated with the state at regularly spaced discrete times, $t = 1, 2, ..., t$. Denote the actual state at t as q_t. The probabilistic specification of current state and the predecessor states is required to a full probabilistic description of this system.

Given a first order Markov Chain, this probabilistic description of this system becomes the probabilistic specification of the current and the predecessor state, i.e.,

$$P(q_t = S_j | q_{t-1} = S_i, q_{t-2} = S_k, \ldots) = P(q_t = S_j | q_{t-1} = S_i). \tag{3.1}$$

Equation (3.1) is independent of time, thereby the transition probability a_{ij} of state switching from state i to state j is

$$a_{ij} = P(q_t = S_j | q_{t-1} = S_i),\ 1 \leq i, j \leq N, \tag{3.2}$$

where $a_{ij} \geq 0$ and $\sum_{j=1}^{N} a_{ij} = 1$. The above stochastic process is observable, because the output of the process is the set of states at each instant of time, where each state corresponds to a observation [39]. Given a observation sequence O as $O = q_1, q_2, ..., q_t$ and a model, the probability of O can be expressed and evaluated as

$$\begin{aligned} P(O|Model) &= P(q_1, q_2, ..., q_t | Model) \\ &= P(q_1) \cdot P(q_2|q_1) \cdot \ldots P(q_t|q_{t-1}) \\ &= \pi_{q_1} \cdot a_{q_1 q_2} \cdot \ldots a_{q_{t-1} q_t}. \end{aligned}$$

where $\pi_{q_1} = P(q_1 = q_1)$ is the initial state probabilities and q_1 is in of a set of $\{S_1, S_2, \ldots, S_N\}$. The initial state probabilities are

$$\pi_i = P(q_1 = S_i),\ 1 \leq i \leq N.$$

Given an example of a Markov process in Figure 3.1, the model for a stock market movement has three states, *Bull*, *Even*, and *Bear*, and three corresponding observations *up*, *unchanged*, and *down*. Given a sequence of observations,

$$O = \{up, up, down, up, unchanged, unchanged, down, down, unchanged\},$$

we can easily verify the state sequence produced these observations is

$$Q = \{Bull, Bull, Bear, Bull, Even, Even, Bear, Bear, Even\}.$$

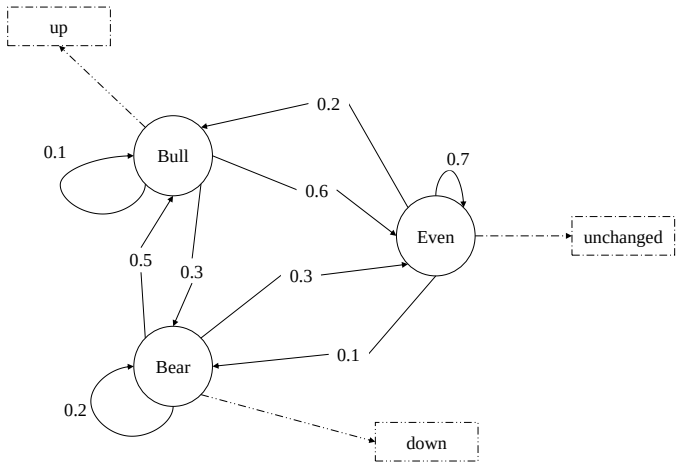


Figure 3.1: Markov process example

If we denote states *Bull*, *Even*, and *Bear* as s_1, s_2, s_3, the probability of this observation sequence is

$$\begin{aligned}
P(O|Model) &= P(s_1, s_1, s_3, s_1, s_2, s_2, s_3, s_3, s_2|Model) \\
&= P(s_1)P(s_1|s_1)P(s_3|s_1)P(s_1|s_3)P(s_2|s_1)P(s_2|s_2)P(s_3|s_2)P(s_3|s_3)P(s_2|s_3) \\
&= \pi_1 \cdot a_{11} \cdot a_{13} \cdot a_{31} \cdot a_{12} \cdot a_{22} \cdot a_{23} \cdot a_{33} \cdot a_{32} \\
&= 1 \cdot (0.1) \cdot (0.3) \cdot (0.5) \cdot (0.6) \cdot (0.7) \cdot (0.1) \cdot (0.2) \cdot (0.3) \\
&= 3.7800 \times 10^{-5}.
\end{aligned}$$

The above Markov model illustrates a simple intuition of financial market: index booms in the bull market, remain unfaltering in even market, and descend in the bear market. However, index trends directionally, i.e., up, down, or sideways no matter in bull, even, or bear market[33]. For instance, the index cannot only drop but also inflate or remains unchanged in the bear market. Therefore, a extension of the above model is required to describe our financial market. A extended model for a stock market movement has three states, *Bull*, *Even*, and *Even*, the model for a stock market movement in Figure 3.2 has three states, *Bull*, *Even*, and *Even*, whereas one of this three states can emit one observation from a set of three observations *up*, *unchanged*, and *down*. Considering we have the same observation sequence as that of the above Markov model,

$$O = \{up, up, down, up, unchanged, unchanged, down, down, unchanged\}.$$

we cannot identify exactly what state sequence brings out the above observation sequence. Therefore, the state sequence is 'hidden' [38, 6]. The extended model has this 'hidden' phenomenon called the HMM.

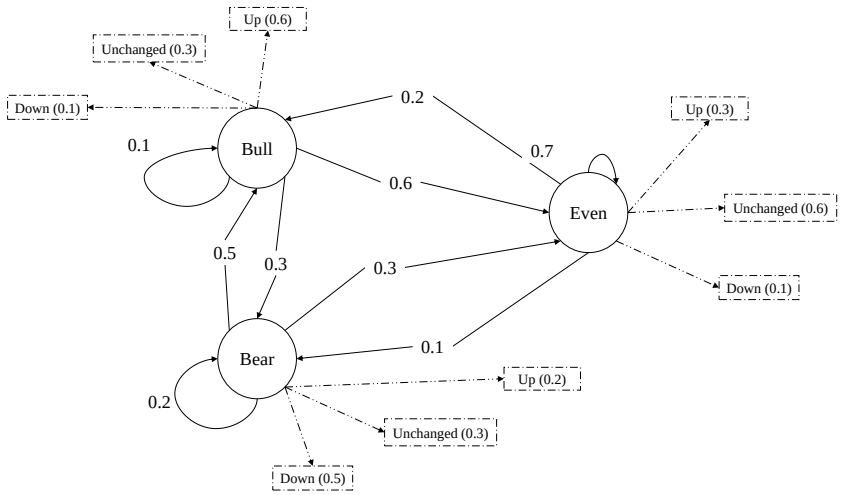


Figure 3.2: Hidden Markov model example

3.1.2 The HMM

The HMM requires two predetermined parameters at the initial. It denotes the number of states in the model as N and the number of distinct observation per state as M. A complete specification of an HMM requires specification of these two model parameters, N and M [39]. The definition of a HMM is as follows: The complete parameter set of the model is

$$\lambda = (\mathbf{A}, \mathbf{B}, \pi), \tag{3.3}$$

where $\mathbf{A}$ is the transition matrix that represents each state transition probability distribution, $\mathbf{B}$ is the emission matrix that represents each observation symbol probability distribution, and π is the prior probability array that presents each the initial state distribution. S is the state set and V is the observation set:

$$S = \{s_1, s_2, \ldots, s_N\}, \tag{3.4}$$

$$V = \{v_1, v_2, \ldots, v_M\}. \tag{3.5}$$

It denotes a fixed state sequence of length T as Q, and the corresponding observation sequence as O:

$$Q = (q_1, q_2, \ldots, q_T), \tag{3.6}$$

$$O = (o_1, o_2, \ldots, o_T). \tag{3.7}$$

The transition matrix $\mathbf{A}$, storing the probability of state j ensuing state i, is time-independent:

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \ldots & a_{1N} \\ a_{21} & a_{22} & \ldots & a_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ a_{N1} & a_{N2} & \ldots & a_{NN} \end{bmatrix}, \tag{3.8}$$

$$a_{ij} = P(q_t = s_j | q_{t-1} = s_i),\ 1 \leq i, j \leq N.$$

The emission matrix $\mathbf{B}$, storing the probability of observation k emitting from state j, is time-independent:

$$\mathbf{B} = \begin{bmatrix} b_1(1) & b_1(2) & \dots & b_1(M) \\ b_2(1) & b_2(2) & \dots & b_2(M) \\ \vdots & \vdots & \ddots & \vdots \\ b_N(1) & b_N(2) & \dots & b_N(M) \end{bmatrix}, \tag{3.9}$$

$$b_i(k) = P(x_t = v_k | q_t = s_i),\ 1 \leq i \leq N,\ 1 \leq k \leq M.$$

The prior probability array, storing each initial state distribution, is

$$\begin{aligned} \pi &= (\pi_1, \pi_2, \dots, \pi_N), \\ \pi_i &= P(q_1 = s_i),\ 1 \leq i \leq N. \end{aligned} \tag{3.10}$$

The HMM has two assumptions [38, 39, 16, 6]:

- The Markov process. The Markov assumption presumes that the current state is dependent of the previous state and is independent of the other processor states:

$$P(q_t | q_1, q_2, \dots, q_{t-1}) = P(q_t | q_{t-1}). \tag{3.11}$$

- The time-independence. The independence assumption presumes that the current observation at time t is dependent only on the current state and is independent of previous observation:

$$P(o_t | o_1, o_2, \dots, o_{t-1}, q_1, q_2, \dots, q_t) = P(o_t | q_t). \tag{3.12}$$

Given a HMM λ, and a sequence of observations O, the probability of the observations O for a specific state sequence Q is

$$P(O|Q, \lambda) = \prod_{t=1}^{T} P(o_t | q_t, \lambda) = b_{q_1}(o_1) \cdot b_{q_2}(o_2) \cdots b_{q_T}(o_T), \tag{3.13}$$

and the probability of the state sequence is

$$P(Q|\lambda) = \prod_{t=1}^{T} P(o_t|q_t, \lambda) = \pi_1 \cdot a_{q_1 q_2} \cdot a_{q_2 q_3} \cdots a_{q_{T-1} q_T}. \tag{3.14}$$

Therefore, the probability of the observations given the model is

$$\begin{aligned} P(O|\lambda) &= \sum_{Q} P(O|Q, \lambda) P(Q|\lambda) \\ &= \sum_{q_1 \cdots q_T} \pi_{q_1} b_{q_1}(o_1) a_{q_1 q_2} b_{q_2}(o_2) \cdots a_{q_{T-1} q_T} b_{q_T}(o_T). \end{aligned} \tag{3.15}$$

3.1.3 An Example

As an example, consider a Markov model with three states of volatility and five possible emissions. Three states of volatility are low, moderate, and high. These five potential emissions are 1, 2, 3, 4, and 5. Under the low volatility state, the probabilities of each emission are 0.80, 0.05, 0.05, 0.05, and 0.05 receptively. Under the moderate state, the probabilities of each emission are 0.10, 0.20, 0.35, 0.20, and 0.15 receptively. Under the high state, the probabilities of each emission are 0.05, 0.10, 0.15, 0.25, and 0.45 receptively.

The model generates a sequence of numbers from the set $\{1, 2, 3, 4, 5\}$ with the following rules:

- Beginning in low volatility state, the transition rules of this state are:
  - If the emission is 1, the state will remain low volatility state.
  - If the emission is 2, the state will transit to moderate volatility state.
  - If the emission is 3, 4, or 5, the state will transit to high volatility state.
- In moderate volatility state, the transition rules are:
  - If the emission is 1, the state will transit to low volatility state.

- If the emission is 3, 4, or 5, the state will remain moderate volatility state.
- If the emission is 2, the state will transit to high volatility state.

- In high volatility state, the transition rules are:
  - If the emission is 2, the state will transit to low volatility state.
  - If the emission is 1 or 4, the state will transit to moderate volatility state.
  - If the emission is 3 or 5, the state will remain high volatility state.

The state diagram for this Markov model has three states, low, moderate, and high, as shown in Figure 3.3. The transition matrix is

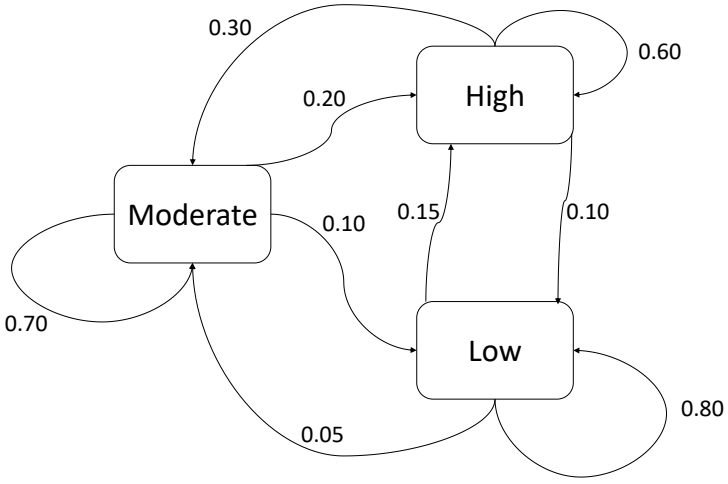


Figure 3.3: The state digram of three volatility states

$$\mathbf{A} = \begin{bmatrix} 0.80 & 0.05 & 0.15 \\ 0.10 & 0.70 & 0.20 \\ 0.10 & 0.30 & 0.60 \end{bmatrix}.$$

The emission matrix is

$$\mathbf{B} = \begin{bmatrix} 0.80 & 0.05 & 0.05 & 0.05 & 0.05 \\ 0.10 & 0.20 & 0.35 & 0.20 & 0.15 \\ 0.05 & 0.10 & 0.15 & 0.25 & 0.45 \end{bmatrix}.$$

We are aware of the rule of how these emissions generate from each state and also know how these states switch with the known emissions, thus this model is not hidden. Supposing we do not know this rule, what we can observe are sequences of emissions. If we notice more ones than any other number given the observation, we can suspect the model is under low volatility state. The probability of the outcome of 1 under the low volatility state is 0.8. However, we are not sure, because we neither know how the emission generates from each state nor know how state switches given the emission.

In a perspective of Hidden Markov model, the sequence of emissions is the only information. Therefore, one may consider following questions:

- How can we estimate the transition matrix and emission matrix?
- What is the most likely state path?
- What are the forward probability and the posterior probability?

Assume we knew the sequence of emissions, S, generates from a HMM with $\mathbf{A}$ and $\mathbf{B}$, we can estimate the transition matrix and emission matrix. The length of sequence is 1000. The last 100 observations are shown in Table 3.1. We use the Baum-Welch algorithm to estimate the transition matrix $\hat{\mathbf{A}}$ and emission matrix $\hat{\mathbf{B}}$ based on the sequence of emissions, S. In this example the estimated transition matrix $\hat{\mathbf{A}}$ is

$$\hat{\mathbf{A}} = \begin{bmatrix} 0.60 & 0.12 & 0.28 \\ 0.20 & 0.79 & 0.01 \\ 0.42 & 0.05 & 0.53 \end{bmatrix}.$$

The estimated emission matrix is

$$\hat{\mathbf{B}} = \begin{bmatrix} 0.00 & 0.01 & 0.13 & 0.34 & 0.52 \\ 0.81 & 0.09 & 0.04 & 0.04 & 0.02 \\ 0.17 & 0.31 & 0.45 & 0.06 & 0.00 \end{bmatrix}.$$

Given the estimated transmission matrix and the estimated emission matrix, the log

Table 3.1: The last 100 observations

Index	Obs.	Index	Obs.	Index	Obs.	Index	Obs.	Index	Obs.
901	3	921	5	941	1	961	3	981	4
902	4	922	3	942	3	962	5	982	3
903	3	923	3	943	4	963	4	983	1
904	3	924	5	944	2	964	5	984	2
905	1	925	1	945	4	965	2	985	1
906	2	926	5	946	4	966	3	986	5
907	1	927	3	947	4	967	1	987	3
908	5	928	1	948	2	968	4	988	1
909	5	929	4	949	2	969	3	989	1
910	2	930	2	950	3	970	5	990	3
911	5	931	2	951	3	971	5	991	3
912	3	932	1	952	3	972	5	992	5
913	4	933	1	953	5	973	4	993	5
914	4	934	1	954	5	974	1	994	2
915	5	935	1	955	5	975	3	995	5
916	2	936	4	956	2	976	3	996	1
917	3	937	4	957	5	977	3	997	4
918	3	938	5	958	3	978	4	998	5
919	5	939	2	959	4	979	2	999	1
920	5	940	2	960	1	980	4	1000	4

of the probability of a given sequence can be calculated in 3.2. We can find the same

Table 3.2: The last 100 observations: identification

Obs.	$log\ P(o_t\|\lambda)$	Obs.	$log\ P(o_t\|\lambda)$	Obs.	$log\ P(o_t\|\lambda)$	Obs.	$log\ P(o_t\|\lambda)$	Obs.	$log\ P(o_t\|\lambda)$
3	-2.6247	5	-2.3949	1	-0.4097	3	-2.6247	4	-2.3211
4	-2.3211	3	-2.6247	3	-2.6247	5	-2.3949	3	-2.6247
3	-2.6247	3	-2.6247	4	-2.3211	4	-2.3211	1	-0.4097
3	-2.6247	5	-2.3949	2	-2.5992	5	-2.3949	2	-2.5992
1	-0.4097	1	-0.4097	4	-2.3211	2	-2.5992	1	-0.4097
2	-2.5992	5	-2.3949	4	-2.3211	3	-2.6247	5	-2.3949
1	-0.4097	3	-2.6247	4	-2.3211	1	-0.4097	3	-2.6247
5	-2.3949	1	-0.4097	2	-2.5992	4	-2.3211	1	-0.4097
5	-2.3949	4	-2.3211	2	-2.5992	3	-2.6247	1	-0.4097
2	-2.5992	2	-2.5992	3	-2.6247	5	-2.3949	3	-2.6247
5	-2.3949	2	-2.5992	3	-2.6247	5	-2.3949	3	-2.6247
3	-2.6247	1	-0.4097	3	-2.6247	5	-2.3949	5	-2.3949
4	-2.3211	1	-0.4097	5	-2.3949	4	-2.3211	5	-2.3949
4	-2.3211	1	-0.4097	5	-2.3949	1	-0.4097	2	-2.5992
5	-2.3949	1	-0.4097	5	-2.3949	3	-2.6247	5	-2.3949
2	-2.5992	4	-2.3211	2	-2.5992	3	-2.6247	1	-0.4097
3	-2.6247	4	-2.3211	5	-2.3949	3	-2.6247	4	-2.3211
3	-2.6247	5	-2.3949	3	-2.6247	4	-2.3211	5	-2.3949
5	-2.3949	2	-2.5992	4	-2.3211	2	-2.5992	1	-0.4097
5	-2.3949	2	-2.5992	1	-0.4097	4	-2.3211	4	-2.3211

sequence with the same $log\ P(o_t|\lambda)$. In Table 3.3, the same probability indicates to the same observation sequence. What if the observe sequence is with length more than 2 observations, how can we find the similar sequence? Or if we have a unique sequence, how can we find the similar sequence of this unique sequence. Then we

Table 3.3: The last 100 observations: the same sequence with the same probability

Obs.	$log\ P(1_t\|\lambda)$	Obs.	$log\ P(2_t\|\lambda)$	Obs.	$log\ P(3_t\|\lambda)$	Obs.	$log\ P(4_t\|\lambda)$	Obs.	$logP(5_t\|\lambda)$
1	-0.4097	2	-2.5992	3	-2.6247	4	-2.3211	5	-2.3949

can use the most similar probability to find the similar observations sequence. This can be illustrated in the following example.

Assume that we had several unique observation sequences:

$$O_1 = \{1, 2, 1, 3, 1, 2, 1, 1, 1\},$$

$$O_2 = \{1, 3, 3, 3, 1, 2, 2, 2, 1\},$$

$$O_3 = \{2, 1, 3, 3, 1, 2, 3, 3, 1\},$$

$$O_4 = \{3, 1, 2, 1, 2, 3, 3, 1, 2\},$$

$$O_5 = \{3, 3, 3, 2, 1, 2, 3, 2, 2\},$$

$$O_6 = \{3, 3, 3, 3, 1, 2, 3, 2, 3\},$$

$$O_7 = \{1, 2, 1, 3, 2, 2, 1, 1, 1\}.$$

How can we find the most similar pattern of O_7 from O_1 to O_6? If the O_7 is the same observations sequence as the O_1, then we have:

$$O_1 = \{1, 2, 1, 3, 1, 2, 1, 1, 1\},$$

$$O_2 = \{1, 3, 3, 3, 1, 2, 2, 2, 1\},$$

$$O_3 = \{2, 1, 3, 3, 1, 2, 3, 3, 1\},$$

$$O_4 = \{3, 1, 2, 1, 2, 3, 3, 1, 2\},$$

$$O_5 = \{3, 3, 3, 2, 1, 2, 3, 2, 2\},$$

$$O_6 = \{3, 3, 3, 3, 1, 2, 3, 2, 3\},$$

$$O_7 = \{1, 2, 1, 3, 1, 2, 1, 2, 1\}.$$

It is easy to figure out $O_1 = O_7$. Although using the HMM model to find the similar pattern is a repetitive work in this case, we can verify whether HMM can do well in

a n-length observations sequences ($n \geq 2$). In this case, the estimated transmission matrix is

$$\hat{\mathbf{A}} = \begin{bmatrix} 0.3128 & 0.3470 & 0.3402 \\ 0.2289 & 0.6414 & 0.1297 \\ 0.5271 & 0.2180 & 0.2548 \end{bmatrix},$$

and the estimated emission matrix is

$$\hat{\mathbf{B}} = \begin{bmatrix} 0.2361 & 0.2331 & 0.5308 \\ 0.3328 & 0.1977 & 0.4695 \\ 0.4761 & 0.4666 & 0.0573 \end{bmatrix}.$$

Given the estimated HMM, we can calculate the probability of observations for each sequence as shown in Table 3.4. In Table 3.4, $log\ P(O_7|\lambda)$ equals to $log\ P(O_1|\lambda)$.

Table 3.4: The probability of the occurrence of a given observation sequence

Obs.	$log\ P(O\|\lambda)$
O_1	-10.0844
O_2	-9.9382
O_3	-9.5651
O_4	-10.1584
O_5	-10.2035
O_6	-9.4157
O_7	-10.0844

Thus, we can presume that O_7 and O_1 were the same observation sequence. In fact, O_7 and O_1 are the same one. Now, we go back to the unique observation sequences, where

$$O_1 = \{1, 2, 1, 3, 1, 2, 1, 1, 1\},$$

$$\vdots$$

$$O_7 = \{1, 2, 1, 3, 2, 2, 1, 1, 1\}.$$

In this case, we can hardly tell which pattern from O_1 to O_6 is similar to the pattern O_7. However, we can estimate a new HMM and calculate the probabilities of their occurrences. In this case of unique observations sequence, the estimated transmission

matrix is

$$\hat{\mathbf{A}} = \begin{bmatrix} 0.1115 & 0.7679 & 0.1206 \\ 0.2489 & 0.1298 & 0.6213 \\ 0.2163 & 0.4651 & 0.3187 \end{bmatrix},$$

and the estimated emission matrix is

$$\hat{\mathbf{B}} = \begin{bmatrix} 0.4171 & 0.4770 & 0.1059 \\ 0.6622 & 0.3095 & 0.0283 \\ 0.3851 & 0.2978 & 0.3171 \end{bmatrix}.$$

Given the estimated model, we can obtain the probabilities of the occurrences of these unique observations sequences as shown in Table 3.5 For these unique observations

Table 3.5: The probability of the occurrence of a given observations sequence

Obs.	$log\ P(O\|\lambda)$	$\|log\ P(O_7\|\lambda) - log\ P(O_i\|\lambda)\|$
O_1	-7.7154	0.4592
O_2	-10.6649	2.4902
O_3	-12.0390	3.8643
O_4	-11.7904	3.6158
O_5	-13.4506	5.2760
O_6	-15.1084	6.9338
O_7	-8.1747	0.0000

sequences, their probabilities of occurrence give a HMM are distinct from each other. However, we can use

$$\min_i |log\ P(O_7|\lambda) - log\ P(O_i|\lambda)| \tag{3.16}$$

to find the most similar pattern of O_7. In Table 3.5, O_1 is the solution for Equation (3.16). Therefore, O_1 is the most similar observations sequences of O_7.

3.2 Clustering Multivariate Data Using a Single HMM

In this section, we investigate a new method to identify similar data and group them into clusters using a single HMM. The idea of it is the identification of similar data

that can be used to forecast time series [27]. However, this kind of method may not be reasonable to forecast time series. We will evaluate its efficiency and look for some other ways to refine this methodology.

Clustering is a process in which, given a set of unlabeled dataset, the data items are grouped into different clusters based on some measure of similarity. Here, the log-likelihood values for each dataset items are used to define the measure of similarity.

In Hassan's model, the single HMM is used to define the measurement of similarity of the dataset [27]. Then, based on the similarity, the data items are labeled and grouped into different clusters. The functions of HMM are as follows:

- the log-likelihood values represent how well the data fits the trained HMM;
- the log-likelihood values represent how different data items are similar;
- based on different log-likelihood values, the dataset can be grouped into different clusters.

3.2.1 Generating Likelihood Values Using HMM

Initially, a single HMM is built by defining the similarity of data items via log-likelihood values. As a rule of the thumb, the number of states can be chosen as the same number of variables in each of the data items. Thus, k-state HMM is chosen, because there are k instances in each data vector. Assume that the input data is D.

Before training an HMM, we have to define the initial parameter values of A, B and π. For the parameter values of A, initialize the transition provabilities a_{ij} such that

$$\sum_i a_{ij} = 1$$

for all j by using randomly generated numbers. Similarly, the π is initialized by

using randomly generated numbers such that

$$\sum_{i}^{N} \pi_i = 1.$$

To obtain initial values for the observation emission probability matrix B, the use of the segmental k-means algorithm has been considered. The segmental k-means algorithm is described as

- choose a random N group of data vectors among m data vectors;
- assign each of N groups of data to one of the states in HMM (the number of groups in the HMM is set to be equal to the number of states).

For group i (or state i), the mean vector is

$$\hat{u}_i = \frac{1}{k_i} \sum_{D_t \in i} O_t,$$

where k_i is the number of data vectors in group i. Meanwhile, the covariance value is

$$\hat{V}_i = \frac{1}{k_i} \sum_{D_t \in i} (D_t - \hat{u}_i)^T (D_t - \hat{u}_i),$$

where O_t is continuous observation ($D_t \in D$). The initial observation emission probability distributions for each of the training data vectors for each state is

$$\hat{b}_i(D_t) = \frac{1}{(2\pi)^{k/2} |\hat{V}|^{1/2}} exp[-\frac{1}{2}(D_t - \hat{u}_i)\hat{V}_i^{-1}(D_t - \hat{u}_i)^T],$$

where $1 \leq i \leq k$. Previous steps are used to form the initialization part. After initialization, we need to re-estimate the initially chosen parameter values of the HMM parameters. The re-estimation method here is the Baum-Welch algorithm, which is described as follows. To describe the whole Baum-Welch algorithm, the

forward-backward algorithm is used to calculate the probability of an observation sequence with relatively less computation burden. The forward-backward algorithm is descried below

- The forward variable $\alpha_t(i)$ is defined as

$$\alpha_t(i) = Pr(O_1, O_2, \ldots, O_t, S_{t=i}|\lambda),$$

  where $\alpha_t(i)$ indicates the probability of the partial observation at time t and the state i is reached at the same time t, given a model λ.

- Initialize

$$\alpha_1(i) = \pi_i b_i(O_1)$$

  for all states i, where $i \in 1, \ldots, N$.

- Obtain the values of $\alpha_t(j)$ for each time unit t, $t = 2, \ldots, T$, and all state j, $j = 1, \ldots, N$, as

$$\alpha_t(j) = [\sum_i \alpha_{t-1}(i)a_{ij}]b_j(O_t),$$

  where $i = 1, \ldots, N$.

- obtain the resultant probability of the observation sequence for the given model λ by the following equation

$$Pr(O|\lambda) = \sum_i \alpha_T(i).$$

- The backward algorithm computes the probability of the observation sequence for the given λ, in a similar way to that of the forward algorithm. In this case, a backward variable $\beta_t(i)$ is introduced as

$$\beta_t(i) = Pr(O_{t+1}, O_{t+2}, \ldots, O_T|S_{t=i}, \lambda),$$

which provides the probability of the partial observation sequence at time $t+1$ to the time T, given the stat i at time t and the model λ. As the forward variable and the final probability is obtained by using the following equation

$$Pr(O|\lambda) = \sum_i \pi_i b_i(O_1)\beta_1(i),$$

where $i = 1, \ldots, N$.

Either of the forward or backward algorithms can be used to calculate $Pr(O|\lambda)$ for evaluation or re-estimation purpose.

Given the model λ, the forward-backward algorithm can be used to calculate the probability of generating the observation sequence. Assume we knew the model λ, the parameter values, the probabilities produced by the model parameters can be evaluate given the observation sequence. Given the current probabilities and the probabilities for the obtained observation sequence, we can further estimate more proper parameter for the model.

Initialize an model, λ, by choosing some non-zero parameter values. The a posterior probability of transitions γ_{ij}, form state i to state j is calculated using

$$\begin{aligned}
\gamma_t(i,j) &= Pr(S_t = i, S_{t+1} = j|O, \lambda) \\
&= \frac{\alpha_t(i)a_{ij}b_j(O_{t+1})\beta_{t+1}(j)}{Pr(O|\lambda)} \\
&= \frac{\alpha_t(i)a_{ij}b_j(O_{t+1})\beta_{t+1}(j)}{\sum_i \alpha_T(i)}.
\end{aligned}$$

The value of the variable $\gamma_t(i)$, represent the a posterior probability of state i at time t for the observation sequence and the model λ, is calculated using

$$\begin{aligned}
\gamma_t(i) &= Pr(S_t = i|O, \lambda) \\
&= \frac{\alpha_t(i)\beta_t(i)}{\sum_i \alpha_T(i)}.
\end{aligned}$$

The Baum-Welch algorithm is as follows

- The new calculated values of $\gamma_t(i)$, at time $t = 1$, are usually assumed to be the new estimate $\bar{\pi}$ of the initial state probabilities π_i,

$$\bar{\pi}_i = \gamma_t(i),$$

  where $t = 1$.

- The new estimate of the probability of the transition form state i to state j is

$$\bar{b}_j(k) = \frac{\sum_{t \in O_t = v_k} \gamma_t(j)}{\sum_{t=1}^{T} \gamma_t(j)}.$$

- The new estimate of the value of $\bar{a}_i j$ is as

$$\bar{a}_{ij} = \frac{\sum_{t=1}^{T-1} \gamma_t(i,j)}{\sum_{t=1}^{T-1} \sum_j \gamma_t(i,j)}$$

$$= \frac{\sum_{t=1}^{T-1} \gamma_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)}.$$

- Repeat previous steps until

$$Pr(O|\bar{\lambda}) \leq Pr(O|\lambda)$$

  is satisfied.

To handle more than one observation sequence, n, the re-estimation equations become

$$\bar{a}_{ij} = \frac{\sum_n \sum_{t-1}^{T_n - 1} \gamma_t^n(i,j)}{\sum_n \sum_{t=1}^{T_n - 1 \sum_j \gamma_t^n(i,j)}},$$

$$\bar{b}_k = \frac{\sum_n \sum_{t \in O_t^n = v_k} \gamma_t^n(j)}{\sum_n \sum_{t=1}^{T_n} \gamma_t^n(j)}.$$

The re-estimated parameter values determine the final HMM. The likelihood function is represented by $Pr(D|\lambda)$, where D represents the continuous observation sequence and λ is the final HMM. The next step is to calculate the likelihood values.

After training HMM, use the forward algorithm to calculate the likelihood values for each of the data vectors x_i, $i = 1, \ldots, n$.

For the m k-dimensional data vectors, we can obtain m distinct likelihood values.

The likelihood or logarithm of likelihood values, known as log-likelihood values, may be interpreted as follows.

- One data item consists of k-dimensional data $D_h = (x_{1h}, x_{2h}, \ldots, x_{kh})$,
- Another data item consists of k-dimensional data $D_j = (x_{1j}, x_{2h}, \ldots, x_{kj})$;
- If log-likelihood values l_h and l_j corresponding to the data items D_h and D_j are close to each other, then we may infer that the data items D_h and D_j should belong to the same cluster.

3.3 HMM-Based Financial Time Series Forecasting

Technical analysis believes that the historical data will interpret the future performance; technical analysis needs the statistics and machine learning tools to recognize distinct patterns. Thus the data patterns can be used to train the HMM and later use the trained HMM to calculate the log-likelihood values of the new pattern; then, to find the date that shares the closest value with the log-likelihood values of the present pattern and use the change between this data point and the following data point to do forecasting of the present data point.

The steps for forecasting using HMM can be described as

- Build, train HMM, and obtain log-likelihood values for patterns;

- Locate similar data patterns to the current ones;
- Obtain forecast value using the change of price of matched pattern and neighboring pattern.

It should be noted that the observation sequence consists of daily trading volume and stock prices open, high, low and close: i.e., each data pattern in the dataset comprises of the daily training volume,open, high, low and closing stock prices. These five features are used as predictors for the next days stock price.

3.3.1 Methodology of HMM Forecasting

Assume that we had T_1 data items in the initial training phase, T_2 data items totally, the rolling windows size is s, the number of states and the number of clusters of K-means algorithm are set to be equal to the number of variables. The HMM forecasting can be illustrated in details

- In the training phase, use K-means clustering algorithm to label data items. For instance, data items would be labeled into any one from $1, \ldots, G_1$ by their similarity in this case, where L_1, the number of clusters, is predetermined.
- Then, transform all data items in the training phase into integer among $1, \ldots, G_1$. After this transformation, all data items, represented by a vector for each one, can be represented by a integer instead. For instance, k-dimension vectors $D_1, D_2, \ldots, D_{T_1}$ can be represent by $I_1, I_2, \ldots, I_{T_1}$, where $I_i \in 1, \ldots, G_1$, for any i. Assume there are L_2 attributes of D_t. For simplification, set the number of cluster equal to the number of attributes, $G1 = G2$. For any t, $D_t = [attribute_t^1, attribute_t^2, attribute_t^3, \ldots, attribute_t^{L_2}]$.
- Treat these "integers" time series, formed by I_i, as the recognition pattern. However, the rolling window size, w, is considered to demarcate the length of

each pattern. The recognition pattern can be shown as can be shown as

1: is $[I_1, I_2, \ldots, I_w]$,

2: is $[I_2, I_3, \ldots, I_{w+1}]$,

3: is $[I_3, I_4, \ldots, I_{w+2}]$,

...

i: is $[I_i, I_{i+1}, \ldots, I_{i+w-1}]$,

...

For example, when the rolling window is predetermined as 20, the first recognition pattern is $I_1, I_2, \ldots, I_{20}$. The first three pattern can be shown as

1: is $[I_1, I_2, \ldots, I_{20}]$,

2: is $[I_2, I_3, \ldots, I_{21}]$,

3: is $[I_3, I_4, \ldots, I_{22}]$.

- Re-frame the T_1 k-dimension vectors $D_1, D_2, \ldots, D_{T_1}$ to $(T_1 - w)$ recognition pattern series k-dimension vectors $X_w, X_{w+1}, \ldots, X_{T_1}$ where $X_k = I_{k-w+1}, I_{k-w+2}, \ldots, I_k$, where $w \leq k \leq T_1$. For example,

X_1 is $[I_{1-w+1}, I_{1-w+2}, \ldots, I_1]$,

X_2 is $[I_{2-w+1}, I_{2-w+2}, \ldots, I_2]$,

X_3 is $[I_{3-w+1}, I_{3-w+2}, \ldots, I_3]$,

...

X_t is $[I_{t-w+1}, I_{t-w+2}, \ldots, I_t]$,

...

X_{T_1} is $[I_{T_1-w+1}, I_{T_1-w+2}, \ldots, I_{T_1}]$.

- Treat X_1, X_2, ..., X_{T_1} as a new variable in the training phase. If one wants to forecast any information of

$$D_{T_1+1} = [attribute^1_{T_1+1}, attribute^2_{T_1+1}, \ldots, attribute^{G_2}_{T_1+1}],$$

for instance, forecasting $\hat{attribute}^1_{T_1+1}$. As the technique analysis believes that history tends to repeat itself, our algorithm tends to find the similarity between recent recognition pattern and the previous recognition patterns. Therefore, we need to find the most similar to X_{T_1+1} pattern from X_1, X_2, ..., X_{T_1}, where X_t represent the recognition pattern with rolling window size rws at time index t. Then, HMM classification method can be applied to find the most similar pattern in this circumstance.

- Train HMM by inputing X_1, X_2, ..., X_{T_1} and calculate log-likelihood values L_m of m^th data pattern from X_1, X_2, ..., X_{T_1}: I_1, I_2, ..., I_{T_1}.

- Search the most nearest value of I_{T_1} from I_1, I_2, ..., I_{T_1}. Assume that the index most similar log-likelihood value is m^*, $1 \leq m^* \leq T_1$. Then, X_{m^*} is thought to be the most similar recognition pattern of X_{T_1+1}.

- Assume that the process from pattern X_{T_1} to $X_{T_1}+1$ will be same as the process from X_{m^*} to X_{m^*+1}. The forecasting of one attribute, i.e. $\hat{attribute}^1_{T_1+1}$, of X_{T_1+1} can be describe as

$$\hat{attribute}^1_{T_1+1} = \frac{attribute^1_{m^*+1} - attribute^1_{m^*}}{attribute^1_{m^*}} attribute^1_{T_1}.$$

For forecasting any other attributes, the forecasting can be executed using the same formula

$$\hat{attribute}_{T_1+1} = \frac{attribute_{m^*+1} - attribute_{m^*}}{attribute_{m^*}} attribute_{T_1},$$

where $attribute_t$ can be $open_t$, $high_t$, low_t, $close_t$, or $volumn_t$.

3.3.2 Methodology of HMM Multivariate Volatility Forecasting

We assume that n financial assets are considered and that k is the rolling window size. The conditional volatility matrix at time index t is

$$\mathbf{H}_t = \begin{bmatrix} \sigma_{11,t} & \sigma_{12,t} & \cdots & \sigma_{1n,t} \\ \sigma_{21,t} & \sigma_{22,t} & \cdots & \sigma_{2n,t} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{n1,t} & \sigma_{n2,t} & \cdots & \sigma_{nn,t} \end{bmatrix},$$

where $\sigma_{ii,t} = \sigma_{it}^2$ and $\sigma_{ij,t} = \sigma_{ji,t}$ if $i \neq j$. The number of attributes of volatility matrix is $(n+1)n/2$. In terms of the attributes of the volatility matrix, the volatility matrix can be represented as

$$\mathbf{H}_t = \begin{bmatrix} a_{1,t} & a_{2,t} & a_{3,t} & \cdots & a_{n,t} \\ a_{2,t} & a_{n+1,t} & a_{n+2,t} & \cdots & \vdots \\ a_{3,t} & a_{n+2,t} & a_{2n,t} & \cdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & a_{n(n+1)/2-1,t} \\ a_{n,t} & a_{2n-1,t} & \cdots & a_{n(n+1)/2-1,t} & a_{n(n+1)/2,t} \end{bmatrix},$$

where $a_{1,t}$, $a_{2,t}$, $\ldots$, $a_{n(n+1)/2,t}$ is $((n+1)n/2)$ attributes which represents variances or covariances of respective assets in volatility matrix. Those attributes can give all information of volatility matrix, thus attributes vector can be introduced to represent volatility matrix in another way as

$$\mathbf{V}_t = vech(\mathbf{H}_t).$$

Assume the number of historical data vectors for forecasting is triple the number of rolling window size, $3k$, and current decision time index is t, then information with $3k-$length day need to be considered and matrix $\mathbf{VA}$ is introduce to save all

information. Information matrix can be described as

$$\mathbf{VA}_t = \begin{bmatrix} \mathbf{V}_{t-3k+1} \\ \mathbf{V}_{t-3k-2} \\ \mathbf{V}_{t-3k-3} \\ \vdots \\ \mathbf{V}_t \end{bmatrix} = \begin{bmatrix} a_{1,t-3k+1} & a_{2,t-3k+1} & \cdots & a_{n(n+1)/2,t-3k+1} \\ a_{1,t-3k+2} & a_{2,t-3k+2} & \cdots & a_{n(n+1)/2,t-3k+2} \\ a_{1,t-3k+3} & a_{2,t-3k+3} & \cdots & a_{n(n+1)/2,t-3k+3} \\ \vdots & \vdots & \ddots & \vdots \\ a_{1,t} & a_{2,t} & \cdots & a_{n(n+1)/2,t} \end{bmatrix}.$$

Group each row of $\mathbf{VA}_t$ into L clusters by K-means clustering algorithm and each row will be labeled with respective index from $1, 2, \ldots, L$, i.e., row i will be labeled with index I_i ($I_i \in 1, 2, \ldots, L$). Labeled information matrix can be described as

$$[\mathbf{VA}_t|\mathbf{I}_t] = \begin{bmatrix} a_{1,t-3k+1} & a_{2,t-3k+1} & \cdots & a_{n(n+1)/2,t-3k+1} & I_{t-3k+1} \\ a_{1,t-3k+2} & a_{2,t-3k+2} & \cdots & a_{n(n+1)/2,t-3k+2} & I_{t-3k+2} \\ a_{1,t-3k+3} & a_{2,t-3k+3} & \cdots & a_{n(n+1)/2,t-3k+3} & I_{t-3k+3} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{1,t} & a_{2,t} & \cdots & a_{n(n+1)/2,t} & I_t \end{bmatrix}.$$

Given rolling window size k, the transformed recognition pattern matrix is

$$\mathbf{T}_t = \begin{bmatrix} \mathbf{X}_{t-2k} \\ \mathbf{X}_{t-2k+1} \\ \mathbf{X}_{t-2k+2} \\ \vdots \\ \mathbf{X}_t \end{bmatrix} = \begin{bmatrix} I_{t-3k+1} & I_{t-3k+2} & \cdots & I_{t-2k} \\ I_{t-3k+2} & I_{t-3k+3} & \cdots & I_{t-2k+1} \\ I_{t-3k+3} & I_{t-3k+4} & \cdots & I_{t-2k+2} \\ \vdots & \vdots & \ddots & \vdots \\ I_{t-k+1} & I_{t-k+2} & \cdots & I_t \end{bmatrix},$$

where $\mathbf{X}_i$, $i = t-2k, t-2k+1, dots, t$, is the transformed recognition pattern at time index i with k transformed information of k sequential time points. Use $\mathbf{T}_t$ to train HMM model and decode the likelihood value ll_i, similarity measure, of each transformed recognition pattern $\mathbf{X}_i$. Then the transformed recognition pattern matrix with similarity measure is

$$[\mathbf{T}_t|\mathbf{L}_t] = \begin{bmatrix} I_{t-3k+1} & I_{t-3k+2} & \cdots & I_{t-2k} & ll_{t-2k} \\ I_{t-3k+2} & I_{t-3k+3} & \cdots & I_{t-2k+1} & ll_{t-2k+1} \\ I_{t-3k+3} & I_{t-3k+4} & \cdots & I_{t-2k+2} & ll_{t-2k+2} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ I_{t-k+1} & I_{t-k+2} & \cdots & I_t & ll_t \end{bmatrix}.$$

Find the most similar pattern by solving following problem

$$\min_{i \in \mathbf{R}_t} ||ll_t - ll_i||_F^2,$$

$$\mathbf{R}_t = \{t-2k, t-2k+1, \ldots, t-1\}.$$

Assume that i^* is the solution of this problem, which means the recognition pattern at time index t is very similar to the recognition pattern at time index i^*, thus the one-day-ahead attributes vector is

$$\hat{\mathbf{V}}_{t+1} = [\frac{a_{1,i^*+1}}{a_{1,i^*}} a_{1,t}, \frac{a_{2,i^*+1}}{a_{2,i^*}} a_{2,t}, \ldots, \frac{a_{n(n+1)/2,i^*+1}}{a_{n(n+1)/2,i^*}} a_{n(n+1)/2,t}]$$

and the one-day-ahead volatility matrix is

$$\hat{\mathbf{H}}_t = \begin{bmatrix} \frac{a_{1,i^*+1}}{a_{1,i^*}} a_{1,t} & \frac{a_{2,i^*+1}}{a_{2,i^*}} a_{2,t} & \cdots & \frac{a_{n,i^*+1}}{a_{n,i^*}} a_{n,t} \\ \frac{a_{2,i^*+1}}{a_{2,i^*}} a_{2,t} & \frac{a_{n+1,i^*+1}}{a_{n+1,i^*}} a_{n,t} & \cdots & \frac{a_{2n-1,i^*+1}}{a_{2n-1,i^*}} a_{2n-1,t} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{a_{n,i^*+1}}{a_{n,i^*}} a_{n,t} & \frac{a_{2n-1,i^*+1}}{a_{2n-1,i^*}} a_{2n-1,t} & \cdots & \frac{a_{n(n+1)/2,i^*+1}}{a_{n(n+1)/2,i^*}} a_{n(n+1)/2,t} \end{bmatrix}.$$

In addition, more about attributes and estimation of forecasting errors are shown as follows:

- More attributes included. The attributes can be more than volatilities and covariances, the distinct entries of volatility matrix. Other attributes can be prices or indexes (open, high, low, close), volumes, interest rate, currency exchange and so on.

- Estimation of forecasting errors. The forecasting errors is considered with k sequential time indexes. In other words, k sequential volatility matrices are considered together to calculate the forecasting errors. Given that

$$\mathbf{H}_{T_0}^{real}(k) = [\mathbf{H}_{T_0-k+1}^{real}, \mathbf{H}_{T_0-k+2}^{real}, \ldots, \mathbf{H}_{T_0}^{real}],$$

$$\hat{\mathbf{H}}_{T_0}^{hmm}(k) = [\hat{\mathbf{H}}_{T_0-k+1}^{hmm}, \hat{\mathbf{H}}_{T_0-k+2}^{hmm}, \ldots, \hat{\mathbf{H}}_{T_0}^{hmm}].$$

where $\mathbf{H}_{T_0}^{real}(k)$ and $\hat{\mathbf{H}}_{T_0}^{hmm}(k)$ are the $n \times (k \times n)$ matrices consisting of k sequential realized volatility matrices and k sequential forecasted volatility matrices by HMM multivariate forecasting algorithm respectively . The covariance matrices generated by the HMM multivariate forecasting algorithm using k time-series samples can be written respectively as

$$TE_{T_0}^{hmm} = \|\mathbf{H}_{T_0}^{real}(k) - \hat{\mathbf{H}}_{T_0}^{hmm}(k)\|_F,$$

where $\|.\|_F$ is the Frobenius norm.

3.4 Empirical Results

In this section, we present univariate forecasting and multivariate forecasting by applying proposed HMM forecasting algorithm. In the univariate forecasting example, we use HMM forecasting algorithm to forecast the *close* price of Heng Sang Index (HSI) based on *open*, *high*, *low*, *close*, and *volume*. In the multivariate forecasting example, we apply the proposed algorithm to test the multivariate samples, and compare the HMM forecasting algorithm with our FMVM forecasting algorithm.

3.4.1 Hang Seng Index Forecasting: A Univariate Forecasting Example

The HMM forecasting is evaluated based on HSI. These daily trading samples, from 2 February 2015 to 1 March 2016, are obtained from the Blomberg databases. The forecasting results of the HMM forecasting algorithm are compared with the realized results. This period includes the upward tendency, downward tendency and mean reversion period. Thus the HMM forecasting can be tested whether it is applicable in different situations. Five attributes are mentioned in this case: they are *open*, *high*, *low*, *close*, and *volume*. The forecasting target is the attribute, *close*.

- In the training phase, use K-means clustering algorithm to label data items. For simplification, the number of clusters is fixed as the same as the number of the attribute, 5. Data items would be labeled into any one from $1, \ldots, 5$ by their similarity in this case, where $L = 5$.

- Then, transform all data items in the training phase into integer among $1, \ldots, 5$. After this transformation, all data items, represented by a vector for each one, can be represented by an integer instead. For instance, k-dimension vectors $D_1, D_2, \ldots, D_{T_1}$ can be represented by $I_1, I_2, \ldots, I_{T_1}$, where $I_i \in 1, \ldots, 5$, for any i. For any t, $D_t = [open_t, high_t, low_t, close_t, volumn_t]$.

- The rolling window size, w, is considered to demarcate the length of each pattern. For example, when the rolling window size, w, is predetermined as 20, the first recognition pattern is $I_1, I_2, \ldots, I_{20}$. The first three patterns can be shown as

  1: is $[I_1, I_2, \ldots, I_{20}]$,

  2: is $[I_2, I_3, \ldots, I_{21}]$,

  3: is $[I_3, I_4, \ldots, I_{22}]$,

  ...

- Re-frame the T_1 20-dimension vectors $D_1, D_2, \ldots, D_{T_1}$ to $(T_1 - w)$ recognition pattern series 20-dimension vectors $X_{rws}, X_{rws+1}, \ldots, X_{T_1}$ where $X_k = I_{k-20+1}, I_{k-20+2}, \ldots, I_k$, where $20 \leq k \leq T_1$. For example,

  X_1 is $[I_{1-w+1}, I_{1-w+2}, \ldots, I_1]$,

  X_2 is $[I_{2-w+1}, I_{2-w+2}, \ldots, I_2]$,

  X_3 is $[I_{3-w+1}, I_{3-w+2}, \ldots, I_3]$,

  ...

X_t is $[I_{t-w+1}, I_{t-w+2}, \ldots, I_t]$,

...

X_{T_1} is $[I_{T_1-w+1}, I_{T_1-w+2}, \ldots, I_{T_1}]$.

- Treat X_1, X_2, ..., X_{T_1} as a new variable in the training phase. If one wants to forecast any information of $D_{T_1+1} = [open_{T_1+1}, high_{T_1+1}, low_{T_1+1}, close_{T_1+1}, vol._{T_1+1}]$, for instance, forecasting $\hat{close}_{T_1+1}$. As the technique analysis believes that history tends to repeat itself, our algorithm tends to find the similarity between recent recognition pattern and the previous recognition patterns. Therefore, we need to find the most similar to X_{T_1+1} pattern from X_1, X_2, ..., X_{T_1}, where X_t represent the recognition pattern with rolling window size rws at time index t. Then, HMM classification method can be applied to find the most similar pattern in this circumstance.

- Train HMM by inputing X_1, X_2, ..., X_{T_1} and calculate log-likelihood values L_m of m^th data pattern from X_1, X_2, ..., X_{T_1}: I_1, I_2, ..., I_{T_1}.

- Search the most nearest value of I_{T_1} from I_1, I_2, ..., I_{T_1}. Assume that the index most similar log-likelihood value is m^*, $1 \leq m^* \leq T_1$. Then, X_{m^*} is thought to be the most similar recognition pattern of X_{T_1+1}.

- Assume that the process from pattern X_{T_1} to $X_{T_1}+1$ will be same as the process from X_{m^*} to X_{m^*+1}. The forecasting of one attribute, i.e. $\hat{close}_{T_1+1}$, of X_{T_1+1} can be described as

$$\hat{close}_{T_1+1} = \frac{close_{m^*+1} - close_{m^*}}{close_{m^*}} close_{T_1}.$$

The final result is shown in Figure 3.4.1. The pattern of the forecasted HSI is very similar to that of the realized HSI, however, there exists latency of the forecasted

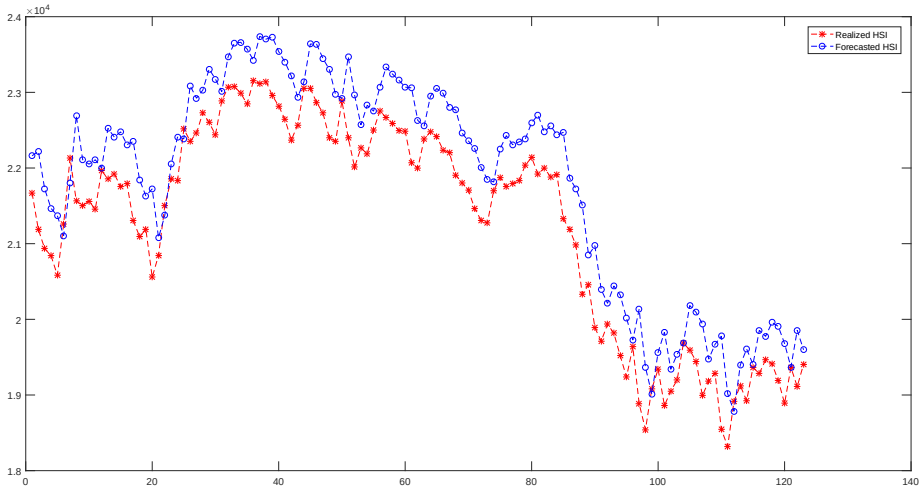


Figure 3.4: The HMM forecasting result of HSI

HSI. The forecasted HSI can represented the tendency of the realized HSI in most time.

The divergence between the realized HSI and the forecasted HSI is shown in Figure 3.4.1. To evaluate the absolute difference between the realized and the forecasted values, the absolute difference is shown in Figure 3.4.1. The divergence cannot directly represent the relative difference of HSI. Thus, the bias ratio of the realized results, $\frac{\text{Absolute difference}}{\text{HSI}}$, is shown in Figure 3.4.1. The mean and standard deviation of the bias ration is also shown in Figure 3.4.1. The mean of bias ratio is 0.0271 and the standard deviation of bias ration is 0.0143. The upper bound is 0.0414 and the lower bound is 0.0128.

3.4.2 Volatility Forecasting

The HMM forecasting algorithm is also evaluated based on 15 high weighted HSI constituent stocks, which is detailed in Table 3.6. These daily trading samples, from 29 October 2010 to 7 October 2014, are obtained from the Bloomberg database which is collected from 29 October 2010 to 7 October 2014. The HMM forecasting uses the

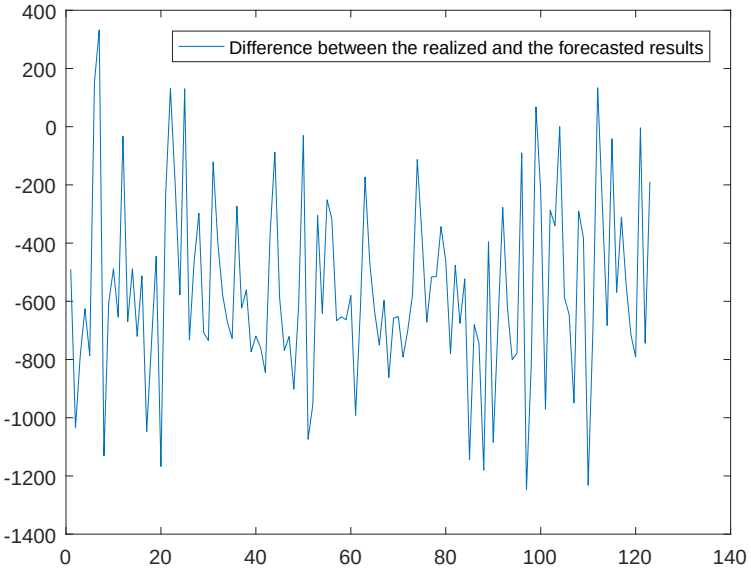


Figure 3.5: Difference between the realized and the forecasted HSI

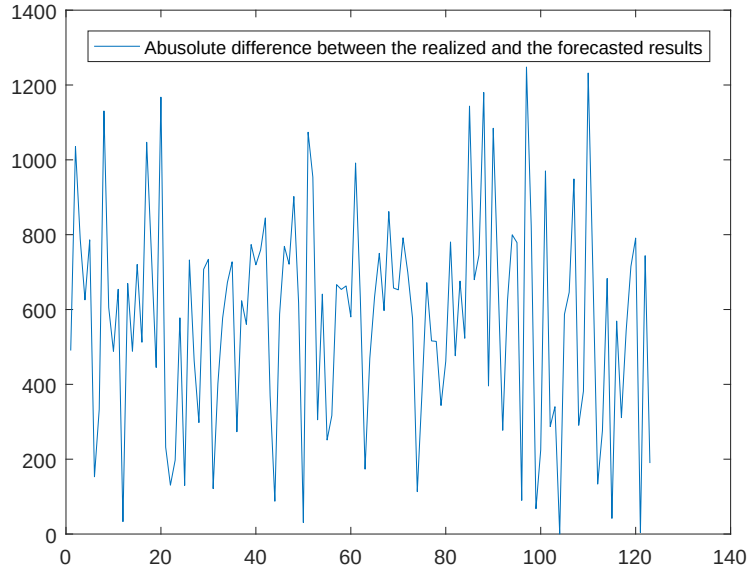


Figure 3.6: The absolute difference between the realized and the forecasted HSI

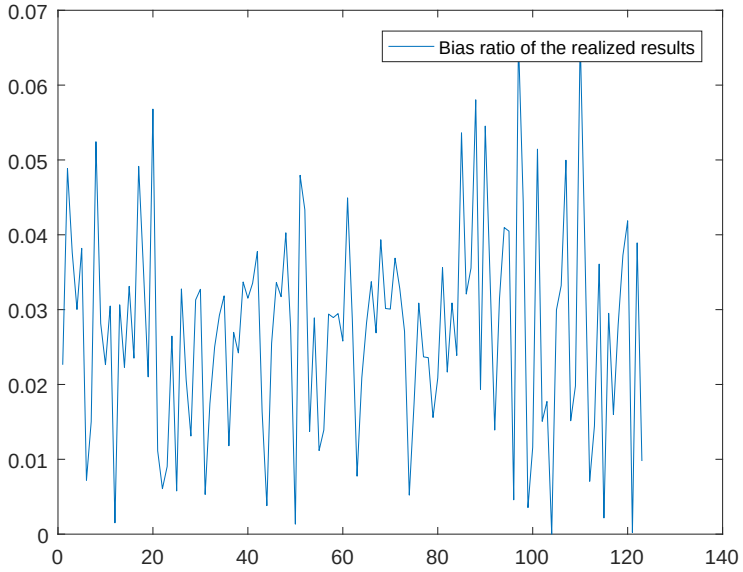


Figure 3.7: Bias ratio of the realized results

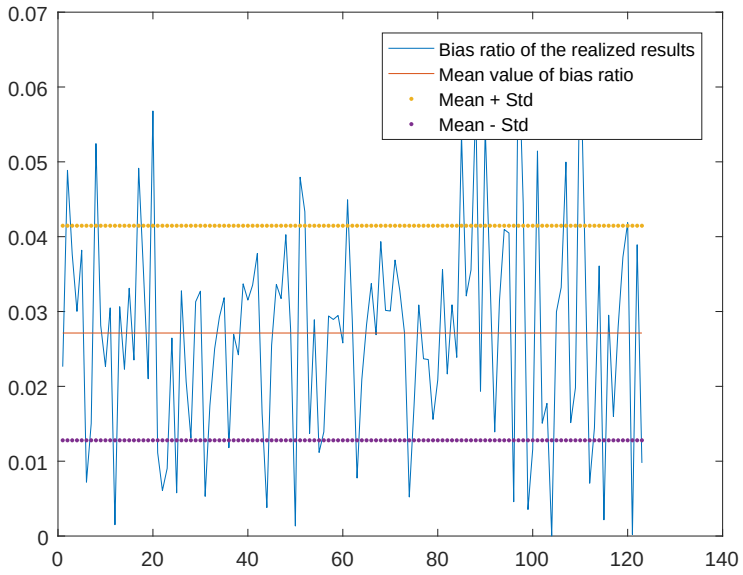


Figure 3.8: Bias ratio of the realized results with boundaries

same data set that used in the empirical results of FMVM. Thus, we can compare HMMforecasting algorithm to proposed HMVM. The HMM forecasting algorithm is also used to forecast the one-day-ahead volatility matrices based on different rolling window size with $k = 20, 30, \ldots, 120$. Also we compared the results obtained by the HMM forecasting algorithm with the existing multivariate volatility approach, the AMVM [36] and with our proposed HMM-based forecasting algorithm.

Table 3.6: Selected stocks in each case

The Low-Dim Case	The High-Dim Case	ordinal number
'5.HK'	'5.HK'	1
'939.HK'	'939.HK'	2
'1299.HK'	'1299.HK'	3
'941.HK'	'941.HK'	4
	'1398.HK'	5
	'3988.HK'	6
	'883.HK'	7
	'857.HK'	8
	'386.HK'	9
	'2628.HK'	10
	'3.HK'	11
	'151.HK'	12
	'762.HK'	13
	'494.HK'	14
	'3328.HK'	15
4 assets	15 assets	

3.4.3 Univariate Forecasting

A Low-Dimensional Example: Forecasting Volatilities with Rolling Window Size 20

For simplicity, we consider as simple case with 4 constituent stocks of HSI: '5.HK', '939.HK', '1299.HK', and '941.HK', shown in Table 3.6. Tag '5.HK', '939.HK', '1299.HK', and '941.HK' by label 1, 2, 3, and 4. Thus the volatility of '5.HK', '1299.HK', and '941.HK' can be labeled as σ_1^2, σ_2^2, σ_3^2, and σ_4^2. Their covariance can be represented by σ_{12}, σ_{13}, σ_{14}, σ_{23}, σ_{24}, σ_{34} ($\sigma_{ij} = \sigma_{ji}$, if $i \neq j$). We evaluate the HMM volatility forecasting algorithm with different rolling window sizes. The rolling

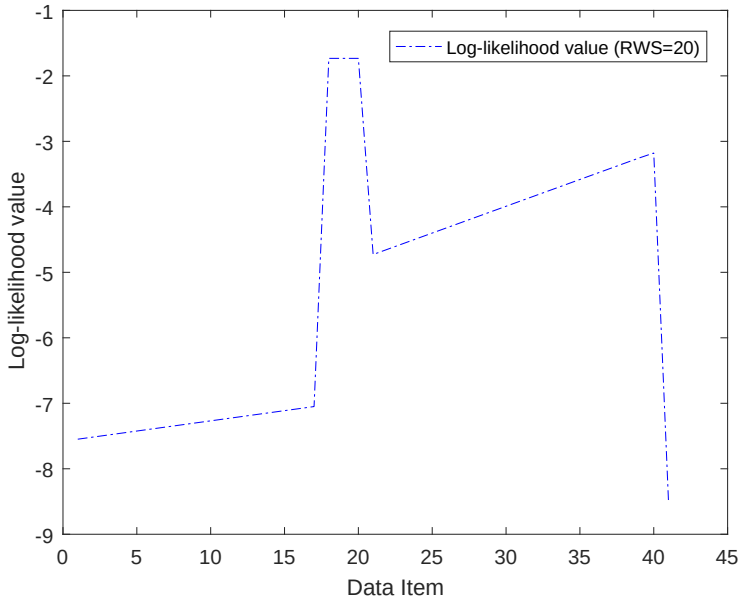


Figure 3.9: Date item with log-likelihood values (RWS: 20, N: 4)

window size is varied from $k = 20$ to 120 in a step of 10. To explain the forecasting volatilities procedure clearly, the numerical result of some intermediate steps are given in this part. Those results present can show the performance of the HMM volatility forecasting directly. In this example, 4 assets can constitute 4 variances (volatilities), σ_1^2, σ_2^2, σ_3^2, σ_4^2 and 6 covariances, σ_{12}, σ_{13}, σ_{14}, σ_{23}, σ_{24}, σ_{34}.

The log-likelihood values of any two data items can show how much they share the similarity: the closer the log-likelihood values are, more similar the two data items are. The data labels with their log-likelihood values are shown in Figure 3.4.3. The data labels for a sample are ranking in time-series order with the respective log-likelihood values. Rank their log-likelihood values in ascending order, the ascending values is shown in Figure 3.4.3. In operation phase, use the similar dynamic progress of the most similar pattern to forecast the following volatility or covariance. The '1299.HK', and '941.HK'

The volatilities, σ_1^2, σ_2^2, σ_3^2, σ_4^2, are estimated on a basis of 20 trading days.

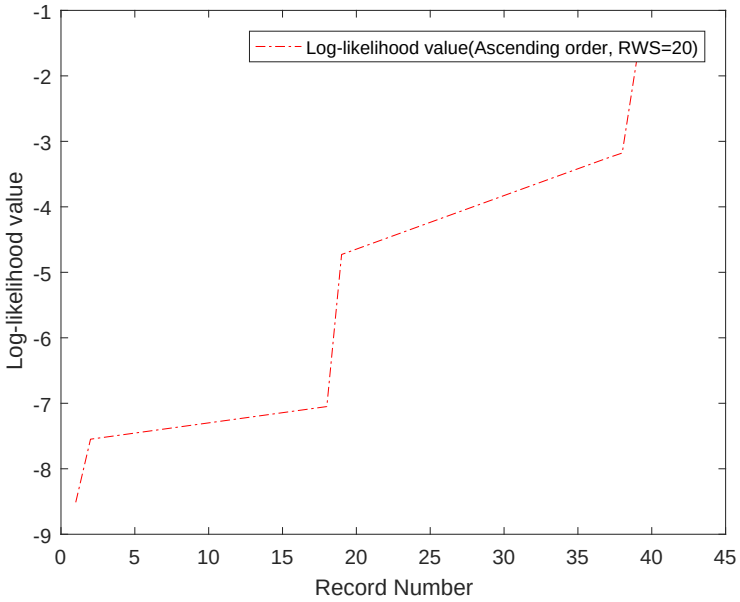
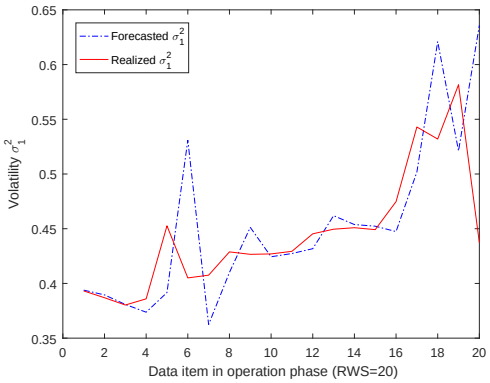
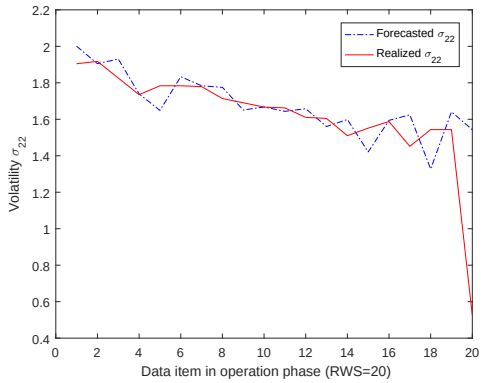


Figure 3.10: Log-likelihood values in ascending order (RWS: 20, N: 4)

Therefore, the rolling window size in this case is also set as 20. The realized and forecasted volatilities of '5.HK' and those of '939.HK' are shown in Figure 3.11. In the same way, the realized and forecasted volatilities of the '1299.HK', and '941.HK' are shown in Figure 3.12. In the variance forecasting, some peaks of forecasted volatilities occur when using the HMM volatility forecasting algorithm. Those 'peaks' cause the high forecasting errors in some trading days. Some delays of HMM forecasting occur at the turnover of the tendency of volatility. The covariance, which describes volatile attribute of any two financial assets, of any two from '5.HK', '939.HK', '1299.HK', and '941.HK' can also test the usefulness of HMM forecasting algorithm. Those realized and the forecasted results of each covariance, σ_{12}, σ_{13}, σ_{14}, σ_{23}, σ_{24}, σ_{34}, are shown in Figure 3.13 to 3.15. The drawbacks of the HMM forecasting reoccur in forecasting covariances: the delay of the switching tendency and the 'peak' of forecasting still exist.

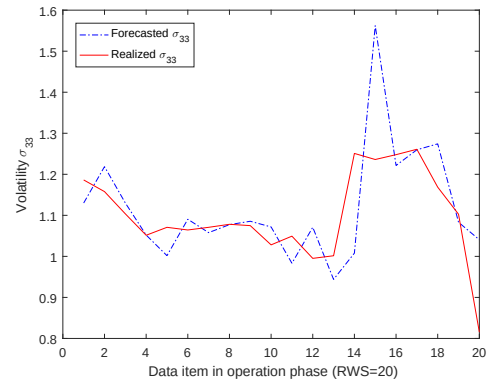


(a) The volatility of '5.HK'

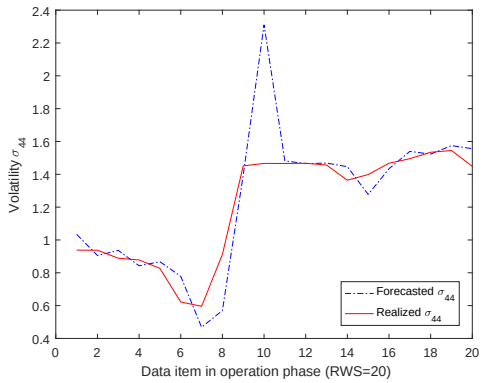


(b) The volatility of '939.HK'

Figure 3.11: Difference between the realized and the forecasted results (RWS:20, N:4, P1)



(a) The volatility of '1299.HK'

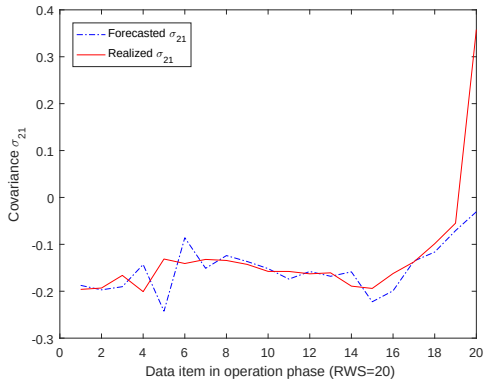


(b) The volatility of '941.HK'

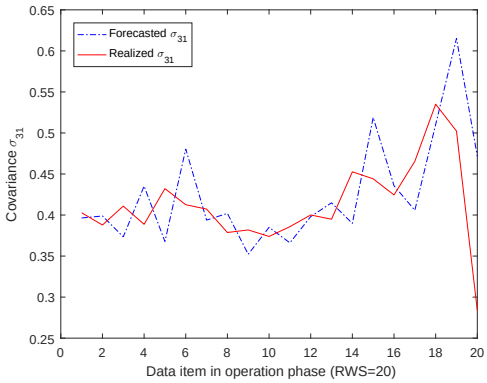
Figure 3.12: Difference between the realized and the forecasted results (RWS: 20, N: 4, P2)

A High-Dimensional Example: Forecasting Volatilities with Rolling Window Size 120

In high-dimensional case, we consider 15 constituent stocks of HSI: '5.HK', '939.HK', '1299.HK', '941.HK', '1398.HK', '3988.HK', '883.HK', '857.HK', '386.HK', '2628.HK', '3.HK', '151.HK', '762.HK', shown in Table 3.6. Tag from '5.HK', '939.HK', '1299.HK' to '941.HK' by label 1, 2, ..., 15. These 15 stocks can compose 15 volatilities and

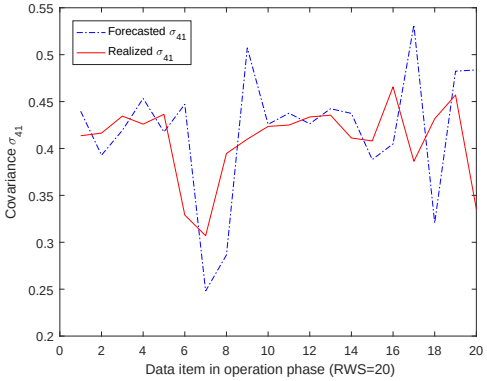


(a) The covariance of '5.HK' and '939.HK'

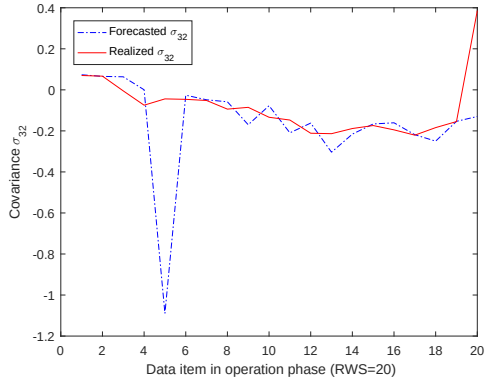


(b) The volatility of '5.HK' and '1299.HK'

Figure 3.13: Difference between the realized and the forecasted results (RWS: 20, N: 4, P3)



(a) The covariance of '5.HK' and '941.HK'



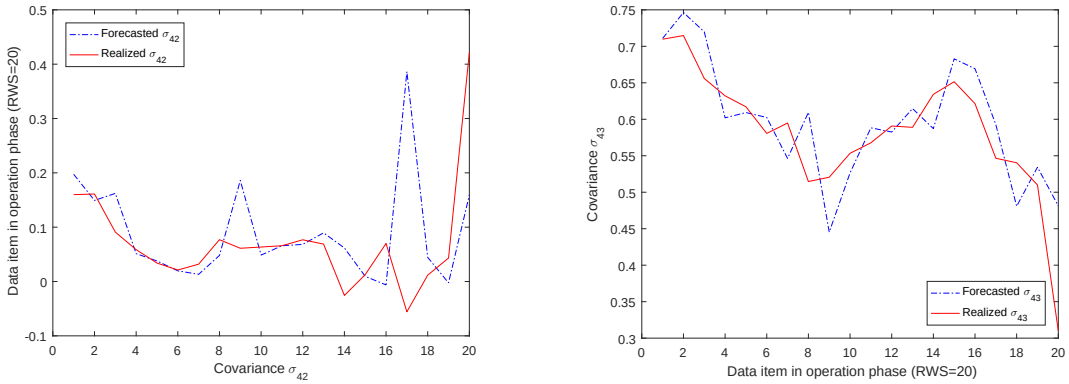
(b) The covariance of '939.HK' and '1299.HK'

Figure 3.14: Difference between the realized and the forecasted results (RWS: 20, N: 4, P4)

105 covariances in their volatility matrix. Therefore, the attribute vector can be

$$\mathbf{V}_{15,t} = (a_{1t}, a_{2t}, \ldots, a_{120,t}),$$

where $a_{1t}, a_{2t}, \ldots, a_{120,t}$ represents $\sigma_{11}, \sigma_{21}, \ldots, \sigma_{15,1}, \ldots, \sigma_{22}, \sigma_{32}, \ldots, \sigma_{15,2}, \ldots, \sigma_{120,120}$ at time index t. The log-likelihood values of any two data items can show how much they share the similarity: the closer the log-likelihood values are, more similar the two data items are. The data labels with their log-likelihood values are shown in Figure 3.4.3. The data labels for a sample are ranking in time-series order with the



(a) The covariance of '939.HK' and '941.HK' (b) The covariance of '1299.HK' and '941.HK'

Figure 3.15: Difference between the realized and the forecasted results (RWS: 20, N: 4, P5)

respective log-likelihood values. Rank their log-likelihood values in ascending order, the ascending values is shown in Figure 3.4.3. In operation phase, use the similar dynamic progress of the most similar pattern to forecast the following volatility or covariance. The '1299.HK', and '941.HK'

In terms of 15 assets, 15 variances and 105 distinct covariances exist in one

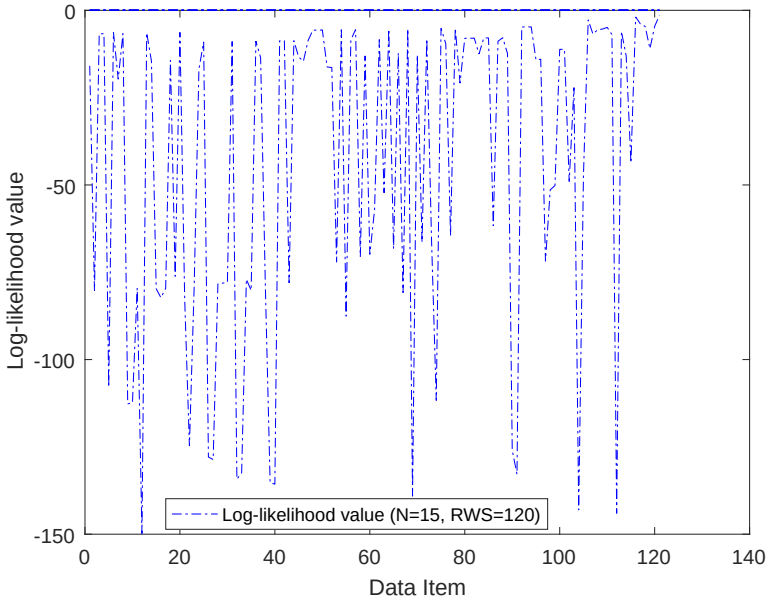


Figure 3.16: Date item with log-likelihood values (RWS: 120, N: 15)

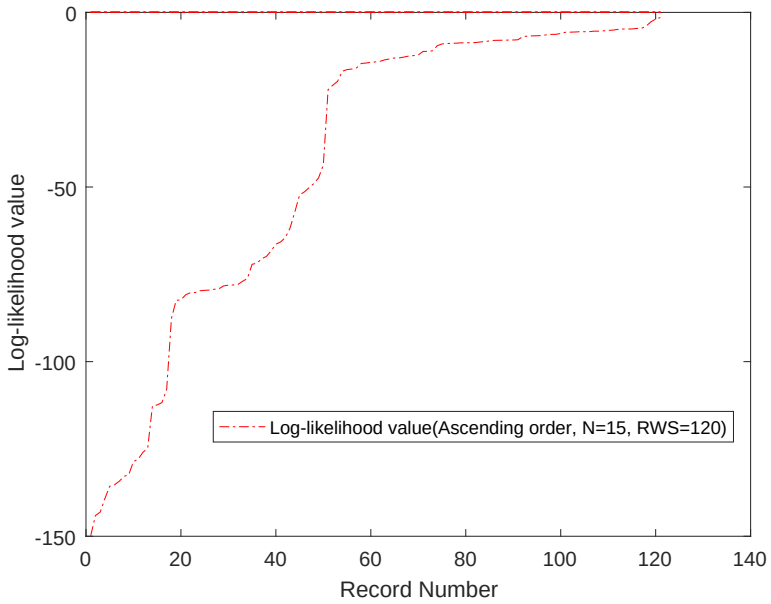
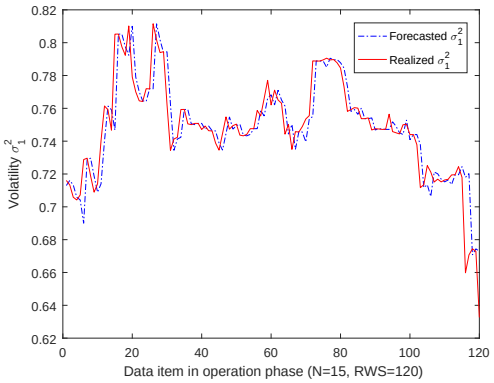


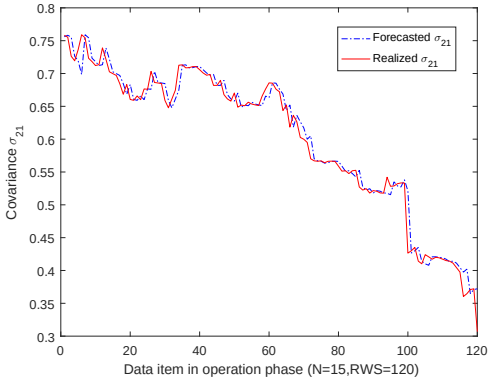
Figure 3.17: Log-likelihood values in ascending order (RWS: 120, N: 15)

volatility matrix. For simplification, we only show the details of first 10 attributes. The volatility, σ_1^2, and covariances, σ_{21}, σ_{31}, ..., $\sigma_{10,1}$, are estimated on a basis of 120 trading days. The realized and forecasted volatilities of '5.HK' and those of covariances of '5.HK' and '939.HK'are shown in Figure 3.18. In the same way, the realized and forecasted covariances of the '5.HK' and '1299.HK', and those of '5.HK' and '941.HK' are shown in Figure 3.19. In the variance or covariance forecasting, some peaks of forecasted volatilities occur when using the HMM volatility forecasting algorithm. Those 'peaks' cause the high forecasting errors in some trading days. Some delays of HMM forecasting occur at the turnover of the tendency of volatility.

Other realized and the forecasted results of each covariance, σ_{51}, σ_{61}, σ_{71}, σ_{81}, σ_{91}, $\sigma_{10,1}$, are shown in Figure 3.20 to 3.22. The drawbacks of the HMM forecasting reoccur in forecasting covariances: the delay of the switching tendency and the 'peak' of forecasting still exist.

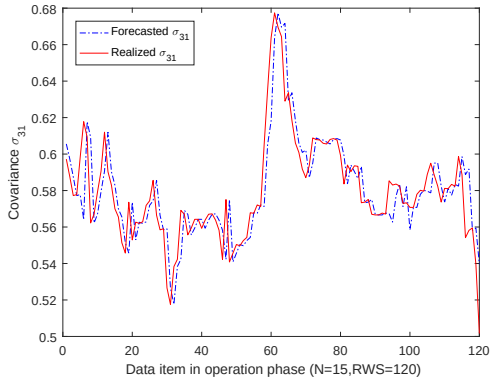


(a) The volatility of '5.HK'

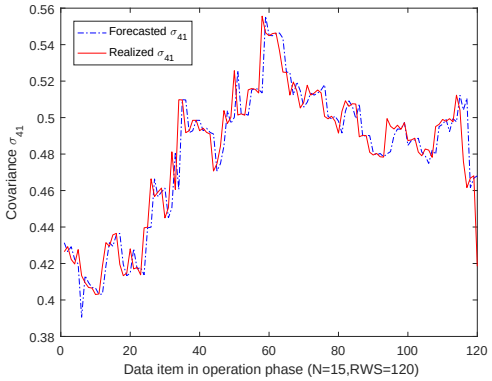


(b) The covariance of '5.HK' and '939.HK'

Figure 3.18: Difference between the realized and the forecasted results (RWS: 120, N: 15, P1)



(a) The covariance of '5.HK' and '1299.HK'

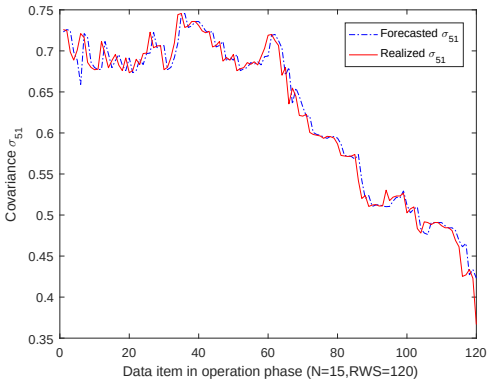


(b) The covariance of '5.HK' and '941.HK'

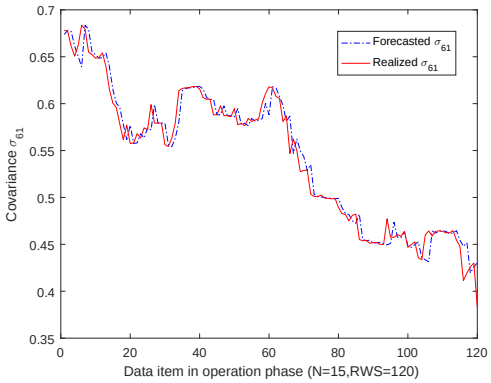
Figure 3.19: Difference between the realized and the forecasted results (RWS: 120, N: 15, P2)

3.4.4 Multivariate Forecasting

The CCC, DCC, ADCC, BEEK, OGARCH, AMVM models and the proposed FMVM were implemented using matlab with toolboxes including the Oxford MFE Matlab toolbox [41]. We evaluate the models with different rolling window sizes. The window size is varied from $k = 20$ to 120 in a step of 10. The date, 7 October 2014, is used as the end window. Both low-dimensional and high-dimensional circumstances are considered.

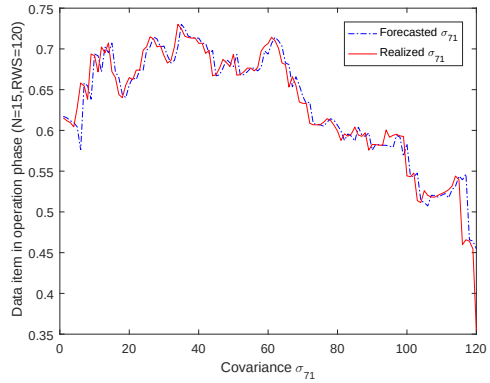


(a) The covariance of '5.HK' and '1398.HK'

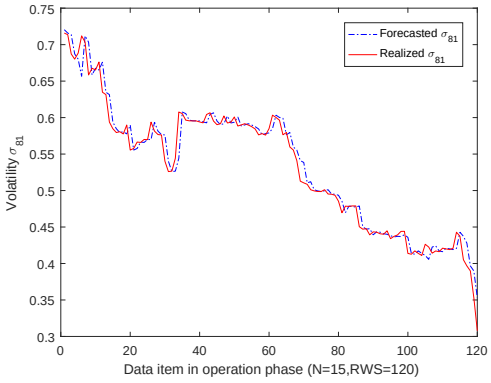


(b) The covariancec of '5.HK' and '3988.HK'

Figure 3.20: Difference between the realized and the forecasted results (RWS: 120, N: 15, P3)



(a) The covariance of '5.HK' and '883.HK'

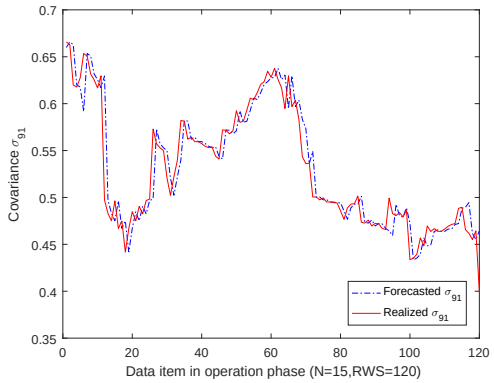


(b) The covariance of '5.HK' and '857.HK'

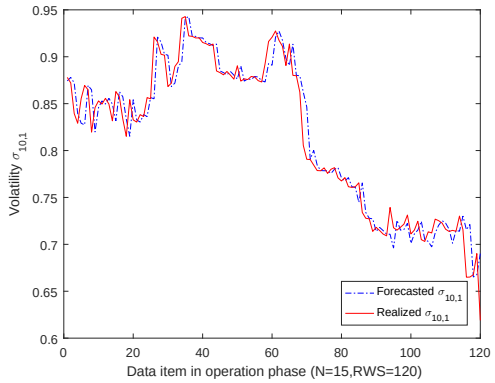
Figure 3.21: Difference between the realized and the forecasted results (RWS: 120, N: 15, P4)

The Low-Dimensional Case: Forecasting Based on the First Attribute

Here, we consider a simple case with 4 stocks which are shown in the first column of Table 2.1. in the HMM forecasting algorithm, K-means clustering algorithm is involved in the transformed pattern recognition part. The clustering results may change each time, thus investigation of whether the forecasting results is steady or not needs to be done. Therefore, the HMM forecasting algorithm is simulated in 20 times. To investigate the influence of the K-means clustering, the HMM fore-



(a) The covariance of '5.HK' and '386.HK'



(b) The covariance of '5.HK' and '2628.HK'

Figure 3.22: Difference between the realized and the forecasted results (RWS: 120, N: 15, P5)

casting algorithm involves in-sample data to save the computation time. However, the comparing classes, the FMVM model and the AMVM model is applied based on out-of-sample data. The tracking errors $TE_T^{hmm}(k)$ with different rolling window size and different simulation order are shown in Table 3.7 and Table 3.8. $TE_T^{hmm}(k)$ is the total tracking errors for the training phase obtained by the HMM multivariate forecasting algorithm with in-sample data. More specific, $TE_{mean}^{hmm}(k)$ and $TE_{mode}^{hmm}(k)$ is the mean and mode of all the simulation results with every rolling windows sized mentioned in this part. Referring to Table 3.7 and Table 3.8, the simulation results show that the uncertainty of K-means clustering algorithm does not affect the HMM forecasting algorithm. The forecasting errors are very close in different simulation. In table 3.9, the HMVM is the best among three forecasting algorithm and the AMVM is better than the HMM forecasting algorithm. This might be this case only involves the forecasting based on the first attribute. If the forecasting based on all attributes, the HMM forecasting algorithm might be better.

Table 3.7: Simulation results of forecasting errors with different rolling windows size (low-dim)

Simulation	20	30	40	50	60	70	80	90	100	110	120
1	10.8488	14.5491	10.9235	15.3039	16.4204	17.1651	17.2774	17.1305	17.2211	17.6132	18.1535
2	11.3438	15.0610	10.9226	14.6879	16.377	17.1404	17.2774	17.1453	17.2211	17.6016	18.1407
3	13.0120	14.7521	10.9175	15.3039	16.4662	17.1585	17.2198	17.1558	17.2175	17.6042	18.1461
4	10.1323	15.0778	10.9181	14.6879	16.4460	17.1235	17.2276	17.1234	17.2299	17.6058	18.1321
5	12.9604	8.3660	10.9277	15.3060	16.4057	17.0701	17.2462	17.1305	17.2211	17.5993	18.1394
6	10.8263	15.2066	10.9240	15.3002	16.4247	17.1626	17.2232	17.1209	17.2299	17.5991	18.1397
7	11.5866	15.0825	10.7589	14.8215	16.4466	17.1626	17.2401	17.1241	17.2093	17.5991	18.1397
8	13.0120	9.5023	10.9235	15.2986	16.4116	17.1585	17.2160	17.1209	17.2290	17.5993	18.1344
9	10.8488	15.2231	10.9275	14.6879	16.3988	17.1379	17.2232	17.1162	17.2290	17.6042	18.1344
10	11.3438	17.7201	10.9248	15.2917	16.3774	17.1626	17.2401	17.1322	17.2290	17.5981	18.1344
11	10.8488	17.2659	10.9181	15.2831	16.4204	17.1635	17.2774	17.1551	17.2282	17.6042	18.1461
12	13.0120	9.6063	10.9234	14.5191	16.4258	17.1651	17.2193	17.1558	17.2418	17.6042	18.1461
13	12.9604	15.0671	10.8733	15.2710	16.4096	17.1585	17.2774	17.1624	17.2299	17.6042	18.1461
14	13.0120	9.2935	10.9200	15.3079	16.4803	17.0626	17.2232	17.1209	17.2211	17.6042	18.1344
15	11.3438	15.2026	10.9277	15.3082	16.4096	17.0701	17.2232	17.1107	17.2228	17.6162	18.1791
16	12.9604	15.0750	10.9545	15.3151	16.4103	17.0416	17.2204	17.1453	17.2175	17.5993	18.1481
17	13.4458	15.1709	10.9226	15.3471	16.4125	17.1635	17.2484	17.1209	17.2299	17.6205	18.1321
18	13.0120	15.0899	10.9212	15.4424	16.4012	17.1585	17.2198	17.1340	17.2503	17.6016	18.1486
19	10.8263	15.1837	10.9212	15.2807	16.4096	17.1626	17.2484	17.1107	17.2282	17.5993	18.1397
20	12.9604	15.1788	10.9249	14.6879	16.4143	17.1262	17.2268	17.1325	17.2418	17.6042	18.1461
mean	12.0148	14.1337	10.9137	15.1226	16.4184	17.1357	17.2388	17.1324	17.2274	17.6041	18.1430
max	13.4458	17.7201	10.9545	15.4424	16.4803	17.1651	17.2774	17.1624	17.2503	17.6205	18.1791
min	10.1323	8.3660	10.7589	14.5191	16.3774	17.0416	17.2160	17.1107	17.2093	17.5981	18.1321
std	1.0901	2.6551	0.0390	0.3020	0.0255	0.0406	0.0223	0.0158	0.0094	0.0060	0.0105

The Low-Dimensional Case: Forecasting Based All Attributes

Here, we consider a simple case with 4 stocks which are shown in the first column of Table 2.1, same stocks in the previous example. The stability of the HMM forecasting algorithm in low dimensional case has been investigated in the previous example. This time, the HMM forecasting algorithm is compared with the HMVM and the AMVM with the same out-of-sample data. The results are shown in Table 3.10. In this case, the HMM forecasting algorithm is improved when considering all attributes in its forecasting. The ranking of the forecasting accuracy is the HMM forecasting algorithm, the FMVM, and the AMVM. Also, the results show the forecasting accuracy become better and better as the rolling windows size is lager and lager. This implies the HMM forecasting algorithm will be improved as more recognition patterns and more attributes considered.

The High-Dimensional Case: Forecasting Based on the First Attribute

Here, we consider a high-dimensional case with 15 stocks which are shown in the second column of Table 2.1. in the HMM forecasting algorithm, K-means clustering algorithm is involved in the transformed pattern recognition part. The clustering

Table 3.8: Simulation results in two digits (low-dim, two decimal)

Simulation	20	30	40	50	60	70	80	90	100	110	120
1	10.85	14.55	10.92	15.30	16.42	17.17	17.28	17.13	17.22	17.61	18.15
2	11.34	15.06	10.92	14.69	16.38	17.14	17.28	17.15	17.22	17.60	18.14
3	13.01	14.75	10.92	15.30	16.47	17.16	17.22	17.16	17.22	17.60	18.15
4	10.13	15.08	10.92	14.69	16.45	17.12	17.23	17.12	17.23	17.61	18.13
5	12.96	8.37	10.93	15.31	16.41	17.07	17.25	17.13	17.22	17.60	18.14
6	10.83	15.21	10.92	15.30	16.42	17.16	17.22	17.12	17.23	17.60	18.14
7	11.59	15.08	10.76	14.82	16.45	17.16	17.24	17.12	17.21	17.60	18.14
8	13.01	9.50	10.92	15.30	16.41	17.16	17.22	17.12	17.23	17.60	18.13
9	10.85	15.22	10.93	14.69	16.40	17.14	17.22	17.12	17.23	17.60	18.13
10	11.34	17.72	10.92	15.29	16.38	17.16	17.24	17.13	17.23	17.60	18.13
11	10.85	17.27	10.92	15.28	16.42	17.16	17.28	17.16	17.23	17.60	18.15
12	13.01	9.61	10.92	14.52	16.43	17.17	17.22	17.16	17.24	17.60	18.15
13	12.96	15.07	10.87	15.27	16.41	17.16	17.28	17.16	17.23	17.60	18.15
14	13.01	9.29	10.92	15.31	16.48	17.06	17.22	17.12	17.22	17.60	18.13
15	11.34	15.20	10.93	15.31	16.41	17.07	17.22	17.11	17.22	17.62	18.18
16	12.96	15.08	10.95	15.32	16.41	17.04	17.22	17.15	17.22	17.60	18.15
17	13.45	15.17	10.92	15.35	16.41	17.16	17.25	17.12	17.23	17.62	18.13
18	13.01	15.09	10.92	15.44	16.40	17.16	17.22	17.13	17.25	17.60	18.15
19	10.83	15.18	10.92	15.28	16.41	17.16	17.25	17.11	17.23	17.60	18.14
20	12.96	15.18	10.92	14.69	16.41	17.13	17.23	17.13	17.24	17.60	18.15
mean	12.01	14.13	10.91	15.12	16.42	17.14	17.24	17.13	17.23	17.60	18.14
max	13.45	17.72	10.95	15.44	16.48	17.17	17.28	17.16	17.25	17.62	18.18
min	10.13	8.37	10.76	14.52	16.38	17.04	17.22	17.11	17.21	17.60	18.13
std	1.09	2.66	0.04	0.30	0.03	0.04	0.02	0.02	0.01	0.01	0.01
mode	13.01	15.08	10.92	15.30	16.41	17.16	17.22	17.12	17.23	17.60	18.15

Table 3.9: Tracking errors of the FMVM, the AMVM and the HMM-based algorithm (low-dim)

RWS k	$TE_{II}^{F}(k)$	$TE_{II}^{A}(k)$	$TE_{mean}^{hmm}(k)$	$TE_{mode}^{hmm}(k)$
20	4.8583	7.9454	12.0148	13.01
30	4.4505	7.3815	14.1337	15.08
40	6.6014	9.6715	10.9137	10.92
50	6.7466	10.8176	15.1226	15.30
60	6.8483	10.3731	16.4184	16.41
70	8.0451	12.0133	17.1357	17.16
80	7.9594	13.0397	17.2388	17.22
90	10.2467	14.1335	17.1324	17.12
100	10.5589	14.4315	17.2274	17.23
110	10.1283	14.7852	17.6041	17.60
120	9.5383	13.6743	18.1430	18.15

Table 3.10: Tracking errors of the FMVM, the AMVM and the HMM-based algorithm (low-dim, all attributes)

RWS k	$TE_{II}^{F}(k)$	$TE_{II}^{A}(k)$	$TE^{hmm}(k)$
20	4.8583	7.9454	2.7832
30	4.4505	7.3815	2.2278
40	6.6014	9.6715	2.4847
50	6.7466	10.8176	2.3918
60	6.8483	10.3731	1.9618
70	8.0451	12.0133	1.9843
80	7.9594	13.0397	2.0036
90	10.2467	14.1335	1.6198
100	10.5589	14.4315	1.7146
110	10.1283	14.7852	1.4480
120	9.5383	13.6743	1.4450

results may change each time, thus investigation of whether the forecasting results is steady or not needs to be done. Therefore, the HMM forecasting algorithm is simulated in 20 times. To investigate the influence of the K-means clustering, the HMM forecasting algorithm involves in-sample data to save the computation time. However, the comparing classes, the FMVM model and the AMVM model is applied based on out-of-sample data. The tracking errors $TE_{T}^{hmm}(k)$ with different rolling window size and different simulation order are shown in Table 3.11 and 3.12. $TE_{T}^{hmm}(k)$ is the total tracking errors for the training phase obtained by the HMM multivariate forecasting algorithm with in-sample data. More specific, $TE_{mean}^{hmm}(k)$ and $TE_{mode}^{hmm}(k)$ is the mean and mode of all the simulation results with every rolling windows sized mentioned in this part.

Referring to Table 3.11 and Table 3.12, the simulation results show that the uncertainty of K-means clustering algorithm does not affect the HMM forecasting algorithm. The forecasting errors are very close in different simulation. In table 3.13, the HMVM is the best among three forecasting algorithm and the AMVM is better than the HMM forecasting algorithm. This might be this case only involves the forecasting based on the first attribute. If the forecasting based on all attributes,

Table 3.11: Simulation results of forecasting errors with different rolling windows size (high-dim)

Simulation	20	30	40	50	60	70	80	90	100	110	120
1	91.8885	55.3793	839.7773	85.2940	82.0625	85.2951	91.5574	96.1251	99.2492	99.6343	102.3337
2	91.8885	55.2579	839.8575	100.1592	82.2236	85.1728	100.5265	96.0500	97.4358	101.4063	102.2740
3	65.6424	55.2579	838.8812	87.9835	81.9495	85.3986	101.9195	96.4401	99.3486	99.6202	104.0646
4	65.6424	54.1040	839.8546	87.9897	82.0254	85.5637	100.5265	96.4779	99.4288	99.4886	102.2733
5	479.4283	55.3793	839.9576	66.3139	82.2236	85.6143	98.4934	96.1276	99.4008	99.6494	102.2303
6	91.8885	55.3793	839.7773	87.9897	82.0430	86.0466	91.5574	96.3380	99.4497	99.6494	102.2720
7	392.9276	55.2579	840.3552	88.6758	82.2236	85.5637	98.3849	94.9320	99.3352	99.6494	102.2303
8	91.8885	53.7214	839.8575	88.6758	81.5488	85.6601	100.3989	96.4779	97.4442	99.5320	102.2682
9	392.9276	76.3668	840.0506	88.6758	82.2236	85.5637	92.3905	96.4779	99.3352	100.3899	102.2291
10	73.0100	53.7214	536.6419	87.9897	82.2236	86.1089	101.9195	96.1211	555.5268	99.5849	102.2525
11	91.8885	55.2524	839.8575	88.2184	82.2236	85.7782	91.5574	96.3380	99.4276	99.6494	102.2238
12	56.6853	55.2579	839.9205	87.9897	82.2236	85.4040	95.3844	96.4401	99.3486	99.5888	102.2867
13	309.1451	63.8147	839.7773	87.9897	82.2236	85.4505	100.3989	96.4779	99.3352	99.5950	102.1405
14	65.6424	76.3668	839.8575	88.6758	82.0625	85.5534	92.3905	96.4779	99.4288	99.5718	102.2238
15	65.6424	55.3793	838.8812	100.1592	82.2236	86.1138	100.3989	94.9320	97.2129	99.6202	102.2303
16	91.8885	53.7214	840.3540	87.9897	82.9796	85.5593	91.5574	96.4779	99.4276	99.5093	102.2291
17	56.6853	53.9577	536.6419	88.2184	82.2236	86.0466	100.4293	96.1276	99.3795	99.5473	102.2238
18	57.5364	59.5460	839.9576	94.6963	82.0625	85.4506	98.8869	96.4779	99.3678	99.5950	102.3094
19	91.8885	55.3793	839.8575	94.9992	82.0625	85.6062	100.3989	96.4779	99.4288	99.5768	102.2733
20	73.0100	59.5460	536.6419	88.2184	82.2236	85.5534	100.3209	96.4779	99.4008	99.5950	102.2531
mean	139.8572	57.9023	794.3379	88.8451	82.1628	85.6252	97.4699	96.2136	121.8856	99.7226	102.3411
max	479.4283	76.3668	840.3552	100.1592	82.9796	86.1138	101.9195	96.4779	555.5268	101.4063	104.0646
min	56.6853	53.7214	536.6419	66.3139	81.5488	85.1728	91.5574	94.9320	97.2129	99.4886	102.1405
std	133.7207	6.7726	111.0668	6.7075	0.2517	0.2669	4.0338	0.4648	102.0712	0.4369	0.4076
mode	91.8885	55.3793	839.8575	87.9897	82.2236	85.5637	91.5574	96.4779	99.4288	99.6494	102.2303

Table 3.12: Simulation results of forecasting errors with different rolling windows size (high-dim, two decimal)

Simulation	20	30	40	50	60	70	80	90	100	110	120
1	91.89	55.38	839.78	85.29	82.06	85.30	91.56	96.13	99.25	99.63	102.33
2	91.89	55.26	839.86	100.16	82.22	85.17	100.53	96.05	97.44	101.41	102.27
3	65.64	55.26	838.88	87.98	81.95	85.40	101.92	96.44	99.35	99.62	104.06
4	65.64	54.10	839.85	87.99	82.03	85.56	100.53	96.48	99.43	99.49	102.27
5	479.43	55.38	839.96	66.31	82.22	85.61	98.49	96.13	99.40	99.65	102.23
6	91.89	55.38	839.78	87.99	82.04	86.05	91.56	96.34	99.45	99.65	102.27
7	392.93	55.26	840.36	88.68	82.22	85.56	98.38	94.93	99.34	99.65	102.23
8	91.89	53.72	839.86	88.68	81.55	85.66	100.40	96.48	97.44	99.53	102.27
9	392.93	76.37	840.05	88.68	82.22	85.56	92.39	96.48	99.34	100.39	102.23
10	73.01	53.72	536.64	87.99	82.22	86.11	101.92	96.12	555.53	99.58	102.25
11	91.89	55.25	839.86	88.22	82.22	85.78	91.56	96.34	99.43	99.65	102.22
12	56.69	55.26	839.92	87.99	82.22	85.40	95.38	96.44	99.35	99.59	102.29
13	309.15	63.81	839.78	87.99	82.22	85.45	100.40	96.48	99.34	99.59	102.14
14	65.64	76.37	839.86	88.68	82.06	85.55	92.39	96.48	99.43	99.57	102.22
15	65.64	55.38	838.88	100.16	82.22	86.11	100.40	94.93	97.21	99.62	102.23
16	91.89	53.72	840.35	87.99	82.98	85.56	91.56	96.48	99.43	99.51	102.23
17	56.69	53.96	536.64	88.22	82.22	86.05	100.43	96.13	99.38	99.55	102.22
18	57.54	59.55	839.96	94.70	82.06	85.45	98.89	96.48	99.37	99.59	102.31
19	91.89	55.38	839.86	95.00	82.06	85.61	100.40	96.48	99.43	99.58	102.27
20	73.01	59.55	536.64	88.22	82.22	85.55	100.32	96.48	99.40	99.59	102.25
mean	139.86	57.90	794.34	88.85	82.16	85.63	97.47	96.21	121.89	99.72	102.34
max	479.43	76.37	840.36	100.16	82.98	86.11	101.92	96.48	555.53	101.41	104.06
min	56.69	53.72	536.64	66.31	81.55	85.17	91.56	94.93	97.21	99.49	102.14
std	133.72	6.77	111.07	6.71	0.25	0.27	4.03	0.46	102.07	0.44	0.41
mode	91.89	55.38	839.86	87.99	82.22	85.56	91.56	96.48	99.43	99.65	102.23

the HMM forecasting algorithm might be better.

The High-Dimensional Case: Forecasting Based All Attributes

Here, we consider a simple case with 15 stocks which are shown in the first column of Table 2.1, same stocks in the previous example. The stability of the HMM forecasting algorithm in high dimensional case has been investigated in the previous example. This time, the HMM forecasting algorithm is compared with the HMVM and the

Table 3.13: Tracking errors of the FMVM, the AMVM and the HMM-based algorithm (high-dim)

RWS k	$TE_{II}^{F}(k)$	$TE_{II}^{A}(k)$	$TE_{mean}^{hmm}(k)$	$TE_{mode}^{hmm}(k)$
20	4.8583	7.9454	12.0148	13.01
30	4.4505	7.3815	14.1337	15.08
40	6.6014	9.6715	10.9137	10.92
50	6.7466	10.8176	15.1226	15.30
60	6.8483	10.3731	16.4184	16.41
70	8.0451	12.0133	17.1357	17.16
80	7.9594	13.0397	17.2388	17.22
90	10.2467	14.1335	17.1324	17.12
100	10.5589	14.4315	17.2274	17.23
110	10.1283	14.7852	17.6041	17.60
120	9.5383	13.6743	18.1430	18.15

AMVM with the same out-of-sample data. The results is shown in Table 3.14. In this case, the HMM forecasting algorithm is improved when considering all attributes in its forecasting. The ranking of the forecasting accuracy is the HMM forecasting algorithm, the FMVM, and the AMVM. Also, the results show the forecasting accuracy become better and better as the rolling windows size is lager and lager. This implies the HMM forecasting algorithm will be improved as more recognition patterns and more attributes considered.

Table 3.14: Tracking errors of the FMVM, the AMVM and the HMM-based algorithm (high-dim, all attributes)

RWS k	$TE_{II}^{F}(k)$	$TE_{II}^{A}(k)$	$TE^{hmm}(k)$
20	18.4214	33.4936	27.8650
30	17.1039	33.4648	12.9244
40	20.5884	40.1193	11.9522
50	31.0473	48.5467	18.1909
60	29.7626	51.8209	14.6072
70	30.0937	57.6992	13.4278
80	26.4884	58.3126	13.3793
90	29.4302	64.1372	11.2650
100	22.8918	68.6020	10.3618
110	21.6531	69.0370	9.8495
120	21.2291	70.0627	9.5463

3.4.5 Conclusion

In this chapter, the HMM-based forecasting algorithm is proposed to fulfill an idea from a technical analysis perspective. The proposed HMM-based forecasting algorithm attempts to find the most similar pattern from the historical data and makes a prediction in the belief that the history will repeat itself. The proposed forecasting algorithm evaluates the similarities of the historical patterns based on their probabilities of their occurrence under an HMM model. The prediction of a future pattern is made by repeat the historical movement of the successor of the most similar pattern of the current pattern. The effectiveness of the proposed HMM-based forecasting algorithm is evaluated based on 15 stocks from the Hang Seng Index. A comparison of the AMVM, FMVM and the HMM-forecasting algorithm is made, given the same data used in Chapter 2. The empirical results show that the HMM-forecasting algorithm can obtain better accuracy in volatility forecasting than the AMVM. The effectiveness of the HMM forecasting algorithm is varying when the number of input attributes is changing. The HMM forecasting algorithm performs worse than the FMVM when the HMM-based method is forecasting based on the similarity concerning one attribute. However, it can perform better than the FMVM when it considers all input attributes.

Chapter 4

The Application of Fuzzy Volatility Model

The purpose of this chapter is to investigate the application of fuzzy volatility model. To start with, we present a quantitative trading product design. Then, we integrate the implementation of the fuzzy volatility model for the product design. As the framework of the product is assumed, the risk management of the product should be taken into consideration. Usually, practitioners need to manage a portfolio after constructing different financial products in a period. Then, portfolio management is taken into account afterward. Therefore, the volatility matrix, a crucial part of portfolio risk management, should be accurately forecasted. To tackle the difficulty of forecasting volatility matrix, the methods as the fuzzy volatility modeling technique and the fuzzy HMM technique are applied to the product design. Finally, detailed implementation procedures are also put forward through the case analysis.

4.1 Background: Source of Inspiration

Pairs trading is a popular trading strategy invented by Wall Street quant Nunzio Tartaglia and his fellow in 1980s, and this procedure has been conducted for nearly thirty years [21]. The intrinsic idea of pair trading is to identify a pair of financial assets those have same or similar risk exposure and price movement in the past.

Regarding this trading idea, distance method, and co-integration method, namely the traditional methods here, are widely conducted. Distance method uses the distance between the standardized prices of two financial assets as the mispricing measurement and conducts the trading instruction when the measurement satisfies the trigger conditions. Similarly, the co-integration method enters into long or short position when the difference of those returns is relatively large and closes its position when the difference is reverting to its mean level. More details can be found in the literature by Engle[19].

Nowadays, people were debating about the usefulness of the traditional methods. While some asserted that one could not receive an abnormal return, some still earned nearly 10% annual return by using this strategy. The traditional methods are the assumptions: mean-reverting property and jointly normal distributed property of the returns. Once the latter property is assumed, the linear correlation can completely describe dependency and the distribution of spread or difference are stationary under different circumstances of prices and returns [46]. However, returns in financial markets cannot always follow those properties, which may cause few trading opportunities and low return consequently [28, 12]. Considering this drawback, a new technique should be considered in pairs trading. An alternative way to solve this problem is to relax the assumption of linear dependence structure. Copulas theory is considerable in this case because copulas can describe the nonlinear dependence structure of two random variables with more robust and accurate description. This method can find joint distribution of two marginal distributions of two random variables, and it was originally founded in the book by Sklar [42]. As one knows the historical returns of two stocks, she can estimate the marginal distribution of each stock with the data of historical returns and then estimate a particular kind of copula of this two marginal distributions.

4.1.1 Trading Strategy Using Copulas Method

Xie mentioned that the mispricing measurement used in the traditional method is robust only if the validity of joint normal distribution assumption holds [47]. A method having the properties as comparability and consistency is desirable to remove the restrictions of the traditional method. While comparability means that the numerical value of this measurement can reflect the degree of mispricing at any given time point, consistency means that the measurement should be consistent and comparable over the different period and return level. To satisfy those two properties, a measurement is proposed to capture the degree of mispricing [46].

Definition Given time t, the returns of two stocks are $R_t^X = r_t^X$ and $R_t^Y = r_t^Y$. Define $P(R_t^X < r_t^X | R_t^Y = r_t^Y)$, the conditional probability of stock X's return smaller than the current realization r_t^X given the stock Y's return equal to current realization r_t^Y as the mis-pricing index of X given Y, denoted as $MI_{X|Y}$. Similarly, we have $MI_{Y|X}$ [47].

Theorem 1 Assuming E and $E^{'}$ are two possible scenario of the stock prices at time t. $R_t^X = r_t^X$ and $R_t^Y = r_t^Y$ in scenario E while $R_t^X = r_t^{X^{'}}$ and $R_t^Y = r_t^{Y^{'}}$ in scenario $E^{'}$. $E(R_t^X) = E(R_t^Y) = R_t$. $r_t^Y = r_t^{Y^{'}}$, $\varepsilon_t^X = \delta_t^X$, $\varepsilon_t^Y = \delta_t^Y$,$\varepsilon_t^{X^{'}} = \delta_t^{X^{'}}$, and $\varepsilon_t^{Y^{'}} = \delta_t^{Y^{'}}$. $MI_{X|Y} > MI_{X|Y^{'}}$ if and only if $\delta_t^X - \delta_t^Y = \delta_t^{X^{'}} - \delta_t^{Y^{'}}$.

Proof

$$
\begin{aligned}
MI_{X|Y} &= P(R_t^X < r_t^X | R_t^Y = r_t^Y) \\
&= P(R_t^X - R_t^Y < r_t^X - R_t^Y | R_t^Y = r_t^Y) \\
&= P(R_t^X - R_t^Y < r_t^X - r_t^Y | R_t^Y = r_t^Y) \\
&= P(((R_t^X - R_t) - (R_t^Y - R_t)) < r_t^X - r_t^Y | R_t^Y = r_t^Y) \\
&= P(\varepsilon_t^X - \varepsilon_t^Y < r_t^X - r_t^Y | R_t^Y) \\
&= P(\varepsilon_t^X - \varepsilon_t^Y < (r_t^X - r_t) - (r_t^Y - r_t) | R_t^Y = r_t) \\
&= P(\varepsilon_t^X - \varepsilon_t^Y < \delta_t^X - \delta_t^Y | R_t^Y = r_t)
\end{aligned}
$$

Despite non-observation of detailed numerical values of ε_t^X and ε_t^Y, the differences, δ_t^X and δ_t^Y, are captured by the difference between the realization return and its mean return respectively. It can be seen that $MI_{X|Y}$ is a monotonic increasing function of $\delta_t^X - \delta_t^Y$ given that $R_t^Y = r_t^Y$, which shows that a higher value of $MI_{X|Y}$ indicates a higher degree of mispricing. Thus, this satisfies the comparability property.

The consistency property $MI_{X|Y}$ of can be showed as: the measurement $MI_{X|Y}$ indicates the likelihood that the spread, $\varepsilon_t^X - \varepsilon_t^Y$, is smaller than the current realization, $(r_t^X - r_t) - (r_t^Y - r_t)$, given the information. Since the likelihood is comparable across different time periods and return levels, it can be concluded that the measurement $MI_{X|Y}$ is consistency [47].

Theorem 2 If the current realized returns of security X and security Y equal to the expected return $R_t = r_t$, then $MI_{X|Y} = 0.5$.

Proof

$$
\begin{aligned}
MI_{X|Y} &= P(R_t^X < r_t^X | R_t^Y = r_t^Y) \\
&= P(R_t^X < r_t | R_t^Y = r_t^Y) \\
&= P(R_t^X - R_t^Y < r_t - r_t^Y | R_t^Y = r_t^Y) \\
&= P((R_t^X - R_t) - (R_t^Y - R_t) < r_t - r_t | R_t^Y = r_t^Y) \\
&= P(\varepsilon_t^X - \varepsilon_t^Y < r_t - r_t | R_t^Y = r_t^Y) \\
&= P(\varepsilon_t^X - (R_t^Y - R_t) < r_t - r_t | R_t^Y = r_t^Y) \\
&= P(\varepsilon_t^X - 0 < r_t - r_t | R_t^Y = r_t^Y) \\
&= P(\varepsilon_t^X < r_t - r_t | R_t^Y = r_t^Y) \\
&= P(\varepsilon_t^X < 0 | R_t^Y = r_t^Y) \\
&= P(\varepsilon_t^X < 0) \\
&= 0.5
\end{aligned}
$$

This suggests that when the error terms of the returns of the two stocks equal to 0, $MI_{X|Y}$ should be equal to 0.5, which means that under fair pricing, $MI_{X|Y}$ is 0.5. Similarly, it can be proved that under fair pricing, $MI_{Y|X}$ is also 0.5.

Corollary

If $MI_{X|Y} > 0.5$, then stock X is overpriced relative to stock Y.

If $MI_{X|Y} = 0.5$, then stock X is fairly priced relative to stock Y.

If $MI_{X|Y} < 0.5$, then stock X is underpriced relative to stock Y.

It has been discussed that the new measurement $MI_{X|Y}$ has the property of comparability and consistency. Under fair pricing, the $MI_{X|Y}$ is 0.5, and this means that stock X is fairly priced relative to stock Y when $MI_{X|Y} = 0.5$. When $MI_{X|Y} > 0.5$, stock X is overpriced relative to stock Y, and when $MI_{X|Y} < 0.5$, the stock X is

under-priced than stock Y [46].

Mis-Pricing Index under Copulas Framework

The procedure to obtain $MI_{X|Y}$ or $MI_{Y|X}$ using copula technique is provided below. The marginal distributions of the returns of stock X and Y are assumed to be F_1 and F_2, while joint distribution is H.

Theorm 3 (Sklar's Theorem) If $F(\cdot)$ is a n-dimensional cumulative distribution function for the random variables $X_1, X_2, \cdots, X_n$ with continuous margins $F_1, ..., F_n$, then there exists a copula function C, such that

$$F(X_1, X_2, \cdots, X_n) = C(F_1(x_1), \cdots, F_n(x_n)).$$

If the margins are continuous, then C is unique. If any distribution is discrete, then C is uniquely defined on the support of the joint distribution.

Let $U = F_1(R^X)$ and $V = F_2(R^Y)$. According to Sklar's theorem, we can always identify a unique copula C which satisfies the following equation

$$H(r^X, r^Y) = C(u, v) = C(F_1(x_1), F_2(x_2)).$$

Theorm 4 Given current realization of returns of stock X and stock Y , r^X and r^Y), we have

$$MI_{X|Y} = \frac{\partial C(u, v)}{\partial v},$$

and

$$MI_{Y|X} = \frac{\partial C(u, v)}{\partial u}.$$

Proof

Given r^X and r^Y, the mispricing index of X given Y can be calculated as

$$MI_{X|Y} = P(R^X < r^X | R^Y = r^Y) = P[F_1(R^X) < F_1(r^X))|F_2(R^Y) = F_2(r^Y))].$$

Let $u = F_1(F^X)$ and $v = F_2(r^Y)$, then

$$MI_{X|Y} = P(U < u|V = v) = \frac{\partial C(u,v)}{\partial v}.$$

Similarly, we obtain that

$$MI_{Y|X} = P(V < v|U = u) = \frac{\partial C(u,v)}{\partial u}.$$

Copulas-Based Pairs Trading Algorithm

Similar to the distance method and co-integration method, the trading strategy using copulas-based technique still consists of two periods, which are formation period and trading period. Below is the trading algorithm assuming pair X and Y is already selected [47]. The trading strategy is illustrated below.

Step 1. Calculate daily returns for each stock during formation period. Estimate the marginal distributions of returns of stock X and stock Y, which are F_1 and F_2 separately.

Step 2. After obtaining the marginal distributions, estimate the copula described by Sklar's theorem to link the joint distribution H with margins F_1 and F_2, denoted as C.

Step 3. During trading period, daily closing prices p_t^X and p_t^Y are used to calculate the daily returns (r_t^X, r_t^Y). Therefore, $MI_{X|Y}$ and $MI_{Y|X}$ can be calculated for every day in the trading period by using the estimated copula C.

Step 4. Construct short position in stock X and long position in stock Y on the day that $MI_{X|Y} > \Delta_1$ and $MI_{Y|X} > \Delta_2$ if no positions in X or Y is held. Construct long position in stock X and short position in stock Y on the day that $MI_{X|Y} > \Delta_2$ and $MI_{Y|X} > \Delta_1$ if no positions in X or Y is held. The market value of short positions should always equal to the market value of long positions at the start. All positions are closed if $MI_{X|Y}$ reaches Δ_3 or $MI_{Y|X}$ reaches Δ_4; $\Delta_1, \Delta_2, \Delta_3, \Delta_4$ where all pre-determined threshold values are.

4.1.2 Tail-Dependence Method with Copulas

An Pairs-Selecting Strategy

Estimate the cumulative distribution function of each log-return by using a kernel smoothing density which is estimate based on a normal kernel function. Kernel function is a function satisfying

$$\int_{-\infty}^{+\infty} k(x)dx = 1.$$

Let random vector x be as $x = (x_1, x_2, \cdots, x_N)$. And $k_n(\cdot), n = 1, 2, \cdots, N$, is a univariate function of random vector x. Use the continued product of $k_n(x_n), n = 1, 2, \ldots, N$ to construct a function illustrated as

$$k(x) = \sum_{n=1}^{N} k_n(x_n).$$

It can also be described as

$$k(x; h) = \sum_{n=1}^{N} k_n(\frac{x_n}{h_n}).$$

Let $\{Y\}_{t=1}^{T}$ be a $N \times 1$ dimensional time series vector and $Y_t = (Y_{1t}, Y_{2t}, \ldots, Y_{Nt})'$. $F(\cdot)$ and $f(\cdot)$ are the joint distribution function and the density function of Y_t re-

spectively. The cumulative distribution function and probability density function of marginal distribution of Y_{nt} can be denoted as $F_n(\cdot)$ and $f_n(\cdot)$. Give $u_n \in (0,1)$. Regarding the canonical maximum likelihood estimation, the empirical probability density function is

$$\hat{f}_n(y_n) = \frac{1}{Th_n} \sum_{n=1}^{T} k_n(\frac{y_n - Y_{nt}}{h_n}),$$

for $n = 1, \ldots, n$. As Y_t is at point $y = (y_1, y_2, \ldots, y_N)'$, the estimate of its joint distribution function $f(\cdot)$ is

$$\hat{f}(y) = \frac{1}{T|h|} \sum_{n=1}^{T} k(y - Y_t; h),$$

where $|h|$ is the value of the determinant of $|h|$. The estimate of the marginal distribution function of Y_{nt} at point y_n is

$$\hat{F}_n(y_n) = \int_{-\infty}^{y_n} \hat{f}_n(x)dx.$$

Similarly, the estimate of the joint distribution function of Y_t at point $y = (y_1, y_2, \ldots, y_N)'$ is

$$\hat{F}(y) = \int_{-\infty}^{y_1} \int_{-\infty}^{y_2} \int_{-\infty}^{y_3} \ldots \int_{-\infty}^{y_n} \hat{f}(x)dx.$$

If we choose normal kernel function as $k_n(x) = \phi(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}$, then $\hat{F}_n(y_n)$ and $\hat{F}(y)$ can be illustrated as

$$\hat{F}_n(y_n) = \frac{1}{T} \sum_{1}^{T} \phi(\frac{y_n - Y_{nt}}{h_n}),$$

$$\hat{F}(y) = \frac{1}{T} \sum_{1}^{T} \prod_{n=1}^{N} \Phi(\frac{y_n - Y_{nt}}{h_n}),$$

which $\phi(\cdot)$ and $\Phi(\cdot)$ represent the probability density function and cumulative distribution function of standard normal distribution.

By using kernel smoothing density estimation, we can gain two cumulative distribution functions of two log return series. After constructing this two cumulative distribution functions, we can estimate the dependent parameters of Clayton copula and Gumbel copula by using canonical maximum likelihood method (CML).

Before illustrating CML method, we have to introduce the inference functions for margins (IFM) method because they are similar to each other with a little difference. By knowing the estimate of marginal distribution at point x_{ni}, use maximum likelihood method to estimate parameters, θ_1 and θ_2, which can be represented as

$$\hat{\theta_1} = \arg\max \sum_{1}^{T} \ln(\hat{f_{n_1}}(x_{ni}); \theta_1),$$

$$\hat{\theta_2} = \arg\max \sum_{1}^{T} \ln(\hat{f_{n_2}}(x_{ni}); \theta_2).$$

Consequently, the the estimate of the unknown parameter α is

$$\hat{\alpha}\arg\max \sum_{1}^{T} \ln[c(F_{n1}\hat{(}x_{nt}), F_{n2}\hat{(}y_{nt}); \alpha)].$$

The above is known as CML method.

After estimation, it is considerable to test whether Clayton copula or Gumbel copula can fit the sample well by using Kolmogorov-Smirnov test. If those copulas can fit the sample well, then compute the upper-tail and lower-tail dependence coefficient of those two log-returns series. Accordingly, form an assets-selecting strategy based on the upper-tail and lower-tail dependence and establish a stock-programing-trading strategies.

If those copulas can fit the sample well, then compute the upper-tail and lower-tail dependence coefficient of those two log-returns series. The relationship between

the lower-tail dependence coefficient and the dependent parameter of Clayton copula is

$$\lambda_{C_{Cl}^{lo}} = 2^{\frac{1}{\alpha_{Cl}}}$$

and the dependent parameter of Gumbel copula and the upper-tail dependence coefficient is

$$\lambda_{C_{Gu}^{up}} = 2 - 2^{\alpha_{Gu}}$$

where α_{Cl} and α_{Gu} are the dependent parameter of Clayton copula and that of Gumbel copula respectively.

Tail-Dependence Pairs Trading Strategy

Tail-dependence pairs trading algorithm is another copulas-based method in pairs trading field [52]. We have revised their trading strategies and the new algorithm is shown below.

Step 1. Calculate daily returns for each stock during formation period. Estimate the marginal distribution F_1 of returns of stock X and the marginal distribution F_2 of returns of stock Y respectively.

Step 2. After obtaining the marginal distributions, estimate the Gumbel copula C_{Gu} and Clayton copula C_{CL} based on margins F_1 and F_2.

Step 3. Compute upper tail and lower tail dependence coefficients to verify the section of the pairs.

Step 4. During trading period, daily closing prices p_t^X and p_t^Y are used to calculate the daily returns (r_t^X, r_t^Y).

Step 5. Calculate the tail-dependence conditional probabilities

$$Pl_{X|Y} = Pr(R_t^X < r_t^X | R_t^Y < r_t^Y),$$
$$Pl_{Y|X} = Pr(R_t^Y < r_t^Y | R_t^X < r_t^X),$$
$$Pu_{X|Y} = Pr(R_t^X > r_t^X | R_t^Y > r_t^Y),$$
$$Pu_{Y|X} = Pr(R_t^Y > r_t^Y | R_t^X > r_t^X).$$

Construct short position in stock X and long position in stock Y on the day that $Pl_{X|Y} > \Delta_1^{lo}$, $Pl_{Y|X} > \Delta_2^{lo}$, as well as no positions in X or Y are held; in the same way, construct long position in stock X and short position in stock Y on the day that $Pu_{X|Y} > \Delta_3^{up}$, $Pu_{Y|X} > \Delta_4^{up}$, and no positions in X or Y is held.

4.1.3 Relative Method with Copulas

The estimation procedure of copulas is the same as that we illustrated in the previous part. However, we have a new measurement to capture mispricing other than those methods in the literature [51] and [52]. At the same time point, the number, n_1, of historical returns R_t^X of stock X, smaller (larger) than the realization return r_t^X, should not be too much different from n_2 of returns R_t^Y smaller (larger) than the realization return r_t^Y. If n_1 is far from n_2, then stock X might be mis-priced and the arbitrage opportunity exists. The algorithm is illustrated below

Step 1. Count the amount of historical returns satisfied $R_t^X < r_t^X$, denoted as n_1. Count the number of historical returns satisfied $R_t^Y < r_t^Y$, denoted as n_2; similarly we have n_3 of $R_t^X > r_t^X$ and n_4 of $R_t^Y > r_t^Y$.

Step 2. The algorithm with a trigger point, $\Delta^* \in (0, 1)$, is described below

- If $n_1 > n_2$ and $\frac{n_2}{n_1} >= \Delta^*$ then one may short-sell stock Y and, close the short position on the next trading day.

- Else if, $n_1 < n_2$ and $\frac{n_1}{n_2} >= \Delta^*$, then one may short-sell stock X and, close the short position on the next trading day.

- Else if, $n_3 < n_4$ and $\frac{n_3}{n_4} >= \Delta^*$, then one may long stock X and, close the long position on the next trading day.

- Else if, $n_4 < n_3$ and $\frac{n_4}{n_3} >= \Delta^*$, then one may long stock Y and, close the long position on the next trading day.

Here, Δ^* is a predetermined value. We can find a particular interval of it by conducting back-testing.

4.2 The Quantitative Trading Product Design

This section introduces how to design a financial product through the idea of the fundamental trading strategy, the way to form products, the formation of a portfolio of these underlying products, and its risk management.

4.2.1 The Fundamentals of the Product and its Design Idea

This subsection aims to show how to construct a quantitative trading product. The process is as follows: a product can be formed by a trading strategy and the underlying assets; as issuing the product to investors, the cash pool can be used to trade the underlying assets under the instruction of the trading strategy. By executing the trading strategy, the fund manager aims to obtain the return projected in the back testing; after the investment period, return the net value of product as it set in the investment agreement. This part is illustrated in Subsection 4.1.

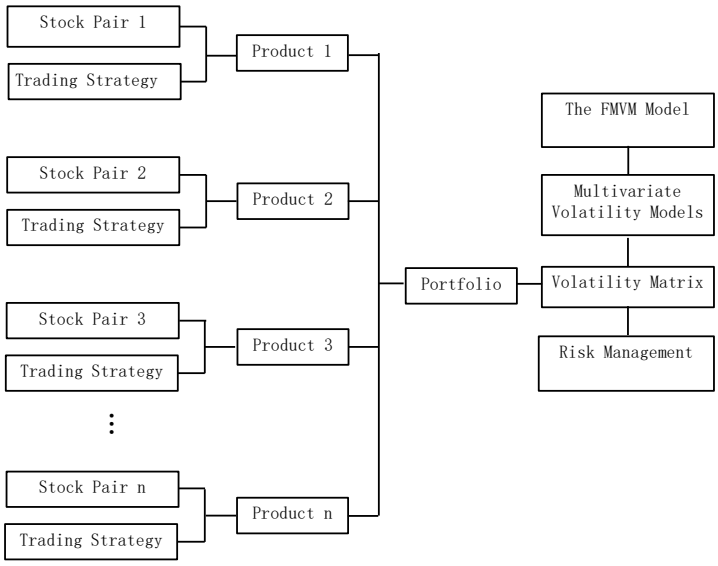


Figure 4.1: The diagram of product design

The core of the product is the trading strategy. The trading strategy and investment idea define the risk characters of the quantitative trading product. Thus, the risk management should be considered in the product design as shown in 4.1. As issuing more than one product, a portfolio is constructed at the same time. The risk character of the portfolio is defined by the volatility matrix. Based on the volatility matrix, one can make a decision with other risk management models. Thus, the application of the FMVM model and fuzzy HMM model can be shown here.

4.2.2 Copulas-Based Two-Sided-Long-Short Strategy

Tail-dependence pairs trading algorithm is an alternative copula-based method in pairs trading field [52]. The strategy shows a positive result. However, they do not consider what to do once the strategy gives a wrong signal; meanwhile, they only close the position on the following trading day only for simplification reason. Here the work refine the previous strategy: 1) revise the timing strategy, 2) add the

stop-loss strategy to this strategy.

Step 1: Calculate daily returns for each stock during formation period. Estimate the marginal distributions of returns of stock X and stock Y, which are F_1 and F_2 separately.

Step 2: After obtaining the marginal distributions, estimate the Gumbel and Clayton copulas to link the joint distribution C_{Gu} and C_{CL} with margins F_1 and F_2.

Step 3: Compute upper tail and lower tail dependence coefficients to verify the section of the pairs. Verify whether the upper tail dependence parameter satisfies $\lambda_{C_{Cl}^{lo}} > k_1$; similarly, verify whether the lower tail dependence parameter satisfies $\lambda_{C_{Gu}^{lo}} > k_2$.

Step 4: Calculate the tail-dependence conditional probabilities

$$Pl_{X|Y} = Pr(R_t^X < r_t^X | R_t^Y < r_t^Y),$$
$$Pl_{Y|X} = Pr(R_t^Y < r_t^Y | R_t^X < r_t^X),$$
$$Pu_{X|Y} = Pr(R_t^X > r_t^X | R_t^Y > r_t^Y),$$
$$Pu_{Y|X} = Pr(R_t^Y > r_t^Y | R_t^X > r_t^X).$$

Step 5: The timing of opening position. If there is no position of security, then

- construct short position of security X, if $Pl_{X|Y} > \Delta_1^{lo}$;
- construct long position of security X, if $Pu_{X|Y} > \Delta_3^{up}$;
- construct long position of security Y, if $Pl_{Y|X} > \Delta_2^{lo}$;
- construct short position of security Y, if $Pu_{Y|X} > \Delta_4^{up}$.

Step 6: The timing of closing position:

- close any position, if the strategy shows a wrong signal.

– close the long position of security X, if $Pl_{X|Y} > \Delta_1^{lo}$;

– close the short position of security X, if $Pu_{X|Y} > \Delta_3^{up}$;

– close the long position of security Y, if $Pl_{Y|X} > \Delta_2^{lo}$;

– close the short position of security Y, if $Pu_{Y|X} > \Delta_4^{up}$.

Here Δ_1^{lo}, Δ_2^{lo}, Δ_3^{up}, and Δ_4^{up} are pre-determined threshold values, which can be found in the training phase.

Parameter Setting

In the copulas-based trading strategy, the upper tail dependence coefficient and lower tail dependence coefficient of a pair of financial instruments are estimated by means of Clayton copula and Gumbel copula; as being illustrated before, the lower tail dependence of Clayton copula and the upper tail dependence coefficient of Gumbel Coupla are

$$\lambda_{C_{Cl}^{lo}} = 2^{\frac{1}{\alpha_{Cl}}},$$

$$\lambda_{C_{Gu}^{up}} = 2 - 2^{\alpha_{Gu}},$$

where the selection criterion is "If $\lambda_{C_{Cl}^{lo}} > k_1$ and $\lambda_{C_{Gu}^{up}} > k_2$, then form a pair consists of asset X and asset Y." k_1 and k_2 are predetermined.

In the timing strategy, Δ_1^{lo}, Δ_2^{lo}, Δ_3^{up}, and Δ_4^{up} are predetermined. They are thresholds of the tail-event probability

$$Pl_{X|Y} = Pr(R_t^X < r_t^X | R_t^Y < r_t^Y),$$

$$Pl_{Y|X} = Pr(R_t^Y < r_t^Y | R_t^X < r_t^X),$$

$$Pu_{X|Y} = Pr(R_t^X > r_t^X | R_t^Y > r_t^Y),$$

$$Pu_{Y|X} = Pr(R_t^Y > r_t^Y | R_t^X > r_t^X),$$

where $Pl_{X|Y}$, $Pl_{Y|X}$, $Pu_{X|Y}$, and $Pu_{Y|X}$ should satisfy $0 \leq Pl_{X|Y} \leq 1$, $0 \leq Pl_{Y|X} \leq 1$,

$0 \leq Pu_{X|Y} \leq 1$, and $0 \leq Pu_{Y|X} \leq 1$. Similarly, the thresholds should also follow $0 \leq \Delta_1^{lo} \leq 1$, $0\Delta_2^{lo} \leq 1$, $0 \leq \Delta_3^{up} \leq 1$, and $0 \leq \Delta_4^{up} \leq 1$.

4.2.3 Copulas-Based One-Sided-Long Strategy

The previous strategy is designed for a two-sided market with a long-short portfolio in Section 4.2.2, i.e., long a stock in Chinese mainland market and short sell its counterpart in Hong Kong market or short sell one in Chinese mainland market and long its counterpart in Hong Kong market. However, high transaction fees and wrong trading instructions occurred both in long and short positions. For a Chinese mainland investor, he or she can take a long position in a mainland stock with a reference system of the tail dependence between the mainland banking sector and the banking industry in Hong Kong. Thus, we can form a new strategy:

Step 1: Calculate daily returns for each stock during formation period. Estimate the marginal distributions of returns of stock X and stock Y, which are F_1 and F_2 separately.

Step 2: After obtaining the marginal distributions, estimate the Gumbel and Clayton copulas to link the joint distribution C_{Gu} and C_{CL} with margins F_1 and F_2.

Step 3: Compute upper tail and lower tail dependence coefficients to verify the section of the pairs. Verify whether the upper tail dependence parameter satisfies $\lambda_{C_{Cl}^{lo}} > k_1$; similarly, verify whether the lower tail dependence parameter satisfies $\lambda_{C_{Gu}^{lo}} > k_2$.

Step 4: Calculate the tail-dependence conditional probabilities

$$Pu_{X|Y} = Pr(R_t^X > r_t^X | R_t^Y > r_t^Y),$$
$$Pl_{X|Y} = Pr(R_t^X < r_t^X | R_t^Y < r_t^Y).$$

Step 5: The timing of opening position. If there is no position of security, then

- construct long position of security X, if $Pu_{X|Y} > \Delta^{up}$.

Step 6: The timing of closing position.

- close any position, if the strategy shows a wrong signal;
- close the long position of security X, if $Pl_{X|Y} > \Delta^{lo}$;

where Δ^{up} and Δ^{lo} are pre-determined threshold values satisfying $0 \geq \Delta^{up}, \Delta^{low} \geq 1$, which can be found in the training phase

4.3 The Application of the FMVM to Portfolio Management

This section shows how the FMVM can apply to portfolio management. We assume the realized volatility matrix is the benchmark for our risk measurement. The framework of this application is shown in Figure 4.2. Assume that the term of return series, s, is fixed and the number of asset is n, the return series matrix $\mathbf{R}_t$ is defined as

$$\mathbf{R}_t = \begin{bmatrix} r_{1,t-s+1} & r_{2,t-s+1} & \cdots & r_{n,t-s+1} \\ r_{1,t-s+2} & r_{2,t-s+2} & \cdots & r_{n,t-s+2} \\ \vdots & \vdots & \ddots & \vdots \\ r_{1,t} & r_{2,t} & \cdots & r_{n,t} \end{bmatrix}.$$

Do not miss up $\mathbf{R}_t$ here with $\mathbf{R}_t$ in Chapter 2. $\mathbf{R}_t$ here describes the returns series matrix. The same notation in Chapter 2 denotes the conditional correlation matrix. Input the returns series matrix $\mathbf{R}_t$ to the FMVM illustrated in Chapter 2, the forecasted volatility matrix $\hat{\mathbf{H}}_{t+1}$ can be retrieved from the FMVM. Given the returns series matrix $\mathbf{R}_t$ and the forecasted volatility matrix $\hat{\mathbf{H}}_{t+1}$, the forecasted optimal

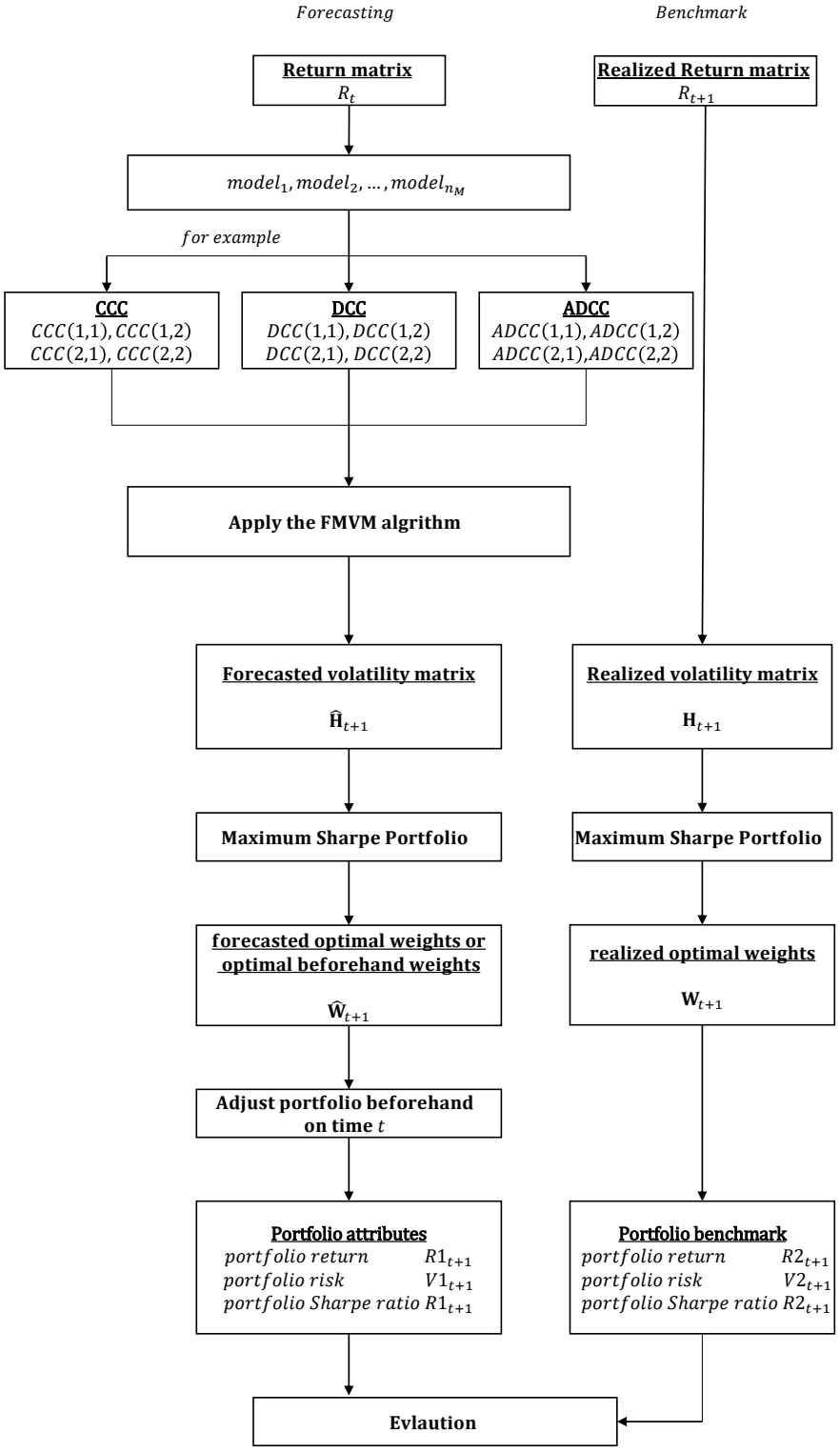


Figure 4.2: The application of the FMVM in portfolio management

weights of each assets, $\hat{w}_{i,t+1}$, can be obtained via Equation 4.1.

$$\begin{aligned} \min \ & \hat{\mathbf{W}}_{t+1}^{T} \hat{\mathbf{H}}_{t+1} \hat{\mathbf{W}}_{t+1}, \\ s.\ t. \ & \hat{\mathbf{W}}_{t+1} \mathbf{R}_t = \bar{r}_t, \\ & \sum_{i=1}^{n} \hat{w}_{i,t+1} = 1, \\ & 0 \leq \hat{w}_{i,t+1} \leq 0.5, \end{aligned} \tag{4.1}$$

where $\hat{w}_{i,t+1}$ is the forecasted optimal weight of the asset i in the portfolio and $\bar{r}_t$ is the current portfolio return. Each weight cannot be larger than 0.5 and the short-sell of an asset is prohibited in our case. The constraint of none short-sell is reasonable in our case: it is unreasonable to short-sell any product that is constituted by one of our trading strategies. The maximum weight is placed to control the tail events of one asset: the "black swan" happens without any foreboding. We need do best preparation no matter when the worst case might happen. Therefore, we define the maximum weight of an asset.

Based on the returns series matrix $\mathbf{R}_t$ and the forecasted volatility matrix $\hat{\mathbf{H}}_{t+1}$, the forecasted optimal weights of each asset, $\hat{w}_{i,t+1}$, can be obtained in another way: maximize the Sharpe ratio and release the constraint of portfolio return. In this case, the cost of funds is also included. Usually, the cost of funds is represented by the risk-free rate r_f. In participation, r_f in Equation 4.2 could be the real cost of the fund house.

$$\begin{aligned} \max \ & \frac{\hat{\mathbf{W}}_{t+1} \mathbf{R}_t - r_f}{\hat{\mathbf{W}}_{t+1}^{T} \hat{\mathbf{H}}_{t+1} \hat{\mathbf{W}}_{t+1}}, \\ s.\ t. \ & \sum_{i=1}^{n} \hat{w}_{i,t+1} = 1, \\ & 0 \leq \hat{w}_{i,t+1} \leq 0.5. \end{aligned} \tag{4.2}$$

Given the forecasted optimal weights matrix $\hat{\mathbf{W}}_{t+1}$, the performance of the beforehand adjustment can be evaluated in time $t+1$ as the realized volatility matrix is available. The portfolio return and portfolio variance after beforehand adjustment can be used to evaluate the performance. The portfolio return after beforehand adjustment can be described as

$$\bar{r}_{p,t+1} = \hat{\mathbf{W}}_{t+1}\mathbf{R}_{t+1}$$

and the portfolio variance after beforehand adjustment can be described as

$$\sigma^2_{p,t+1} = \hat{\mathbf{W}}^T_{t+1}\mathbf{H}_{t+1}\hat{\mathbf{W}}_{t+1}.$$

Furthermore, the Sharpe ration under beforehand adjustment is

$$S.R._{p,t+1} = \frac{\bar{r}_{p,t+1} - r_f}{\sigma_{p,t+1}}.$$

The benchmarks of the beforehand adjustment are all retrieved from the realized information such as the realized optimal weights, the realized portfolio return and the realized portfolio variance under a specified optimal portfolio models. The model can be the minimum-variance model with a specific return

$$\begin{aligned} \min \ & \mathbf{W}^T_{t+1}\mathbf{H}_{t+1}\mathbf{W}_{t+1}, \\ s.\,t. \ & \mathbf{W}_{t+1}\mathbf{R}_{t+1} = \bar{r}_{t+1}, \\ & \sum_{i=1}^{n} w_{i,t+1} = 1, \\ & 0 \leq w_{i,t+1} \leq 0.5, \end{aligned}$$

or maximum-Sharpe-ratio model

$$\max \ \frac{\mathbf{W}_{t+1}\mathbf{R}_t - r_f}{\mathbf{W}_{t+1}^T \mathbf{H}_{t+1} \mathbf{W}_{t+1}},$$

$$s.\ t.\ \ \sum_{i=1}^{n} w_{i,t+1} = 1,$$

$$0 \leq w_{i,t+1} \leq 0.5.$$

Therefore, the benchmark portfolio return is

$$\bar{r}_{p,t+1}^{real} = \mathbf{W}_{t+1}\mathbf{R}_{t+1},$$

and the benchmark portfolio variance is

$$\sigma_{p,t+1}^{2,real} = \mathbf{W}_{t+1}^T \mathbf{H}_{t+1} \mathbf{W}_{t+1}.$$

Furthermore, the benchmark Sharpe ratio is

$$S.R._{p,t+1}^{real} = \frac{\bar{r}_{p,t+1}^{real} - r_f}{\sigma_{p,t+1}^{real}}.$$

4.4 The Empirical Result: Trading Strategies

In this section, we show the empirical result of the trading strategies with a different version. Finally, the application of the previous volatility modeling will be discussed together with the financial instruments with the trading strategies.

4.4.1 Copulas-Based Two-Sided-Long-Short Strategy

Copulas-based two-sided-long-short strategy is formed by the method and data illustrated in Zhang's Strategy [52]. In this subsection, the back-testing result will be shown.

The data will be divided into two parts: one is for the training phase; another one is for the validation phase. The data in the training phase is used to form the

thresholds Δ_1^{lo}, Δ_2^{lo}, Δ_3^{up}, and Δ_4^{up}. For simplicity, let $0 \leq \Delta_1^{lo} = \Delta_2^{lo} = \Delta_3^{up} = \Delta_4^{up} \leq 1$. Once $Pl_{X|Y} > \Delta_1^{lo}$, $Pl_{Y|X} > \Delta_2^{lo}$, $Pu_{X|Y} > \Delta_3^{up}$, or $Pu_{Y|X} > \Delta_4^{up}$, the trading strategy will be trigged. Let initial values of $0\Delta_1^{lo} = \Delta_2^{lo} = \Delta_3^{up} = \Delta_4^{up} = 0.001$, step width equals to 0.001, and the termination is 0.999. In the training phase, find the optimal interval of the thresholds, and test these intervals in the validation stage.

In this section, the companies listing both on China stock market and Hong Kong stock market have been chosen to form a strategy. Those selected companies are China Merchants Bank (CMB), Minsheng Bank (MB), Bank of Communications (BOCs), Agricultural Bank China (ABC), Industrial and Commercial Bank of China (ICBC), China Construction Bank (CCB), Bank of China (BOC), and China Citic Bank (CITIC). Unilateral transaction data will be deleted and only the transaction data keep on the intersected trading days of two markets. The stock data is retried from Nov. 28, 2007 to Nov. 27, 2014. In the training phase, the data from Nov. 28, 2007 to Nov. 27, 2013 is used to active the selection strategy, which verifies whether the pair shall be selected. The data from Nov. 28, 2012 to Nov. 27, 2013 is used to train the thresholds in the trading strategy. The data from Nov. 28, 2013 to Nov. 27, 2014 is used to validate the thresholds.

Fitting Copulas

The stocks share high tail dependence can be selected to form the trading pairs. The parameters of Copulas and tail coefficients are shown in 4.1. The parameters of Copulas have passed K-S test at 95% confidence level.

Generally, the lower tail dependencies of those pairs are higher than their upper tail dependence, which shows a similar result to previous research [52]. In an economic perspective, the stocks listed in a different market should represent the same intrinsic value of this company. Thus the dependence structure between the return series of different markets shall share a strong relationship. Also, the stochastic

Table 4.1: Copulas coefficients and tail dependence coefficients

Pairs	α_{Cl}	α_{Gu}	$\lambda_{C_{Cl}^{lo}}$	$\lambda_{C_{Gu}^{up}}$
CMB	0.7997	1.5172	0.4203	0.4209
CCB	0.8234	1.4945	0.4309	0.4099
MB	0.6637	1.4167	0.3519	0.3686
BOCs	0.7393	1.4733	0.3916	0.3992
ABC	0.5557	1.3577	0.2873	0.3338
ICBC	0.6399	1.3972	0.3385	0.3577
BOC	0.1007	1.0275	0.0010	0.0368
CITIC	0.6077	1.3512	0.3196	0.3297

progress of prices of the same company should be similar. In addition, they may have a strong tail dependence. In 4.1, the tail dependence of the BOC pair is relatively smaller than the other pairs; thus, the BOC pair should be omitted.

Result in Training and Validation Phases

The result of the training phase can be found in 4.2. In the training, the threshold intervals should determined such that each pair can get relatively higher return in the training phase. If the optimal interval can be stable in different phases, then this

Table 4.2: Copulas-based two-sided-long-short strategy in the training Phase

Pairs	Optimal threshold interval	Return	success rate
CMB	[0.869,0.890]	[5.41%6.57%]	[54.55%57.63%]
CCB	[0.6660.687]	[2.90%3.60%]	[61.76%63.16%]
MB	[0.5380.554]	[20.39%21.86%]	[52.74%53.10%]
BOCs	[0.8130.823]	[8.40%9.84%]	[50.63%56.25%]
ABC	[0.9320.939]	[10.63%10.84%]	[56.25%57.58%]
ICBC	[0.9540.961]	0.30%	50.00%
BOC	-	-	-
CITIC	[0.8240.870]	[13.20%15.77%]	[47.14%48.68%]

strategy can be also stable in its operation. In the terms of the optimal threshold interval, those pairs whose return are higher than the SHIBOR in corresponding term, 4.40%, are the CMB pair, the MB pair, the BOCs pair, the ABC pair and the CITIC pair; in which, the returns of the ABC pair, the CITIC pair and the BOCs pair are higher than 10%. Table 4.3 shows the optimal threshold interval, return,

Table 4.3: Copulas-based two-sided-long-short strategy in the validation Phase

Pairs	Optimal threshold interval	Return	success rate
CMB	[0.692,0.697]	[8.98%10.23%]	[50.27%50.54%]
CCB	[0.6150.637]	[20.02%20.19%]	[51.85%53.85%]
MB	[0.4130.428]	[21.32%22.86%]	[53.13%53.74%]
BOCs	[0.7280.755]	[23.62%20.01%]	[52.26%53.78%]
ABC	[0.9270.940]	[17.06%19.56%]	[49.19%50.40%]
ICBC	[0.3770.457]	[19.33%23.87%]	[49.19%,50.40%]
BOC	-	-	-
CITIC	[0.8580.870]	[18.73%20.74%]	[48.39%49.47%]

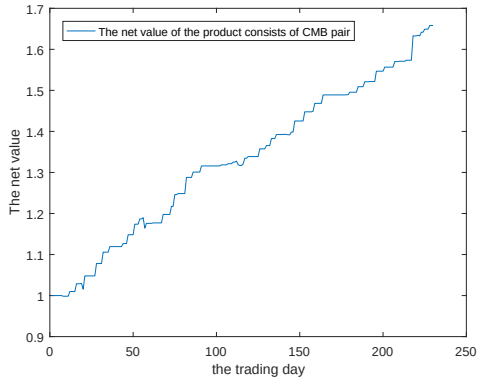
and the success rate in the unrefined tail dependence strategy.

4.4.2 Copulas-Based One-Sided-Long Strategy

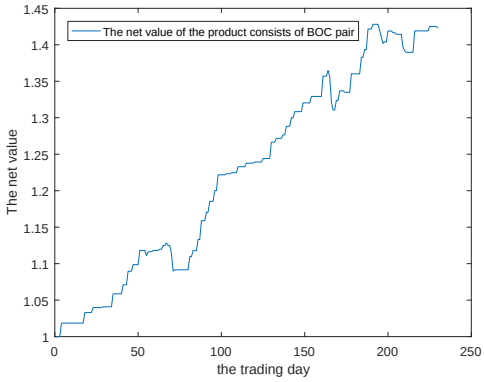
Table 4.4 shows these characters in three stages. The parameter stabilities are tested in three stages: the optimal parameters are found in the first stage and also tested in the second stages. The optimal parameters in the first stage and the second stage are also tested in the third stage. The first stage is the in-sample data. The optimal parameters found in the previous stage are verified in the current stage based on out-of-sample data. From Table 4.3, the CMB, BOC, BOCs, MB, and CITIC pairs show well parameter stabilities. Oppositely, the parameters of the ABC, ICBC, and CCB pairs change rapidly in different stages. The implication of the parameter stabilities is "more stable the parameters are, more stable the net value of the pair may be". This can be tested in the net values of different pairs in Figures 4.3 to 4.6. The curve of the net values of the CMB, BOC, BOCs, MB, and CITIC pairs are very steady. However, a jump of the net value curve of the ABC pair occurs in 4.4(b). The net value curves of the ICBC and the CCB pairs fluctuate rapidly in Figures 4.5(a) and 4.5(b). Thus the performance of this trading strategy is based on the parameter stabilities of the selected pairs. In Table 4.3, the return of Copulas-based strategy is higher than that of buy and hold strategy in all cases.

Table 4.4: Copulas-based single-long strategy empirical results

Pairs	First stage			Second stage			Third stage		
	O.P.	R.	B.&H.R	O.P.	R.	B.&H.R	O.P.	R.	B.&H.R.
CMB	(0.6, 0.8)	84.30	-20.81	(0.7, 0.6)	57.43	48.05	(0.7, 0.7)	121.61	-5.11
				(0.6, 0.8)	54.20		(0.6, 0.8)	99.25	
							(0.7, 0.6)	112.04	
BOC	(0.7, 0.7)	33.03	-0.22	(0.7, 0.7)	59.85	27.19	(0.6, 0.6)	86.10	-33.71
				(0.7, 0.7)	59.85		(0.7, 0.7)	72.24	
							(0.7, 0.7)	72.24	
BOCs	(0.7, 0.6)	166.30	12.47	(0.5, 0.7)	108.82	33.90	(0.5, 0.7)	294.44	59.69
				(0.7, 0.6)	73.11		(0.7, 0.6)	200.13	
							(0.5, 0.7)	294.44	
ABC	(0.5, 0.9)	20.26	-12.26	(0.5, 0.6)	54.02	48.98	(0.8, 0.5)	84.78	-21.46
				(0.5, 0.9)	38.60		(0.5, 0.9)	37.45	
							(0.5, 0.6)	84.51	
ICBC	(0.8, 0.8)	20.15	-16.61	(0.4, 0.9)	77.13	39.35	(0.6, 0.7)	105.57	-19.57
				(0.8, 0.8)	43.91		(0.8, 0.8)	66.92	
							(0.4, 0.9)	60.68	
CCB	(0.6, 0.8)	43.90	-16.49	(0.6, 0.9)	62.04	50.31	(0.6, 0.7)	134.47	-29.12
				(0.6, 0.8)	50.79		(0.6, 0.8)	55.13	
							(0.6, 0.9)	34.12	
MB	(0.6, 0.8)	74.32	-27.27	(0.5, 0.8)	71.57	46.47	(0.8, 0.6)	101.84	-16.14
				(0.6, 0.8)	54.90		(0.6, 0.8)	78.10	
							(0.5, 0.8)	80.61	
CITIC	(0.5, 0.8)	97.64	10.27	(0.7, 0.7)	118.44	35.41	(0.5, 0.8)	243.60	-27.35
				(0.5, 0.8)	95.13		(0.5, 0.8)	243.60	
							(0.7, 0.7)	189.64	



(a) The volatility of '5.HK'

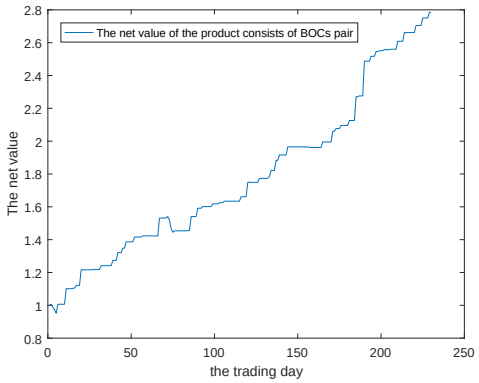


(b) The volatility of '939.HK'

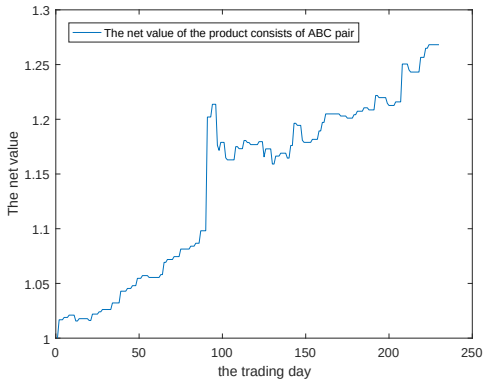
Figure 4.3: The net values of the CMB pair and the BOC pair

4.5 The Empirical Result: the Application of the Proposed Volatility Model

The empirical results show the outcome of adjustment of the weight of the portfolio guided by the FMVM in different schemes: the minimum-variance portfolio and the maximum-Sharpe-ratio portfolio. We assume we adjust our portfolio based on the forecasted optimal weights of the following trading day, where the forecasted optimal weights are produced by the portfolio management models corresponding with the forecasted volatility matrix. Then, this adjustment in advance can be tested by the

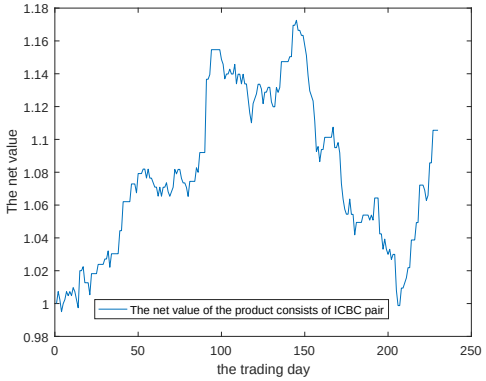


(a) The volatility of '1299.HK'

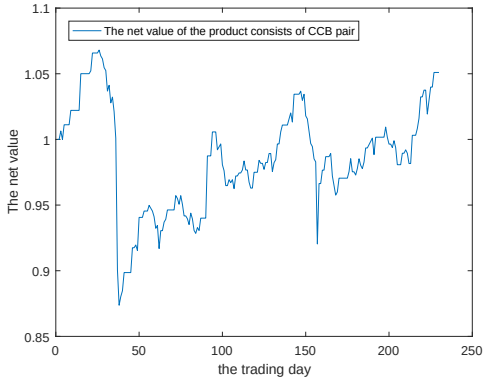


(b) The volatility of '941.HK'

Figure 4.4: The net values of the BOCs pair and the ABC pair



(a) The volatility of '1299.HK'

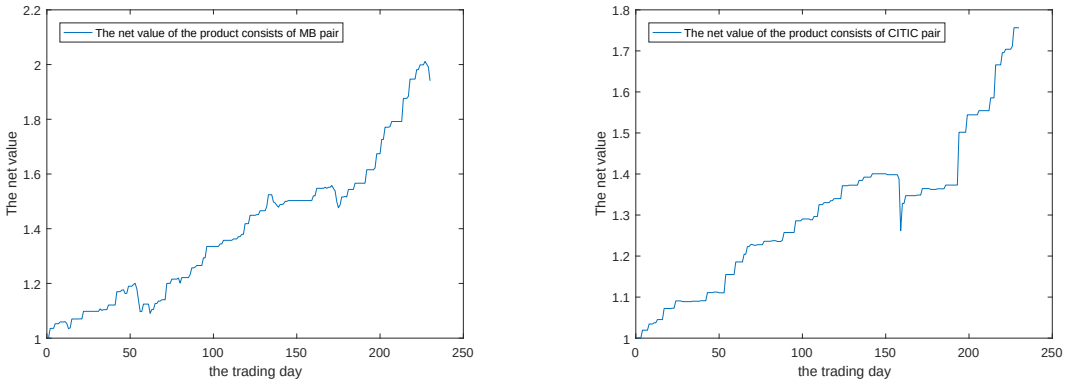


(b) The volatility of '941.HK'

Figure 4.5: The net values of the ICBC pair and the CCB pair

realized benchmark, which can be calculated in the following trading day.

We apply our FMVM to volatilty matrix forecasting. The individual models involved are $CCC(1,1)$, $CCC(1,2)$, $CCC(2,1)$, $CCC(2,2)$, $DCC(1,1)$, $DCC(1,2)$, $DCC(2,1)$, $DCC(2,2)$, $ADCC(1,1)$, $ADCC(1,2)$, $ADCC(2,1)$, and $ADCC(2,2)$. Given the rolling widows size 189, 20 volatility matrices of last 20 trading days are forecasted under the FMVM scheme. The volatility matrix forecasted on each trading day is applied to the minimum-variance portfolio and the maximum-Sharpe-ratio portfolio for portfolio management purpose. The return constraint in the minimum-variance portfolio is set to be current annualized return on decision day.



(a) The volatility of '1299.HK' (b) The volatility of '941.HK'

Figure 4.6: The net values of the MB pair and the CITIC pair

Table 4.5 and Table 4.8 show the outcomes of adjustment in advance under the minimum-variance portfolio and the maximum-Sharpe-ratio portfolio respectively. The benchmarks of these outcomes are given in 4.6 and Table 4.9. In these tables, "P.R." is shored for portfolio return. "P.V." is shorted for portfolio variance. "P.S.R." is shorted for portfolio Sharpe ratio. Under the minimum-variance portfo-

Table 4.5: A fusion of the minimum-variance portfolio and the FMVM

Day	W1	W2	W3	W4	W5	W6	W7	W8	P.R.	P.V.	P.S.R.
1	0.49	0.13	0.02	0.08	0.14	0.06	0.04	0.05	0.38	0.0011	10.90
2	0.29	0.20	0.05	0.10	0.13	0.05	0.08	0.08	0.38	0.0017	8.59
3	0.29	0.20	0.05	0.10	0.14	0.05	0.09	0.09	0.37	0.0018	8.28
4	0.31	0.21	0.06	0.11	0.13	0.05	0.09	0.02	0.37	0.0019	8.10
5	0.32	0.22	0.06	0.11	0.14	0.02	0.04	0.09	0.38	0.0019	8.25
6	0.29	0.20	0.06	0.11	0.12	0.05	0.08	0.09	0.38	0.0018	8.62
7	0.35	0.23	0.08	0.11	0.11	0.05	0.06	0.00	0.39	0.0021	8.10
8	0.33	0.20	0.05	0.11	0.10	0.06	0.07	0.07	0.40	0.0018	8.92
9	0.31	0.22	0.06	0.11	0.11	0.05	0.05	0.08	0.40	0.0020	8.66
10	0.26	0.23	0.06	0.13	0.13	0.03	0.08	0.09	0.41	0.0020	8.67
11	0.33	0.22	0.08	0.11	0.11	0.05	0.07	0.03	0.41	0.0020	8.78
12	0.25	0.22	0.07	0.11	0.12	0.04	0.09	0.09	0.42	0.0020	8.97
13	0.26	0.22	0.08	0.11	0.11	0.05	0.08	0.09	0.41	0.0020	8.78
14	0.25	0.22	0.07	0.11	0.12	0.05	0.09	0.09	0.42	0.0020	8.87
15	0.25	0.22	0.07	0.11	0.12	0.05	0.09	0.09	0.42	0.0020	8.92
16	0.25	0.22	0.08	0.11	0.11	0.05	0.09	0.09	0.42	0.0020	9.09
17	0.25	0.21	0.08	0.11	0.11	0.05	0.11	0.09	0.43	0.0019	9.22
18	0.28	0.23	0.10	0.10	0.09	0.06	0.12	0.02	0.45	0.0022	9.06
19	0.25	0.21	0.08	0.09	0.09	0.07	0.12	0.09	0.45	0.0020	9.74
20	0.26	0.21	0.09	0.10	0.09	0.07	0.10	0.09	0.44	0.0020	9.48

lio, the outcomes of the beforehand adjustment are shown in Table 4.5. The benchmarks are shown in Table 4.6. The returns of under beforehand adjustment are arranged from 37% to 45%, in contrast, its benchmarks are arranged from 51% to 54%. The variances under the beforehand adjustment are arranged from 0.11 to 0.21%, in contrast, its benchmarks are arranged from 0.20 to 0.21%. The Sharpe

Table 4.6: The realized results of the minimum-variance portfolio

Day	W1	W2	W3	W4	W5	W6	W7	W8	P.R.	P.V.	P.S.R.
1	0.25	0.24	0.06	0.13	0.11	0.03	0.10	0.08	0.41	0.0021	8.52
2	0.25	0.24	0.06	0.12	0.12	0.03	0.10	0.08	0.40	0.0020	8.36
3	0.25	0.24	0.06	0.12	0.12	0.03	0.10	0.08	0.40	0.0021	8.45
4	0.25	0.25	0.07	0.12	0.11	0.03	0.10	0.08	0.42	0.0021	8.66
5	0.25	0.25	0.07	0.12	0.11	0.03	0.10	0.08	0.41	0.0021	8.64
6	0.25	0.24	0.07	0.12	0.12	0.03	0.10	0.07	0.42	0.0021	8.66
7	0.25	0.24	0.07	0.12	0.11	0.03	0.10	0.07	0.42	0.0021	8.69
8	0.23	0.25	0.07	0.13	0.12	0.03	0.10	0.08	0.43	0.0022	8.73
9	0.23	0.25	0.07	0.13	0.12	0.03	0.10	0.08	0.43	0.0022	8.81
10	0.23	0.25	0.07	0.13	0.12	0.03	0.10	0.08	0.43	0.0022	8.86
11	0.23	0.24	0.08	0.13	0.12	0.03	0.09	0.07	0.42	0.0021	8.76
12	0.23	0.24	0.08	0.13	0.12	0.03	0.10	0.07	0.43	0.0021	8.83
13	0.22	0.24	0.08	0.13	0.12	0.03	0.10	0.07	0.43	0.0021	8.85
14	0.22	0.24	0.08	0.13	0.12	0.03	0.10	0.07	0.43	0.0021	8.91
15	0.22	0.24	0.07	0.13	0.13	0.03	0.10	0.07	0.43	0.0021	8.87
16	0.23	0.24	0.08	0.13	0.13	0.03	0.10	0.07	0.42	0.0021	8.84
17	0.23	0.24	0.08	0.13	0.12	0.03	0.10	0.07	0.43	0.0021	8.88
18	0.23	0.24	0.08	0.13	0.12	0.03	0.10	0.07	0.43	0.0021	8.88
19	0.23	0.24	0.08	0.13	0.12	0.03	0.10	0.07	0.43	0.0021	8.89
20	0.23	0.24	0.08	0.13	0.13	0.04	0.09	0.07	0.41	0.0020	8.76

ratios under the beforehand adjustment are arranged from 8.10 to 10.90, in contrast, its benchmarks are arranged from 8.36 to 8.91. In Table 4.7, the average of each attribute is given on the beforehand adjustment and its benchmark. The average return after adjustment is 1% smaller than that of the benchmark, where only 15% of the beforehand adjustment are no worse than the benchmark. The average risk variance after adjustment is 0.02% lower than the benchmark, and 90% of them outperforms the benchmark variances. The Sharpe ratio of the beforehand adjustment is 8.90 in the average, 0.16 higher than that of the benchmark. The Sharpe ratios of the beforehand adjustment are 55% better than that of the benchmark.

Table 4.7: The comparison of the beforehand adjustment to the benchmarks under the minimum-variance portfolio

	Avg. Ret.	Outperform Ratio	Avg. Var.	Outperform Ratio	Avg. S.R	Outperform Ratio
Beforehand	0.41	15%	0.0019	90%	8.90	55%
Benchmark	0.42	85%	0.0021	10%	8.74	45%

Under the maximum-Sharpe-ratio portfolio, the outcomes of the beforehand adjustment are shown in Table 4.8. The benchmarks are shown in Table 4.9. The returns of under beforehand adjustment are arranged from 49% to 53%, in the contrast, its benchmarks are arranged from 51% to 54%. The variances under the beforehand adjustment are arranged from 0.13 to 0.31%, in the contrast, its benchmarks are arranged from 0.27 to 0.31%. The Sharpe ratios under the beforehand adjustment are

Table 4.8: A fusion of the maximum-Sharpe-ratio portfolio and the FMVM

Day	W1	W2	W3	W4	W5	W6	W7	W8	P.R.	P.V.	P.S.R.
1	0.50	0.16	0.07	0.07	0.03	0.01	0.09	0.08	0.49	0.0013	12.88
2	0.38	0.20	0.11	0.08	0.00	0.00	0.13	0.10	0.50	0.0025	9.69
3	0.38	0.20	0.11	0.07	0.00	0.00	0.13	0.11	0.50	0.0025	9.57
4	0.41	0.22	0.12	0.08	0.00	0.00	0.14	0.03	0.50	0.0028	9.21
5	0.41	0.22	0.12	0.08	0.00	0.00	0.06	0.11	0.50	0.0027	9.23
6	0.38	0.20	0.12	0.08	0.00	0.00	0.12	0.11	0.51	0.0026	9.65
7	0.45	0.24	0.14	0.08	0.00	0.00	0.08	0.01	0.52	0.0031	8.95
8	0.41	0.19	0.11	0.07	0.00	0.00	0.12	0.10	0.52	0.0024	10.15
9	0.39	0.22	0.12	0.07	0.00	0.00	0.09	0.11	0.52	0.0027	9.64
10	0.32	0.22	0.13	0.08	0.01	0.00	0.13	0.11	0.53	0.0028	9.64
11	0.41	0.23	0.13	0.08	0.01	0.00	0.10	0.04	0.52	0.0028	9.46
12	0.31	0.22	0.13	0.08	0.02	0.00	0.13	0.11	0.53	0.0028	9.71
13	0.30	0.23	0.13	0.09	0.01	0.00	0.12	0.12	0.52	0.0028	9.45
14	0.30	0.23	0.13	0.08	0.01	0.00	0.14	0.11	0.53	0.0028	9.62
15	0.30	0.23	0.13	0.08	0.01	0.00	0.13	0.11	0.53	0.0028	9.62
16	0.30	0.21	0.13	0.08	0.02	0.00	0.14	0.11	0.53	0.0027	9.72
17	0.29	0.21	0.13	0.08	0.02	0.01	0.16	0.11	0.53	0.0027	9.81
18	0.31	0.23	0.14	0.09	0.03	0.04	0.15	0.02	0.50	0.0027	9.25
19	0.27	0.20	0.12	0.07	0.03	0.05	0.15	0.11	0.52	0.0024	10.03
20	0.29	0.21	0.12	0.07	0.02	0.05	0.13	0.11	0.51	0.0025	9.76

Table 4.9: The realized results of the maximum-Sharpe-ratio portfolio

Day	W1	W2	W3	W4	W5	W6	W7	W8	P.R.	P.V.	P.S.R.
1	0.31	0.25	0.11	0.10	0.00	0.00	0.14	0.09	0.51	0.0029	9.11
2	0.31	0.25	0.11	0.09	0.00	0.00	0.14	0.09	0.51	0.0029	8.99
3	0.31	0.25	0.11	0.09	0.00	0.00	0.14	0.10	0.51	0.0030	9.08
4	0.31	0.24	0.12	0.09	0.01	0.00	0.14	0.09	0.51	0.0029	9.14
5	0.31	0.25	0.12	0.09	0.01	0.00	0.13	0.09	0.52	0.0030	9.17
6	0.31	0.25	0.12	0.09	0.01	0.00	0.13	0.10	0.52	0.0030	9.23
7	0.30	0.25	0.12	0.09	0.01	0.00	0.13	0.10	0.52	0.0029	9.24
8	0.29	0.25	0.12	0.09	0.01	0.00	0.14	0.09	0.54	0.0031	9.30
9	0.29	0.25	0.12	0.09	0.03	0.00	0.13	0.09	0.53	0.0030	9.33
10	0.29	0.25	0.12	0.09	0.03	0.00	0.13	0.10	0.53	0.0030	9.36
11	0.29	0.24	0.13	0.09	0.02	0.00	0.13	0.09	0.53	0.0030	9.36
12	0.28	0.24	0.13	0.10	0.02	0.00	0.14	0.09	0.53	0.0029	9.39
13	0.28	0.24	0.13	0.09	0.02	0.00	0.14	0.09	0.53	0.0029	9.47
14	0.28	0.24	0.13	0.09	0.03	0.00	0.14	0.09	0.53	0.0029	9.49
15	0.28	0.24	0.13	0.09	0.02	0.00	0.14	0.09	0.53	0.0029	9.51
16	0.28	0.24	0.13	0.09	0.02	0.00	0.14	0.09	0.53	0.0029	9.45
17	0.28	0.24	0.12	0.09	0.03	0.00	0.14	0.09	0.52	0.0028	9.44
18	0.28	0.23	0.12	0.09	0.04	0.00	0.14	0.09	0.52	0.0028	9.41
19	0.28	0.23	0.13	0.09	0.04	0.00	0.14	0.09	0.52	0.0028	9.42
20	0.29	0.23	0.13	0.09	0.04	0.00	0.12	0.09	0.51	0.0027	9.40

arranged from 8.95 to 12.88, in the contrast, its benchmarks are arranged from 8.99 to 9.49. In Table 4.10, the average of each attribute is given with respect to the beforehand adjustment and its benchmark. The average return after adjustment is very close to that of the benchmark, where only 15% of the beforehand adjustment are no worse than the benchmark. The average risk variance after adjustment is lower than the benchmark and 90% of them outperforms the benchmark variances. Regarding to the Sharpe ratio, the average Sharpe ratio is 0.43 lager than the benchmark value. The beforehand adjustment can be better than the benchmark in 85% of time. Given

Table 4.10: The comparison of the beforehand adjustment to the benchmarks under the maximum-Sharpe-ratio portfolio

	Avg. Ret.	Outperform Ratio	Avg. Var.	Outperform Ratio	Avg. S.R	Outperform Ratio
Beforehand	0.52	15%	0.0026	95%	9.75	85%
Benchmark	0.52	85%	0.0029	5%	9.32	15%

these results, a beforehand adjustment, given forecasted volatility matrix produced by the FMVM and the maximum-Sharpe-ratio portfolio, can obtain a portfolio with better Sharpe ratio comparing to the benchmark. This application of the FMVM shows that our FMVM can help us to maintain a relatively lower risky portfolio.

Chapter 5

Conclusions and Future Work

This chapter draws conclusions on the thesis and points out some possible research directions related to the work done in this thesis.

5.1 Conclusions

The focus of the thesis has been placed on various multivariate volatility forecasting algorithms for a better volatility forecasting. Specifically, two research problems have been investigated in detail.

1. In this thesis, the FMVM is proposed to improve forecasting accuracy when developing multivariate volatility models. The proposed FMVM attempts to overcome the limitations of the existing AMVM which requires redundant computation on multivariate volatility models. The proposed FMVM classifies the individual volatility models into smaller scale clusters by using fuzzy C-means clustering algorithm. It then selects the most representative model with the lowest tracking error in each cluster and determines the optimal weight to each selected model. Based on the determined weights, some forecasting patterns are fed into the multivariate volatility model. Hence, repetitive but unnecessary computational effort can be avoided in the FMVM. The effectiveness of the proposed FMVM is evaluated based on 15 stocks from the Hang Seng

Index. The empirical results show that the FMVM can obtain better accuracy in volatility forecasting. Also, the number of volatility models involved in the FMVM is less than those required by the AMVM. Hence, a less computational effort is required by the FMVM than that of the AMVM.

2. The Hidden Markov Model (HMM) based forecasting algorithm is proposed to increase the forecasting accuracy, which can be used in both univariate and multivariate volatility forecasting. Attributes can be classified into recognition patterns. Based on the similarity of recognition pattern, the HMM-based forecasting algorithm can give forecasting results based on more information than other methods. Most volatility matrix forecasting methods are restricted to the dynamic processes of volatilities and covariances, whereas the HMM-based forecasting algorithm can include other attributes, for instance, trading volume, prices (open, high, low, close), some macroeconomic factors than volatiles and covariances. The HMM forecasting algorithm, considering other attributes, can obtain better prediction results than those generated by the FMVM.

5.2 Future Work

Related topics for the future research work are listed below.

1. The first orientation is described in Figure 5.1. Although the FMVM and the HMM forecasting algorithm are accurate forecasting techniques for univariate and multivariate forecasting in Chapter 2 and 3, the forecasting errors of the FMVM are relatively higher than that of the HMM forecasting algorithm and the forecasting delay is also relatively higher caused by the vacancy of the process description. However the FMVM and the HMM forecasting algorithms have complementary characteristics. Thus using some fuzzy rules and fuzzy logics one may coordinate those two algorithms to enhance the forecasting

accuracy. This fuzzy combination can be applied to volatility forecasting.

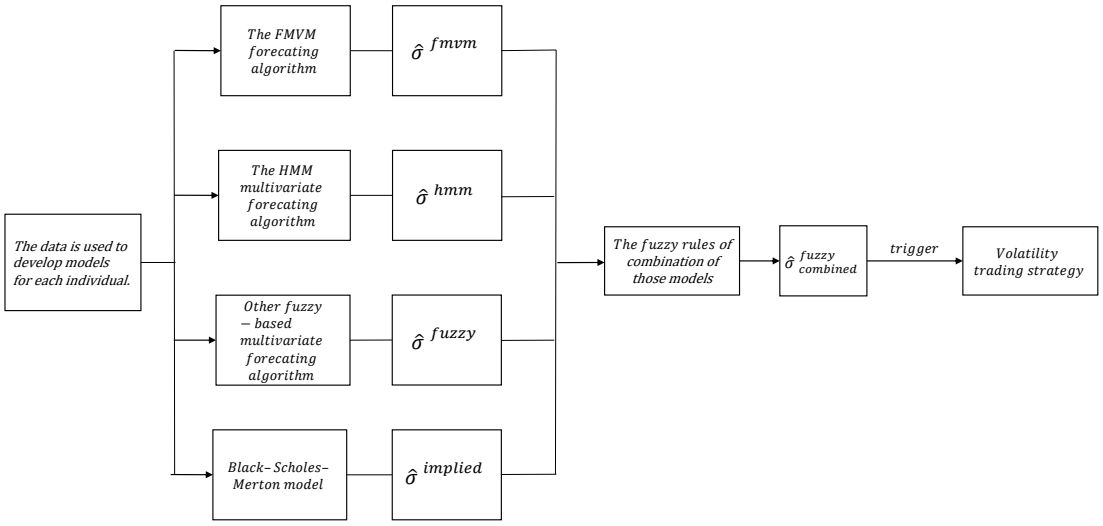


Figure 5.1: The future work of the first orientation

2. Regarding multivariate volatility matrices forecasting, the FMVM provides a dynamic process description with relatively higher forecasting errors. Meanwhile, the HMM forecasting forecasts as more accurate without sound process description. Therefore, in the future work, trying to figure out a more cooperative way to multivariate volatility forecasting is indeed meaningful. Such orientation is described in Figure 5.2.

3. Although the application and its empirical results are given in Chapter 4, the fuzzy fusion model is not involved. In the future, other fuzzy forecasting methods can be tested in investigating more in the relationship between allocation prediction and volatility forecasting. The results will be insightful for risk management and portfolio management. This research orientation is shown in Figure 5.3

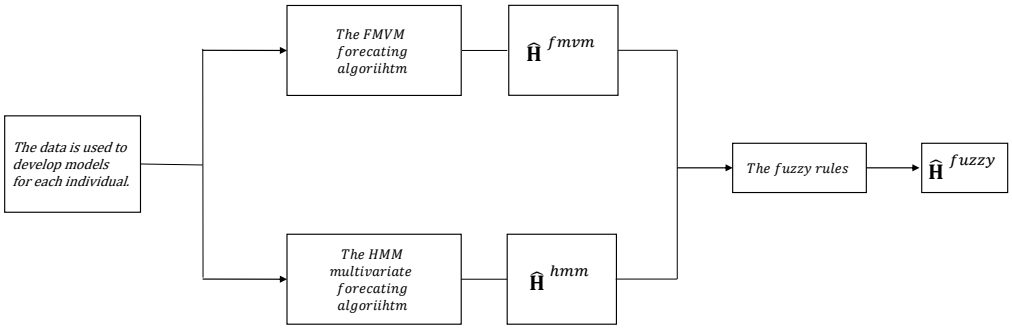


Figure 5.2: The mechanism of the second orientation

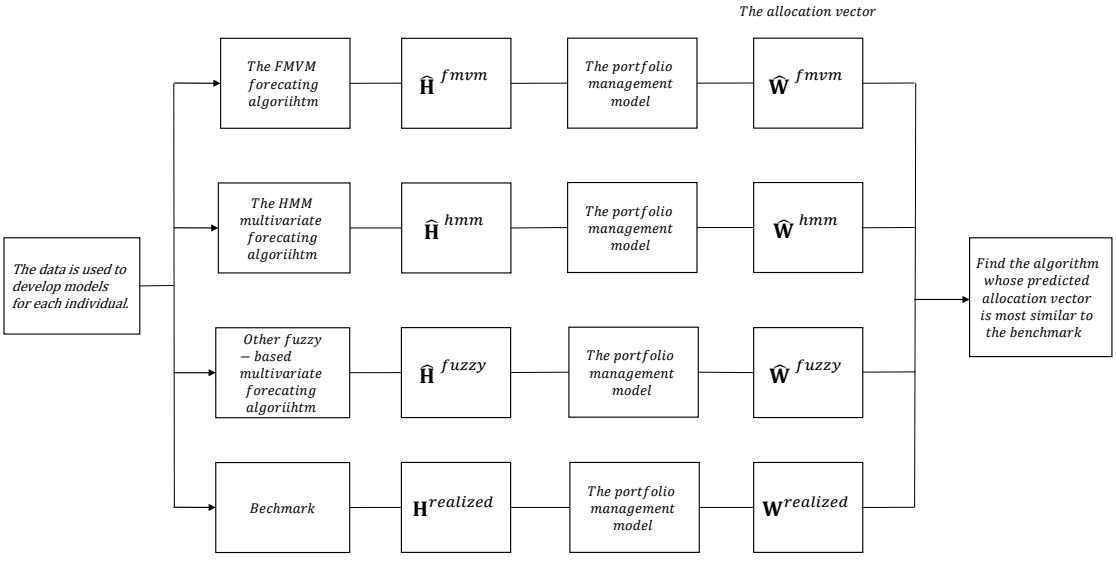


Figure 5.3: The mechanism of the third orientation

4. Add the buffering phenomenon to multivariate volatility models. The structure of volatilities are different for different structures of returns: the market trends and the extreme of returns will separate the volatility in different regimes. Only define the regime by thresholds will cause the regime unshifting problem. Current multivariate models are based on the description of the dynamic process of diagonal elements of volatility matrix, i.e., volatilities, and the description of the dynamic process of the linear correlation [30, 45, 54, 55, 29]. However, the buffering phenomenon is only considered in the univariate case. In univariate case, the buffered model can fit better than that of the classical models. It is reasonable to extend this buffering phenomenon to every diagonal entry of a volatility matrix. Further, we can use the Buffered form to modify the dynamic process of correlation terms. Thus, we can propose a new multivariate volatility model, a multivariate buffered autoregressive model with generalized autoregressive conditional heteroscedasticity.

Bibliography

[1] Alexander, C. (2001), "Orthogonal garch," *Mastering risk*, 2, 21–38.

[2] Andersen, T. G., Bollerslev, T., Diebold, F. X., and Labys, P. (2003), "Modeling and forecasting realized volatility," *Econometrica*, 71, 579–625.

[3] Batuwita, R. and Palade, V. (2010), "FSVM-CIL: fuzzy support vector machines for class imbalance learning," *IEEE Transactions on Fuzzy Systems*, 18, 558–571.

[4] Batuwita, R., Palade, V., and Bandara, D. C. (2011), "A customizable fuzzy system for offline handwritten character recognition," *International Journal on Artificial Intelligence Tools*, 20, 425–455.

[5] Bezdek, J. C., Ehrlich, R., and Full, W. (1984), "FCM: The fuzzy c-means clustering algorithm," *Computers & Geosciences*, 10, 191–203.

[6] Blunsom, P. (2004), "Hidden markov models," *Lecture notes, August*, 15, 18–19.

[7] Bollerslev, T. (1986), "Generalized autoregressive conditional heteroskedasticity," *Journal of econometrics*, 31, 307–327.

[8] Bollerslev, T. (1990), "Modelling the coherence in short-run nominal exchange rates: a multivariate generalized ARCH model," *The review of economics and statistics*, 72, 498–505.

[9] Cappiello, L., Engle, R. F., and Sheppard, K. (2006), "Asymmetric dynamics in the correlations of global equity and bond returns," *Journal of Financial econometrics*, 4, 537–572.

[10] Chen, S.-M. and Chang, Y.-C. (2010), "Multi-variable fuzzy forecasting based on fuzzy clustering and fuzzy rule interpolation techniques," *Information sciences*, 180, 4772–4783.

[11] Christoffersen, P. F. and Diebold, F. X. (2000), "How relevant is volatility forecasting for financial risk management?" *Review of Economics and Statistics*, 82, 12–22.

[12] Crook, J. and Moreira, F. (2011), "Checking for asymmetric default dependence in a credit card portfolio: A copula approach," *Journal of Empirical Finance*, 18, 728–742.

[13] Dalman, H. (2016a), "An interactive fuzzy goal programming algorithm to solve decentralized bi-level multiobjective fractional programming problem," *arXiv preprint arXiv:1606.00927*.

[14] Dalman, H. (2016b), "Uncertain programming model for multi-item solid transportation problem," *International Journal of Machine Learning and Cybernetics*, pp. 1–9.

[15] Dalman, H., Güzel, N., and Sivri, M. (2016), "A fuzzy set-based approach to multi-objective multi-item solid transportation problem under uncertainty," *International Journal of Fuzzy Systems*, 18, 716–729.

[16] Eddy, S. R. (1996), "Hidden markov models," *Current opinion in structural biology*, 6, 361–365.

[17] Engle, R. (2002), "Dynamic conditional correlation: A simple class of multivariate generalized autoregressive conditional heteroskedasticity models," *Journal of Business & Economic Statistics*, 20, 339–350.

[18] Engle, R. F. (1982), "Autoregressive conditional heteroscedasticity with estimates of the variance of United Kingdom inflation," *Econometrica: Journal of the Econometric Society*, 50, 987–1007.

[19] Engle, R. F. and Granger, C. W. (1987), "Co-integration and error correction: representation, estimation, and testing," *Econometrica: journal of the Econometric Society*, 55, 251–276.

[20] Engle, R. F. and Kroner, K. F. (1995), "Multivariate simultaneous generalized ARCH," *Econometric theory*, 11, 122–150.

[21] Gatev, E., Goetzmann, W. N., and Rouwenhorst, K. G. (2006), "Pairs trading: Performance of a relative-value arbitrage rule," *Review of Financial Studies*, 19, 797–827.

[22] Glosten, L. R., Jagannathan, R., and Runkle, D. E. (1993), "On the relation between the expected value and the volatility of the nominal excess return on stocks," *The journal of finance*, 48, 1779–1801.

[23] Hassan, M. R. (2009), "A combination of hidden Markov model and fuzzy model for stock market forecasting," *Neurocomputing*, 72, 3439–3446.

[24] Hassan, M. R. and Nath, B. (2005), "Stock market forecasting using hidden Markov model: a new approach," in *5th International Conference on Intelligent Systems Design and Applications, 2005. ISDA'05. Proceedings. 5th International Conference on*, pp. 192–196, IEEE.

[25] Hassan, M. R., Nath, B., and Kirley, M. (2007), "A fusion model of HMM, ANN and GA for stock market forecasting," *Expert Systems with Applications*, 33, 171–180.

[26] Hassan, M. R., Nath, B., Kirley, M., and Kamruzzaman, J. (2012), "A hybrid of multiobjective Evolutionary Algorithm and HMM-Fuzzy model for time series prediction," *Neurocomputing*, 81, 1–11.

[27] Hassan, R. (2007), "Hybrid HMM and soft computing modeling with applications to time series analysis," Ph.D. thesis, The University of Melbourne.

[28] Kat, H. M. (2003), "The dangers of using correlation to measure dependence," *The Journal of Alternative Investments*, 6, 54–58.

[29] Li, G., Guan, B., Li, W. K., and Yu, P. L. (2015), "Hysteretic autoregressive time series models," *Biometrika*, 102, 717–723.

[30] Liu, J., Li, W., and Li, C. (1997), "On a threshold autoregression with conditional heteroscedastic variances," *Journal of Statistical Planning and inference*, 62, 279–300.

[31] Liu, J., Yiu, K.-F. C., Siu, T. K., and Ching, W.-K. (2013), "Optimal investment-reinsurance with dynamic risk constraint and regime switching," *Scandinavian Actuarial Journal*, 2013, 263–285.

[32] Merton, R. C. (1980), "On estimating the expected return on the market: An exploratory investigation," *Journal of financial economics*, 8, 323–361.

[33] Murphy, J. J. (1999), *Technical analysis of the financial markets: A comprehensive guide to trading methods and applications*, Penguin.

[34] Nelson, D. B. and Cao, C. Q. (1992), "Inequality constraints in the univariate GARCH model," *Journal of Business & Economic Statistics*, 10, 229–235.

[35] Pathak, A. and Pal, N. R. (2016), "Clustering of mixed data by integrating fuzzy, probabilistic, and collaborative clustering framework," *International Journal of Fuzzy Systems*, 18, 339–348.

[36] Pesaran, M. H., Schleicher, C., and Zaffaroni, P. (2009), "Model averaging in risk management with an application to futures markets," *Journal of Empirical Finance*, 16, 280–305.

[37] Philippe, J. (2001), *Value at risk: the new benchmark for managing financial risk*, New York: McGraw-Hill Professional.

[38] Poritz, A. B. (1988), "Hidden Markov models: A guided tour," in *International Conference on Acoustics, Speech, and Signal Processing, 1988. ICASSP-88., 1988 International Conference on*, vol. 1, pp. 7–13, IEEE.

[39] Rabiner, L. R. (1989), "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, 77, 257–286.

[40] Serguieva, A. and Hunter, J. (2004), "Fuzzy interval methods in investment risk appraisal," *Fuzzy Sets and Systems*, 142, 443–466.

[41] Sheppard, K. (2009), "MFE MATLAB function reference financial econometrics," *Unpublished paper, Oxford University, Oxford. Available at: http://www. kevinsheppard. com/images/9/95/MFE_Toolbox_Documentation. pdf.*

[42] Sklar, M. (1959), *Fonctions de répartition à n dimensions et leurs marges*, Université Paris 8.

[43] Van Den Broek, K. (2014), "Long-term insurance products and volatility under the Solvency II Framework," *European Actuarial Journal*, 4, 315–334.

[44] Wang, S., Yiu, K., and Mak, K.-L. (2013), "Optimal inventory policy with fixed and proportional transaction costs under a risk constraint," *Mathematical and Computer Modelling*, 58, 1595–1614.

[45] Wong, C. and Li, W. (1997), "Testing for threshold autoregression with conditional heteroscedasticity," *Biometrika*, 84, 407–418.

[46] Xie, W. and Wu, Y. (2013), "Copula-based pairs trading strategy," in *Asian Finance Association (AsFA) 2013 Conference. doi*, vol. 10.

[47] Xie, W., Liew, R. Q., Wu, Y., and Zou, X. (2014), "Pairs trading with copulas," *The Journal of Trading*, 11, 41–52.

[48] Yiu, K., Wang, S., and Mak, K. (2008), "Optimal portfolios under a value-at-risk constraint with applications to inventory control in supply chains," *Journal of industrial and management optimization*, 4, 81–94.

[49] Yiu, K.-F. C. (2004), "Optimal portfolios under a value-at-risk constraint," *Journal of Economic Dynamics and Control*, 28, 1317–1334.

[50] Yiu, K.-F. C., Liu, J., Siu, T. K., and Ching, W.-K. (2010), "Optimal portfolios with regime switching and value-at-risk constraint," *Automatica*, 46, 979–989.

[51] Zhang, G., Cheng, K., Lu, F., and Shouyang, W. (2011), "Program trading strategy wtih Copula function," *System Engineering—Theory and Practice*, 31, 599–605.

[52] Zhang, K. and Wei, Z. (2014), "Copulas-based algorithmic trading strategy and its risk measurement," *Statistics and Decision*, 1, 56–58.

[53] Zhang, T., Chen, L., and Chen, C. P. (2016), "Clustering algorithm based on spatial shadowed fuzzy C-means and I-Ching operators," *International Journal of Fuzzy Systems*, 18, 609–617.

[54] Zhu, K., Yu, P. L., and Li, W. K. (2014), "Testing for the buffered autoregressive processes," *Statistica Sinica*, 24, 971–984.

[55] Zhu, K., Li, W. K., and Yu, P. L. (2015), "Buffered autoregressive models with conditional heteroscedasticity: An application to exchange rates," *Journal of Business & Economic Statistics*, 1, 1–15.