NUMERICAL INVESTIGATION OF MALAYSIAN STOCKS USING ORDER ONE UNIVERSAL PORTFOLIO STRATEGIES

PANG SOOK THENG

DOCTOR OF PHILOSOPHY IN SCIENCE

LEE KONG CHIAN FACULTYOF ENGINEERING AND SCIENCE
UNIVERSITI TUNKU ABDUL RAHMAN
SEPTEMBER 2017

NUMERICAL INVESTIGATION OF MALAYSIAN STOCKS USING ORDER ONE UNIVERSAL PORTFOLIO STRATEGIES

By

PANG SOOK THENG

A dissertation submitted to the Department of Mathematical and Actuarial Sciences,
Lee Kong Chian Faculty of Engineering and Science,
Universiti Tunku Abdul Rahman,
in partial fulfilment of the requirements for the degree of Doctor of Philosophy in Science
September 2017

ABSTRACT

NUMERICAL INVESTIGATION OF MALAYSIAN STOCKS USING ORDER ONE UNIVERSAL PORTFOLIO STRATEGIES

PANG SOOK THENG

Universal portfolio (UP) is an important investment strategy from a theoretical point of view due to Cover's research, which proves that UP can achieve very good return for a set of stocks in the long run. The exact algorithm for Cover's UP takes memory of order $O(n^{m-1})$ where n is the number of trading days and m is the number of stocks. It becomes useless for long term daily trading. To tackle the memory consumption issues, C. P. Tan in year 2013 proposed to limit UP to finite order. This research set the goal to promote active daily trading in Malaysia stock market using types of Tan's finite order UP and its generalisation.

The goal of our research is to study Malaysia stock market using the four proposed finite order universal portfolios comprehensively and refer to constant rebalanced portfolio (CRP) as a benchmark. The research is conducted with 95 stocks data selected from KLSE (Kuala Lumpur Stock Exchange) from 1 January 2000 to 31 December 2015. The UPs are Multinomial UP, Multivariate Normal UP, Brownian-motion generated UP and UP generated by Ornstein-Uhlenbeck process. This research limits to order one UP and 3 stocks, $m = 3$. Using the basic probability theory, we have formalised the four proposed finite order universal portfolio.

To test our proposed four finite order universal portfolios, they are first benchmarked against Cover's Dirichlet $(\alpha_1, \alpha_2, \alpha_3)$ universal portfolio strategy. Using 3 sets of stock data selected randomly from KLSE for the period from 1 January, 2003 to 30 November 2004, empirical result showed that the finite order universal portfolio can perform as good as Cover's universal portfolio with computationally

better in speed and memory.

For the comprehensive study of the above 95 selected stocks, we first find the good parameters for each proposed universal portfolios by using the combination of the 10 selected most active stocks from the above 95 selected stocks. From numerical experiment, the good performance of the parameters is identified for the purpose of further analysis.

Next, numerical experiment for the four classes of universal portfolio strategies are studied for short term (1 year period), middle term (4 years and 8 years periods) and long term (12 years and 16 years periods) data. We employed the four portfolio strategies with their best performing parameters identified to the above 5 groups of data to learn the performance of these methods and used CRP as a benchmark. Every 3 stocks data generated by combination of the selected 95 stocks are used for study. The empirical results showed that the performances of the proposed four strategies are better than CRP in 1 year and 4 years period groups. However, the four strategies did poorly in 8 years, 12 years and 16 years period groups. When comparing to KLCI, the result shows that finite order UP performs poorly.

The research indicates that the order one universal portfolio may help promote active online trading for Malaysia stock market in short term, only if the right combination of stocks are chosen.

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my supervisor, Dr. Chang Yun Fah and my co-supervisor, Dr. Liew How Hui, for their guidance, support, motivation, encouragement and advice. I thank my family and friends for supporting me throughout all my studies. Lastly, I would like to thank Dr. Tan Choon Peng for guiding me in the beginning of my research.

PANG SOOK THENG

APPROVAL SHEET

This dissertation entitled "**NUMERICAL INVESTIGATION OF MALAYSIAN STOCKS USING ORDER ONE UNIVERSAL PORTFOLIO STRATEGIES**" was prepared by PANG SOOK THENG and submitted as partial fulfilment for the requirements for the degree of Doctor of Philosophy in Science at Universiti Tunku Abdul Rahman.

Approved by:

____

(Supervisor: Dr Chang Yun Fah) Date: ....

Department of Mathematical and Actuarial Sciences

Lee Kong Chian Faculty of Engineering and Science

Universiti Tunku Abdul Rahman

____

(Co-Supervisor: Dr Liew How Hui) Date: ....

Department of Mathematical and Actuarial Sciences

Lee Kong Chian Faculty of Engineering and Science

Universiti Tunku Abdul Rahman

LEE KONG CHIAN FACULTY OF ENGINEERING AND SCIENCE

UNIVERSITI TUNKU ABDUL RAHMAN

Date:

SUBMISSION OF THESIS

It is hereby certified that ***Pang Sook Theng*** (ID No: ***09UED01975***) has completed this thesis entitled " ***Numerical Investigation of Malaysian Stocks using Order One Universal portfolio Strategies***" under the supervision of ***Dr. Chang Yun Fah*** (Supervisor) and ***Dr. Liew How Hui*** (Co-supervisor) from the Department of Mathematical and Actuarial Sciences, Lee Kong Chian Faculty of Engineering and Science.

I understand that University will upload softcopy of my thesis in pdf format into UTAR Institutional Repository, which may be made accessible to UTAR community and public.

Yours truly,

____

(Pang Sook Theng)

DECLARATION

I, PANG SOOK THENG hereby declare that the dissertation is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at UTAR or other institutions.

Signature : ______________________

Name : ______________________

Date : ______________________

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

CRP	Constant rebalanced portfolio
UP	Universal Portfolio
MUP	Multinomial Universal Portfolio
MVNUP	Multivariate Normal Universal Portfolio
BMUP	Brownian-motion Process Generated Universal Portfolio
OUUP	Ornstein Uhlenbeck Process Generated Universal Portfolio

CHAPTER 1

INTRODUCTION

Portfolio is a grouping of financial assets such as stocks, bonds and cash equivalents, as well as funds counterparts, including mutual, exchange-traded and closed funds. Portfolio selection, aiming to optimise the allocation of wealth across a set of assets, is a fundamental research problem in finance (Li & Hoi 2012).

There are two major philosophies in portfolio selection. One philosophy is the mean-variance theory, which assumes that each asset follows some normal distribution and an optimal portfolio selection can be obtained based on the tolerance of risk of an investor. Another philosophy is the capital growth theory, which focuses on optimising the expected cumulative wealth for a periodic (e.g. daily) portfolio selection.

The mean-variance theory is well-establised and widely used in practice. The capital growth theory, although less popular, is probably more meaningful to a developing country such as Malaysia, to promote active stock trading.

There are many portfolio selection strategies based on the capital growth theory. Li & Hoi (2012) has summarised the state-of-art portfolio selection algorithms as follows:

Classifications	Algorithm	Representative References
Benchmarks	Buy And Hold Best Stocks Constant Rebalanced Portfolios	 Kelly (1956); Cover (1991)
The winner	Universal Portfolios Exponential Gradient	Cover (1991); Cover & Ordentlich (1996) Helmbold et al. (1998)

The universal portfolio algorithm is a portfolio selection algorithm from the field of machine learning and information theory. The algorithm learns adaptively from historical data and maximizes log-optimal growth rate in the long run. The algorithm rebalances the portfolio at the beginning of each trading period. At the beginning of first trading period it starts with a naive diversification. In the following trading periods the portfolio composition depends on the historical total return of all possible Constant-Rebalanced Portfolios (CRP).

This research will investigate a variation of the universal portfolio proposed by Cover (1991) and the capital growth will be compared to Constant Rebalanced Portfolios, CRP as a benchmark.

In this chapter, we will first state the problem setting, then we will investigate the relevant research and literatures and finally, an outline of the thesis is given.

1.1 Problem Setting

Consider an m-stock market. Let $\mathbf{x}_n = (x_{n,i})$ be the stock-price-relative vector on the n^{th} trading day, where $x_{n,i}$ denotes the stock-price relative of stock i on day n, which is defined to be the ratio of the closing price to its opening price on day n, for $i = 1, 2, \cdots m$. We assume that $x_{n,i} \geq 0$ for all $i = 1, 2 \cdots m$ and $n = 1, 2 \cdots$. The portfolio vector $\mathbf{b}_n = (b_{n,i})$ on day n is the investment strategy used on day n when $b_{n,i}$ is the proportion of the current wealth invested on stock i for $i = 1, 2 \cdots m$, with $0 \leq b_{n,i} \leq 1$ and $\sum_{i=1}^{m} b_{n,i} = 1$.

The simplest strategy is to "buy-and-hold" stocks using some portfolio $\mathbf{b} = (b_1, b_2, \cdots, b_m)$, where $b_i \geq 0$, $\sum b_i = 1$. In particular, when $\mathbf{b} = (\frac{1}{m}, \cdots, \frac{1}{m})$, the buy-and-hold strategy is called a "uniform buy-and-hold". Suppose that the initial wealth is S_0, then the wealth accumulate at the end of the nth day is

$$S_n = S_0(b_1 x_{1,1} x_{2,1} \cdots x_{n,1} + b_2 x_{1,2} x_{2,2} \cdots x_{n,2} + \cdots + b_m x_{1,m} \cdots x_{n,m}) \quad (1.1)$$

We say that a portfolio selection algorithm "beats the market" when it outperforms uniform buy-and-hold strategy on a given market. Buy-and-hold strategy relies on the tendency of successful markets to grow. Much of modern portfolio theories focus on how to choose a good $\mathbf{b}$ for the buy-and-hold strategy, such as the famous Markowitz portfolio theory.

An alternative approach to the static buy-and-hold is to dynamically change the portfolio during the trading period. One of the example of this approach is constant rebalancing portfolio (CRP), namely, fix a portfolio $\mathbf{b}$ and reinvest your money each day according to $\mathbf{b}$. The paper presents constant rebalanced portfolio first introduced by Cover (1991). Under the assumption that the daily market vectors are observations of identically and independently distributed random variables, Cover (1991) showed that constant rebalanced portfolio performs at least as good as the best online portfolio selection algorithm. There are several weaknesses of this approach. One is that it is extremely hard to find the optimal weights and the second weakness is that in downtrend market CRP tends to trend down.

A *constant-rebalanced portfolio* (CRP) is a portfolio $\mathbf{b}_j = (\mathbf{b})$ that is constant over the trading days and the wealth at the end of nth trading day is

$$S_n = S_0 \prod_{j=1}^{n} \mathbf{b}^t \mathbf{x}_j. \tag{1.2}$$

Cover (1991) pointed out that the active trading strategy will perform better than "buy-and-hold" in the long run. An active trading strategy involves a sequence of universal portfolio vector $\mathbf{b}_n = (b_{n,1}, \cdots, b_{n,m})$, which are associated with the n trading days. Here, the value $b_{n,i,}$, $i = 1, 2, \cdots, m$ is the proportion of the wealth of the asset i on the nth day and $\sum_{i=1}^{m} b_{n,i} = 1$. The wealth at the end of n^{th} trading day S_n is given by

$$S_n = S_0 \prod_{j=1}^{n} \mathbf{b}_j^t \mathbf{x}_j. \tag{1.3}$$

The goal of our research is to study Malaysia stock market using the four proposed universal portfolios comprehensively and refer to CRP as a benchmark. The four universal portfolio strategies are investigated through a set of numerical experiments concerning 95 stocks data selected from Kuala Lumpur Stock Exchange.

1.2 Literature Review

The selection of assets in the portfolio is not unique which depends on the special characteristics of the market. The relation and movement of assets in the portfolio will directly affect the expected return and variance for analysis.

Portfolio theory was first developed mathematically by Markowitz (1952). Markowitz portfolio provides a method to analyse how good in a given portfolio is based on only the means and the variance of the returns of the assets contained in the portfolio. This led to the efficient frontier where the investor could choose his preferred portfolio depending on his risk preference. The Markowitz portfolio theory shows that the selected portfolio can achieve the goal of return by ensuing that it does not exceed the tolerance risk level of the investor. This idea is later known as the expected sector for expected return-variance rule (EV-rule). Sharpe (1963) extends Markowitz work on the portfolio analysis. A simplified model of the relationships among securities for practical application of the Markowitz portfolio analysis technique is provided by Sharpe (1963).

There is a new online learning algorithm for portfolio selection on alternative measure of price relative called Cyclically Adjusted Price Relative (CAPR) proposed by Nkomo & Kabundi (2013). The CAPR is derived from a simple state-space model of stock prices and have proved that the CAPR, unlike the standard raw price relative widely used in the machine literature, has well defined and desirable statistical properties that makes it better suited for nonparametric mean reversion strategies.

Agarwal et al. (2006) studied on-line investment algorithms first proposed by Agarwal & Hazan (2005) and extended by Hazan et al. (2006) which achieve almost the same wealth as the best constant-rebalanced portfolio determined in hindsight. These algorithms are the first to combine optimal logarithmic regret bounds with efficient deterministic computability and based on the Newton method for offline optimization.

Cover (1991) proposed the Universal Portfolio (UP) strategy and showed empirically that based on the New York Stock Exchange data over a 22 year period, this universal portfolio perform well on two- stock portfolios. Cover & Ordentlich (1996) further investigate universal portfolio and redefined the algorithm as μ-Weighted universal portfolio and study universal portfolio with side information using conditional probability. So far, side information is not easy to determine and was not proved to be much superior, therefore, this research will not consider side information.

Kalai & Vempala (2002) pointed out that the algorithm to calculate Cover & Ordentlich (1996)'s universal portfolio strategy accurately has time (or space) complexity of $O(n^{m-1})$, where n is the number of trading days and m is the number of stocks

Several studies are conducted with the objective of saving computation time and computer-memory space. The universal portfolio is first studied by Helmbold et al. (1998) in employing multiplicative update rule in the portfolio derived from a frame rule in Kivinen & Warmuth (1997). The algorithm is easier to be implemented and requires less computer memory for implementation compare to Cover's universal portfolio. Since the algorithm uses multiplicative updates on the weight distribution and does not involve any integration, it is more straightforward to implement than Cover's algorithm.

Next, Ishijima (2001) developed several methods for computing universal portfolios and conducted numerical experiments to show its possibility for the practical use. Ishijima applied Monte Carlo Methods for calculating universal portfolios. By the virtue of Monte Carlo, they are not time-consuming even if increasing the universe of assets. Besides, Ishijima also showed how to generate Dirichlet variates in the feasible region of constant portfolios, attention have been paid to the generation of uniform variates which is special case of Dirichlet Variates.

The recent study with the above objective is the finite-order universal portfolio introduced by Tan (2013). This finite order universal portfolio of order ν depends on the most recent ν days of the stock-price data, assuming that the assets are stocks.

Tan (2002) also studied the asymptotic performance of the parametric family of Cover-Ordentlich universal portfolios with respect to the best constant rebalanced portfolio. In a recent paper, Tan & Lim (2010) analysed the empirical performance of the parametric family of Helmbold universal portfolios. Tan & Pang (2011) compared the performances of the two parametric families of universal portfolios. Some empirical results are obtained based on some stock data sets from the local stock exchange and the results seem to indicate that the Cover-Ordentlich universal portfolio can outperform the Helmbold universal portfolio by a proper choice of parameters.

The performance of the Dirichlet universal portfolio can be improved by varying the parameter vector periodically after a fixed number of days, which is known as a trading day. After fixing a trading period, Tan & Pang (2012) showed that using a cyclic constrained-search algorithm, a new parametric vector is chosen that improve upon the wealth achieved in the previous period. They studied on some selected stock data sets from the local stock exchange and results shown that higher returns in wealth are achieved for the parameter-varying universal portfolio over the constant-parameter universal portfolio.

In Tan & Chu (2012), the investment performance of a pair of universal portfolios generated by the quadratic divergence with respect to a symmetric, positive definite matrix and its inverse is studied. They have selected three stock-data sets from Kuala Lumpur Stock Exchange and the empirical performance of the three-stock portfolios is presented. The empirical results showed that the use of complementary pairs of universal portfolios may increase the wealth of investors.

Tan & Lim (2013) have demonstrated another application of the mixture-current-run universal portfolio in extracting the best daily wealth due to best parameter from the same parametric family of universal portfolios. The empirical study showed that with the three stock data sets selected for the Kuala Lumpur Stock Exchange, the universal portfolio that using the best current-run parameter seem to outperform the original universal portfolio that using the best parameter from hindsight.

In Tan et al. (2015), a study of the empirical performance of the universal portfolios generated by certain reciprocal functions of the price relatives is presented. The portfolios are obtained from the zero-gradient sets of specific logarithmic objective functions containing the estimated daily growth rate of the investment wealth. The Malaysian companies selected for this empirical study consist of five stock data sets and trading period ranging form 1 March 2006 until 2 August 2012, consisting of a total of 1500 trading days.

Yeoh et al. (2010) studied the time-varying world integration of the Malaysia stock market using a Kalman Filter approach. They estimated the time-varying world integration of the Malaysia stock market and examined if the paths of the time-varying integration match the economic events of the country, where they employed weekly time series data for the period between February 1988 and September 2009 . The results showed that by using the Kalman Filter technique, the changes in the level of market integration coincided with the economic events that took place in the country and provided some evidence to the practical application and suitability of the Kalman Filter technique in studying stock market integration.

The relationship between the portfolio risk and the number of stocks in a portfolio for a given portfolio return across portfolios of the Malaysian stocks have been studied by Gupta et al. (2001). The Random Diversification Approach based on the Statman (1987) technique has been used in their study with a sample of 213 stocks traded on the Kuala Lumpur Stock Exchange(KLSE) are considered to form sets of portfolios.

Borodin et al. (2004) studied the novel algorithm for actively trading stocks.

The empirical results on historical markets provide strong evidence that this type of technical trading can beat the market and moreover can beat the best stock in the market.

From the above studies, especially the studies by Tan and his students, the dataset chosen for studying the performances of parameter for respective universal portfolios only involved small combinations of stocks and short term trading period from Malaysian stocks. Therefore, a comprehensive study of universal portfolio involve of large number of stocks, middle term and long term period will be carried out for investigation in our study.

1.3 Outline

This thesis mainly consists of seven chapters together with an introduction and conclusion. In Chapter 2, the data included in this study comprises of 95 stocks data collected from KLSE, whose was available from the Yahoo Finance historical data. The database contains of daily opening prices, daily closing prices, daily high and low, and the volume of transaction are collected .

In Chapter 3, we introduce the finite-order universal portfolios generated by two probability distributions and universal portfolios generated by stochastic process due to Tan (2013). This type of universal portfolio depends only on the positive moments of the generating probability distributions. The finite-order universal portfolio of order ν depends on the most recent ν days of the stock-price data, assuming that our assets are stocks. The constraint is there is no transaction cost in each portfolio. In order to reduce the extensive memory and long computational time, we choose the low order universal portfolio, where the order ν is a small integer, where ν = 1, 2, 3.

There are many programming languages for the implementation of the universal portfolio strategies. Python is a suitable choice because it is easy to use, powerful, and versatile. In Chapter 4, the algorithms of four proposed order one universal portfolios are written in Python code. Also, we used Dirichlet$(\alpha_1, \alpha_2, \alpha_3)$ universal portfolio strategy as a benchmark to compare with our four proposed universal portfolio strategies. The modified algorithm of Diriclet universal portfolio was obtained by Tan (2004b). We now chosen this modified algorithm for computing the three stock universal portfolio generated by Diriclhet$(\alpha_1, \alpha_2, \alpha_3)$ distribution where $\alpha_j > 0$ for $j = 1, 2, 3$. This modified algorithm is written in python using numpy module.

In Chapter 5, we present an experimental study of the four universal portfolios strategies, namely the finite order Multinomial Universal Portfolio, the finite order Multivariate Normal universal portfolio, universal portfolio generated by Brownian-motion and universal portfolio generated by Ornstein-Uhlenbeck processes. 10 most active stocks data are chosen among the 95 selected stocks from Kuala Lumpur Stock Exchange. The trading period is from 1 January 2000 to 31 December 2015. The above order one portfolio are run on 3 stock-price data sets generated by the combination of the 10 selected most active stocks. The wealths achieved after the n trading days by the above portfolio strategies is compare to the CRP strategies. The well performing parameters of the above four strategies are selected for further study in Chapter 6.

In Chapter 6, the overview of the analysis are carried out. The 95 stocks with 16 years of trading period divided into 5 groups, there are 1 year trading period (short term) ranging from 1 January 2015 to 31 December 2015, 4 years period group (middle term) started from 1 January 2012 to 31 December 2015, 8 years period group (middle term) ranging from 1 January 2008 to 31 December 2015, 12 years period group (long term) started from 1 January 2004 to 31 December 2015 and 16 years period group (long term) from 1 January 2000 to 31 December 2015. These 5 groups of data are run using the proposed four universal portfolio strategies with the well performing respective parameter obtained in Chapter 5. The wealth obtained is compared to its respective benchmark, CRP. In first section of Chapter 6, the price relatives for a given stock used is the ratio price defined in section Problem Setting in this chapter. In conclusion, the empirical results showed that the performances of the proposed four strategies are better than CRP in 1 year and 4 years period groups. However, the four strategies did poorly in 8 years, 12 years and 16 years period groups.

In order to compare to benchmark, KLCI, the price relatives for a given stock is redefined as the ratio of today closing price to the day before closing price which is discussed in Chapter 2. The wealth obtained is compared to its respective benchmark, CRP and KLCI. In conclusion, only around 25% of the results showed that the performances of the proposed four strategies are better than CRP in short term, middle term and long terms period investment. This concluded that the four proposed universal portfolio strategies did poorly when compared to KLCI.

CHAPTER 2

DATA COLLECTION

In this chapter, we explain how the data were obtained and processed. This research will work solely on the analysis of the stock data from Bursa Malaysia (formerly known as KLSE), which is introduced in Section 2.1. We then explain how we pick the stocks from Bursa Malaysia in Section 2.2 and how Python was used to process the stock data for later use in Section 2.3.

2.1 Bursa Malaysia

A National Bank (Bank Negara) report in 2005 indicated that compared to other exchange markets, Malaysia was ranked twenty-third in the world. Bursa Malaysia is a prominent centre of stock markets in South East Asia. It was previously known as Kuala Lumpur Stock Exchange (KLSE) from its founding in 1930 when the Singapore Stockbrokers' Association was set up as a formal organisation dealing in securities in Malaya (`http://www.bursamalaysia.com/corporate/about-us/corporate-history/`).

When Singapore broke away from Malaya federation in 1966 and became an independent country, the stock exchange remained its function until 1973, where the Stock Exchange of Malaysia and Singapore was divided into the Kuala Lumpur Stock Exchange Berhad (KLSE) and the Stock Exchange of Singapore

(SES) with the cessation of currency interchangeability between Malaysia and Singapore.

On 14 April, 2004, the KLSE was renamed Bursa Malaysia Berhad, following a demutualisation exercise, the purpose of which was to enhance competitive position and to respond to global trends in the exchange sector by making the Exchange more customer-driven and market-oriented.

On 18 March, 2005, Bursa Malaysia was listed on the Main Board of Bursa Malaysia Securities Berhad. On 6 July 2009, Bursa Malaysia introduced an enhancement of the Kuala Lumpur Composite Index (KLCI). The KLCI adopted the FTSE's global index standards and became known as the FTSE Bursa Malaysia KLCI. The FTSE Bursa Malaysia KLCI adopted the internationally accepted index calculation methodology to provide a more investable, tradable and transparent managed index. The constituents free float adjusted only by the investable portion included in the index calculation. With this new method the constituents of the *FTSE Bursa Malaysia KLCI*, also known as the *FBM KLCI*, shrunk from 100 to 30 companies to enhance the tradability of the index, while remaining representative of the market (Sinakalai & Suppayah 2011).

The FBM KLCI comprises thirty largest companies in Malaysia by full market capitalisation on the Bursa Malaysia's main board. When it was launched on the 6 July 2009, it replaced the old KLCI. It started at the closing value of the old KLCI on 3rd July 2009 and inherited the full history of the old KLCI index (Wikipedia 2016*b*).

2.2 Data Description

The KLCI was relatively new compare to famous index such as Dow Jones Industrial Average (`https://en.wikipedia.org/wiki/Dow_Jones_Industrial_Average`) and FTSE 100 Index (Wikipedia 2016*a*) which dated back to 1885 and 1984 respectively.

Since the FTSE 100 Index consists of 100 companies listed on the London Stock Exchange with the highest market capitalisation, we were trying to find 100 companies with the highest market capitalisation in Malaysia but from the 100 top listed companies in the main board of Bursa Malaysia given by (Capital n.d.). However, only 95 of them have data available from the Yahoo Finance (Yahoo n.d.). Five stocks listed in (Capital n.d.), i.e. JT International Bhd, IJM Land Bhd, UOA Development Bhd, Malaysia Airlines System Bhd and Kulim Malaysia Bhd were excluded because the historical data are not available in Yahoo Finance.

We fixed our data to begin from the beginning of the year 2000. However, not all the 95 selected stocks were listing since 2000. We will tabulate the data obtained from Yahoo Finance based on the category of the stock and the earliest trading date to 31 December 2015. The average volume is calculated as the average of volume of the earliest trading day (not earlier than 1 Jan 2000) to the volume of 31 December 2015.

Table 2.1: 95 Selected Stocks Based On The Category Of The Stock

Category	Code	Stock Name	Average Volume	Earliest Trading Date
Construction	5398	Gamuda Bhd	4986282	1 Jan 2000
	3336	IJM Coparation Bhd	3248926	31 Dec 2007
Consumer product	2836	Galsberg Brewery Malaysia Bhd	190800	1 Jan 2000
	3026	Dutch Lady Milk Industries Bhd	12200	1 Jan 2000
	3255	Guinness Anchor Bhd	166200	1 Jan 2000
	3689	Fraser and Neave Holdings Bhd	59650	1 Jan 2000
	3719	Penasonic Manufacturing Malaysia Bhd	116850	1 Jan 2000
	4006	Oriental Holdings Bhd	154100	1 Jan 2000
	4065	PPB Group Bhd	408600	1Jan 2000
	4162	Bristish American Tabacco (M)	323250	1 Jan 2000
	4405	Tan Chong Motor Holdings Bhd	273000	1 Jan 2000
	4588	UMW Holdings Bhd	691000	1 Jan 2000
	4707	Nestle (Malaysia) Bhd	61650	1 Jan 2000
	5131	Zhulian Corporation Bhd	26714150	27 Apr 2007
	7052	Padini Holdings Bhd	1059650	20 Sept 2006
	7084	QL Resources Bhd	2086150	31 Mar 2000

Table 2.1 (Continue) 95 Selected Stocks Based On The Category Of The Stock

Category	Code	Stock Name	Average Volume	Earliest Trading Date
Finance	1015	AMMB Holdings Bhd	3188500	1 Jan 2000
	1023	CIMB Group Holdings Bhd	6675600	1 Jan 2000
	1066	RHB Capital Bhd	3368950	1 Jan 2000
	1155	Malayan Banking Bhd	10407650	1 Jan 2000
	1171	Malaysia Buiding Society Bhd	1120150	1 Jan 2000
	1295	Public Bank Bhd	1687800	1 Jan 2000
	1818	Bursa Malaysia Bhd	51271350	18 Mar 2005
	2488	Alliance Financial Group Bhd	6770900	1 Jan 2000
	5053	OSK Holdings Bhd	574850	1 Jan 2000
	5139	Aeon Credit Service M. Bhd	10358700	12 Dec 2007
	5185	Affin Holdings Bhd	504450	1 Jan 2000
	5819	Hong Leong Bank Bhd	729700	1 Jan 2000
	6688	Hwang Capital(Malaysia) Bhd	242850	1 Jan 2000
	8621	PLI Capital Bhd	119950	1 Jan 2000
Hotel	5517	Shangri-La Hotels Malaysia Bhd	69100	1 Jan 2000
IPC	5031	Time Dotcom Bhd	630100	13 Mar 2001
	6947	DIGI Com Bhd	6094600	31 Dec 2007

Table 2.1 (Continue) 95 Selected Stocks Based On The Category Of The Stock

Category	Code	Stock Name	Average Volume	Earliest Trading Date
Industry Product	3026	Dutch Lady Milk Industries Bhd	1974250	1 Jan 2000
	4324	Shell Refining Company Bhd	50400	1 Jan 2000
	4383	Jaya Tiasa Holdings Bhd	7059050	1 Jan 2000
	5012	TA Ann Holdings Bhd	1291650	1 Jan 2000
	5183	Petronas Chemicaks Group Bhd	322414600	26 Nov 2010
	5168	Harta Lega Holdings Bhd	41772450	17 Apr 2008
	6033	Petronas Gas Bhd	598050	1 Jan 2000
	7106	Supermax	12656000	8 Aug 2000
	7113	Top Glove Corparation Bhd	4141700	27 Mar 2001
	7153	Kossan Rubber Industries Bhd	2929050	1 Jan 2000
Plantation	1899	Batu Kawan Bhd	2550	1 Jan 2000
	1961	IOI Corp. Bhd	7423350	1 Jan 2000
	2054	TDM Bhd	1261250	1 Jan 2000
	2089	United Plantation Bhd	34950	1 Jan 2000
	2216	IJM Plantation Bhd	10168800	2 July 2003
	2291	Genting Plantation Bhd	228700	1 Jan 2000
	2445	Kuala Lumpur Kepong Bhd	572400	1 Jan 2000
	2593	United Malacca Bhd	67300	31 Dec 2007
	2771	Boustead Holdings Bhd	332200	1 Jan 2000
	5027	Kim Loong Resources Bhd	415100	31 Dec 2007

Table 2.1 (Continue) 95 Selected Stocks Based On The Category Of The Stock

Category	Code	Stock Name	Average Volume	Earliest Trading Date
Plantation	5126	Sarawak Oil Palms Bhd	182300	31 Dec 2007
	5135	Sarawak Plantation Bhd	7823200	28 Aug 2007
	5138	Hap Seng Plantation Holdings Bhd	33316100	16 Nov 2007
	5222	Telda Global Ventures Holdings Bhd	133072250	28 June 2012
	9059	TSH Resources Bhd	556050	1 Jan 2000
Properties	1591	IGB Corparation Bhd	144800	21 Sept 2006
	2976	Wing Tai Malaysia Bhd	122250	1 Jan 2000
	5148	UEM Sunrise Bhd	9561550	1 Jan 2000
	5211	Sunway Bhd	3063150	24 Aug 2011
	8583	Mah Sing Group Bhd	314300	1 Jan 2000
	8664	SP Setia Bhd	742950	1 Jan 2000
REIST	5176	Sunway Real Estate Investment Trust	36566800	8 July 2010
	5180	Capital and Malaysia Mall Trust	6009700	16 July 2010
	5212	Pavillion Real Estate	98995300	7 Dec 2011
	5227	IGB Real Estate Investment Trust	118431450	21 Sept 2012
Trading Services	1562	Berjaya Sports Toto Bhd	1871300	1 Jan 2000
	2194	MMC Coparation Bhd	328200	1 Jan 2000
	3034	Hap Seng Consilidated Bhd	850200	1 Jan 2000
	3182	Genting Bhd	13869700	1 Jan 2000
	3859	magnum Bhd	2109600	31 Dec 2007
	4197	Sime Darby Bhd	1650850	1 Jan 2000

Table 2.1 (Continue) 95 Selected Stocks Based On The Category Of The Stock

Category	Code	Stock Name	Average Volume	Earliest Trading Date
Trading Sevices	4502	Media Prime Bhd	193950	1 Jan 2000
	4634	Pos Malaysia Bhd	9523550	1 Jan 2000
	4677	YTL Coparation Bhd	5907750	1 Jan 2000
	4863	Telekom Malaysia Bhd	4929150	1 Jan 2000
	5014	Malaysia Airports Holdings Bhd	1799650	1 Jan 2000
	5090	Media Chinese International Limited	134200	31 Dec 2007
	5099	Air Asia Bhd	94589600	22 Nov 2004
	5141	Dagang Enterprise Holdings Bhd	568050	3 Mar 2009
	5186	Malaysia Marine and Heavy Eng Holdings Bhd	71266600	29 Oct 2010
	5209	Gas Malaysia Bhd	43652950	6 Nov 2012
	5210	Bumi Armada Bhd	281719400	21 July 2011
	5218	Sapura kencana Petroleum Bhd	110011600	17 May 2012
	5225	IHH Healthcare Bhd	199803750	25 July 2012
	5347	Tenaga Nasional Bhd	5677850	31 Dec 2007
	5657	Pakson Holdings Bhd	465350	1 Jan 2000
	5681	Petronas Dagangan Bhd	301200	1 Jan 2000
	6012	Maxis Bhd	154892350	19 Nov 2009
	6351	Amway Malaysia Holdings Bhd	21350	1 Jan 2000
	6399	Astro Malaysia Holdings Bhd	6912200	31 Dec 2012
	6599	Aeon Co.(M) Bhd	650650	1 Jan 2000
	7277	Dialog Group Bhd	3811900	1 Jan 2000

To select good performing parameters for the universal portfolio strategies developed in Chapter 3, we will pick 10 most active (highest average volume) and long-term (having at least 10 years being listed in the stock exchange) stocks from each categories in the above table. This is possible for all categories except the REIST category, where all stocks are listed less than 10 years. Therefore, we picked two stocks from the trading services category and did not pick any stocks from the REIST category.

Table 2.2: 10 Most Active Stocks

Category	stock code	stock name	Average Volume	Earliest Trading Date
Construction	5398	Gamuda Bhd	4986282.33	1 Jan 2000
Consumer Product	7084	QL Resources Berhad	2086150	31 Mar 2000
Finance	1818	Bursa Malaysia Bhd	51271350	18 Jan 2005
Hotel	5517	Shangri-La Hotels Malaysia Bhd	69100	1 Jan 2000
Industry Products	7106	Supermax Corporation Bhd	12656000	1 Jan 2000
IPC	5031	Time Dotcom Bhd	630100	13 Mar 2001
Plantation	2216	IJM Plantation Bhd	10168800	2 July 2003
Properties	5148	UEM Sunrise Bhd	9561550	1 Jan 2000
Trading Services	5099	Air Asia Bhd	94589600	22 Nov 2004
Trading Services	3182	Genting Bhd	13869700	1 Jan 2000

2.3 Generating Price Ratios

The data of all the 95 stocks were downloaded from Yahoo Finance in CSV format (`https://en.wikipedia.org/wiki/Comma-separated_values`). Each stock consists the daily opening prices, the daily closing prices, the daily high and low, and the volume of transaction. However, what we are interested is the ratio of the stock price. We assume that it is possible to *buy at the opening* and *sell at the closing* and use the ratio of closing price to its opening price as price relative which we defined in problem setting in Chapter 1, to study the performance of the universal portfolio strategies we investigate in Chapter 3. We also assume that the opening price of day n is same as the closing price on day $n-1$ for the respective stock.

Since there are 95 stocks, the price ratio of each of the stock data will be calculated. For example, the price ratio for stock with code 5398 (GAMUDA, which belongs to the construction category) can be generated by running the following Python script.

```
import pandas as pd

df = pd.read_csv("CONSTRUCTION/5398GAMUDA.csv")
df['Date'] = pd.to_datetime(df['Date'],format="%m/%d/%y")
df['DailyROI'] = df['Close']/df['Open']
newdf = df[['Date','DailyROI']]
newdf.to_csv("construction new/5398roi.csv")
```

However, in order to compare the performances of four proposed strategies with benchmark KLCI which we will investigate in Chapter 6, the price relative is defined in problem setting (refer Chapter 1) are redefined as the ratio of closing price on day n to the closing price of day $n-1$, where the code in line number 5 are changed accordingly . If the stock splits or bonus issue was announced by the respective company, the stock's price has either increased or decreased sharply, the price relative will follow the ratio calculated as defined in problem setting of Chapter 1.

We then combine all the 95 files with all the price ratios for all 95 stocks (listed in the variable `stockslist`) into a large CSV file using the following Python script.

```
1  stockslist = [... list_of_csv_files ...]
2
3  import pandas as pd
4
5  maindf = pd.read_csv("IPC new/5031roi.csv")
6  maindf['Date'] = pd.to_datetime(maindf['Date'])
7  maindf = maindf.drop(maindf.columns[0],axis=1)
8  for thefile in stockslist:
9      df = pd.read_csv(thefile)
10     df['Date'] = pd.to_datetime(df['Date'])
11     df = df.drop(df.columns[0],axis=1)
12     df.columns = [df.columns[0], df.columns[1]
13        + thefile[:-4].replace(" new/","_")]
14     maindf = pd.merge(left=maindf,right=df,on="Date",how='outer')
15 maindf = maindf.sort(['Date'])
16 maindf.to_csv("stockslist.csv")
```

CHAPTER 3

ORDER ν UNIVERSAL PORTFOLIO STRATEGIES

In this chapter, we first simplified the formulation of μ- weighted universal portfolio Cover & Ordentlich (1996) and order ν universal portfolio Tan (2013) using probability theory and further extend to probability generated order ν universal portfolio to stochastic process based universal portfolio. We begin the section to generate the two order ν universal portfolio by probability distribution, namely Multinomial universal portfolio and Multivariate Normal universal portfolio, follow by two universal portfolio generated by stochastic process, they are universal portfolio generated by Brownian-motion and universal portfolio generated by Ornstein-Uhlenbeck process.

Consider a financial market with m stocks, the wealth of a fund is invested over the m shares in the market for a sequence of n days trading. Let $\mathbf{x}_n = (x_{n,1}, x_{n,2}, \cdots, x_{n,m})$ be the price relative vectors of the n^{th} trading day. Here, $x_{n,1}, x_{n,2}, \cdots, x_{n,m}$ are the ratio of the closing price to the opening price for m stocks respectively on n^{th} trading day.

Assuming that a portfolio is self-financing and no short is allowed then the variation of the wealth of the n^{th} trading day is related to the following factor

$$b_{n,1}x_{n,1} + b_{n,2}x_{n,2} + \cdots + b_{n,m}x_{n,m} \tag{3.1}$$

where $b_{n,1} + b_{n,2} + \cdots + b_{n,m} = 1$ and $0 \leq b_{n,i} \leq 1$ for $1 \leq i \leq m$.

Let S_n be the wealth at the n^{th} trading day and let S_0 be the initial wealth before the first trading day. By introducing the column vector notation $\mathbf{b}_n = (b_{n1}, \cdots, b_{n,m})$ from linear algebra (Goodaire 2003), (3.1) can be written as $\mathbf{b}_n^t\mathbf{x}_n$, where t denotes transpose. Therefore, the total wealth at the end of n^{th} trading day becomes

$$S_n = S_0\mathbf{b}_1^t\mathbf{x}_1\mathbf{b}_2^t\mathbf{x}_2\cdots\mathbf{b}_n^t\mathbf{x}_n \quad = S_0\prod_{j=1}^{n}\mathbf{b}_j^t\mathbf{x}_j \tag{3.2}$$

Universal portfolio strategy is a kind of online portfolio selection strategy (Li & Hoi 2012) and proposed in Cover (1991). Cover & Ordentlich (1996) redefined the universal portfolio strategy to μ-weigthted Universal Portfolio strategy where μ is a given distribution on the space of valid portfolios Δ_m (Li & Hoi 2012). The following are the wealth and update scheme used in Cover & Ordentlich (1996).

$$\mathbf{b}_{n+1} = \frac{\int_{\Delta_m}\mathbf{b}S_n(\mathbf{b})d\mu(\mathbf{b})}{\int_{\Delta_m}S_n(\mathbf{b})d\mu(\mathbf{b})} \tag{3.3}$$

$$S_n = \int_{\Delta_m}S_n(\mathbf{b})d\mu(\mathbf{b}) \tag{3.4}$$

where $S_n(\mathbf{b}) = \prod_{j=1}^{n}\mathbf{b}_j^t\mathbf{x}_j$.

Let $\mathbf{Y} = (Y_1, \cdots, Y_m)$ be an m-dimensional random vector such that $0 \leq Y_i \leq 1$ and $Y_1 + \cdots + Y_m = 1$. By using the notation of expected value (Wackerly et al. 2000) and replacing the random vector in (3.3) with $\mathbf{Y}$, it can be written as

$$\mathbf{b}_{n+1} = \frac{E[\mathbf{Y}\prod_{j=1}^{n}\mathbf{Y}^t\mathbf{x}_j]}{E[\prod_{j=1}^{n}\mathbf{Y}^t\mathbf{x}_j]} = \frac{E[\mathbf{Y}(\mathbf{Y}^t\mathbf{x}_n)(\mathbf{Y}^t\mathbf{x}_{n-1})\cdots(\mathbf{Y}^t\mathbf{x}_1)]}{E[(\mathbf{Y}^t\mathbf{x}_n)(\mathbf{Y}^t\mathbf{x}_{n-1})\cdots(\mathbf{Y}^t\mathbf{x}_1)]}. \tag{3.5}$$

The k component of vector $\mathbf{b}_{n+1}$ is

$$b_{n+1,k} = \frac{E[Y_k(\mathbf{Y}^t\mathbf{x}_n)(\mathbf{Y}^t\mathbf{x}_{n-1})\cdots(\mathbf{Y}^t\mathbf{x}_1)]}{E[(\mathbf{Y}^t\mathbf{x}_n)(\mathbf{Y}^t\mathbf{x}_{n-1})\cdots(\mathbf{Y}^t\mathbf{x}_1)]} \tag{3.6}$$

where $k = 1, 2, \cdots, m$.

3.1 Order ν Universal Portfolio

The idea of using a probability distribution to generate a universal portfolio was first proposed by Cover (1991). The Cover-Ordentlich universal portfolio (Cover & Ordentlich 1996) is a moving-order universal portfolio. This moving-order universal portfolio are not practical in the sense that as the number of stocks in the portfolio increases, the implementation time and the computer storage requirements grow exponentially fast. To improve on the time and memory storage performance, Tan (2013) proposed finite-order universal portfolio generated by some probability distribution. This type of universal portfolio depends only on the positive moments of the generating probability distribution. Our study only focus on Multinomial distribution generated universal portfolio and Multivariate normal distribution generated universal portfolio.

Tan (2013) introduced a variation of Cover's universal portfolio to deal with the inefficiency of Cover's universal portfolio. We will call it a *Probability-Based Order-ν Universal Portfolio*. In contrast to Cover, Tan (2013) introduced a random vector such that $Y_i > 0$ and define

$$b_{n+1} = \frac{E[(\mathbf{Y}(\mathbf{Y}^t\mathbf{x}_n)(\mathbf{Y}^t\mathbf{x}_{n-1})\cdots(\mathbf{Y}^t\mathbf{x}_{n-\nu+1})]}{E[(Y_1 + Y_2 + \cdots + Y_m)(\mathbf{Y}^t\mathbf{x}_n)(\mathbf{Y}^t\mathbf{x}_{n-1})\cdots(\mathbf{Y}^t\mathbf{x}_{n-\nu+1})]} \tag{3.7}$$

For k component, (3.7) can be written as

$$b_{n+1,k} = \frac{E[(Y_k(\mathbf{Y}^t\mathbf{x}_n)(\mathbf{Y}^t\mathbf{x}_{n-1})\cdots(\mathbf{Y}^t\mathbf{x}_{n-\nu+1})]}{E[(Y_1+Y_2+\cdots+Y_m)(\mathbf{Y}^t\mathbf{x}_n)(\mathbf{Y}^t\mathbf{x}_{n-1})\cdots(\mathbf{Y}^t\mathbf{x}_{n-\nu+1})]} \tag{3.8}$$

First, the wealth formula (3.2) is written as

$$S_n = (\mathbf{b}_n^t\mathbf{x}_n)(\mathbf{b}_{n-1}^t\mathbf{x}_{n-1})\cdots(\mathbf{b}_{n-\nu+1}^t\mathbf{x}_{n-\nu+1})S_{n-\nu} \tag{3.9}$$

when $\nu = 1$,

$$S_n = (\mathbf{b}_n^t\mathbf{x}_n)\cdot S_{n-1} \tag{3.10}$$

when $\nu = 2$,

$$S_n = (\mathbf{b}_n^t\mathbf{x}_n)(\mathbf{b}_{n-1}^t\mathbf{x}_{n-1})\cdot S_{n-2} \tag{3.11}$$

Although this method eliminates some inefficiency in Cover's universal portfolio, it has bootstrap issues. We pick simple constant rebalanced portfolio to bootstrap $S_0, S_1, \cdots, S_\nu$.

3.1.1 General Algorithm

The algorithm for computing (3.8) can be defined by expanding the numerator expression as follows:

$$
\begin{aligned}
&E[Y_k(\mathbf{Y}^t\mathbf{x}_n)\cdots(\mathbf{Y}^t\mathbf{x}_{n-\nu+1})] \\
&= E[Y_k \prod_{j=n-\nu+1}^{n}(\sum_{i=1}^{m} Y_i x_{j,i})] \\
&= E[Y_k \sum_{i_1=1}^{m}\sum_{i_2=1}^{m}\cdots\sum_{i_\nu=1}^{m} Y_{i_1}\cdots Y_{i_\nu} x_{n,i_1}\cdots x_{n-\nu+1,i_\nu}] \\
&= \sum_{i_1=1}^{m}\sum_{i_2=1}^{m}\cdots\sum_{i_\nu=1}^{m} x_{n,i_1}\cdots x_{n-\nu+1,i_\nu} E[Y_k Y_{i_1}\cdots Y_{i_\nu}]
\end{aligned}
\tag{3.12}
$$

The expanding of denominator of (3.8) is as follow:

$$
\begin{aligned}
&E[(Y_1+\cdots+Y_m)(\mathbf{Y}^t\mathbf{x}_n)\cdots(\mathbf{Y}^t\mathbf{x}_{n-\nu+1})] \\
&= \sum_{j=1}^{m} E[Y_j(\mathbf{Y}^t\mathbf{x}_n)\cdots(\mathbf{Y}^t\mathbf{x}_{n-\nu+1})] \\
&= \sum_{j=1}^{m}\sum_{i_1=1}^{m}\sum_{i_2=1}^{m}\cdots\sum_{i_\nu=1}^{m} x_{n,i_1}\cdots x_{n-\nu+1,i_\nu} E[Y_j Y_{i_1}\cdots Y_{i_\nu}] \\
&= \sum_{i_1=1}^{m}\sum_{i_2=1}^{m}\cdots\sum_{i_\nu=1}^{m} x_{n,i_1}\cdots x_{n-\nu+1,i_\nu} \sum_{j=1}^{m} E[Y_j Y_{i_1}\cdots Y_{i_\nu}]
\end{aligned}
\tag{3.13}
$$

when $\nu = 1$, (3.8) becomes

$$
b_{n+1,k} = \frac{\sum_{i=1}^{m} x_{n,i} E[Y_k Y_i]}{\sum_{i=1}^{m} x_{n,i}(\sum_{j=1}^{m} E[Y_j Y_i])}
\tag{3.14}
$$

When $\nu = 1$, $m = 3$, let the denominator be ζ, then

$$
\begin{aligned}
\zeta &= x_{n,1}E[Y_1Y_1] + x_{n,2}E[Y_1Y_2] + x_{n,3}E[Y_1Y_3] \\
&+ x_{n,1}E[Y_2Y_1] + x_{n,2}E[Y_2Y_2] + x_{n,3}E[Y_2Y_3] \\
&+ x_{n,1}E[Y_3Y_1] + x_{n,2}E[Y_3Y_2] + x_{n,3}E[Y_3Y_3]
\end{aligned}
\tag{3.15}
$$

The (3.14) becomes

$$b_{n+1,k} = \zeta^{-1}(x_{n,1}E[Y_kY_1] + x_{n,2}E[Y_kY_2] + x_{n,3}E[Y_kY_3]) \tag{3.16}$$

for $k = 1, 2, 3$, and the wealth of day $n + 1$ is defined as

$$S_{n+1} = (\mathbf{b}_{n+1}^t \mathbf{x}_{n+1})S_n \tag{3.17}$$

When $\nu = 2$, (3.8) becomes

$$b_{n+1,k} = \frac{x_{n,1}x_{n-1,1}E[Y_kY_1Y_1] + x_{n,1}x_{n-1,2}E[Y_kY_1Y_2] + \cdots + x_{n,m}x_{n-1,m}E[Y_kY_1Y_m]}{\sum_{j=1}^{m}(x_{n,1}x_{n-1,1}E[Y_jY_1Y_1] + x_{n,1}x_{n-1,2}E[Y_jY_1Y_2] + \cdots + x_{n,m}x_{n-1,m}E[Y_jY_1Y_m])} \tag{3.18}$$

and $\nu = 2$ and $m = 3$, let the denominator be ζ, then

$$\begin{aligned}
\zeta = (&x_{n,1}x_{n-1,1}E[Y_1Y_1Y_1] + x_{n,1}x_{n-1,2}E[Y_1Y_1Y_2] + x_{n,3}x_{n-1,3}E[Y_1Y_1Y_3] \\
&+ x_{n,2}x_{n-1,1}E[Y_1Y_2Y_1] + x_{n,2}x_{n-1,2}E[Y_1Y_2Y_2] + x_{n,2}x_{n-1,3}E[Y_1Y_2Y_3] \\
&+ x_{n,3}x_{n-1,1}E[Y_1Y_3Y_1] + x_{n,3}x_{n-1,2}E[Y_1Y_3Y_2] + x_{n,3}x_{n-1,3}E[Y_1Y_3Y_3] \\
&+ x_{n,1}x_{n-1,1}E[Y_2Y_1Y_1] + x_{n,1}x_{n-1,2}E[Y_2Y_1Y_2] + x_{n,3}x_{n-1,3}E[Y_2Y_1Y_3] \\
&+ x_{n,2}x_{n-1,1}E[Y_2Y_2Y_1] + x_{n,2}x_{n-1,2}E[Y_2Y_2Y_2] + x_{n,2}x_{n-1,3}E[Y_2Y_2Y_3] \\
&+ x_{n,3}x_{n-1,1}E[Y_2Y_3Y_1] + x_{n,3}x_{n-1,2}E[Y_2Y_3Y_2] + x_{n,3}x_{n-1,3}E[Y_2Y_3Y_3] \\
&+ x_{n,1}x_{n-1,1}E[Y_3Y_1Y_1] + x_{n,1}x_{n-1,2}E[Y_3Y_1Y_2] + x_{n,3}x_{n-1,3}E[Y_3Y_1Y_3] \\
&+ x_{n,2}x_{n-1,1}E[Y_3Y_2Y_1] + x_{n,2}x_{n-1,2}E[Y_3Y_2Y_2] + x_{n,2}x_{n-1,3}E[Y_3Y_2Y_3] \\
&+ x_{n,3}x_{n-1,1}E[Y_3Y_3Y_1] + x_{n,3}x_{n-1,2}E[Y_3Y_3Y_2] + x_{n,3}x_{n-1,3}E[Y_3Y_3Y_3]
\end{aligned} \tag{3.19}$$

and (3.18) becomes

$$
\begin{aligned}
b_{n+1,k} &= \zeta^{-1}(x_{n,1}x_{n-1,1}E[Y_kY_1Y_1] + x_{n,1}x_{n-1,2}E[Y_kY_1Y_2] \\
&+ x_{n,3}x_{n-1,3}E[Y_kY_1Y_3] + x_{n,2}x_{n-1,1}E[Y_kY_2Y_1] \\
&+ x_{n,2}x_{n-1,2}E[Y_kY_2Y_2] + x_{n,2}x_{n-1,3}E[Y_kY_2Y_3] \\
&+ x_{n,3}x_{n-1,1}E[Y_kY_3Y_1] + x_{n,3}x_{n-1,2}E[Y_kY_3Y_2] \\
&+ x_{n,3}x_{n-1,3}E[Y_kY_3Y_3])
\end{aligned}
\tag{3.20}
$$

for $k = 1, 2, 3$ and the wealth of day $n + 1$ is define as

$$S_{n+1} = (\mathbf{b}_{n+1}^t\mathbf{x}_{n+1})(\mathbf{b}_n^t\mathbf{x}_n)S_{n-1} \tag{3.21}$$

When $\nu = 3$, (3.8) becomes

$$b_{n+1,k} = \frac{\sum_{i_1=1}^m \sum_{i_2=1}^m \sum_{i_3=1}^m x_{n,i_1}x_{n-1,i_2}x_{n-2,i_3}E[Y_KY_{i_1}Y_{i_2}Y_{i_3}]}{\sum_{j=1}^m(\sum_{i_1=1}^m \sum_{i_2=1}^m \sum_{i_3=1}^m x_{n,i_1}x_{n-1,i_2}x_{n-2,i_3}E[Y_jY_{i_1}Y_{i_2}Y_{i_3}]})\tag{3.22}$$

The expansion of the algorithm for order three are similar to order one and two. When order increases, the algorithm will be changing accordingly as above and become larger due to the increasing of number of moments.

The wealth of day $n + 1$ is defined as

$$S_{n+1} = (\mathbf{b}_{n+1}^t\mathbf{x}_{n+1})(\mathbf{b}_n^t\mathbf{x}_n)(\mathbf{b}_{n-1}^t\mathbf{x}_{n-1})S_{n-2} \tag{3.23}$$

3.1.2 Multinomial Universal Portfolio (MUP)

In Wackerly et al. (2000), when the random vector $(Y_1, Y_2, ..., Y_m)$ has a joint multinomial distribution with parameters $N, p_1, p_2, \cdots, p_{m-1}$ where $0 < p_i < 1$ for $i = 1, 2, \cdots, m-1$ and $0 < p_m = 1 - p_1 - p_2 - \cdots - p_{m-1} < 1$; N is a positive integer larger than the number of stocks in the market, m. Consider joint probability function $f(y_1, y_2, \cdots, y_m)$ of Multinomial distribution :

$$f(y_1, y_2, \cdots y_m) = \begin{pmatrix} & N & & \\ y_1 & y_2 & \cdots & y_m \end{pmatrix} p_1^{y_1} p_2^{y_2} \cdots p_m^{y_m} \tag{3.24}$$

where the multinomial coefficient

$$\begin{pmatrix} & N & & \\ y_1 & y_2 & \cdots & y_m \end{pmatrix} = \frac{N!}{y_1! y_2! \cdots y_m!} \tag{3.25}$$

and $y_i = 0, 1, 2, \cdots, N$ for $i = 1, 2, ..., m$ subject to $y_1 + y_2 + \cdots + y_m = N$. The $E[Y_1 \cdots Y_m]$ in Section 3.1.1 can be found from the joint moment generating function.

$$M(\tau_1, \tau_2 \cdots, \tau_m) = (p_1 e^{t_1} + p_2 e^{t_2} + \cdots + p_m e^{t_m})^N \tag{3.26}$$

When $\nu = 1$, the $E[Y_k Y_{i_1} \cdots Y_{i_\nu}]$ in (3.12) can replace by the following moments:

$$
\begin{aligned}
&E(Y_k) = Np_k \\
&E(Y_k^2) = N(N-1)p_k^2 + Np_i \\
&E(Y_kY_i) = N(N-1)p_kp_i \qquad \text{for} \quad i \neq k
\end{aligned} \tag{3.27}
$$

When $\nu = 2$, the $E[Y_kY_{i_1} \cdots Y_{i_\nu}]$ in (3.12) can be replace by the following moments:

$$
\begin{aligned}
&E(Y_k^2) = N(N-1)p_k^2 + Np_k \\
&E(Y_kY_i) = N(N-1)p_kp_i && \text{for} \quad k \neq i \\
&E(Y_k^3) = N(N-1)(N-2)p_k^3 + 3N(N-1)p_k^2 + Np_k \\
&E(Y_k^2Y_i) = N(N-1)(N-2)p_k^2p_i + N(N-1)p_kp_i && \text{for} \quad k \neq i \\
&E(Y_kY_iY_j) = N(N-1)(N-2)p_kp_ip_j, && \text{for} \quad k \neq i, k \neq j, i \neq j
\end{aligned} \tag{3.28}
$$

When $\nu = 3$, the $E[Y_kY_{i_1} \cdots Y_{i_\nu}]$ in (3.12) can replace by the following moments:

$$
\begin{aligned}
&E(Y_k^3) = N(N-1)(N-2)p_k^3 + 3N(N-1)p_k^2 + Np_k \\
&E(Y_k^2 Y_i) = N(N-1)(N-2)p_k^2 p_i + N(N-1)p_k p_i && \text{for} \quad k \neq i \\
&E(Y_k Y_i Y_j) = N(N-1)(N-2)p_k p_i p_j && \text{for} \quad k \neq i, k \neq j, i \neq j \\
&E(Y_k^4) = N(N-1)(N-2)(N-3)p_k^4 + 6N(N-1)(N-2)(N-3)p_k^3 \\
&+ 3N(N-1)P_k^2 + Np_k \\
&E(Y_k^3 Y_i) = N(N-1)(N-2)(N-3)p_k^3 p_i + 3N(N-1)(N-2)p_k^2 p_i \\
&+ N(N-1)p_k p_i && \text{for} \quad k \neq i \\
&E(Y_k^2 Y_i^2) = N(N-1)(N-2)(N-3)p_k^2 p_i^2 + N(N-1)(N-2)p_k^2 p_i \\
&+ N(N-1)(N-2)p_k p_i^2 + N(N-1)p_k p_i && \text{for} \quad k \neq i \\
&E(Y_k^2 Y_i Y_j) = N(N-1)(N-2)(N-3)p_k^2 p_i p_j \\
&+ N(N-1)(N-2)p_k p_i p_j && \text{for} \quad (i,j,k) distinct \\
&E(Y_k Y_i Y_j Y_l) = N(N-1)(N-2)(N-3)p_k p_i p_j p_l && \text{for} \quad (i,j,k,l) distinct.
\end{aligned}
\tag{3.29}
$$

In particular, when $m = 3$, order =1, $k = 1, 2, 3$ and $i = 1, 2, 3$, (3.27) becomes

$$
\begin{aligned}
E(Y_1) &= Np_1 \\
E(Y_2) &= Np_2, \\
E(Y_3) &= Np_3 \\
E(Y_1^2) &= N(N-1)p_1^2 + Np_1 \\
E(Y_2^2) &= N(N-1)p_2^2 + Np_2 \\
E(Y_3^2) &= N(N-1)p_3^2 + Np_3 \\
E(Y_1 Y_2) &= N(N-1)p_1 p_2 \\
E(Y_1 Y_3) &= N(N-1)p_1 p_3 \\
E(Y_2 Y_3) &= N(N-1)p_2 p_3
\end{aligned}
\tag{3.30}
$$

Similarly, when order increasing, all the moments can be found by using partial derivatives of the joint moment generating function of (3.26).

3.1.3 Multivariate Normal Universal Portfolio (MVNUP)

Consider the random vector $\mathbf{Y} = (Y_1, Y_2, \cdots, Y_m)$ with a joint multivariate normal probability density function defined as follow:

$$f(\mathbf{Y}) = \frac{e^{-\frac{1}{2}(\mathbf{Y}-\boldsymbol{\mu})^t K^{-1}(\mathbf{Y}-\boldsymbol{\mu})}}{(\sqrt{2\pi})^n |K|^{1/2}}, \tag{3.31}$$

Where K is the covariance matrix of Y, $E(\mathbf{Y}) = \boldsymbol{\mu} = (\mu_1, \mu_2, \cdots, \mu_m)$ is the mean vector. We say Y has the multivariate normal distribution $N(\boldsymbol{\mu}, K)$, $|\cdot|$ means determinate. In our study, we choose $m = 3$, the Multivariate Normal distribution $N(\boldsymbol{\mu}, K)$ is special form of $\boldsymbol{\mu} = (\mu_1, \mu_2, \mu_3)$ and K = diag $(\sigma_1^2, \sigma_2^2, \sigma_3^2)$, where Y_1, Y_2, Y_3 are independent. Hence, the joint moment generating function is defined as:

$$M(t_1, t_2, t_3) = e^{((t_1\mu_1 + t_2\mu_2 + t_3\mu_3) + \frac{1}{2}(\sigma_1^2 t_1^2 + \sigma_2^2 t_2^2 + \sigma_3^2 t_3^2))} \tag{3.32}$$

When $\nu = 1$, the $E[Y_k Y_{i_1} \cdots Y_{i_\nu}]$ in (3.12) can be obtained from the following:

$$\begin{aligned}
&E(Y_k) = \mu_k \\
&E(Y_k^2) = \mu_k^2 + \sigma_k^2 \\
&E(Y_k Y_i) = \mu_k \mu_i \qquad \text{for} \quad i \neq k \\
&E(Y_k Y_j) = \mu_k \mu_j \qquad \text{for} \quad j \neq k \\
&E(Y_i Y_j) = \mu_i \mu_j \qquad \text{for} \quad i \neq j
\end{aligned} \tag{3.33}$$

for $k = 1, 2, 3$.

Similarly, for order two and three, all the moments of order two and three of the universal portfolio can be obtained by taking appropriate partial derivative of (3.32).

3.2 Stochastic Process Generated Universal Portfolios

In this section, the order-ν universal portfolio will be generalised to *stochastic process based order-ν universal portfolio*.

Definition 3.1. A stochastic process $\{X(t), t \in T\}$ is a collection of random variables. That is, for each $t \in T, X(t)$ is a random variable. The index t is often interpreted as time and, as a result, we refer to $X(t)$ as the state of the process at time t.

The set T is called the $indexset$ of the process. When T is a countable set the stochastic process is said to be a $discrete$-$time\ process$. If T is an interval of the real line, the stochastic process is said to be a $continuous$-$time\ process$. For instance, $\{X_n, n = 0, 1, ...\}$ is a discrete-time stochastic process indexed by the nonnegative integers; while $\{X(t), t \geq 0\}$ is a continuous-time stochastic process indexed by the nonnegative real numbers. The state space of a stochastic process is defined as the set of all possible values that the random variables $X(t)$ can assume. Thus, a stochastic process is a family of random variables that describes the evolution through time of some (physical) process (Ross 2007).

Let $\{Y_{n1}\}_{n=1}^{\infty}$, $\{Y_{n2}\}_{n=1}^{\infty}$,$\cdots$, $\{Y_{nm}\}_{n=1}^{\infty}$ be m given independent stochastic processes. For a fixed positive ν, the ν-order universal portfolio $\{\mathbf{b}_{n+1}\}$ generated by the m given stochastic processes is defined as:

$$b_{n+1,k} = \frac{E[Y_{nk}(\mathbf{Y}_n^t\mathbf{x}_n)(\mathbf{Y}_{n-1}^t\mathbf{x}_{n-1})\cdots(\mathbf{Y}_{n-(\nu-1)}^t\mathbf{x}_{n-(\nu-1)})]}{\sum_{j=1}^m E[Y_{nj}(\mathbf{Y}_n^t\mathbf{x}_n)(\mathbf{Y}_{n-1}^t\mathbf{x}_{n-1})\cdots(\mathbf{Y}_{n-(\nu-1)}^t\mathbf{x}_{n-(\nu-1)})]} \tag{3.34}$$

for $k = 1, 2, \cdots, m$ and where the vector $\mathbf{Y}_l = (Y_{l1}, \cdots, Y_{lm})$ for $l = 1, 2, \cdots$.

The formula in (3.34) is the generalisation of (3.8) because it is a special case of $Y_{nj} = Y_j, j = 1, \cdots, m$ independent of the trading day n.

Expanding the numerator and denominator of (3.34), $b_{n+1,k}$ can be written as

$$\frac{\sum_{i_1=1}^m\sum_{i_2=1}^m\cdots\sum_{i_\nu=1}^m(x_{ni_1}x_{n-1,i_2}\cdots x_{n-\nu+1,i_\nu})E[Y_{nk}Y_{ni_1}Y_{n-1,i_2}\cdots Y_{n-\nu+1,i_\nu}]}{\sum_{j=1}^m\{\sum_{i_1=1}^m\sum_{i_2=1}^m\cdots\sum_{i_\nu=1}^m(x_{ni_1}x_{n-1,i_2}\cdots x_{n-\nu+1,i_\nu})E[Y_{nj}Y_{n_{i_1}}Y_{n-1,i_2}\cdots Y_{n-\nu+1,i_\nu}]\}} \tag{3.35}$$

The derivation is similar to (3.12) and (3.13).

3.2.1 Brownian Motion, Stationary Process and Weakly Stationary Process

Let us state the definitions of various stochastic processes (Ross 2007) that we will be needed for later subsection.

Definition 3.2. A stochastic process $\{X(t), t \geq 0\}$ is said to be a *Brownian motion process* with drift coefficient μ and variance parameter σ^2 if

1. $X(0) = 0$,

2. $\{X(t), t \geq 0\}$ has stationary and independent increments;

3. for every $t > 0$, $X(t)$ is normally distributed with mean μt and variance $\sigma^2 t$, i.e. $X(t) \sim$Normal$(\mu t, \sigma^2 t)$.

When $\mu = 0, \sigma = 1$, $\{X(t), t \geq 0\}$ is it called standard Brownian motion with $E(X(t)) = 0$ and $V(X(t)) = t$.

Definition 3.3. A stochastic process $X(t)$, $t \geq 0$ is called a *Gaussian process*, or a normal process if $X(t_1), \cdots, X(t_n)$ has a multivariate normal distribution for all $t_1, \cdots, t_n$.

Definition 3.4. A stochastic process $\{X(t), t > 0\}$ is said to be a *stationary process* if for all n, s, $t_1, \cdots, t_n$, the random vector $X(t_1), \cdots, X(t_n)$ and $X(t_1 + s), \cdots, X(t_n + s)$ have the same joint distribution.

Definition 3.5. A stochastic process $\{Y_r\}_{r=1}^{\infty}$ is said to be *weakly stationary* if $E(Y_r) = \mu$, independent of the time r and $cov(Y_r, Y_{r+s})$ does not depend of the r but depends on the time difference s only.

The weakly stationary process is a generalisation of the stationary process. A Brownian-motion is the integral of white noise Gaussian process whereas Ornstein-Uhlenbeck process is both a stationary process and a Gaussian.

Both Brownian-motion process and Ornstein Uhlenbeck process are stationary process. Since a stationary process is also a weakly stationary process, they are also weakly stationary.

Definition 3.6. If $\{W_r\}$ is a standard Brownian-motion process, then $\{Z_r\}$ defined by $Z_r = e^{\frac{-\alpha r}{2}} W(e^{\alpha r})$ for $\alpha > 0$ is a zero-mean weakly stationary process with covariance function $\text{cov}[Z_r, Z_{r+s}] = e^{\frac{-\alpha s}{2}}$. The process $\{Z_r\}$ is known as the Ornstein-Uhlenbeck process.

3.2.2 Brownian-motion Process Generated Universal Portfolio (BMUP)

From (3.35), $E[Y_{s_1 i_1} Y_{s_2 i_2} \cdots Y_{s_u i_u}] = E[Y_{s_1 i_1}] E[Y_{s_2 i_2}] \cdots E[Y_{s_u i_u}]$ if the u integers $i_1, i_2, \cdots, i_u$ are distinct. Otherwise $E[Y_{s_1 j} Y_{s_2 j} \cdots Y_{s_u j}]$ is determined by using the moment-generating function of $Y_{s_i j}, Y_{s_2 j}, \cdots, Y_{s_u j}$.

Specifically,

$E[(Y_{s_1 j} Y_{s_2 j} \cdots Y_{s_u j})(Y_{r_1 k} Y_{r_2 k} \cdots Y_{r_p k})] = E[Y_{s_1 j} Y_{s_2 j} \cdots Y_{s_u j}] \times E[Y_{r_1 k} Y_{r_2 k} \cdots Y_{r_p k}]$

for $j \neq k$.

$$E\left[\prod_{i=1}^{q} (Y_{s_{i_1} j_i} Y_{s_{i_2} j_i} \cdots Y_{s_{i u_i} j_i})\right] = \prod_{i=1}^{q} E(Y_{s_{i_1} j_i} Y_{s_{i_2} j_i} \cdots Y_{s_{i u_i} j_i}) \quad (3.36)$$

for any set of distinct integers $j_1, j_2, \cdots, j_q$.

When $\{Y_{n1}\}_{n=1}^{\infty}$, $\{Y_{n2}\}_{n=1}^{\infty}$, $\cdots$, $\{Y_{nm}\}_{n=1}^{\infty}$ are independent Brownian motion with positive drift coefficients $\mu_1, \mu_2, \cdots, \mu_m$ and variance parameters $\sigma_1^2, \sigma_2^2, \cdots,$ σ_m^2 respectively. According to Ross (2007), the process $\{Y_{nl}\}$ has stationary and independent increments, where Y_{nl} has a normal distribution with mean $n\mu_l$ and variance $n\sigma_l^2$ for $l = 1, 2, \cdots, m$ and $n = 1, 2, \cdots$. The covariance of $Y_{n_1 l}$ and $Y_{n_2 l}$ is given by

$$\mathrm{Cov}(Y_{n_1 l}, Y_{n_2 l}) = n_1 \sigma_l^2 \quad \text{for} \quad 0 < n_1 \leq n_2. \tag{3.37}$$

Furthermore, the ν random variables $Y_{n-\nu+1,l}, Y_{n-\nu+2,l}, \cdots, Y_{nl}$ have a joint multivariate normal distribution with mean vector $\boldsymbol{\mu}_l = (\mu_{l_1}, \mu_{l_2}, \cdots, \mu_{l_\nu}) > \mathbf{0}$ and $\boldsymbol{\nu} \times \boldsymbol{\nu}$ covariance matrix

$$K_l = \sigma_l^2 L = \sigma_l^2 (\lambda_{ij}) \quad \text{for} \quad l = 1, 2, \cdots, m \tag{3.38}$$

where

$$\mu_{ik} = (n - \nu + k)\mu_i \quad \text{for} \quad k = 1, 2, \cdots, \nu \tag{3.39}$$

and

$$\lambda_{ij} = \begin{cases} n - \nu + i & \text{if } i \leq j, \\ n - \nu + j & \text{if } i > j. \end{cases} \tag{3.40}$$

for $i, j = 1, 2, \cdots, \nu$.

Note that the lambda matrix $L = (\lambda_{ij})$ in (3.38) does not depend on l. The components λ_{ij} are the covariance components of $Y_{n-\nu+1,l}, \cdots, Y_{n,l}$ that depend on time n. The means μ_{ik} are nonnegative for $n \geq \nu = 1$, $k = 1, 2, \cdots, \nu$ and $l = 1, 2, \cdots, m$. Similarly, $\lambda_{ij} \geq 0$ for all $i, j = 1, 2, \cdots, \nu$ if $n \geq \nu - 1$.

In particular, when $\nu = 1$ and $m = 3$, (3.35) becomes

$$b_{n+1,k} = \frac{x_{n,1}E(Y_{nk}Y_{n1}) + x_{n,2}E(Y_{nk}Y_{n2}) + x_{n,3}E(Y_{nk}Y_{n3})}{\sum_{j=1}^{3}[x_{n,1}E(Y_{nj}Y_{n1}) + x_{n,2}E(Y_{nj}Y_{n2}) + x_{n,3}E(Y_{nj}Y_{n3})]} \tag{3.41}$$

for $k = 1, 2, 3$

Since the Brownian process are assumed to be independent, from (3.37) to (3.40), when $\nu = 1$, we have

$$\begin{aligned} E(Y_{nk}^2) &= n(\sigma_k^2 + n\mu_k^2) \\ E(Y_{nk}Y_{ni}) &= E(Y_{nk})E(Y_{ni}) = n^2\mu_k\mu_i \qquad \text{for} \quad k \neq i \\ E(Y_{nk}Y_{nj}) &= E(Y_{nk})E(Y_{nj}) = n^2\mu_k\mu_j \qquad \text{for} \quad k \neq j \end{aligned} \tag{3.42}$$

3.2.3 Asymptotic Behaviour of the Brownian-Motion-Generated Universal Portfolio

In this subsection, we will try to show that for sufficient large n, the Brownian-motion generated universal portfolio will behave like the constant rebalance portfolio (define in Chapter 1).

Previously, it is stated that the ν random variables $Y_{n-\nu+1,l}, Y_{n-\nu+2,l}, \cdots, Y_{nl}$ are assumed to have a joint multivariate normal distribution with mean vector $\boldsymbol{\mu}_l = (\mu_{l1}, \cdots, \mu_{l\nu}) > \mathbf{0}$ given by (3.39) and covariance matrix (3.37). Let $\boldsymbol{\tau} = (\tau_1, \cdots, \tau_v)$.

The moment-generating function $M_l(\tau_1, \cdots, \tau_\nu)$ of the v random variables is given by

$$M_l(\boldsymbol{\tau}) = E(e^{\sum_{i=1}^{\nu} \tau_i Y_{n-\nu+i,l}}) = e^{\mu_l^t \boldsymbol{\tau} + \frac{1}{2}\boldsymbol{\tau}^t K_l \boldsymbol{\tau}} \tag{3.43}$$

for $l = 1, 2, \cdots, m$.

Lemma 3.1. The moment $E[Y_{n-\nu+k_1,l} Y_{n-\nu+k_2,l} \cdots Y_{n-\nu+k_\nu,l}]$ is a polynomial in n of degree u where the coefficients of n^u is μ_l^u and the integers $1 \leq k_i \leq \nu$ for $i = 1, \cdots, u, l = 1, \cdots, m$.

Proof. Note that

$$\boldsymbol{\mu}_l^t \boldsymbol{\tau} + \frac{1}{2}\boldsymbol{\tau}^t K_l \boldsymbol{\tau} = \sum_{i=1}^{\nu} \mu_{li}\tau_i + \frac{1}{2}[\sigma_i^2 \sum_{i=1}^{\nu} \lambda_{ii}\tau_i^2 + 2\sigma_l^2 \sum_{i<j} \lambda_{ij}\tau_i\tau_j], \tag{3.44}$$

the first and second order derivatives of $M_l(\boldsymbol{\tau})$ are given by :

$$\frac{\partial}{\partial \tau_{k_1}} M_l(\boldsymbol{\tau}) = e^{\boldsymbol{\mu}_l^t \boldsymbol{\tau} + \frac{1}{2}\boldsymbol{\tau}^t K_l \boldsymbol{\tau}} [\mu_{lk_l} + \sigma_l^2 \sum_{j=1}^{\nu} \lambda_{k_1 j}\tau_j], \tag{3.45}$$

$$\begin{aligned}\frac{\partial^2}{\partial \tau_{k_1}\partial \tau_{k_2}} M_l(\boldsymbol{\tau}) = {} & e^{\boldsymbol{\mu}_l^t \boldsymbol{\tau} + \frac{1}{2}\boldsymbol{\tau}^t K_l \boldsymbol{\tau}} [\mu_{lk_l} + \sigma_l^2 \sum_{j=1}^{\nu} \lambda_{k_1 j}\tau_j][\mu_{lk_2} + \sigma_l^2 \sum_{j=1}^{\nu} \lambda_{k_2 j}\tau_j] \\ & + e^{\boldsymbol{\mu}_l^t \boldsymbol{\tau} + \frac{1}{2}\boldsymbol{\tau}^t K_l \boldsymbol{\tau}} [\sigma_l^2 \lambda_{k_1 k_2}],\end{aligned} \tag{3.46}$$

furthermore

$$\left.\frac{\partial}{\partial \tau_{k_1}} M_l(\boldsymbol{\tau})\right|_{\tau=0} = \mu_{kl_1} = (n - \nu + k_1)\mu_l \tag{3.47}$$

and

$$
\begin{aligned}
&\frac{\partial}{\partial\tau_{k_1}\partial\tau_{k_2}}M_l(\boldsymbol{\tau})|_{\tau=0} = \mu_{kl_1}\mu_{lk_2} + \sigma_l^2\lambda_{k_1k_2l} \\
&= \begin{cases} \mu_l^2(n-\nu+k_1)(n-\nu+k_2) + \sigma_l^2(n-\nu+k_1) & \text{if } k_i \le k_2, \\ \mu_l^2(n-\nu+k_1)(n-\nu+k_2) + \sigma_l^2(n-\nu+k_2) & \text{if } k_1 > k_2. \end{cases}
\end{aligned} \tag{3.48}
$$

It is observed that $\frac{\partial}{\partial\tau_{k_1}}M_l(\boldsymbol{\tau})\Big|_{\tau=0}$ and $\frac{\partial}{\partial\tau_{k_1}\partial\tau_{k_2}}M_l(\boldsymbol{\tau})\Big|_{\tau=0}$ are polynomials in n with degree 1 and 2 respectively. For any nonnegative integers $s_2, s_2, \cdots, s_\nu$ summing up to u, it follows that

$$
E[Y_{n-\nu+1,l}^{s_1}Y_{n-\nu+2,l}^{s_2}\cdots Y_{n,l}^{s_\nu}] = \frac{\partial^u M_l(\boldsymbol{\tau})}{\partial\tau_1^{s_1}\partial\tau_2^{s_2}\cdots\partial\tau_\nu^{s_\nu}}\Bigg|_{\tau=0} \tag{3.49}
$$

where the differential operator $\frac{\partial^{s_i}}{\partial\tau_i^{s_i}}$ is omitted is $s_1 = 0$. For integers $1 \le k_i \le \nu$ possibly repeated, it is straight forward to deduce that $\frac{\partial^u M_l(\boldsymbol{\tau})}{\partial\tau_{k_1}\partial\tau_{k_2}\cdots\partial\tau_{k_u}}$ is a sum of products where each product is of the form

$$
\begin{aligned}
&e^{\mu_l^t\boldsymbol{\tau}+\frac{1}{2}\boldsymbol{\tau}^tK_l\boldsymbol{\tau}}[\mu_{lr_1} + \sigma_l^2\sum_j \lambda_{r_1j}\boldsymbol{\tau}_j]\cdots[\mu_{lr_p} + \sigma_l^2\sum_j \lambda_{r_pj}\boldsymbol{\tau}_j] \times [\sigma_l^2\lambda_{r_pr_{p+2}}] \\
&\quad \cdots [\sigma_l^2\lambda_{r_{q-1}r_q}].
\end{aligned} \tag{3.50}
$$

for some positive integers $r_1, \cdots, r_p, r_{p+1}, \cdots, r_q$ such that each integer $1 \le r_i \le \nu$ and $q \le u$.

The product (3.50) evaluated at $\boldsymbol{\tau} = 0$ is $\mu_{lr_1} \cdots \mu_{lr_p} \sigma^2 \lambda_{r_p r_{p+1}} \cdots \sigma^2 \lambda_{r_{q-1} r_q}$

$$\begin{aligned} &= \mu_l^p (n - \nu + r_1) \cdots (n - \nu + r_p) \sigma_l^{2(q-p)} (n - \nu + min(r_p; r_{p+1})) \\ &\quad \cdots (n - \nu + min(r_{q-1}, r_q)) \end{aligned} \tag{3.51}$$

The right hand side is a polynomial in n of degree $q \leq u$. The only product in $\left.\frac{\partial^u M_l(\boldsymbol{\tau})}{\partial \tau_{k_1} \partial \tau_{k_2} \cdots \partial \tau_{k_u}}\right|_{\tau=0}$ containing the highest power of n is $\mu_l^u (n - \nu + k_1)(n - \nu + k_2) \cdots (n - \nu + k_u)$ and the coefficient of n^u is μ_l^u. or any integers $p \leq s \leq u$, the coefficient of n^s is of the form

$$\mu_l^p \sigma_l^{2(s-p)} c_1 c_2 \cdots c_{u-s} \tag{3.52}$$

for some negative constants $c_1, c_2, \cdots c_{u-s}$ which do not depend on n and

$$|c_i| \in \{1, 2, \cdots, \nu - 1\} \quad \text{for} \quad i = 1, 2, \cdots, u - s. \tag{3.53}$$

□

Lemma 3.2. The moment $E[\prod_{i=1}^q (Y_{s_{i_1} j_i} \cdots Y_{s_{i u_i} j_i})]$ for any set of distinct integers $j_1, j_2, \cdots, j_q, 1 \leq j_i \leq m, n - \nu + 1 \leq s_{i_k} \leq n, u_1 + u_2 + \cdots + u_q = \nu + 1$, is a polynomial in n of degree ν + 1 where the coefficient of $n^{\nu+1}$ is $\mu_{j_i}^{u_1} \mu_{j_2}^{u_2} \cdots \mu_{j_q}^{u_q}$.

Proof. It follows from 3.1 that $E[Y_{s_{i_1} j_1} Y_{s_{i_2} j_i} \cdots Y_{s_{i u_i} j_i}]$is a polynomial in n of degree u_i and the coefficient of $n^{u_i}$is $\mu_{ji}^{u_i}$. From (3.35), $E[\prod_{i=1}^q (Y_{s_{i_1} j_1} \cdots Y_{s_{i u} j_1})]$ is a polynomial in n of degree $\sum_{i=1}^q u_i = \nu + 1$ and the coefficient of $n^{\sum_{i=1}^q u_i}$ is $\prod_{i=1}^q \mu_{ji}^{u_i}$. □

Proposition 3.2.1. Consider the universal portfolio $\{b_{n+1}\}$ given by (3.35) generated by m independent Brownian motions $\{Y_{n1}\}_{n=1}^{\infty}, \{Y_{n2}\}_{n=1}^{\infty}, \cdots, \{Y_{nm}\}_{n=1}^{\infty}$, for a fixed integer $\nu \geq 1$. For a large n, the k^{th} component of b_n behaves like

$$b_{n,k} = \frac{\sum_{i_1=1}^{m}\sum_{i_2=1}^{m}\cdots\sum_{i_\nu=1}^{m}(x_{ni_1}x_{n-1,i_2}\cdots x_{n-\nu+1,i_\nu})(\mu_k\mu_{i_1}\mu_{i_2}\cdots\mu_{i_\nu})}{\sum_{j=1}^{m}\{\sum_{i_1=1}^{m}\sum_{i_2=1}^{m}\cdots\sum_{i_\nu=1}^{m}(x_{ni_1}x_{n-1,i_2}\cdots x_{n-\nu+1,i_\nu})(\mu_j\mu_{i_1}\mu_{i_2}\cdots\mu_{i_\nu})\}} \tag{3.54}$$

for $k = 1, 2, \cdots, m$ where $\mu_1, \mu_2, \cdots, \mu_m$ are positive drift coefficients of the m Brownian motions.

Proof. It is clear from Lemma 3.2 that the component $(x_{ni_1}x_{n-2,i_2}\cdots x_{n-\nu+1,i_\nu})$ $\times E[Y_{nk}Y_{ni_1}Y_{n-1,i_2}\cdots Y_{n-\nu+1,i_\nu}]$ given by (3.35) is a polynomial in n of degree $\nu+1$ where $(x_{ni_1}x_{n-2,i_2}\cdots x_{n-\nu+1,i_\nu})(\mu_k\mu_{i_1}\mu_{i_2}\cdots\mu_{i_k})$ is the coefficient of $n^{\nu+1}$. Hence the sum (3.35) is a polynomial in n of degree $\nu+1$ where

$$\sum_{i_1=1}^{m}\sum_{i_2=1}^{m}\cdots\sum_{i_\nu=1}^{m}(x_{ni_1}x_{n-1,i_2}\cdots x_{n-\nu+1,i_\nu})(\mu_k\mu_{i_1}\mu_{i_2}\cdots\mu_{i_k})$$

is the coefficient of $n^{\nu+1}$. Similarly, the denominator of b_{nk} in (3.35) is a polynomial in n of degree $\nu+1$ where the coefficient of $n^{\nu+1}$ is

$$\sum_{j=1}^{m}\left\{\sum_{i_1=1}^{m}\sum_{i_2=1}^{m}\cdots\sum_{i_\nu=1}^{m}(x_{ni_1}x_{n-1,i_2}\cdots x_{n-\nu+1,i_\nu})(\mu_j\mu_{i_1}\mu_{i_2}\cdots\mu_{i_k})\right\}.$$

In conclusion, b_{nk} is a ratio of polynomial in n of the same degree $\nu+1$. Noting that for a fixed integer $\nu \geq 1$, the form quantities:

$$\max_{1\leq i_l\leq m}\left\{x_{ni_1}x_{n-1,i_2}\cdots x_{n-\nu+1,i_\nu}\right\}, \max_{\substack{\mu_{i_l}\in\{\mu_1,\mu_2,\cdots,\mu_m\},\\ 1\leq j\leq \nu+1}}\left\{\mu_{i_1}\mu_{i_2}\cdots\mu_{i_j}\right\},$$

$$\max_{\substack{\sigma_{i_l}\in\{\sigma_1,\sigma_2,\cdots\sigma_l\},\\ 1\leq j\leq\nu}}\{\sigma_{i_1}^2\sigma_{i_2}^2\cdots\sigma_{i_j}^2\}$$

and

$$\max_{\substack{|c_i|\in\{1,2,\cdots,\nu-1\},\\ 1\leq j\leq\nu}}\{|c_1c_2\cdots c_j|\}$$

are bounded the asymptotic behaviour of $b_{n,k}$ is the ratio of the numerator coefficient of $n^{\nu+1}$ to the denominator coefficient of $n^{\nu+1}$. $\square$

This proposition states that b_n converges to some constant when $n \to \infty$

3.2.4 Ornstein Uhlenbeck Process Generated Universal Portfolio (OUUP)

A finite-order universal portfolio generated by a set of independence Brownian motions was studied in Section 3.2.2. In the portfolio, the past price relatives are weighted by the joint moments of the Brownian motions which depend on the Brownian motion parameters and the sampled times. In Section 3.2.3, asymptotically, over a long time, this portfolio converges to a constant rebalanced portfolio. We will investigate a generalisation of Brownian-motion, the Ornstein-Uhlenbeck process to see it has a better property.

According to Ross (2007), a stochastic process $\{Y_r\}_{r=1}^{\infty}$ is said to be weakly stationary if $E(Y_r) = \mu$, independent of the time r and $cov(Y_r, Y_{r+s})$ does not depend of the r but depends on the time difference s only. For m given weakly stationary processes $\{Y_{n1}\}, \{Y_{n2}\}, \cdots \{Y_{nm}\}$, $E(\cdot)$ can be defined by rearranging the product of random variables $(Y_{nk}Y_{ni_1}Y_{n-1,i_2}\cdots Y_{n-\nu+1,i_\nu})$ as m products, where in each product, the random variables come from the same process. For weakly stationary process, $(Y_{nk}Y_{ni_1}Y_{n-1,i_2}\cdots Y_{n-\nu+1,i_\nu})$ can be written as the following

m products:

$$
\begin{aligned}
(Y_{nk}Y_{ni_1}Y_{n-1,i_2}\cdots Y_{n-\nu+1,i_\nu}) &= (Y_{r_1 1}Y_{r_2 1}\cdots Y_{r_n 1})(Y_{u_1 2}Y_{u_2 2}\cdots Y_{u_n 2}) \\
&\times(Y_{v_1 3}Y_{v_2 3}\cdots Y_{v_n 3})\times\cdots\times(Y_{w_1 m}Y_{w_2 m}\cdots Y_{w_n m})
\end{aligned}
\tag{3.55}
$$

for some ordered sequences of time indices $r_1 \geq r_2 > \cdots > r_n; u_1 \geq u_2 > \cdots > u_n; v_1 \geq v_2 > \cdots > v_n; \cdots ; w_1 \geq w_2 > \cdots > w_n$. Taking expected value,

$$
\begin{aligned}
E(Y_{nk}Y_{ni_1}Y_{n-1,i_2}\cdots Y_{n-\nu+1,i_\nu}) &= E(Y_{r_1 1}Y_{r_2 1}\cdots Y_{r_n 1})E(Y_{u_1 2}Y_{u_2 2}\cdots Y_{u_n 2}) \\
&\times E(Y_{v_1 3}Y_{v_2 3}\cdots Y_{v_n 3})\times\cdots\times E(Y_{w_1 m}Y_{w_2 m}\cdots Y_{w_n m})
\end{aligned}
\tag{3.56}
$$

where the functionals E is defined as :

$$
E(Y_{q_1 k}Y_{q_2 k}\cdots, Y_{q_{n_q}k}) = \begin{cases} \displaystyle\prod_{i=1}^{n_q-1} E(Y_{q_i k}Y_{q_{i+1}k}) & \text{if } n_q \text{ is even,} \\ \displaystyle E(Y_{q_{n_q}k})\prod_{i=1}^{n_q-1} E(Y_{q_i k}Y_{q_{i+1}k}) & \text{if } n_q \text{ is odd.} \end{cases}
\tag{3.57}
$$

for $k = 1, 2, \cdots, m; q_1 \geq q_2 > \cdots > q_{n_q}$.

The translated Ornstein-Uhlenbeck process $\{Y_r\}$ is define by $Y_r = Z_r + \mu$ for all r is said to have parameters (μ, α) if $E[Y_r] = \mu$ and $E[Y_r Y_{r+s}] = e^{\frac{-\alpha s}{2}} + \mu^2$ for $s > 0$, $\alpha > 0$. It is also assumed that $\mu > 0$. Let $\{Y_{n1}\}, \{Y_{n2}\}, \cdots \{Y_{nm}\}$ be m given independent (translated) Ornstein-Uhlenbeck process with parameters $(\mu_1, \alpha_1), (\mu_2, \alpha_2), \cdots, (\mu_m, \alpha_m)$ respectively., where all parameters are positive. Consider the universal portfolio (3.35) generated by these process where $E(\cdot)$ is defined by (3.56) and (3.57), namely

$$E(Y_{q_1k}Y_{q_2k}\cdots,Y_{q_{n_q}k}) = \begin{cases} \prod_{i=1}^{n_q-1}[e^{\frac{-\alpha_k(q_i-q_{i+1})}{2}} + \mu_k^2] & \text{if } n_q \text{is even,} \\ \mu_k \prod_{i=1}^{n_q-2}[e^{\frac{-\alpha_k(q_i-q_{i+1})}{2}} + \mu_k^2] & \text{if } n_q \text{ is odd.} \end{cases} \tag{3.58}$$

for $k = 1, 2, \cdots, m : q_1 \geq q_2 > \cdots > q_{n_q}$.

in particular, when $\nu = 1$, $m = 3$, (3.34) becomes

$$b_{n+1,k} = \frac{\sum_{i=1}^{3} x_{n,i}E(Y_{nk}Y_{n,i})}{\sum_{j=1}^{3}(\sum_{i=1}^{3} x_{n,i}E(Y_{nj}Y_{ni}))} \tag{3.59}$$

and the (3.58) becomes

$$\begin{aligned} &E(Y_{nk}Y_{nk}) = E(Y_{nk}^2) = 1 + \mu_k^2 \\ &E(Y_{nk}Y_{ni}) = E(Y_{nk})E(Y_{ni}) = \mu_k\mu_i \qquad \text{for} \quad i \neq k \\ &E(Y_{nk}Y_{nj}) = E(Y_{nk})E(Y_{nj}) = \mu_k\mu_j \qquad \text{for} \quad j \neq k \end{aligned} \tag{3.60}$$

for $k = 1, 2, 3$.

CHAPTER 4

COMPUTER IMPLEMENTATION

There are many programming languages for implementing the universal portfolio strategies. In the previous chapter, Python is a suitable choice because it is easy to use, powerful, versatile and being relatively popular in finance (Tuttle & Butcher 2016).

4.1 Dirichlet Universal Portfolio Strategy

We will first investigate Dirichlet universal portfolio strategy since it was proposed by Cover & Ordentlich (1996). Cover & Ordentlich (1996) presented an algorithm for generating the Dirichlet($\frac{1}{2}, \frac{1}{2}$) two stocks universal portfolio. Chan (2002) modified this algorithm for generating any Dirichlet($\alpha_1, \alpha_2, \cdots, \alpha_m$) universal portfolio for $m = 2, 3$ and 4 stocks. The modified algorithm also obtained by Tan (2004b). We choose this modified algorithm for computing the 3 stocks universal portfolio generated by Dirichlet($\alpha_1, \alpha_2, \alpha_3$) distribution where $\alpha_j > 0$ for $j = 1, 2, 3$. We present the modified algorithm of Chan (2002) for computing the three-stock universal portfolio generated by Dirichlet $(\alpha_1, \alpha_2, \alpha_3)$ distribution where $\alpha_j > 0$ for $j = 1, 2, 3$. Consider a sequence of price-relative vectors $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n$ corresponding to n trading days in a three-stock market, where $x_i = (x_{i,1}, x_{i,2}, x_{i,3})$ for $i = 1, 2, \cdots, n$. We shall define three recursive functions $X_n(l_1, l_2), C_n(l_1, l_2)$ and $Q_n(l_1, l_2)$.

Firstly, we define

$$X_n(l_1, l_2) = \sum_{j^n \in T_n(l_1,l_2)} \prod_{i=1}^{n} x_{ij} \tag{4.1}$$

where $T_n(l_1, l_2)$ is the set of all sequence $j^n = (j_1, j_2, \cdots, j_n) \in \{1, 2, 3\}^n$ with l_1 1's and l_2 2's and $n - l_1 - l_2$ 3's. Secondly, we defined,

$$C_n(l_1, l_2) = \int_{\bar{B}} b_1^{l_1} b_2^{l_2} b_3^{n-l_1-l_2} d\mu(\mathbf{b}) \tag{4.2}$$

where

$$d\mu(\mathbf{b}) = \frac{\Gamma(\alpha_1 + \alpha_2 + \alpha_3)}{\Gamma(\alpha_1)\Gamma(\alpha_2)\Gamma(\alpha_3)} b_1^{\alpha_1-1} b_2^{\alpha_2-1} b_3^{\alpha_3-1} db_1 db_2 \tag{4.3}$$

Finally,

$$Q_n(l_1, l_2) = X_n(l_1, l_2) C_n(l_1, l_2) \tag{4.4}$$

The universal portfolio $\{\mathbf{b}_n\}$ can then be computed as follows:

$$\hat{\mathbf{b}}_n = \frac{1}{\sum_{l_1=1}^{n-1} \sum_{l_2=0}^{n-l_1-1} Q_{n-1}(l_1, l_2)} \begin{pmatrix} \displaystyle\sum_{l_1=0}^{n-1} \sum_{l_2=0}^{n-l_1-1} \frac{(l_1 + \alpha_1)}{(n + \alpha_1 + \alpha_2 + \alpha_3 - 1)} Q_{n-1}(l_1, l_2) \\ \displaystyle\sum_{l_1=0}^{n-1} \sum_{l_2=0}^{n-l_1-1} \frac{(l_2 + \alpha_2)}{(n + \alpha_1 + \alpha_2 + \alpha_3 - 1)} Q_{n-1}(l_1, l_2) \\ \displaystyle\sum_{l_1=0}^{n-1} \sum_{l_2=0}^{n-l_1-1} \frac{(n - l_1 - l_2 + \alpha_3 - 1)}{(n + \alpha_1 + \alpha_2 + \alpha_3 - 1)} Q_{n-1}(l_1, l_2) \end{pmatrix} \tag{4.5}$$

and the unversal capital achieved by $\hat{\mathbf{b}}_n$ is computed as :

$$\hat{S}_n = \sum_{l=0}^{n} Q_l(l_1, l_2) \tag{4.6}$$

Next, we give the recursive relationships of the functions $X_n(l_1, l_2), C_n(l_1, l_2)$ and $Q_n(l_1, l_2)$ and their end-point conditions.

The recursion for $X_n(l_1, l_2)$ is given by :

$$X_n(l_1, l_2) = x_{n1}X_{n-1}(l_1 - 1, l_2) + x_{n2}X_{n-1}(l_1, l_2 - 1) + x_{n3}X_{n-1}(l_1, l_2) \quad (4.7)$$

for $1 \leq l_1 \leq n - 1$ and $1 \leq l_2 \leq n - l_1 - 1$ and the endpoint condition are given by:

$$X_n(l_1, 0) = x_{n1}X_{n-1}(l_1, 0) + x_{n3}X_{n-1}(l_1, 0) \quad (4.8)$$

$$X_n(0, l_2) = x_{n2}X_{n-1}(0, l_2 - 1) + x_{n3}X_{n-1}(0, l_2) \quad (4.9)$$

The recursion for $C_n(l_1, l_2)$ given by :

$$C_n(l_1, l_2) = \frac{l_1 + \alpha_1 - 1}{n + \alpha_1 + \alpha_2 + \alpha_3 - 1} C_{n-1}(l_1 - 1, l_2) \quad (4.10)$$

for $1 \leq l_1 \leq n$ and $0 \leq l_2 \leq n - l_1$.

$$C_n(l_1, l_2) = \frac{l_2 + \alpha_2 - 1}{n + \alpha_1 + \alpha_2 + \alpha_3 - 1} C_{n-1}(l_1, l_2 - 1) \quad (4.11)$$

for $0 \leq l_1 \leq n - 1$ and $1 \leq l_2 \leq n - l_1$

$$C_n(l_1, l_2) = \frac{n - l_1 - l_2 + \alpha_3 - 1}{n + \alpha_1 + \alpha_2 + \alpha_3 - 1} C_{n-1}(l_1, l_2) \quad (4.12)$$

for $0 \leq l_1 \leq n - 1$ and $0 \leq l_2 \leq n - 1_1 - 1$.

The initial condition is

$$C_0(0, 0) = 1. \quad (4.13)$$

The recursive for $Q_n(l_1, l_2)$ given by:

$$
\begin{aligned}
Q_n(l_1,l_2) = x_{n1}&\frac{l_1+\alpha_1-1}{n+\alpha_1+\alpha_2+\alpha_3-1}Q_{n-1}(l_1-1,l_2) \\
+x_{n2}&\frac{l_2+\alpha_2-1}{n+\alpha_1+\alpha_2+\alpha_3-1}Q_{n-1}(l_1,l_2-1) \\
+x_{n3}&\frac{n-l_1-l_2+\alpha_3-1}{n+\alpha_1+\alpha_2+\alpha_3-1}Q_{n-1}(l_1,l_2)
\end{aligned} \tag{4.14}
$$

for $1 \leq l_1 \leq n-1$ and $1 \leq l_2 \leq n-l_1$. The six endpoint conditions are given by :

$$
\begin{aligned}
Q_n(l_1,0) = x_{n1}&\frac{l_1+\alpha_1-1}{n+\alpha_1+\alpha_2+\alpha_3-1}Q_{n-1}(l_1-1,0) \\
+x_{n3}&\frac{n-l_1-l_2+\alpha_3-1}{n+\alpha_1+\alpha_2+\alpha_3-1}Q_{n-1}(l_1,0)
\end{aligned} \tag{4.15}
$$

for $1 \leq l_1 \leq n-1$.

$$
\begin{aligned}
Q_n(0,l_2) = x_{n2}&\frac{l_2+\alpha_2-1}{n+\alpha_1+\alpha_2+\alpha_3-1}Q_{n-1}(0,l_2-1) \\
+x_{n3}&\frac{n-l_1-l_2+\alpha_3-1}{n+\alpha_1+\alpha_2+\alpha_3-1}Q_{n-1}(0,l_2)
\end{aligned} \tag{4.16}
$$

for $1 \leq l_2 \leq n-l_1-1$.

$$
\begin{aligned}
Q_n(l_1,l_2) = x_{n1}&\frac{l_1+\alpha_1-1}{n+\alpha_1+\alpha_2+\alpha_3-1}Q_{n-1}(l_1-1,l_2) \\
+x_{n2}&\frac{l_2+\alpha_2-1}{n+\alpha_1+\alpha_2+\alpha_3-1}Q_{n-1}(l_1,l_2-1)
\end{aligned} \tag{4.17}
$$

for $1 \leq l_1, 1 \leq l_2$ and $l_1+l_2=n$.

$$
\begin{aligned}
Q_n(n,0) &= x_{n1}\frac{n+\alpha_1-1}{n+\alpha_1+\alpha_2+\alpha_3-1} \times Q_{n-1}(n-1,0) \\
Q_n(0,n) &= x_{n2}\frac{n+\alpha_2-1}{n+\alpha_1+\alpha_2+\alpha_3-1} \times Q_{n-1}(0,n-1) \\
Q_n(0,0) &= x_{n3}\frac{n+\alpha_3-1}{n+\alpha_1+\alpha_2+\alpha_3-1} \times Q_{n-1}(0,0)
\end{aligned} \tag{4.18}
$$

The initial condition is

$$
Q_0(0,0) = 1. \tag{4.19}
$$

The computation of the universal portfolio $\hat{\mathbf{b}}_n$ and the universal capital $\hat{S}_n$ through (4.8) and (4.9) respectively can be done solely by computing the quantities $Q_n(l_1, l_2)$ recursively through (4.14) to (4.18) or by computing both $X_n(l_1, l_2)$ and $C_n(l_1, l_2)$ recursively through (4.8) to (4.14). The detailed derivations of the formulae are given in Chan (2002).

This above modified algorithm is written in python using numpy module as follow.

```
from numpy import array, zeros, arange
from time import clock

def Recursive3CCT(x,a1,a2,a3,debug=False,echo=False):
    _N = len(x[0])
    _b = zeros((3,_N+1))
    _S = zeros(_N+1)
    _S[0]=1.
    _x = x

    Qold = array([[1.]])
    Q = zeros((_N+1,_N+1))

    x1 = _x[0]
    x2 = _x[1]
    x3 = _x[2]

    begin = clock()
    for n in range(1,_N+1):
        denom = float(n+a1+a2+a3-1)

        start = clock()

        Q[0,0] = x3[n-1]*(n+a3-1)/denom*Qold[0,0]
        Q[0,n] = x2[n-1]*(n+a2-1)/denom*Qold[0,n-1]
        Q[n,0] = x1[n-1]*(n+a1-1)/denom*Qold[n-1,0]

        for l in range(1,n):
            Q[l,0]  = x1[n-1]*(l+a1-1)/denom*Qold[l-1,0] \
```

```
30                       + x3[n-1]*(n-l+a3-1)/denom*Qold[l,0]
31             Q[0,l]  = x2[n-1]*(l+a2-1)/denom*Qold[0,l-1] \
32                       + x3[n-1]*(n-l+a3-1)/denom*Qold[0,l]
33             l2 = n-l
34             Q[l,l2] = x1[n-1]*(l+a1-1)/denom*Qold[l-1,l2] \
35                       + x2[n-1]*(l2+a2-1)/denom*Qold[l,l2-1]
36
37         for l1 in range(1,n):
38             for l2 in range(1,n-l1):
39                 Q[l1,l2]  = x1[n-1]*(l1+a1-1)/denom*Qold[l1-1,l2]
40                 Q[l1,l2] += x2[n-1]*(l2+a2-1)/denom*Qold[l1,l2-1]
41                 Q[l1,l2] += x3[n-1]*(n-l1-l2+a3-1)/denom*Qold[l1,l2]
42
43         if debug:
44             for i in range(n+1):
45                 for j in range(n+1):
46                     print("%.4f" % Q[i,j],end="")
47                 print()
48             print("-"*70)
49
50         b1 =sum([(l1+a1)/denom*Qold[l1,l2] for l1 in range(0,n) for l2 in range(n-l1)])
51         b2 =sum([(l2+a2)/denom*Qold[l1,l2] for l1 in range(0,n) for l2 in range(n-l1)])
52         b3 = ]sum([(n-l1-l2+a3-1)/denom*Qold[l1,l2]for l1 in range(0,n)forl2 in range(n-l1)])
53         _b[:,n] = array([b1, b2, b3])/Qold.sum()
54         _S[n] = Q.sum()
55         Qold = Q.copy()
56
57         if echo: print("Finish recursive step %3d in %.2f second." % (n,clock()-start))
58
59     print("Total time spent: %g second." % (clock()-begin))
60     return _S[1:]
```

In the above code, from line number 5 to 12 is allocation of data structure to store data. Line number 20 is denominator of (4.10). While line number 24 to 41 is (4.14) to (4.18). The portfolio of (4.5) is from line number 50 to 53. The wealth is calculated by (4.6) and is line number 54.

Next, the four universal portfolios are written in python using the numpy module. We start with order one and order two Multinomial universal portfolio. Refer order one in (3.14) and follow by order two, (3.18) in Chapter 3 . We chosen 3 stocks for our study where $m = 3$.

4.2 Multinomial Generated Universal Portfolio

4.2.1 Order One Multinomial Genarated Universal Portfolio

```
from numpy import array, zeros, arange
from time import clock

def Recursive3Multinomial(x,p1,p2,NN,debug=False,echo=False):

    _N = len(x[0])
    _b = zeros((3,_N+1))
    _S = zeros(_N+1)
    _S[0] = 1.
    _x = x
    _p1,_p2,_p3,_NN = None,None,None,None
    order = 1
    _p1 = p1
    _p2 = p2
    _p3 = max(1.-p1-p2,0.)
    _NN = NN
    EY1 = _NN*_p1
    EY2 = _NN*_p2
    EY3 = _NN*_p3
    EY12 = _NN*_p1*(1.-_p1) + (_NN*_p1)**2
    EY22 = _NN*_p2*(1.-_p2) + (_NN*_p2)**2
    EY32 = _NN*_p3*(1.-_p3) + (_NN*_p3)**2
    EY1Y2 = _NN*(_NN-1.)*_p1*_p2
    EY1Y3 = _NN*(_NN-1.)*_p1*_p3
    EY2Y3 = _NN*(_NN-1.)*_p2*_p3

```

```
27      x1 = list(_x[0])
28      x2 = list(_x[1])
29      x3 = list(_x[2])
30      begin = clock()
31
32      _b[:,0] = array([1., 1., 1.])/3
33      _S[1] = _b[0,0]*x1[0] + _b[1,0]*x2[0] + _b[2,0]*x3[0]
34
35      for n in range(order,_N):
36          if echo: start = clock()
37
38          numer = x1[n]*x1[n-1]*EY12 + x2[n]*x2[n-1]*EY22 + x3[n]*x3[n-1]*EY32
39          numer += (x1[n]*x2[n-1] + x2[n]*x1[n-1])*EY1Y2
40          numer += (x1[n]*x3[n-1] + x3[n]*x1[n-1])*EY1Y3
41          numer += (x2[n]*x3[n-1] + x3[n]*x2[n-1])*EY2Y3
42          denom = _NN*(x1[n-1]*EY1 + x2[n-1]*EY2 + x3[n-1]*EY3)
43
44          dotbx = numer/denom
45          _S[n+1] = dotbx * _S[n]
46
47          b1 = x1[n-1]*EY12 + x2[n-1]*EY1Y2 + x3[n-1]*EY1Y3
48          b2 = x1[n-1]*EY1Y2 + x2[n-1]*EY22 + x3[n-1]*EY2Y3
49          b3 = x1[n-1]*EY1Y3 + x2[n-1]*EY2Y3 + x3[n-1]*EY32
50          _b[:,n] = array([b1, b2, b3])/denom
51
52          if echo: print("Finish recursive step %3d in %.2f second.")
53
54      if debug: print("Total time spent: %g second.")
55
56      return _S[1:]
```

From line number 11 to line number 16 are the parameters of multinomial universal portfolio where $p_1 = p_1, p_2 = p_2$ and $NN = N$. Line number 17 to line number 25 are the moments of the multinomial universal portfolio (3.27) in Chapter 3 . Line number 17 is when $\nu = 1$, refer to (3.27) in Chapter 3. Line number 38 to number 42 represented the numerator and denominator of (3.15) and (3.16) in Chapter 3. Line number 44 to line number 45 are the wealth obtained after $n + 1$ trading days, refer to (3.17) in Chapter 3.

4.2.2 Order Two Multinomial Generated Universal Portfolio

```
from numpy import array, zeros, ones, arange
from time import clock

def Recursive3Multinomial(x,p1,p2,NN,debug=False,echo=False):
    _order = 2

    _N = len(x[0])
    _b = zeros((3,_N+1))
    _S = ones(_N+1)
    _x = x
    _p1,_p2,_p3,_NN = None,None,None,None
    _p1 = p1
    _p2 = p2
    _p3 = max(1.-p1-p2,0.)
    if any([p1<0,p2<0,_p3<0]):
        import sys
        print("Error: Parameter is less than 0!")
        sys.exit(1)

    _p12 = p1**2
    _p22 = p2**2
    _p32 = _p3**2
    _p13 = p1**3
    _p23 = p2**3
    _p33 = _p3**3
    _NN = NN
    EY1 = _NN*_p1
    EY2 = _NN*_p2
    EY3 = _NN*_p3
    EY12 = _NN*_p1*(1.-_p1) + (_NN*_p1)**2
    EY22 = _NN*_p2*(1.-_p2) + (_NN*_p2)**2
    EY32 = _NN*_p3*(1.-_p3) + (_NN*_p3)**2
    EY13 = _NN*(_NN-1)*(_NN-2)*_p13 + 3*_NN*(_NN-1)*_p12 + _NN*_p1
    EY23 = _NN*(_NN-1)*(_NN-2)*_p23 + 3*_NN*(_NN-1)*_p22 + _NN*_p2
    EY33 = _NN*(_NN-1)*(_NN-2)*_p33 + 3*_NN*(_NN-1)*_p32 + _NN*_p3
    EY1Y2 = _NN*(_NN-1.)*_p1*_p2
    EY1Y3 = _NN*(_NN-1.)*_p1*_p3
    EY2Y3 = _NN*(_NN-1.)*_p2*_p3
    EY12Y2 = _NN*(_NN-1)*(_NN-2)*_p12*_p2 + _NN*(_NN-1)*_p1*_p2
    EY22Y1 = _NN*(_NN-1)*(_NN-2)*_p22*_p1 + _NN*(_NN-1)*_p1*_p2
```

```
EY12Y3 = _NN*(_NN-1)*(_NN-2)*_p12*_p3 + _NN*(_NN-1)*_p1*_p3
EY32Y1 = _NN*(_NN-1)*(_NN-2)*_p32*_p1 + _NN*(_NN-1)*_p1*_p3
EY22Y3 = _NN*(_NN-1)*(_NN-2)*_p22*_p3 + _NN*(_NN-1)*_p2*_p3
EY32Y2 = _NN*(_NN-1)*(_NN-2)*_p32*_p2 + _NN*(_NN-1)*_p2*_p3
EY12Y3 = _NN*(_NN-1)*(_NN-2)*_p12*_p3 + _NN*(_NN-1)*_p3*_p1
EY32Y1 = _NN*(_NN-1)*(_NN-2)*_p32*_p1 + _NN*(_NN-1)*_p3*_p1
EY1Y2Y3 = _NN*(_NN-1)*(_NN-2)*_p1*_p2*_p3

x1 = _x[0]
x2 = _x[1]
x3 = _x[2]
begin = clock()

_b[:,0] = array([1., 1., 1.])/3
_S[1] = _b[0,0]*x1[0] + _b[1,0]*x2[0] + _b[2,0]*x3[0]
_b[:,1] = array([1., 1., 1.])/3
_S[2] = _S[1]*(_b[0,1]*x1[1] + _b[1,1]*x2[1] + _b[2,1]*x3[1])

for n in range(_order,_N):
    if echo: start = clock()

    numer = x1[n]*x1[n-1]*x1[n-2]*EY13 + x2[n]*x2[n-1]*x2[n-2]*EY23
    += x3[n]*x3[n-1]*x3[n-2]*EY33
    numer += (x1[n]*x1[n-1]*x2[n-2] + x1[n]*x2[n-1]*x1[n-2]
    +=x2[n]*x1[n-1]*x1[n-2])*EY12Y2
    numer += (x1[n]*x2[n-1]*x2[n-2] + x2[n]*x2[n-1]*x1[n-2]
    +=x2[n]*x1[n-1]*x2[n-2])*EY22Y1
    numer += (x1[n]*x3[n-1]*x1[n-2] + x1[n]*x1[n-1]*x3[n-2]
    +=x3[n]*x1[n-1]*x1[n-2])*EY12Y3
    numer += (x1[n]*x3[n-1]*x3[n-2] + x3[n]*x3[n-1]*x1[n-2]
    +=x3[n]*x1[n-1]*x3[n-2])*EY32Y1
    numer += (x2[n]*x2[n-1]*x3[n-2] + x2[n]*x3[n-1]*x2[n-2]
    +=x3[n]*x2[n-1]*x2[n-2])*EY22Y3
    numer += (x3[n]*x2[n-1]*x3[n-2] + x3[n]*x3[n-1]*x2[n-2]
    +=x2[n]*x3[n-1]*x3[n-2])*EY32Y2
    numer += (x1[n]*x3[n-1]*x2[n-2] + x1[n]*x2[n-1]*x3[n-2]
    +=x2[n]*x3[n-1]*x1[n-2] \
    + x2[n]*x1[n-1]*x3[n-2]+x3[n]*x1[n-1]*x2[n-2]+x3[n]*x2[n-1]*x1[n-2])*EY1Y2Y3
    denom = _NN*(x1[n-1]*x1[n-2]*EY12 + x1[n-1]*x2[n-2]*EY1Y2 + x1[n-1]*x3[n-2]*EY1Y3 \
    + x2[n-1]*x1[n-2]*EY1Y2+x2[n-1]*x2[n-2]*EY22+x2[n-1]*x3[n-2]*EY2Y3+x3[n-1] \
        * x1[n-2]*EY1Y3+x3[n-1]*x2[n-2]*EY2Y3+x3[n-1]*x3[n-2]*EY32)

    dotbx = numer/denom
    _S[n+1] = dotbx * _S[n]
```

```
85
86          b1 = x1[n-1]*x1[n-2]*EY13 + x1[n-1]*x2[n-2]*EY12Y2 + x1[n-1]*x3[n-2]*EY12Y3
87          += x2[n-1]*x1[n-2]*EY12Y2 + x2[n-1]*x2[n-2]*EY22Y1
88          b1 += x2[n-1]*x3[n-2]*EY1Y2Y3 + x3[n-1]*x1[n-2]*EY12Y3
89           += x3[n-1]*x2[n-2]*EY1Y2Y3 + x3[n-1]*x3[n-2]*EY32Y1
90          b2 = x1[n-1]*x1[n-2]*EY12Y2 + x1[n-1]*x2[n-2]*EY22Y1
91          += x1[n-1]*x3[n-2]*EY1Y2Y3 + x2[n-1]*x1[n-2]*EY22Y1
92          += x2[n-1]*x2[n-2]*EY23
93          b2 += x2[n-1]*x3[n-2]*EY22Y3 + x3[n-1]*x1[n-2]*EY1Y2Y3
94          += x3[n-1]*x2[n-2]*EY22Y3 + x3[n-1]*x3[n-2]*EY32Y2
95          b3 = x1[n-1]*x1[n-2]*EY12Y3 + x1[n-1]*x2[n-2]*EY1Y2Y3
96          += x1[n-1]*x3[n-2]*EY32Y1 + x2[n-1]*x1[n-2]*EY1Y2Y3
97          += x2[n-1]*x2[n-2]*EY22Y3
98          b3 += x2[n-1]*x3[n-2]*EY32Y2 + x3[n-1]*x1[n-2]*EY32Y1
99          + =x3[n-1]*x2[n-2]*EY32Y2 + x3[n-1]*x3[n-2]*EY33
100
101         _b[:,n] = array([b1, b2, b3])/denom
102
103     if echo: print("Finish recursive step %3d in %.2f second." % (n,clock()-start))
104
105     print("Total time spent: %g second." % (clock()-begin))
106     return _S[1:]
```

Similary, in order two multinomial universal portfolio code, from line number 11 to line number 26 are the parameters of multinomial universal portfolio where $p_1 = p_1, p_2 = p_2$ and $NN = N$. Line number 27 to line number 47 are the moments of order two multinomial generated universal portfolio, refer to (3.28) in Chapter 3. Line number 62 to number 81 represented the numerator and denominator, refer to (3.19) and (3.20) in Chapter 3. While line number 84 is the wealth obtained after $n + 1$ trading days, (3.21) in Chapter 3.

From the above two python codes have shown that the calculation of numerator and denominator of portfolios become complicated when order increase. These are due to the moments are increasing when order ν increase. Therefore, only the order one of the other three proposed universal portfolios will be shown and discussed in the following section.

4.3 Order One Multivariate Normal Generated Universal Portfolio

```
from numpy import array, zeros, arange
from time import clock

def Recursive3MultivariatNormal(x,p1,p2,p3,NN,debug=False,echo=False):
    _N = len(x[0])
    _b = zeros((3,_N+1))
    _S = zeros(_N+1)
    _S[0] = 1.
    _x = x
    _p1,_p2,_p3,_NN = None,None,None,None
    order = 1
    _p1 = p1
    _p2 = p2
    _p3 = p3
    _p12 = p1**2
    _p22= p2**2
    _p32 = p3**2
    _NN = NN

    EY1 = _p1
    EY2 = _p2
    EY3 = _p3
    EY12 = _p12 + _NN
    EY22 = _p22 + _NN
    EY32 = _p32 + _NN
    EY1Y2 = _p1*_p2
    EY1Y3 = _p1*_p3
    EY2Y3 = _p2*_p3

    x1 = list(_x[0])
    x2 = list(_x[1])
    x3 = list(_x[2])

    begin = clock()
```

```
35
36      _b[:,0] = array([1., 1., 1.])/3
37      _S[1] = _b[0,0]*x1[0] + _b[1,0]*x2[0] + _b[2,0]*x3[0]
38
39      for n in range(order,_N):
40          if echo: start = clock()
41
42          numer = x1[n]*x1[n-1]*EY12 + x2[n]*x2[n-1]*EY22 + x3[n]*x3[n-1]*EY32
43          numer += (x1[n]*x2[n-1] + x2[n]*x1[n-1])*EY1Y2
44          numer += (x1[n]*x3[n-1] + x3[n]*x1[n-1])*EY1Y3
45          numer += (x2[n]*x3[n-1] + x3[n]*x2[n-1])*EY2Y3
46          denom = x1[n-1]*EY12 + x2[n-1]*EY22 + x3[n-1]*EY32
47          denom += (x2[n-1] + x1[n-1])*EY1Y2
48          denom += (x3[n-1] + x1[n-1])*EY1Y3
49          denom += (x3[n-1] + x2[n-1])*EY2Y3
50
51          dotbx = numer/denom
52          _S[n+1] = dotbx * _S[n]
53
54          b1 = x1[n-1]*EY12 + x2[n-1]*EY1Y2 + x3[n-1]*EY1Y3
55          b2 = x1[n-1]*EY1Y2 + x2[n-1]*EY22 + x3[n-1]*EY2Y3
56          b3 = x1[n-1]*EY1Y3 + x2[n-1]*EY2Y3 + x3[n-1]*EY32
57          _b[:,n] = array([b1, b2, b3])/denom
58
59          if echo: print("Finish recursive step %3d in %.2f second.")
60
61      if debug: print("Total time spent: %g second.")
62      return _S[1:]
```

From the above code, line number 10 to line number 18 are the parameters of multivariate normal universal portfolio where $\mu_1 = p_1, \mu_2 = p_2, \mu_3 = p_3$ and $\sigma_1^2 = \sigma_2^2 = \sigma_3^2 = \sigma = NN$. Line number 20 to line number 28 are the moments of the Multivariate normal generated universal portfolio (3.33) in Chapter 3. Line number 42 to number 49 represented the numerator and denominator where the numerator is (3.14) in Chapter 3. While the wealth obtained is in line number 52, refer (3.17) in Chapter 3.

4.4 Order One Brownian-motion Generated Universal Portfolio

```
from numpy import array, zeros, arange
from time import clock

def Recursive3bronianmotion (x,p1,p2,p3,NN,MM,QQ,debug=False,echo=False):
    _N = len(x[0])
    _b = zeros((3,_N+1))
    _S = zeros(_N+1)
    _S[0] = 1.
    _x = x

    order = 1
    _p1 = p1
    _p2 = p2
    _p3 = p3
    _p12 = p1**2
    _p22= p2**2
    _p32 = p3**2
    _NN = NN
    _MM = MM
    _QQ = QQ
    x1 = list(_x[0])
    x2 = list(_x[1])
    x3 = list(_x[2])

    begin = clock()
    _b[:,0] = array([1., 1., 1.])/3
    _S[1] = _b[0,0]*x1[0] + _b[1,0]*x2[0] + _b[2,0]*x3[0]

    for n in range(order,_N):
        if echo: start = clock()

        EY1 = n*_p1
        EY2 = n*_p2
        EY3 = n*_p3
        EY12 = n*n*_p12 + n*_NN*_NN
        EY22 = n*n*_p22 + n*_MM*_MM
        EY32 = n*n* _p32 + n*_QQ*_QQ
        EY1Y2 = n*n*_p1*_p2
        EY1Y3 = n*n*_p1*_p3
        EY2Y3 = n*n*_p2*_p3
```

```
41
42         numer = x1[n]*x1[n-1]*EY12 + x2[n]*x2[n-1]*EY22 + x3[n]*x3[n-1]*EY32
43         numer += (x1[n]*x2[n-1] + x2[n]*x1[n-1])*EY1Y2
44         numer += (x1[n]*x3[n-1] + x3[n]*x1[n-1])*EY1Y3
45         numer += (x2[n]*x3[n-1] + x3[n]*x2[n-1])*EY2Y3
46         denom = x1[n-1]*EY12 + x2[n-1]*EY22 + x3[n-1]*EY32
47         denom += (x2[n-1] + x1[n-1])*EY1Y2
48         denom += (x3[n-1] + x1[n-1])*EY1Y3
49         denom += (x3[n-1] + x2[n-1])*EY2Y3
50
51         dotbx = numer/denom
52         _S[n+1] = dotbx * _S[n]
53
54         b1 = x1[n-1]*EY12 + x2[n-1]*EY1Y2 + x3[n-1]*EY1Y3
55         b2 = x1[n-1]*EY1Y2 + x2[n-1]*EY22 + x3[n-1]*EY2Y3
56         b3 = x1[n-1]*EY1Y3 + x2[n-1]*EY2Y3 + x3[n-1]*EY32
57         _b[:,n] = array([b1, b2, b3])/denom
58
59         if echo: print("Finish recursive step %3d in %.2f second." )
60
61     if debug: print("Total time spent: %g second." )
62     return _S[1:]
```

From the above code, line number 12 to line number 20 are the parameters of Brownian motion generated universal portfolio where $\mu_1 = p_1, \mu_2 = p_2, \mu_3 = p_3$ and $\sigma_1^2 = NN, \sigma_2^2 = MM, \sigma_3^2 = QQ$. Line number 32 to line number 40 are the moments of the universal portfolio generated by Brownian-motion, refer (3.42) in Chapter 3. Line number 42 to number 49 represented the numerator and denominator of the portfolio (3.41) in Chapter 3. While Line number 52 is the calculation of the wealth obtained after $n + 1$ trading days, (3.17) in Chapter 3.

4.5 Order One Universal Portfolio Generated by Ornstein Uhlenbeck Process

```
from numpy import array, zeros, arange
from time import clock

def Recursive3ornstein(x,p1,p2,p3,debug=False,echo=False):

    _N = len(x[0])
    _b = zeros((3,_N+1))
    _S = zeros(_N+1)
    _S[0] = 1.
    _x = x
    _p1,_p2,_p3,_NN,_MM,_QQ =,None,None,None,None,None, None
    Order =1
    _p1 = p1
    _p2 = p2
    _p3 = p3
    _NN = NN
    _MM = MM
    _QQ = QQ
    _p12 = p1**2
    _p22 = p2**2
    _p32 = p3**2
    _p13 = p1**3
    _p23 = p2**3
    _p33 = p3**3

    EY1 = _p1
    EY2 = _p2
    EY3 = _p3
    EY12 = 1 + _p12
    EY22 = 1+ _p22
    EY32 =  1+  _p32
    EY1Y2 = _p1*_p2
    EY1Y3 = _p1*_p3
    EY2Y3 = _p2*_p3

    x1 = list(_x[0])
    x2 = list(_x[1])
    x3 = list(_x[2])
```

```
39
40      begin = clock()
41
42      _b[:,0] = array([1., 1., 1.])/3
43      _S[1] = _b[0,0]*x1[0] + _b[1,0]*x2[0] + _b[2,0]*x3[0]
44
45      for n in range(order,_N):
46          if echo: start = clock()
47
48          numer = x1[n]*x1[n-1]*EY12 + x2[n]*x2[n-1]*EY22 + x3[n]*x3[n-1]*EY32
49          numer += (x1[n]*x2[n-1] + x2[n]*x1[n-1])*EY1Y2
50          numer += (x1[n]*x3[n-1] + x3[n]*x1[n-1])*EY1Y3
51          numer += (x2[n]*x3[n-1] + x3[n]*x2[n-1])*EY2Y3
52          denom = x1[n-1]*EY12 + x2[n-1]*EY22 + x3[n-1]*EY32
53          denom += (x2[n-1] + x1[n-1])*EY1Y2
54          denom += (x3[n-1] + x1[n-1])*EY1Y3
55          denom += (x3[n-1] + x2[n-1])*EY2Y3
56
57          dotbx = numer/denom
58          _S[n+1] = dotbx * _S[n]
59
60          b1 = x1[n-1]*EY12 + x2[n-1]*EY1Y2 + x3[n-1]*EY1Y3
61          b2 = x1[n-1]*EY1Y2 + x2[n-1]*EY22 + x3[n-1]*EY2Y3
62          b3 = x1[n-1]*EY1Y3 + x2[n-1]*EY2Y3 + x3[n-1]*EY32
63
64          _b[:,n] = array([b1, b2, b3])/denom
65
66          if echo: print("Finish recursive step %3d in %.2f second." )
67
68      if debug: print("Total time spent: %g second." )
69      return _S[1:]
```

From the above code, from line number 13 to line number 24 are the parameters of universal portfolio generated by Ornstein Uhlenbeck process where $\mu_1 = p_1, \mu_2 = p_2, \mu_3 = p_3$ and $\alpha_1 = NN, \alpha_2 = MM, \alpha_3 = QQ$. Line number 26 to line number 34 are the moments of the universal portfolio generated by Ornstein uhlenbeck process, refer (3.60) in Chapter 3. Line number 48 to number 55 represented the numerator and denominator of (3.59) in Chapter 3. While, Line number 58 is the wealth obtained after $n + 1$ trading days, (3.17) in Chapter 3.

CHAPTER 5

IDENTIFY "GOOD PARAMETERS"

In the preliminary stage of this research, a numerical experiment was done on the Dirichlet distribution based universal portfolio proposed in Cover (1991) for $m = 3$, i.e. three stocks. The Tan and Chan algorithm (Chan 2002) was found hardly worked for more than 500 trading days, i.e. $n \geq 500$. To resolve this matter, Tan et al. (2012) proposed finite order universal portfolio strategies. Much of Tan's universal porfolio strategies are applied to a few selected stocks (refer to the literature review in Chapter 1). While we will be investigating how various strategies work on all combination of stocks.

5.1 Performance of Five Universal Portfolio Strategies for Special Sets of Stocks

In all these numerical experiments, an initial wealth of 1 is assumed. All calculations with the proposed strategies were obtained by using Python 2.7 and NumPy on a MacBook Pro with MacOS/X Version 10.8.3.

5.1.1 Performance of Dirichlet-Distribution-Based Universal Portfolio

Universal portfolio strategies can only perform well with the right parameters. In this numerical experiment, 3 sets of stocks data were selected randomly from KLSE (Kuala Lumpur Stock Exchange) and Yahoo Finance (Yahoo n.d.) for the period from 1 January, 2003 to 30 November, 2004, consisting of 500 trading days. Each set consists of 3 company stocks. Set A consists of the stocks of Malayan Banking, Genting and Amway(M) Holdings. Set B consists of the stocks of Public Bank, Sunrise and YTL Corporation. Lastly, set C consists of the stocks of Hong Leong Bank, RHB Capital and YTL Corporation. Refer to Appendix A.

Cover & Ordentlich (1996) presented an algorithm for generating the Dirichlet $(\frac{1}{2}, \frac{1}{2})$ two stock universal portfolio. Chan (2002) modified this algorithm for generating any Dirichlet $(\alpha_1, \alpha_2, \cdots, \alpha_m)$ universal portfolio for $m = 2, 3$ and 4 stock. A general algorithm capable of handling the m-stock Dirichlet $(\alpha_1, \alpha_2, \cdots, \alpha_m)$ universal portfolio was proposed by Tan (2004b). We used the modified algorithm of Chan (2002) for computing the three-stock universal portfolio generated by Diriclet $(\alpha_1, \alpha_2, \alpha_3)$ distribution where $\alpha_j > 0$ for $j = 1, 2, 3$, with the starting capital S_0 to be 1 unit and the initial portfolio $b_0 = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$. We can compute the investment capitals achieved by Dirichlet $(\alpha_1, \alpha_2, \alpha_3)$ Cover-Ordentlich universal portfolios for set A, set B and set C. Table 5.1 show the wealth obtained by the Dirichlet $(\alpha_1, \alpha_2, \alpha_3)$ where parameter are chosen as $\alpha_1 = 0.01, \alpha_2 = 0.01$ and $\alpha_3 = 8000$. From Table 5.1, the wealth obtained S_{500} is 1.8750 for data set A, 4.0761 for data set B and 4.0761 for data set C with the computation time in second.

Table 5.1: The wealth S_{500} achieved by the Dirichlet (0.01,0.01,8000) UP for data set A,B and C.

	S_{500}	Computation Time (Seconds)
Set A	1.8750	394.77
Set B	4.0761	394.98
Set C	4.0761	394.38

5.1.2 Empirical Performance of Multinomial Distribution Generated Universal Portfolio

The computation of the Dirichlet universal portfolios requires a substantial amount of computer and longer computation time. To overcome this problem, the memory-saving Dirichlet universal portfolios of finite order is introduced in Tan et al. (2012). The finite-order multinomial universal portfolios having the faster computation time and reduced the computer- memory saving is studied.

A numerical experiment was carried out for the finite-order Multinomial universal portfolios on same three stock-price data sets A, B and C selected from the Kuala Lumpur Stock Exchange. The wealth S_{500} achieved by selected parameters of the multinomial universal portfolio(p_1, p_2, N) for algorithm given in Chapter 3 are shown in Table 5.2, Table 5.3 and Table 5.4 respectively. The computation times of the finite-order Multinomial universal portfolios are 0.014, 0.022 and 0.052 seconds for the first, second and third order respectively. From Table 5.1, the computation times of the Dirichlet universal portfolios are 394.77, 394.98 and 394.38 seconds with the maximum wealth of 1.8750, 4.0761 and 4.0761 for data set A, B and C. Thus, the wealth achieved by the finite-order multinomial universal portfolios are comparable to that of the moving-order Dirichlet universal portfolio with substantial savings in computation time. Note that, the order one Multinomial universal portfolio has better performance.

Table 5.2: The wealth S_{500} achieved by the MNUP (0.8,0.01,600) for data Set A.

	S_{500}	Computation Time (Seconds)
order 1	2.0238	0.014
order 2	2.0362	0.022
order 3	2.0315	0.052

Table 5.3: The wealth S_{500} achieved by the MNUP (0.000001,0.000001,600) for data Set B.

	S_{500}	Computation Time (Seconds)
order 1	4.0761	0.014
order 2	4.0354	0.022
order 3	4.0217	0.052

Table 5.4: The wealth S_{500} achieved by the MNUP (0.000001,0.000001,600) for data Set C.

	S_{500}	Computation Time (Seconds)
order 1	4.0761	0.014
order 2	4.0490	0.022
order 3	4.0218	0.052

5.1.3 Empirical Performance of Multivariate Normal Distribution Generated Universal Portfolio

We now investigate the finite-order multivariate normal universal portfolios. The finite-order multivariate normal universal portfolios are run on same selected three data sets with the selected values of parameter $(\mu_1, \mu_2, \mu_3, \sigma_1^2 = \sigma_2^2 = \sigma_3^2 = \sigma)$ listed in Table 5.5, Table 5.6 and Table 5.7. Empirically, from the Table 5.5, Table 5.6 and Table 5.7, we observed the performance of the finite-order multivariate normal universal portfolios is comparable to that of the Dirichlet universal portfolios and yet requiring substantially less computation time.

Table 5.5: The wealth S_{500} achieved by the MVNUP (7.1,0.8,0.3,1.0) for data Set A.

	S_{500}	Computation Time (Seconds)
order 1	1.9393	0.014
order 2	1.9442	0.024
order 3	1.9405	0.063

Table 5.6: The wealth S_{500} achieved by the MVNUP (0.01,0.01,1000,1.2) for data Set B.

	S_{500}	Computation Time (Seconds)
order 1	4.0761	0.015
order 2	4.0353	0.024
order 3	4.0218	0.062

Table 5.7: The wealth S_{500} achieved by the MVNUP (0.01,0.01,1000,1.2) for data Set C.

	S_{500}	Computation Time (Seconds)
order 1	4.0761	0.015
order 2	4.0490	0.024
order 3	4.0219	0.062

5.1.4 Empirical Performance of Brownian-motion Generated Universal Portfolio

In this subsection, we investigate the universal portfolio generated by three independent Brownian motions with drift coefficients μ_1, μ_2, μ_3 and variance parameters $\sigma_1^2, \sigma_2^2, \sigma_3^2$ respectively are run on the same sets A, B and C. In the following tables, the notation $(\mu_1, \sigma_1^2; \mu_2, \sigma_2^2; \mu_3, \sigma_3^2)$ is used to denote the parameters of the 3 generating Brownian motions. For three sets of selected parameters $(1, 1.4; 1, 1.5; 1, 1.2)$, $(1, 1.4; 10, 2.5; 100, 3.6)$ and $(25, 1.4; 0.5, 1.5; 0.5, 1.2)$ the Brownian motion universal portfolios of orders 1, 2, 3 are run on the data sets A, B and C. Assuming an initial wealth of 1 unit, the wealth S_{500} achieved by the portfolios after 500 trading days and the implementation times for data sets A, B and C are displayed in Table 5.8, Table 5.9 and Table 5.10 respectively. Comparing the wealth achieved by the multinomial universal portfolios in Section 5.1.2 and the wealth achieved in Table 5.8, Table 5.9 and Table 5.10, its is observed that 4.0335, 3.9666 and 3.9194 units are achieved by the orders 1,2 and 3 portfolios respectively for data set B in Table 5.9. The corresponding wealth of 4.0761, 4.0354 and 4.0217 units are achieved by the (0.000001,0.000001,600) multinomial universal portfolio for data set B. In Table 5.8, lower wealth of 1.6832, 1.6832 and 1.5886 units are achieved by the orders 1, 2 and 3 Brownian-motion portfolios respectively for data set A. The corresponding higher wealth of 2.0238, 2.0362 and 2.0315 units are achieved by the (0.8,0.01,600) multinomial universal portfolio for data set A.

Table 5.8: The wealth S_{500} achieved by the BMUP (1,1.4;1,1.5;1,1.2) for data Set A.

	S_{500}	Computation Time (Seconds)
order 1	1.6832	0.042
order 2	1.6832	0.065
order 3	1.5886	0.211

Table 5.9: The wealth S_{500} achieved by the BMUP (1,1.4;10,2.5;100,3.6) for data Set B.

	S_{500}	Computation Time (Seconds)
order 1	4.0335	0.025
order 2	3.9666	0.067
order 3	3.9194	0.208

Table 5.10: The wealth S_{500} achieved by the BMUP (25,1.4;0.5,1.5;0.5,1.2) for data Set C.

	S_{500}	Computation Time (Seconds)
order 1	1.4230	0.027
order 2	1.4142	0.067
order 3	1.4451	0.210

5.1.5 Empirical Performance of Universal Portfolio Generated by Ornstein-Uhlenbeck Process

The Ornstein-Uhlenbeck process based universal portfolio is studied with the three stock-price data sets A, B and C .The parametric vector of this universal portfolio is $(\mu_1, \mu_2, \mu_3; \alpha_1, \alpha_2, \alpha_3)$. By trial and error, parametric vectors (16,0.1,0.5; 0.2,1.2,2), (0.001,0.001,50; 0.2,1.2,2) and (0.005,0.009,99; 0.2,1.2,2) were found to be doing well for data sets A, B and C respectively. The wealth S_{500} and computation time in second after 500 trading days achieved by the orders 1, 2 and

3 portfolios for data sets A, B and C are displayed in Table 5.11,Table 5.12 and Table 5.13, respectively. The wealth achieved by the portfolios is from 2.035 to 2.037, 4.069 to 4.073 and 4.073 to 4.075 for data sets A, B and C respectively.

Table 5.11: The wealth S_{500} achieved by the portfolios generated by the OU process with parametric vector (16,0.1,0.5;0.2,1.2,2) for data Set A.

	S_{500}	Computation Time (Seconds)
order 1	2.0370	0.015
order 2	2.0368	0.072
order 3	2.0358	0.332

Table 5.12: The wealth S_{500} achieved by the portfolios generated by the OU process with parametric vector (0.001,0.001,50;0.2,1.2,2) for data Set B.

	S_{500}	Computation Time (Seconds)
order 1	4.0732	0.010
order 2	4.0711	0.077
order 3	4.0696	0.325

Table 5.13: The wealth S_{500} achieved by the portfolios generated by the OU process with parametric vector (0.005,0.009,99;0.2,1.2,2) for data Set C.

	S_{500}	Computation Time (Seconds)
order 1	4.0746	0.011
order 2	4.0739	0.074
order 3	4.0734	0.334

5.2 Identifying "Good" Parameters

From the above results, four finite order universal portfolios strategies were found can perform better than the Cover Ordentlich Dirichlet universal portfolio for same data sets. Noticed that the four universal portfolios with order one performed slightly better than order two and three. Therefore, this research will be limited to order one universal portfolios.

5.2.1 Representation Data

For the following sections, selection of 10 set of parameters for each universal portfolios are described. Python code are written to obtain the good performance parameters among these 10 set of parameters. We use CRP as a benchmark to evaluate and to compare with these four universal portfolio strategies. A comparison can be made with the wealth obtained by CRP and the wealth achieved by the four proposed universal portfolios. Every one year period starting from year 2000 to year 2015 of the available stock data listed in Table 2.1 in Chapter 2 are used for study. At least 20% of the wealth achieved by proposed universal portfolio performed better than wealth obtained by CRP are analysed. The good performance of the parameters is observed.

5.2.2 Parameters for Multinomial Universal Portfolio

10 set of parameters (p_1, p_2, N) are used for selection of good performances, they are A = (0.8, 0.001, 600), B = (0.001,0.8,600), C =(0.5, 0.5, 600), D = (0.9,0.1, 600), E = (0.1,0.9,600), F = (0.8,0.001,50), G = (0.001,0.8,50), H = (0.05,0.95,99), I = (0.95,0.05,99) and J = (0.9,0.004 10). In Section 5.1.2, (0.8,0.001,600) performed well when ran on 3 selected stocks from KLSE. Therefore, this set of

parameter chosen for the analysis. The other nine set of parameters are formed by varying among (0.8,0.001,600).

5.2.3 Parameters for Multivariate Normal Universal Portfolio

In Section 5.1.3, good performances obtained by order one universal portfolio generated by Multivariate Normal Distribution with the set of (7.1,0.8,0.2,1.0). Therefore, the 10 sets of parameters are selected varying among (7.1,0.8,0.2,1.0), i.e. K = (7.1,0.8,0.2,1.0), L = (0.2,0.8,7.1,1.0), M = (0.8,7.1,0.2,1.0), N = (7.1,0.2, 0.8,1.0), O = (0.8,0.2,7.1,1.0), P = (0.2,7.1,0.8,1.0), Q = (8.1,9.1,0.1,2.0), R = (0.2,8.1,9.1,2.0), S = (8.1,0.,2,9.1,2.0) and T = (9.0,0.1,0.1,2.0).

5.2.4 Parameters for Brownian-motion Generated Universal Portfolio

In Section 5.1.4, good performances parameter of order one universal portfolio generated by Brownian-motion is (1,1.4;10,2.5;100,3.6), therefore, the 10 sets of parameters are formed among (1,1.4;10,2.5;100,3.6), i.e. a = (1,1.4;10,2.5;100,3.6), b = (100,1.4;1,2.5;10,3.6), c = (100,1.4;10,2.5;1,3.6), d = (10,1.4;1,2.5;100,3.6), e = (1,1.4;100,2.5;10,3.6), f = (10,1.4;100,2.5;1,3.6), g = (1,1.4;10,3.6;100,2.5), h = (1,2.5;10,1.4;100,3.6), i = (1,2.5;10,3.6;100,1.4) and j = (1,3.6;10,1.4;100, 2.5).

5.2.5 Parameters for Universal Portfolio Generated by Ornstein Uhlenbeck Process

From Section 5.1.5, one of the parameters selected of is (16,0.1,0.5,0.2,1.2,2) due to good performance is obtained. The other 9 set of parameters we formed by varying among (16,0.1,0.5,0.2,1.2,2). The 10 set of parameters are k = (16,0.1,0.5, 0.2,1.2,2), l = (16,0.5,0.1,0.2,1.2,2), m = (0.5,0.1,16,0.2,1.2,2), n = (0.5,16,0.1,0.2, 1.2,2), o = (0.1,16,0.5,0.2,1.2,2), p = (0.1,0.5,16,0.2,1.2,2), q = (16,0.1,0.5,2,1.2, 0.2), r =(16,0.5,0.1,2,1.2,0.2), s = (50, 0.5, 2, 0.2, 1.2, 2) and t = (50,2,0.5,0.2,1.2,2).

5.3 Numerical Experiment

5.3.1 Computer Program

The chosen parameters in the previous section will be used for numerical computation. The Python code is written as follow:

```
import pandas as pd
import numpy as np

df = pd.read_csv('data00-15/klse100roi.csv')

to_find_list = ["3182", "5148", "5398", "5517", "7106", \
                "7084","1818","5031","2216","5099"]

stock_data.fillna(1.0,inplace=True)

```

```
def CRP(x, param=[]):
    num_cols = len(x)
    _x = [list(x[i]) for i in range(num_cols)]
    num_rows = len(_x[0])
    Se = np.ones(num_rows)
    if len(param)>0 and (sum(param)-1.0)<1e-9:
        Se[0] = sum([param[i]*_x[i][0] for i in range(num_cols)])
        for n in range(num_rows-1):
            daily_roi_p = sum([param[i]*_x[i][n] \
                                   for i in range(num_cols)])
            if np.isnan(daily_roi_p):
                print("\nAt n =", n, "_x =", \
                      [_x[i][n] for i in range(num_cols)])
                return None
            Se[n+1] = Se[n]*daily_roi_p
    else:
        Se[0] = (x1[0]+x2[0]+x3[0])/3
        for n in range(num_rows-1):
            Se[n+1] = Se[n]*(x1[i]+x2[i]+x3[i])/3
            if np.isnan(daily_roi_p):
                print("\nAt n =", n, "_x =", \
                      [_x[i][n] for i in range(num_cols)])
                return None
    return Se

from portfolio2015.recur3multinomial import Recursive3Multinomial
from portfolio2015.recur3multivariatNormal import Recursive3MultivariatNormal
from portfolio2015.recur3bronianmotion import Recursive3bronianmotion
from portfolio2015.recur3ornstein import Recursive3ornstein

for i, j, k in itertools.combinations(stock_roi, 3):
    x = [list(stock_data[i]),list(stock_data[j]),list(stock_data[k])]

    print("%s,%s,%s,CRP,\"%s\"" % (i,j,k,[])),
    crp_res = CRP(x,[1./3,1./3,1./3])
    if crp_res is not None:
        print(",%f"%(crp_res[-1]))
    else:
        print("Data Error")

     parameters = [
        [0.8,   0.001, 600], [0.001, 0.8,   600],
        [0.5,   0.5,   600], [0.9,   0.1,   600],
        [0.1,   0.9,   600], [0.8,   0.001,  50],
```

```
        [0.001, 0.8,    50], [0.05,  0.95,   99],
        [0.95,  0.05,   99], [0.9,   0.004,  10],
    ]
    for p1, p2, NN in parameters:
        print("%s,%s,%s,MultiNomial,\"%s\"" % (i,j,k,[p1,p2,NN])),
        res = Recursive3Multinomial(x,p1,p2,NN)
        if res is not None:
            print(",%f"%res[-1])
        else:
            print("Data Error")

    parameters = [
        [7.1,0.8,0.2,1.0], [0.2,0.8,7.1,1.0],
        [0.8,7.1,0.2,1.0], [7.1,0.2,0.8,1.0],
        [0.8,0.2,7.1,1.0], [0.2,7.1,0.8,1.0],
        [8.1,9.1,0.2,2.0], [0.2,8.1,9.1,2.0],
        [8.1,0.2,9.1,2.0], [9.0,0.1,0.1,2.0],
    ]

    for p1, p2, p3, NN in parameters:
        print("%s,%s,%s,MultivariatNormal,\"%s\"" % (i,j,k,[p1,p2,p3,NN])),
        res = Recursive3MultivariatNormal(x,p1,p2,p3,NN)
        if res is not None:
            print(",%f"%res[-1])
        else:
            print("Data Error")

    parameters = [
        [1,10,100,1.4,2.5,3.6], [100,1,10,1.4,2.5,3.6],
        [100,10,1,1.4,2.5,3.6], [10,1,100,1.4,2.5,3.6],
        [1,100,10,1.4,2.5,3.6], [10,100,1,1.4,2.5,3.6],
        [1,10,100,1.4,3.6,2.5], [1,10,100,2.5,1.4,3.6],
        [1,10,100,2.5,3.6,1.4], [1,10,100,3.6,1.4,2.5],
    ]
    for p1, p2, p3 in parameters:
        print("%s,%s,%s,bronianmotion,\"%s\"" % (i,j,k,[p1,p2,p3])),
        res = Recursive3bronianmotion(x,p1,p2,p3,NN,MM,QQ)
        if res is not None:
            print(",%f"%res[-1])
        else:
            print("Data Error")

    parameters = [
        [16,0.1,0.5,0.2,1.2,2], [16,0.5,0.1,0.2,1.2,2],
```

```
99          [0.5,0.1,16,0.2,1.2,2], [0.5,16,0.1,0.2,1.2,2],
100         [0.1,16,0.5,0.2,1.2,2], [0.1,0.5,16,0.2,1.2,2],
101         [16,0.1,0.5,0.2,1.2,2], [16,0.5,0.1,0.2,1.2,2],
102         [50,0.5,2,0.2,1.2,2],   [50,2,0.5,0.2,1.2,2],
103     ]
104     for p1, p2, p3 in parameters:
105         print("%s,%s,%s,ornstein,\"%s\"" % (i,j,k,[p1,p2,p3])),
106         res = Recursive3ornstein(x,p1,p2,p3,NN,MM,QQ)
107         if res is not None:
108             print(",%f"%res[-1])
109         else:
110             print("Data Error")
```

From the above python code, line number 4 is reading all the 95 stocks data from the CSV file which has been generated in Section 2.3 in Chapter 2. In line number 6 is to pick the 10 most active stocks. While in line number 9 is describing the data collected from Yahoo KLSE stocks are incomplete and sometimes wrong. In line number 41 is combination of 3 stocks which is described in Chapter 4.

The chosen parameters for each portfolios are listed from line number 52 to line number 56 for Multinomial generated universal portfolio, line number 67 to number 71 for Multivariate Normal generated universal portfolio, line number 83 to line number 87 for Brownian-motion generated universal portfolio and line number 98 to line number 102 are for Ornstein Uhlenbeck process generated universal portfolio.

5.3.2 Numerical Result and Analysis

The number of stocks and trading period will be vary based on the available data and trading period listed in Table 2.1 in Chapter 2. From period 1 January 2015 to 31 December 2015, 10 most active stocks data listed in Table 2.2 in Chapter 2 with codes 5398, 7084, 1818, 5517, 7106, 5031, 2216, 5148, 5099 and 3182 will be studied, and the 120 sets consists of 3 stocks data have been generated from the ${}^{10}C_3$ combination. At least 20% out of total 120 sets with the performance better the CRP portfolio strategies will be observed and studied. Only stocks data with codes 5398, 7084, 5517, 7106, 5031, 2216, 5148, 3182 are available for study from year 2004 to year 2015, refer to Table 2.2 in Chapter 2. Total of combination is 56 sets data and 20% of the good performance parameters strategie is observed. From year 2002 to year 2015 only stocks data with codes 5398, 7084, 5517, 7106, 5031, 5148 and 3182 are available with total 35 sets data, and from year 2001 to year 2015, stocks with codes 5398, 7084, 5517, 7106, 5148 and 3182 and the total 20 sets of data are studied. From year 2000 to year 2015, there are 5 stocks with codes 5398, 5517, 7106, 5148 and 3182 available for studied and total only 10 data sets consists on 3 stock data being studied. At least 20% of each universal portfolio strategy perform better than benchmark CRP are recorded, details of result can refer Appendix C. For each trading period, well performance parameters for each proposed universal portfolio are obtained and shown in the following figures.

The meaning of A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, a, b, c, d, e, f, g, h, i, k, l, m, n, o, p, q, r, s and t are given in Section 5.2.2 to Sec 5.2.5. For each parameter setting, the W_n is calculated as the ratio of S_n of selected universal portfolio to S_n of CRP, and then a box-plot is constructed for

W_n. Figure 5.1 to Figure 5.13 compare the box-plots for W_n which are selected from the good parameters of the respective four proposed universal portfolios.

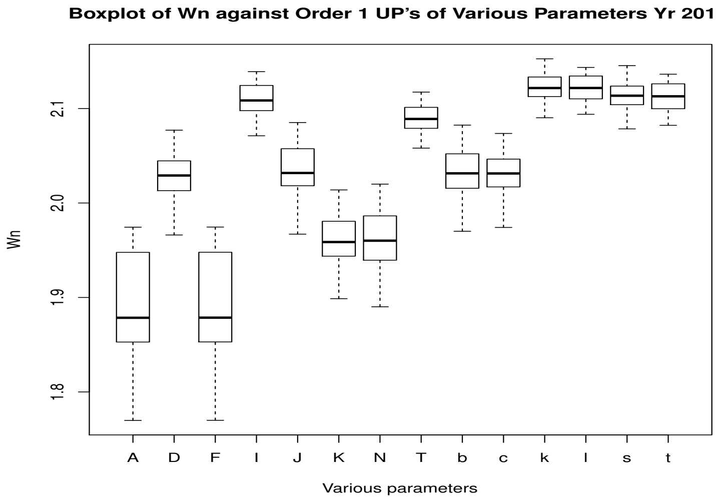


Figure 5.1: Well Performed Parameters of each Universal Portfolio in Year 2015 Period

From Figure 5.1, good performance parameters are observed for selected parameters A, D, F, I and J for Multinomial generated universal portfolio. For Multivariate Normal generated universal portfolio, parameters K, N and T are performed good. While, for Brownian-motion generated universal portfolio, only parameter b and c are good and there are four parameters performed well for universal portfolio generated by Ornstein Uhlenbeck process, there are k, l, s and t. The performances of Ornstein Uhlenbeck process generated universal portfolio performed better than the other three universal portfolio strategies.

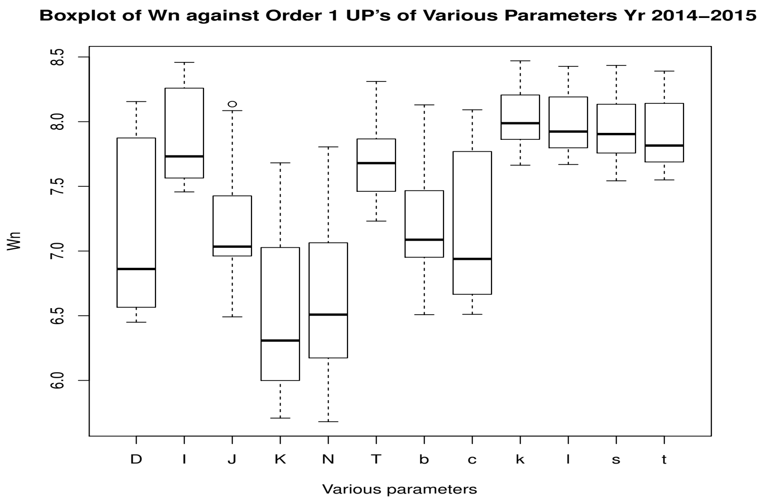


Figure 5.2: Well Performed Parameters of each Universal Portfolio in Year 2014 to 2015 Period

From Figure 5.2, good performance parameters are observed for selected parameters D, I and J for Multinomial generated universal portfolio. For Multivariate Normal generated universal portfolio, parameters K, N and T are performed good. While, for Brownian-motion generated universal portfolio, only parameters b and c are good and there are four parameters performed well for universal portfolio generated by Ornstein Uhlenbeck process, there are k, l, s and t. Among the four strategies, we can observed that the universal portfolio generated by Ornstein Uhlenbeck process again performed better than other strategies.

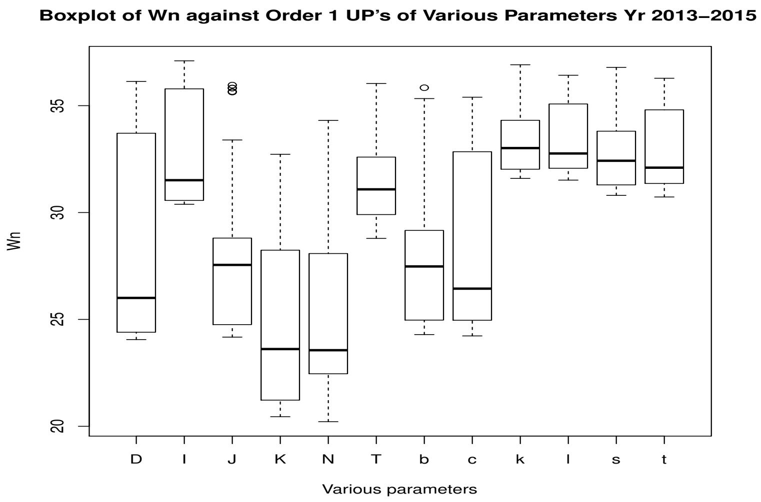


Figure 5.3: Well Performed Parameters of each Universal Portfolio in Year 2013 to 2015 Period

From Figure 5.3, we can observed that the results are similar to Figure 5.2 with good performance parameters are shown for selected parameters D, I and J for Multinomial generated universal portfolio. For Multivariate Normal generated universal portfolio, parameters K, N and T are performed good. While, for Brownian-motion generated universal portfolio, only parameters b and c are good. There are four parameters performed well for universal portfolio generated by Ornstein Uhlenbeck process, there are k, l, s and t and this strategy again better better than the other three strategies.

However, none of the proposed four universal portfolio has at least 20% of the results obtained performed better than CRP for the year 2012 to year 2015, year 2011 to year 2015 and year 2010 to year 2015 periods.

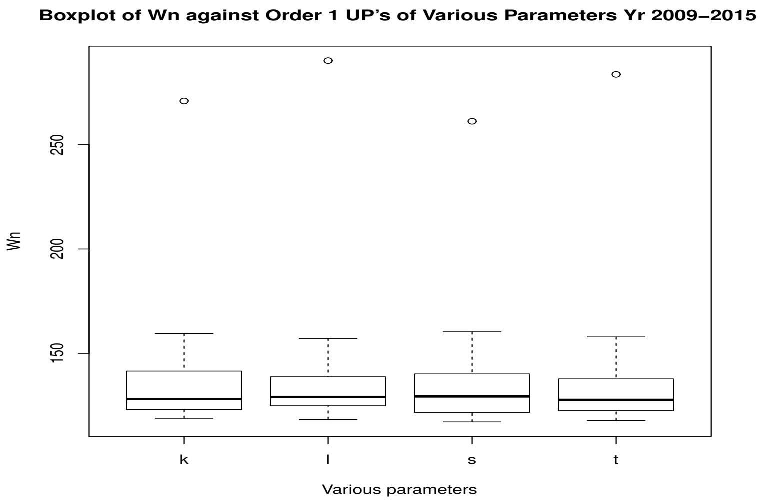


Figure 5.4: Well Performed Parameters of each Universal Portfolio in Year 2009 to 2015 Period

From Figure 5.4, only universal portfolio generated by Ornstein Uhlenbeck process has shown the good performances with parameters k, l, s and t and have showed the similar performances among them.

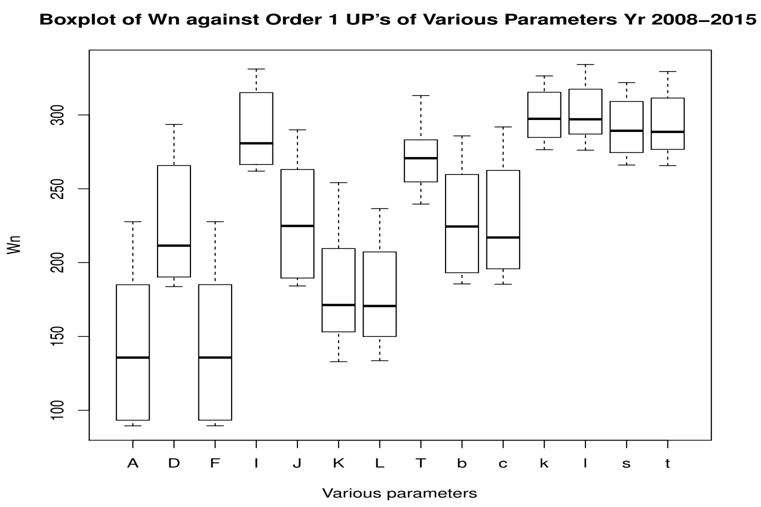


Figure 5.5: Well Performed Parameters of each Universal Portfolio in Year 2008 to 2015 Period

From Figure 5.5, good performance parameters are observed for selected parameters A, D, F, I and J for Multinomial generated universal portfolio. For Multivariate normal generated universal portfolio, parameters K, L and T are performed good. While, for Brownian-motion generated universal portfolio, only parameters b and c are good and there are four parameters performed well for universal portfolio generated by Ornstein Uhlenbeck process, there are k, l, s and t with the performances better than other three strategies.

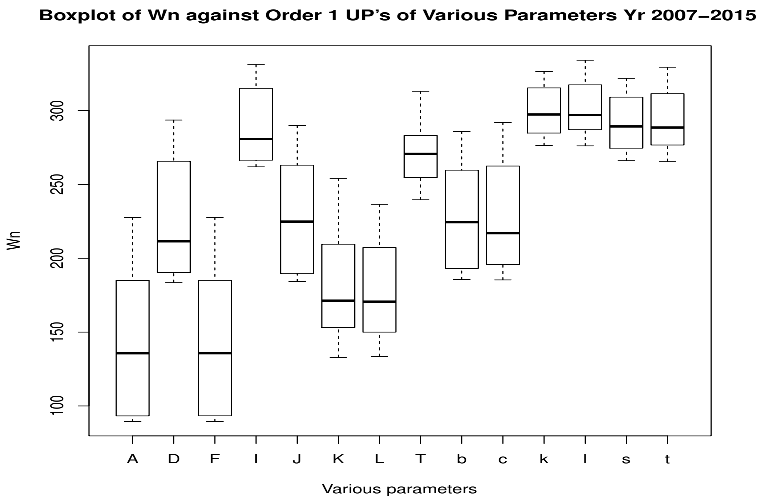


Figure 5.6: Well Performed Parameters of each Universal Portfolio in Year 2007 to 2015 Period

Figure 5.6 showed the results obtained is similar to Figure 5.5, good performance parameters are observed for selected parameters A, D, F, I and J for Multinomial generated universal portfolio. For Multivariate normal generated universal portfolio, parameters K, L and T are performed good. While, for Brownian-motion generated universal portfolio, only parameters b and c are good and there are four parameters performed well for universal portfolio generated by Ornstein Uhlenbeck process, there are k, l, s and t.

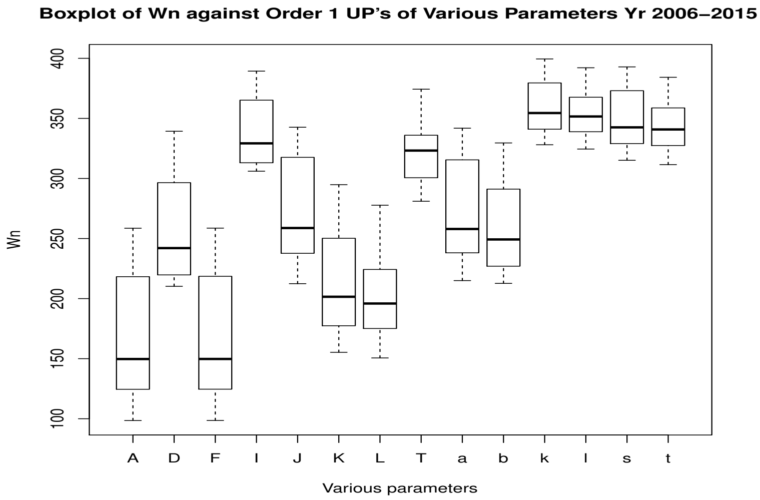


Figure 5.7: Well Performed Parameters of each Universal Portfolio in Year 2006 to 2015 Period

From Figure 5.7, results obtained is similar to previous period, where good performance parameters are observed for selected parameters A, D, F, I and J for Multinomial generated universal portfolio. For Multivariate Normal generated universal portfolio, parameters K, L and T are performed good. While, for Brownian-motion generated universal portfolio, only parameters a and b are good and there are four parameters performed well for universal portfolio generated by Ornstein Uhlenbeck process, there are k, l, s and t.

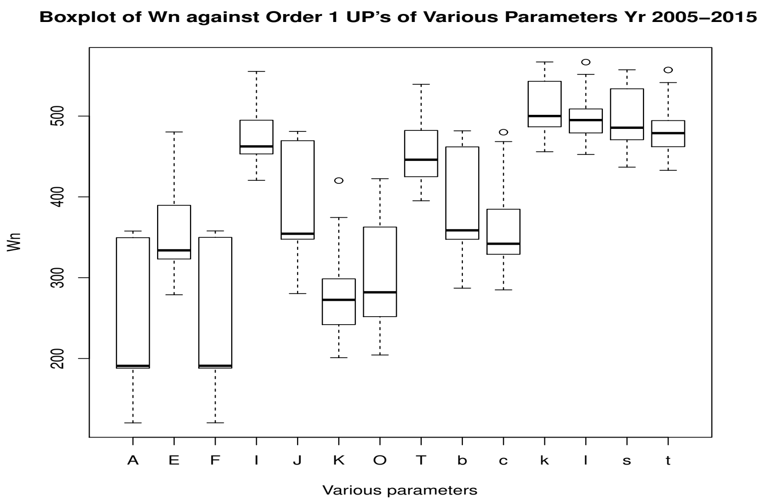


Figure 5.8: Well Performed Parameters of each Universal Portfolio in Year 2005 to 2015 Period

From Figure 5.8, results obtained is slightly different compare to Figure 5.7. Good performance parameters are observed for selected parameters A, E, F, I and J for Multinomial generated universal portfolio. For Multivariate Normal generated universal portfolio, parameters K, O and T are performed good. While, for Brownian-motion generated universal portfolio, only parameters b and c are good and there are four parameters performed well for universal portfolio generated by Ornstein Uhlenbeck process, there are k, l, s and t. Universal portfolio generated by Ornstein Uhlenbeck still performed better than the other three strategies.

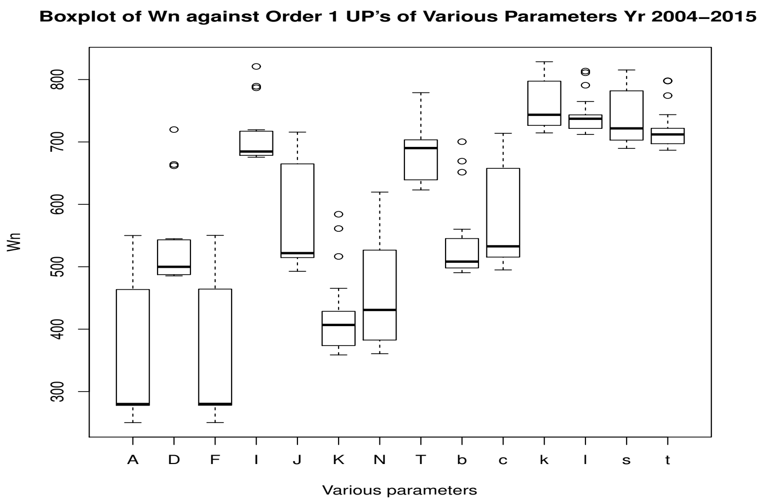


Figure 5.9: Well Performed Parameters of each Universal Portfolio in Year 2004 to 2015 Period

From Figure 5.9, good performance parameters are observed for selected parameters A, D, F, I and J for Multinomial generated universal portfolio. For Multivariate Normal generated universal portfolio, parameters K, N and T are performed good. While, for Brownian-motion generated universal portfolio, only parameters b and c are good and there are four parameters performed well for universal portfolio generated by Ornstein Uhlenbeck process, there are k, l, s and t. Universal portfolio generated by Ornstein Uhlenbeck still performed better than the other three strategies. There are more outliers compared to the previous graphs.

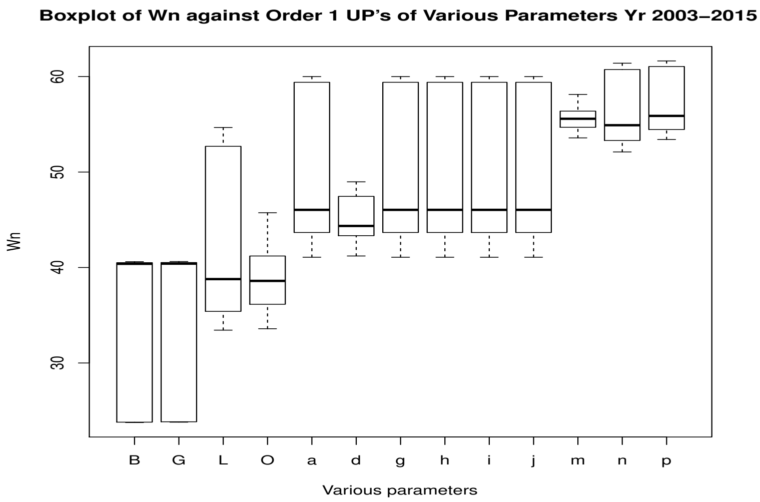


Figure 5.10: Well Performed Parameters of each Universal Portfolio in Year 2003 to 2015 Period

From Figure 5.10, the results obtained by the four proposed strategies are different compared to the previous period. Good performance parameters are observed for selected parameters B and G for Multinomial generated universal portfolio. For Multivariate Normal generated universal portfolio, parameters L and O are performed good. While, for Brownian-motion generated universal portfolio, parameters a, d, g, h, i and j are good and there are three parameters performed well for universal portfolio generated by Ornstein Uhlenbeck process, there are m, n, and p. The universal portfolio generated by Ornstein Uhlenbeck process still performed better than the other three portfolio strategies even with the different parameters.

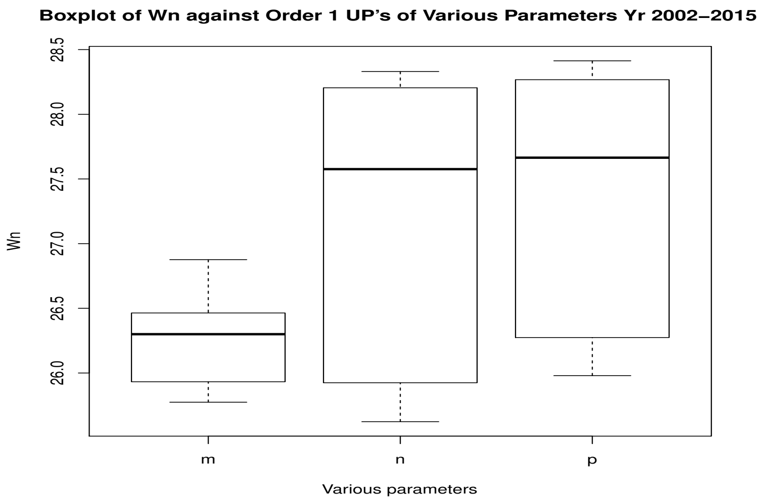


Figure 5.11: Well Performed Parameters of each Universal Portfolio in Year 2002 to 2015 Period

From Figure 5.11, only universal portfolio generated by Ornstein Uhlenbeck process showed the good performances with parameters m, n and p. We can concluded that the universal portfolio generated by Ornstein Uhlenbeck process consistently performed well compare to the other three strategies.

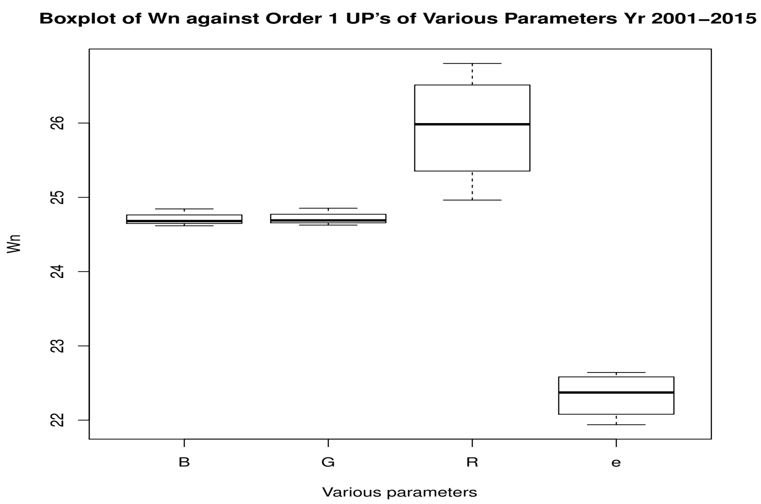


Figure 5.12: Well Performed Parameters of each Universal Portfolio in Year 2001 to 2015 Period

From Figure 5.12, good parameters are observed by B and G for universal portfolio generated by Multinomial, R of universal portfolio generated by Multivariate normal and e for universal portfolio generated by Brownian-motion. However, the universal portfolio generated by Ornstein Uhlenbeck process did not perform well in this period.

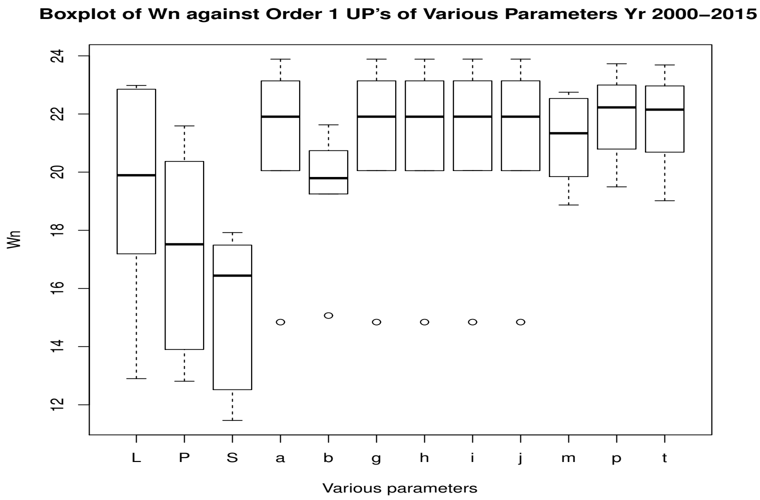


Figure 5.13: Well Performed Parameters of each Universal Portfolio in Year 2000 to 2015 Period

From Figure 5.13, good performance parameters are observed for selected parameters L, P and S for Multivariate Normal generated universal portfolio. While for Brownian-motion generated universal portfolio, parameters a, b, g, h, i and j were performed good and there are three parameters performed well for universal portfolio generated by Ornstein Uhlenbeck process, there are m, p and t. The universal portfolio generated by Brownian-motion is performed better than the other three universal portfolio strategies with 5 set of parameters shown in above figure.

In conclusion, for the further data analysis in next chapter, only the best performing parameter was chosen among all the above good performance parameters for each universal portfolio obtained in each period. Overall, from Figure 5.1 to Figure 5.13, we observed that the parameter for Multinomial generated universal portfolio is I which is (0.95, 0.05, 99) showed the good performances, for universal portfolio generated by Multivariate Normal distribution, parameter T (9.0,0.1,0.1,2.0) is observed with good performances. The other parameters performed are b, which is (100,1.4;1,2.5;10,3.6) for universal portfolio generated by Brownian motion and k which is (16,0.1,0.5,0.2,1.2,2) for universal portfolio generated by Ornstein Uhlenbeck process with good performances.

The detail of the results are shown in Appendix C.

CHAPTER 6

PERFORMANCE OF SELECTED UNIVERSAL PORTFOLIO STRATEGIES

In this chapter, we will study the four classes universal portfolio strategies (refer to Chapter 3) with the best performance parameters identified in Chapter 5. Numerical experiment for the four classes of portfolio strategies will be studied for short term (1 year period), middle term (4 years and 8 years periods) and long term (12 years and 16 years periods). We will employ the four portfolio strategies on the above 5 groups of data to learn the performance of these methods and use CRP as a benchmark. The experiment will be limited to order one universal portfolios and 3 stocks data are chosen for study.

We have identified the 95 most active stocks in KLSE. In Table 2.1 in Chapter 2, the first group of data starting from 1 January 2015 to 31 December 2015 consists of 95 stocks data. The second group is 4 years period starting from 1 January 2012 to 31 December 2015 consists of 90 stocks data where stock data with codes 5222, 5227, 5209, 5225 and 6399 are excluded because there were no transaction from year 2012 to 2014 . The third group is 8 years trading period starting from 1 January 2008 to 31 December 2015 consists of 80 stocks data where the stock data excluded are codes 5183, 5141, 5186, 5211, 5176, 5180, 5212, 5210, 5218, 6912 and all the codes excluded in four years trading period group. The fourth group is 12 years period starting from 1 January 2004 to 31 December 2015 consists of 66 stocks data excluded the stocks with codes 3336, 5131, 7052,1818, 5139, 5168, 6947, 2593, 5027, 5126, 5135, 5138,1591, 3859, 5090, 5099, 5347 and all the stock excluded in 4 years and 8 years period groups.

While, the last group is 16 years of trading period starting from 1 January 2000 to 31 December 2015 consists of 63 stocks data where data are excluded in are codes 7113, 5031, 7106 and all the stock data excluded in 4 years, 8 years and 12 years periods.

6.1 Performance Analysis of Four Universal Portfolio Strategies against CRP

We will investigate the performance of the four universal portfolio strategies by analysing the basic statistics of S_n against the CRP performance. The price relative for a given stock is the ratio of the closing price to its opening price on the same trading day. The summary of the analysis are calculated as log of the wealth obtained. The histogram of the log of the ratio of the wealth obtained by the four universal portfolio against the corresponding benchmark, CRP wealth are plotted.

6.1.1 Statistical Analysis of $\mathbf{n}$-year Investment Based on "log $\mathbf{S_n}$"

After running the Multinomial distribution generated universal portfolio algorithm in Chapter 4, the data stored in "resultmultinomial.csv". The summary of statistical analysis is calculated using R commands as below:

```
> setwd("/Volumes/SAMSUNG/PhD-Thesis/code/result1year")
> df=read.csv("resultmultinomial.csv",header=F)
> x =log(df$V4)
```

```
4 > summary(x)
5      Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
6 -2.99000 -0.30040  0.07203  0.11340  0.47590  3.14600
```

In line number 2, the CSV file consist of the ratio of respective universal portfolio wealth to CRP wealth. While for the CRP strategy, CSV file is read with the log of the wealth obtained by this strategy. By loading other data files obtained in Chapter 4 and performing basic statistical analysis, results of summary of the statistical analysis are listed in the following tables.

Table 6.1: Short Term Period Data Analysis (1 Year Period)

Portfolio Strategies	Min	First Quartile	Median	Mean	Third Quartile	Max
Multinomial Universal Portfolio	-2.9900	-0.3004	0.0720	0.1134	0.4759	3.1460
Multivariate Normal Universal Portfolio	-2.9540	-0.2953	0.0672	0.1074	0.4615	3.0750
Brownian Motion Universal Portfolio	-2.8480	-0.2851	0.05780	0.0939	0.4303	2.8910
OU Process Universal Portfolio	-3.0830	-0.3088	0.0666	0.1075	0.4747	3.1761
$\log(S_n(CRP))$	-1.8430	-0.1040	0.2469	0.3051	0.6390	3.7700

Table 6.2: Middle Term Period Data Analysis (4 Years Period)

Portfolio Strategies	Min	First Quartile	Median	Mean	Third Quartile	Max
Multinomial Universal Portfolio	-3.6420	-0.3956	0.0653	0.1245	0.5532	4.3480
Multivariate Normal Universal Portfolio	-3.5830	-0.3914	0.0579	0.1161	0.5356	4.2520
Brownian Motion Universal Portfolio	-3.4300	-0.3798	0.0446	0.0980	0.49730	4.0050
OU Process Universal Portfolio	-3.7330	-0.4114	0.0552	0.1143	0.5492	4.3960
$\log(S_n(CRP))$	-1.8430	-0.1071	0.2430	0.3070	0.6467	3.7700

From Table 6.1 and Table 6.2, we obtained the median and mean with positive values for the four universal portfolios. This indicate that the performances of the four proposed universal portfolio strategies are better than CRP.

Table 6.3: Middle Term Period Data Analysis (8 Years Period)

Portfolio Strategies	Min	First Quartile	Median	Mean	Third Quartile	Max
Multinomial Universal Portfolio	-8.4310	-0.8883	-0.0226	-0.1555	0.7735	5.8400
Multivariate Normal Universal Portfolio	-9.0200	-0.9381	-0.0247	-0.1704	0.8109	6.1520
Brownian Motion Universal Portfolio	-9.3050	-0.9625	-0.0256	-0.1818	0.8223	6.2600
OU Process Universal Portfolio	-9.3500	-0.9810	-0.0327	-0.1840	0.8335	6.3660
$\log(S_n(CRP))$	-1.5920	-0.0247	0.3121	0.3917	0.7271	3.7700

From Table 6.3, we observed that CRP is performed better than the four universal portfolio strategies.

Table 6.4: Long Term Period Data Analysis (12 Years Period)

Portfolio Strategies	Min	First Quartile	Median	Mean	Third Quartile	Max
Multinomial Universal Portfolio	-inf	-2	0	-inf	1	16
Multivariate Normal Universal Portfolio	-inf	-2	0	-inf	1	15
Brownian Motion Universal Portfolio	-inf	-2	0	-inf	1	16
OU Process Universal Portfolio	-inf	-2	0	-inf	1	16
$\log(S_n(CRP))$	-1.5850	-0.0472	0.2886	0.3830	0.7234	3.7470

Table 6.5: Long Term Period Data Analysis (16 Years Period)

Portfolio Strategies	Min	First Quartile	Median	Mean	Third Quartile	Max
Multinomial Universal Portfolio	-inf	-2	0	-inf	1	40
Multivariate Normal Universal Portfolio	-inf	-2	0	-inf	1	38
Brownian Motion Universal Portfolio	-inf	-2	0	-inf	1	39
OU Process Universal Portfolio	-inf	-2	0	-inf	1	40
$\log(S_n(CRP))$	-1.5850	-0.0548	0.2680	0.3610	0.6762	3.7470

From Table 6.4 and Table 6.5, we observed that the CRP is performed well compared to the four proposed strategies. There are some extreme values showed under mean, minimum value and maximum value of the four strategies . Therefore, the summary values obtained in long term period is meaningless. The maximum values obtained for four proposed strategies are much more larger than CRP. Hence, we can concluded that there are outliers and further analysis without these outlier will be carried out on next section.

In order to view the result graphically, the following histograms for short term period, middle-term period and long term period are plotted using R commands as below.

```
1 > setwd("/Volumes/SAMSUNG/PhD-Thesis/code/result1year")
2 > df=read.csv("resultmultinomial.csv",header=F)
3 > setEPS()
4 > postscript("resultmultinomial.eps")
5 > x=log(df$V4)
6 > coeff=density(x,na.rm=TRUE)
7 > hist(x,freq=FALSE,ylim=c(0,0.25))
8 > lines(coeff)
9 > dev.off()
```

Refer to line number 2, CSV files of each universal portfolio's result with their respective trading period are read and histogram are plotted in line number 7.

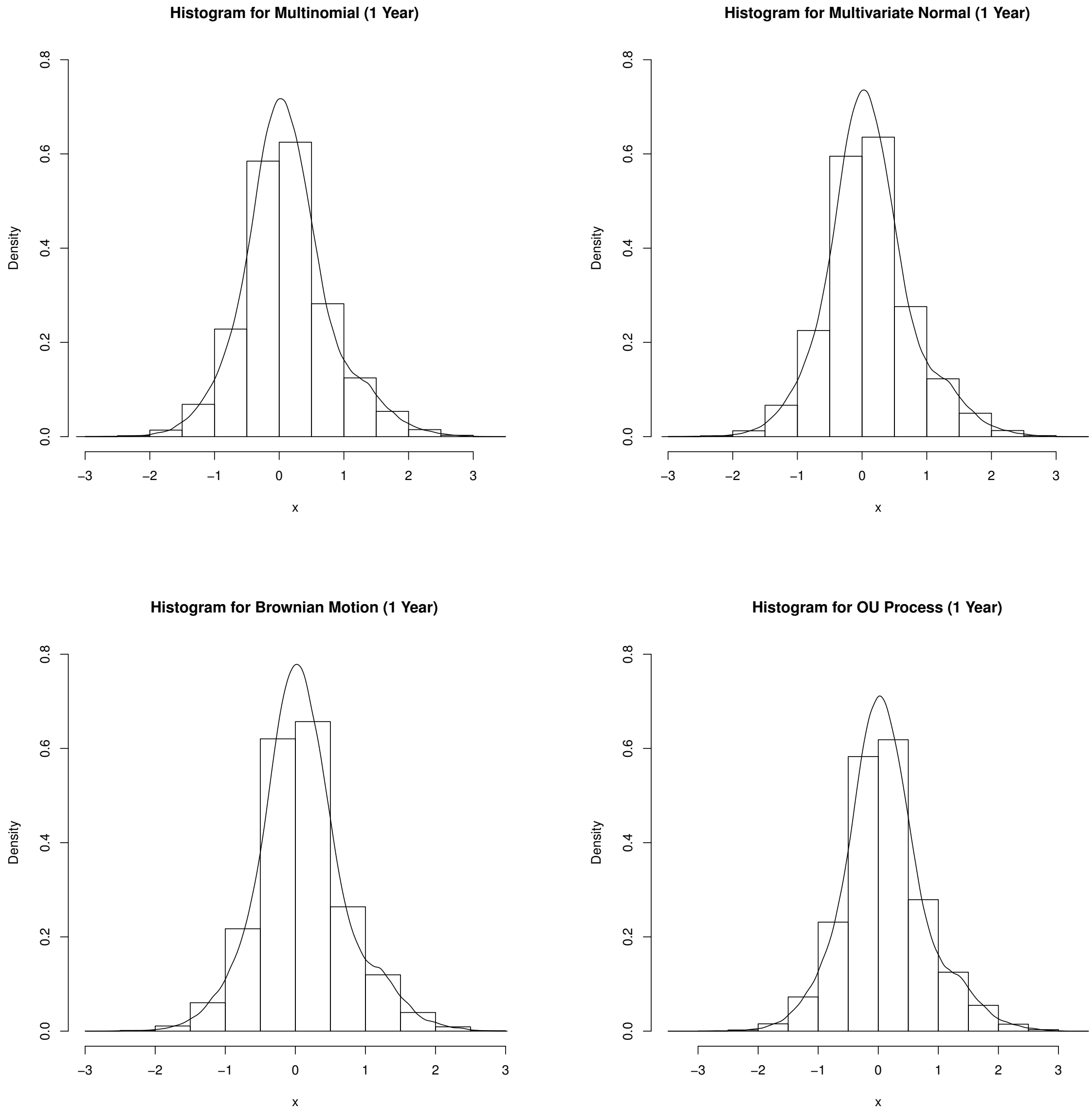


Figure 6.1: Short Term Period Data Analysis (1 Year Period)

Figure 6.1 allows us to make the conclusion that the four graphs are slightly skew to the left, same as the results shown in previous statistical analysis tables, we can concluded that the wealths achieved by the four universal portfolio strategies are better than the wealths achieved by CRP strategy.

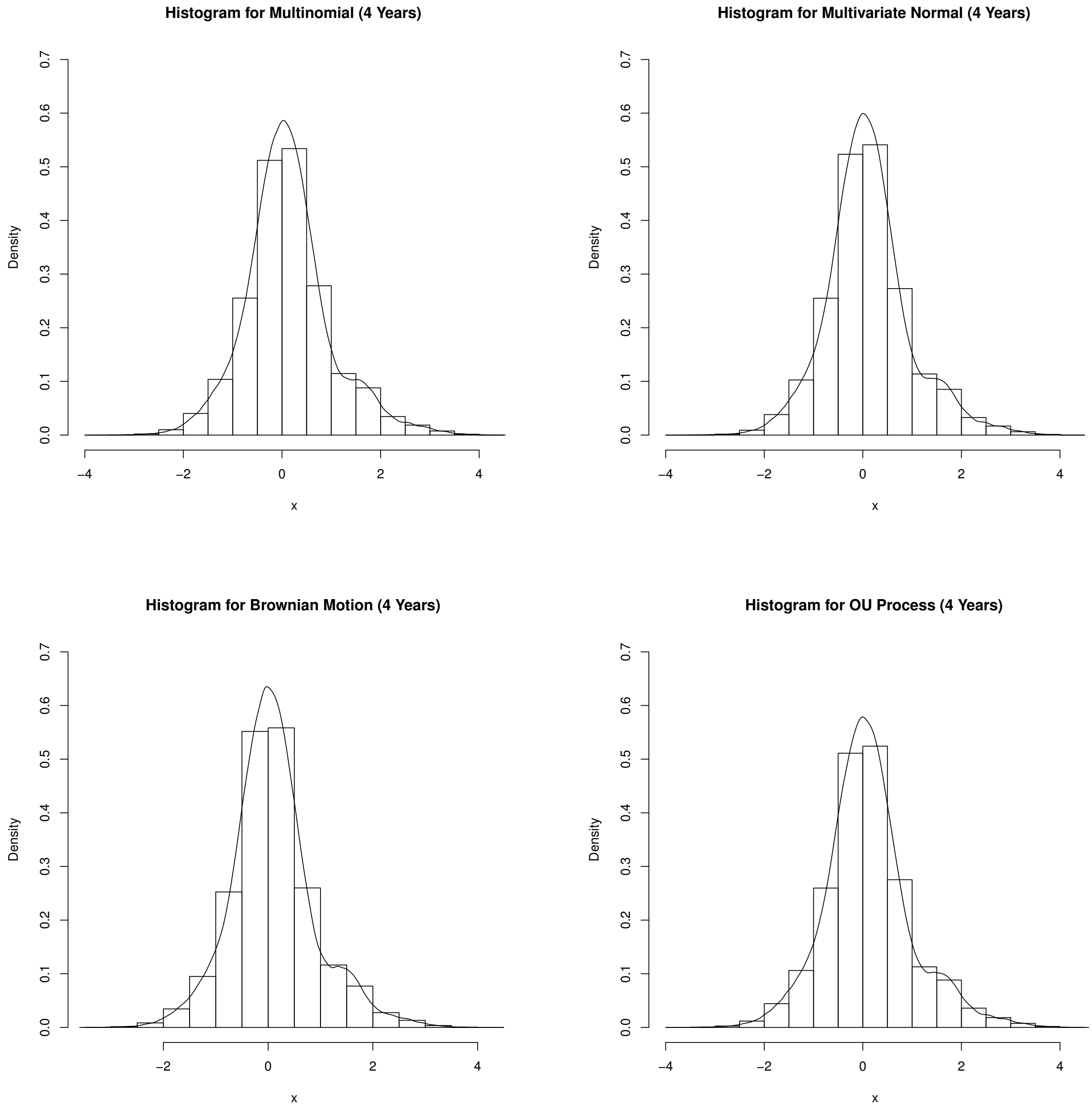


Figure 6.2: Middle Term Period Data Analysis (4 Years Period)

Figure 6.2 show that the four graphs are slightly skew to the left, even the four graphs are pretty close in the shape, but the four strategies still consistently beating the performance of CRP strategy.

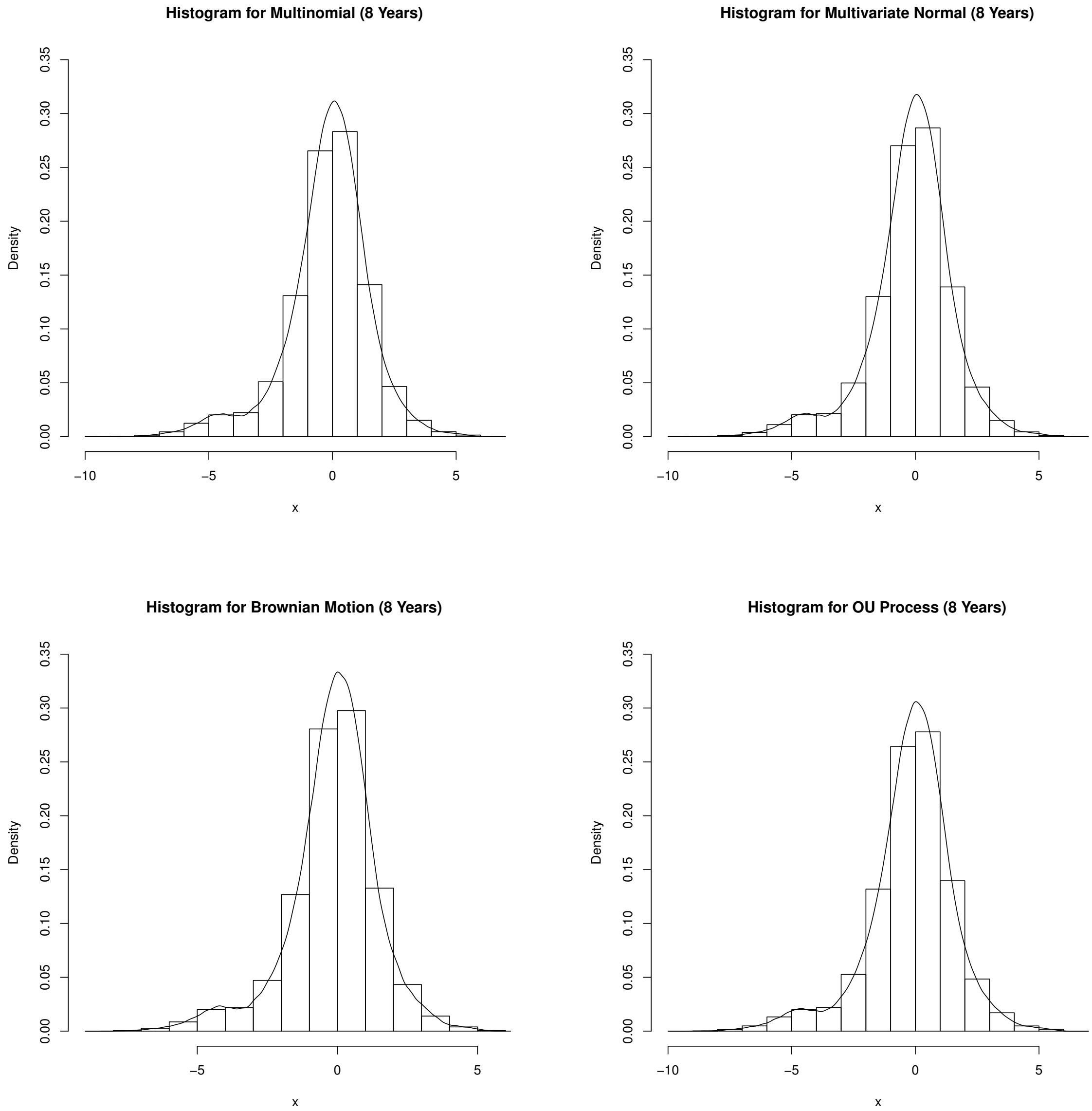


Figure 6.3: Middle Term Period Data Analysis (8 Years Period)

The third group data starting from 1 January 2008 to 31 December 2015. The financial crisis happened at the end of year 2007 and beginning of year 2008 caused a huge impact on all financial markets around the world. The crisis damaged investor confidence had an impact on global stock markets, where securities suffered large losses during the late year 2008 and early year 2009. The analysis of Figure 6.3 allow us to make the following conclusion, the four graphs are slightly skew to the right, from the summary of analysis in table 6.3 also showed that the four strategies are not perform well compared to CRP strategy.

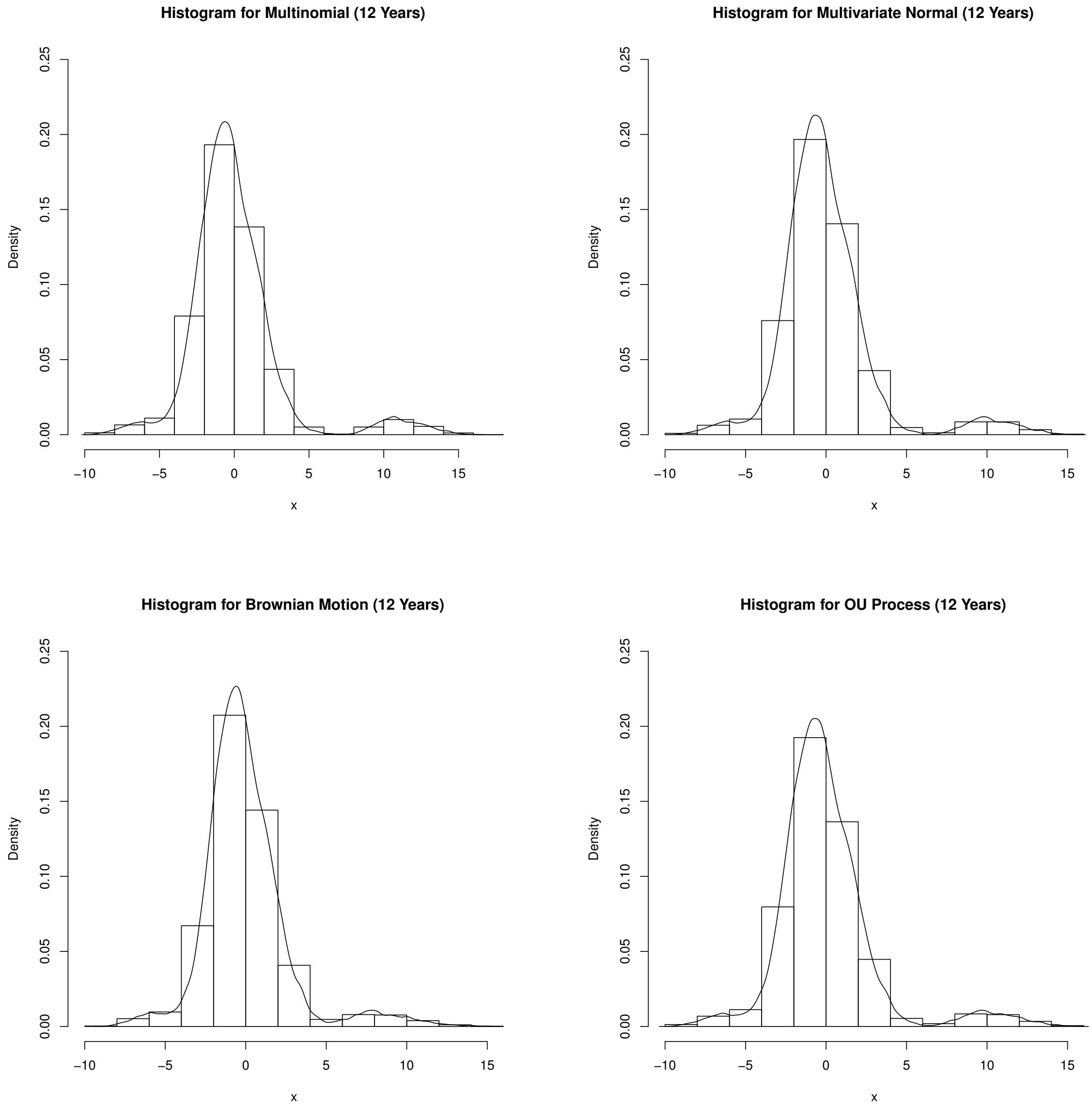


Figure 6.4: Long Term Period Data Analysis (12 Years Period)

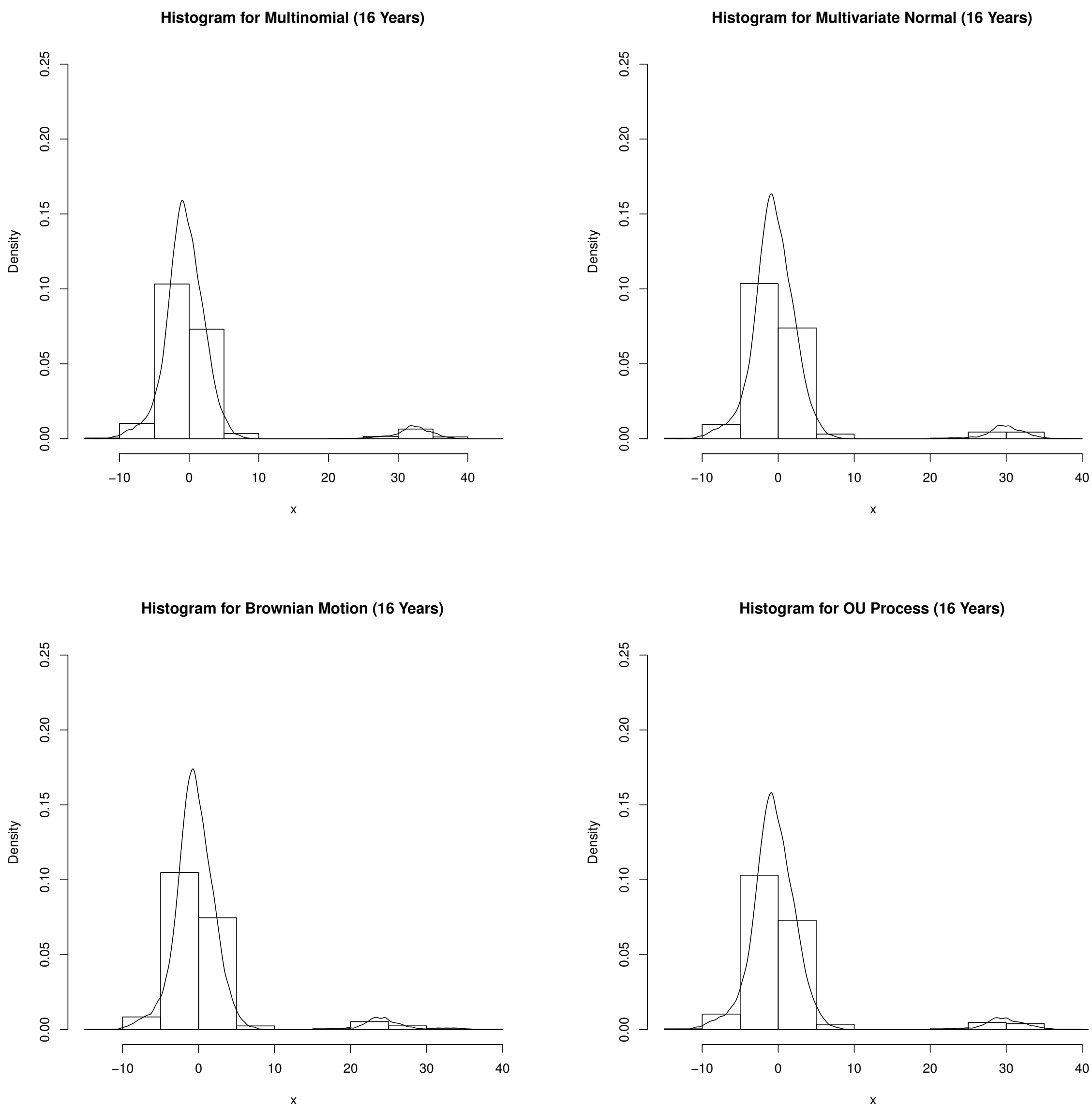


Figure 6.5: Long Term Period Data Analysis (16 Years Period)

Figure 6.4 and Figure 6.5 show that there are outliers in the result obtained. Therefore, further analysis are carry out without the outlier to identify the performances of the four proposed universal portfolio strategies.

6.2 Performance Analysis on Data Without Outliers

In the previous section, we observe that the empirical distribution indicates that for long term (12 years and 16 years), the four order one universal portfolios are performed very badly (i.e. basically losing all money), therefore leading to some outliers where some of the ratios very small which leads to $-inf$ and some of the ratios very big when taking logarithm. Therefore, in this section, we present the analysis for the long term period without the outliers.

Table 6.6: Long Term Period Data Analysis (12 Years Period) without Outliers

Portfolio Strategies	Minimum	First Quartile	Median	Mean	Third Quartile	Maximum
Multinomial Universal Portfolio	-4.9990	-1.6200	-0.4573	-0.3691	0.8410	4.9820
Multivariate Normal Universal Portfolio	-5.0000	-1.8530	-0.4408	-0.3500	0.8290	4.9990
Brownian Motion Universal Portfolio	-5.0000	-1.4940	-0.4199	-0.3222	0.7991	4.9880
OU Process Universal Portfolio	-4.9980	-1.6430	-0.4634	-0.3656	0.8527	4.9900
$\log(S_n(CRP))$	-1.5850	-0.0472	0.2886	0.3830	0.7234	3.7470

Table 6.7: Long Term Period Data Analysis (16 Years Period) without Outliers

Portfolio Strategies	Minimum	First Quartile	Median	Mean	Third Quartile	Maximum
Multinomial Universal Portfolio	-4.9990	-1.9460	-0.5129	-0.3840	1.1190	4.9970
Multivariate Normal Universal Portfolio	-4.9980	-1.8940	-0.4991	-0.3638	1.1150	5.000
Brownian Motion Universal Portfolio	4.9990	-1.7900	-0.4645	-0.3383	-0.1075	4.9970
OU Process Universal Portfolio	-5.0000	-1.9500	-0.5207	-0.3792	1.1360	4.9980
$\log(S_n(CRP))$	-1.5850	-0.0548	0.2680	0.3610	0.6762	3.7470

From the Table 6.6 and Table 6.7, the summary values obtained after removing the outliers show that the performances of CRP in 12 years and 16 years again performed better than the proposed four strategies. The following Figures show the graphs plotted without the outliers for long term period groups (12 years and 16 years).

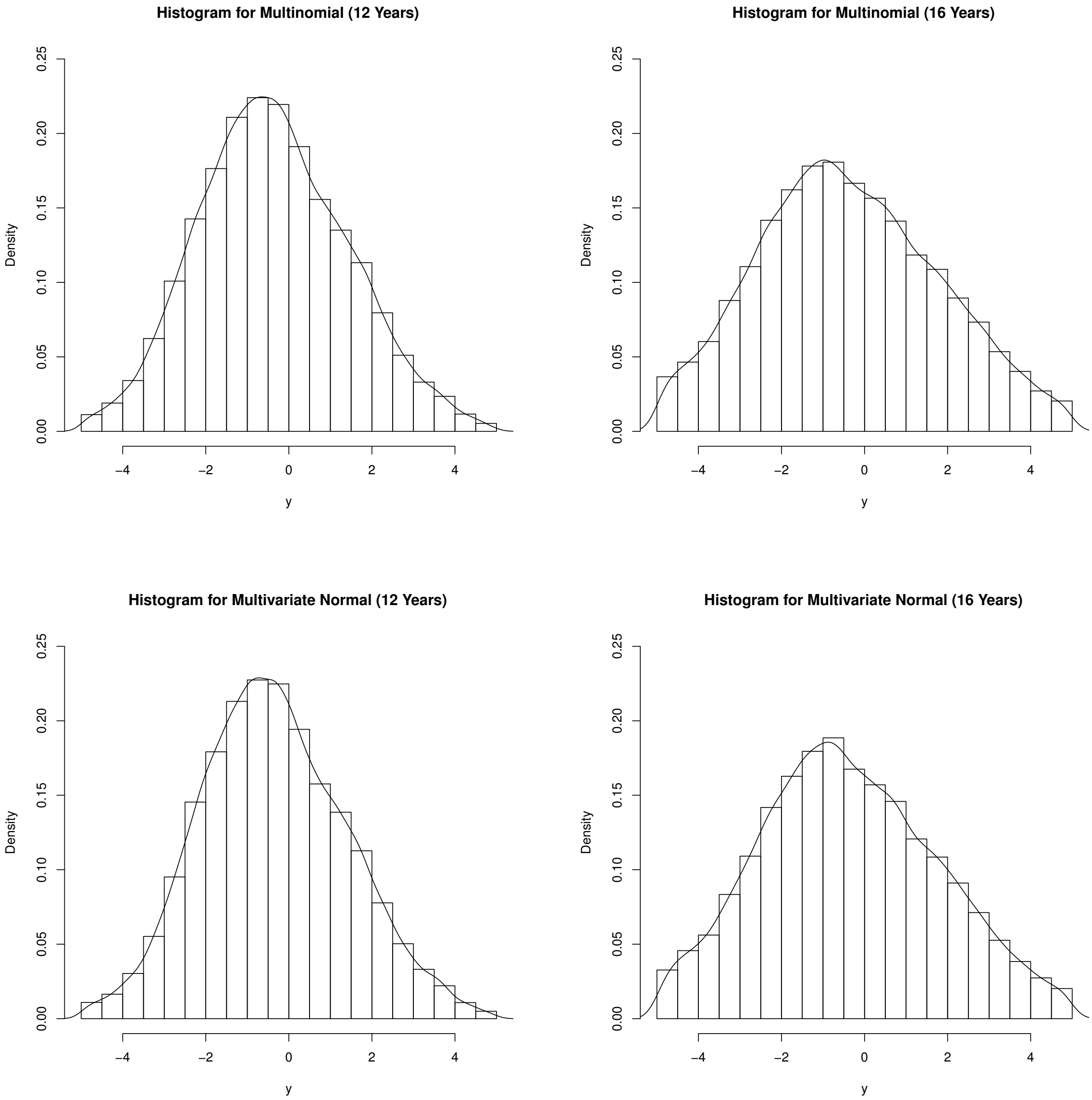


Figure 6.6: Long Term Period Data Analysis (12 and16 Years Period) without Outliers for MNUP and MVNUP

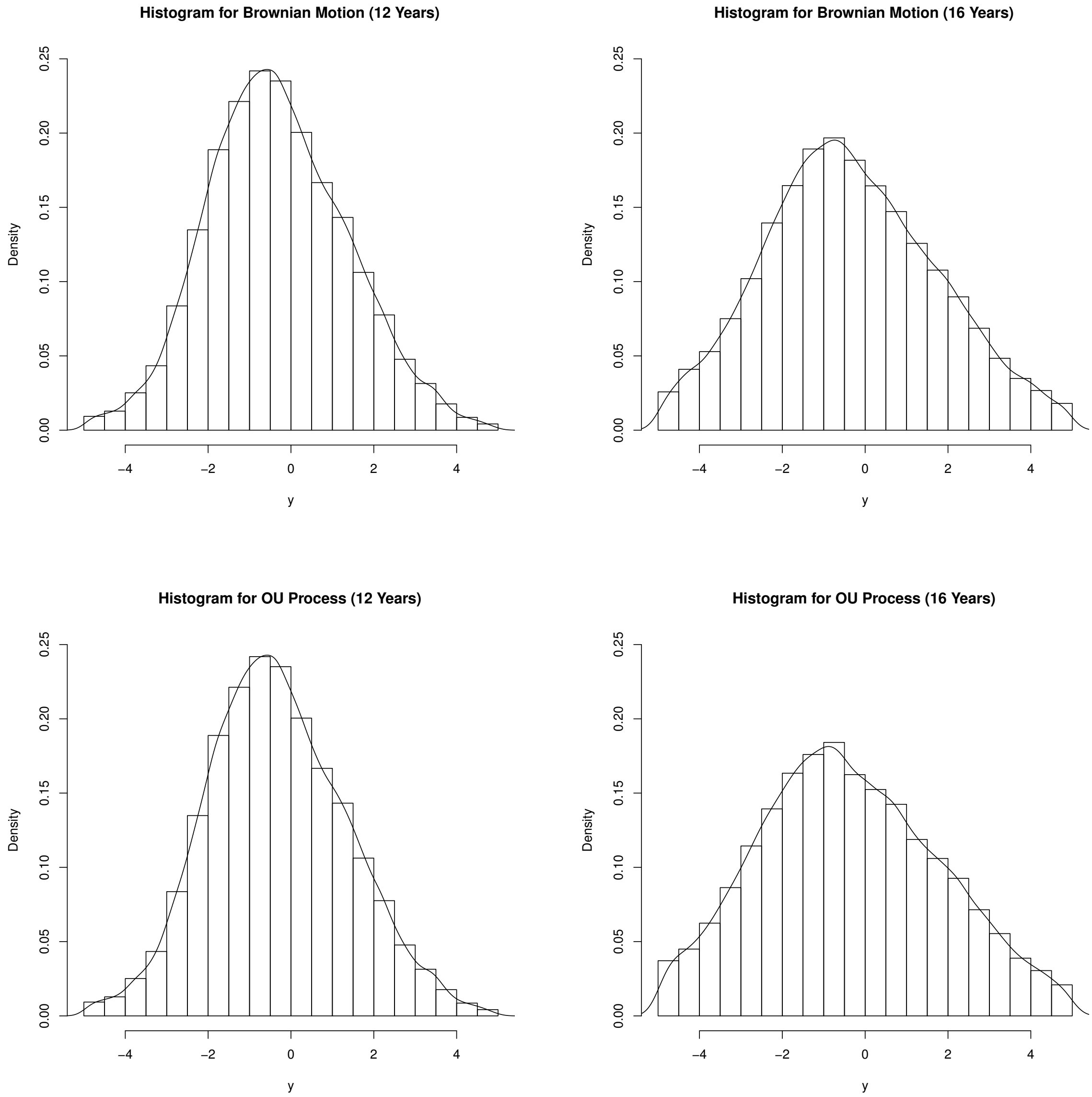


Figure 6.7: Long Term Period Data Analysis (12 and 16 Years Period) without Outliers for BMUP and OUUP

Form the Figure 6.6 and Figure 6.7, the spread of the graphs of four universal portfolio in 16 years period are larger than graphs in 12 years period. The data are clustered around the mean for 12 years period data. Notice that the proposed four universal portfolios performed better in 12 years period group compare to 16 years period group after removing the outliers.

From the above analysis and discussion, we can conclude that the four proposed universal portfolio strategies are perform well for the short term period and the middle term period (only 4 years period). Unfortunately, the four strategies are not perform well in 8 years period (middle term) and long term period.

6.3 Performance Analysis of Four Universal Portfolio Strategies against CRP with Redefined the Price Relative

In this section, we will try to compare the performances of four proposed strategies with benchmark KLCI. The price relatives for a given stock change to the ratio of closing price on the current trading day to the closing price on the day before. However, if the stock splits or bonus issue was announced by the respective company, the stock's price has either increased or decreased sharply, the price relative will be calculated as the ratio of closing price to its opening price on the same trading day. The performance of the benchmark CRP is compared to the KLCI ratio where this ratio is calculated as the log of closing index to its opening index for the respective trading period.

6.3.1 Statistical Analysis of n-year Investment Based on "log $\mathrm{S_n}$" with Redefined the Price Relative

Similar to previous section, the performance of the four universal portfolio strategies are analysed. The statistical analysis of the ratio of S_n against the CRP performance are performed. The summary of statistical analysis of the results obtained are calculated as log of the ratio of S_n. The short term, middle term and long term period statistical analysis are shown in the following tables. The graph of the log of the ratio of the wealth obtained by the four universal portfolio against the corresponding benchmark, CRP wealth are plotted.

Table 6.8: Short Term Period Data Analysis (1 Year Period) with Redefined the Price Relative

Portfolio Strategies	Min	First Quartile	Median	Mean	Third Quartile	Max
Multinomial Universal Portfolio	-4.2260	–1.1140	-0.5467	-0.5935	-0.0218	2.1116
Multivariate Normal Universal Portfolio	-4.0850	-1.0720	-0.5272	-0.5671	-0.0178	2.0690
Brownian Motion Universal Portfolio	-3.6920	-0.9513	-0.4475	-0.4809	0.02168	1.9970
OU Process Universal Portfolio	-4.21500	-1.1100	-0.5498	-0.5896	-0.0249	2.1250
$\log(S_n(CRP))$	0.2045	0.9622	1.5330	2.2890	2.6450	76.4800
$\log(KLCIRatio)$			-0.0395			

First, we analyse the performance of short term period. From the Table 6.8, we obtained the median and mean with negative values for the four portfolios. This indicate that the four proposed universal portfolio strategies are not perform well compare to CRP and KLCI. Noticed that CRP strategy is performed better than KLCI.

Table 6.9: Middle Term Period Data Analysis (4 Years Period) with Redefined the Price Relative

Portfolio Strategies	Min	First Quartile	Median	Mean	Third Quartile	Max
Multinomial Universal Portfolio	-4.3310	-1.1440	-0.5683	-0.6146	-0.0330	2.2011
Multivariate Normal Universal Portfolio	-4.1650	-1.1000	-0.5499	-0.5872	-0.0290	2.0891
Brownian Motion Universal Portfolio	-3.7120	-0.9731	-0.4649	-0.4973	0.0129	1.9970
OU Process Universal Portfolio	-4.3511	-1.1370	-0.5718	-0.6101	-0.0360	2.2176
$\log(S_n(CRP))$	0.2541	0.9693	1.5160	2.3520	2.7210	76.4820
$\log(KLCIRatio)$			0.1052			

Table 6.10: Middle Term Period Years Data Analysis (8 Years Period) with Redefined the Price Relative

Portfolio Strategies	Min	First Quartile	Median	Mean	Third Quartile	Max
Multinomial Universal Portfolio	-4.2103	-1.2100	-0.6189	-0.6625	-0.0632	2.0880
Multivariate Normal Universal Portfolio	-4.1010	-1.1620	-0.5954	-0.6327	-0.0573	2.2134
Brownian Motion Universal Portfolio	-3.5991	-1.02400	-0.5014	-0.5344	-0.0077	1.9945
OU Process Universal Portfolio	-4.2240	-1.2010	-0.6189	-0.6566	-0.0642	2.1311
$\log(S_n(CRP))$	0.2066	0.9916	1.6410	2.5060	2.9150	76.4816
$\log(KLCIRatio)$			0.1590			

Next, we analyse the results obtained for middle term period. From Table 6.9 and Table 6.10, we again obtained the median and mean with negative values for the four universal portfolios strategies. This indicate that the four proposed universal portfolio strategies still performed poorly when compare to benchmark CRP and KLCI. Furthermore, CRP still performed better than KLCI.

Table 6.11: Long Term Period Years Data Analysis (12 Years Period) with Redefined the Price Relative

Portfolio Strategies	Min	First Quartile	Median	Mean	Third Quartile	Max
Multinomial Universal Portfolio	-4.2212	-1.2850	-0.6719	-0.7072	-0.0840	2.1157
Multivariate Normal Universal Portfolio	-4.0978	-1.2850	-0.6719	-0.7072	-0.0840	2.0988
Brownian Motion Universal Portfolio	-3.7100	-1.0960	-0.5505	-0.5750	-0.0186	1.9876
OU Process Universal Portfolio	-4.2219	-1.2830	-0.6753	-0.7045	-0.0849	2.1250
$\log(S_n(CRP))$	0.2088	0.9716	1.6750	2.6250	3.0660	74.8850
$\log(KLCIRatio)$			0.7576			

Table 6.12: Long Term Period Data Analysis (16 Years Period) with Redefined the Price Relative

Portfolio Strategies	Min	First Quartile	Median	Mean	Third Quartile	Max
Multinomial Universal Portfolio	-4.2159	-1.2970	-0.6807	-0.7191	-0.0939	2.2213
Multivariate Normal Universal Portfolio	-4.0798	-1.2490	-0.6610	-0.6903	-0.0893	2.0645
Brownian Motion Universal Portfolio	-3.7001	-1.6970	-0.5542	-0.5807	-0.0268	1.9897
OU Process Universal Portfolio	-4.2139	-1.2890	-0.6844	-0.7134	-0.0951	2.1245
$\log(S_n(CRP))$	0.2044	0.9768	1.6790	2.6430	3.0790	74.8250
$\log(KLCIRatio)$			0.7332			

The analysis of results obtained for long term periods are showed in Table 6.11 and Table 6.12. Both tables still showed the negative values of mean and median. Hence, we concluded that the CRP is performed well compared to the four proposed universal portfolio strategies for long term period.

In order to view the result graphically, the following histograms for short term period, middle-term period and long term period analysis are plotted.

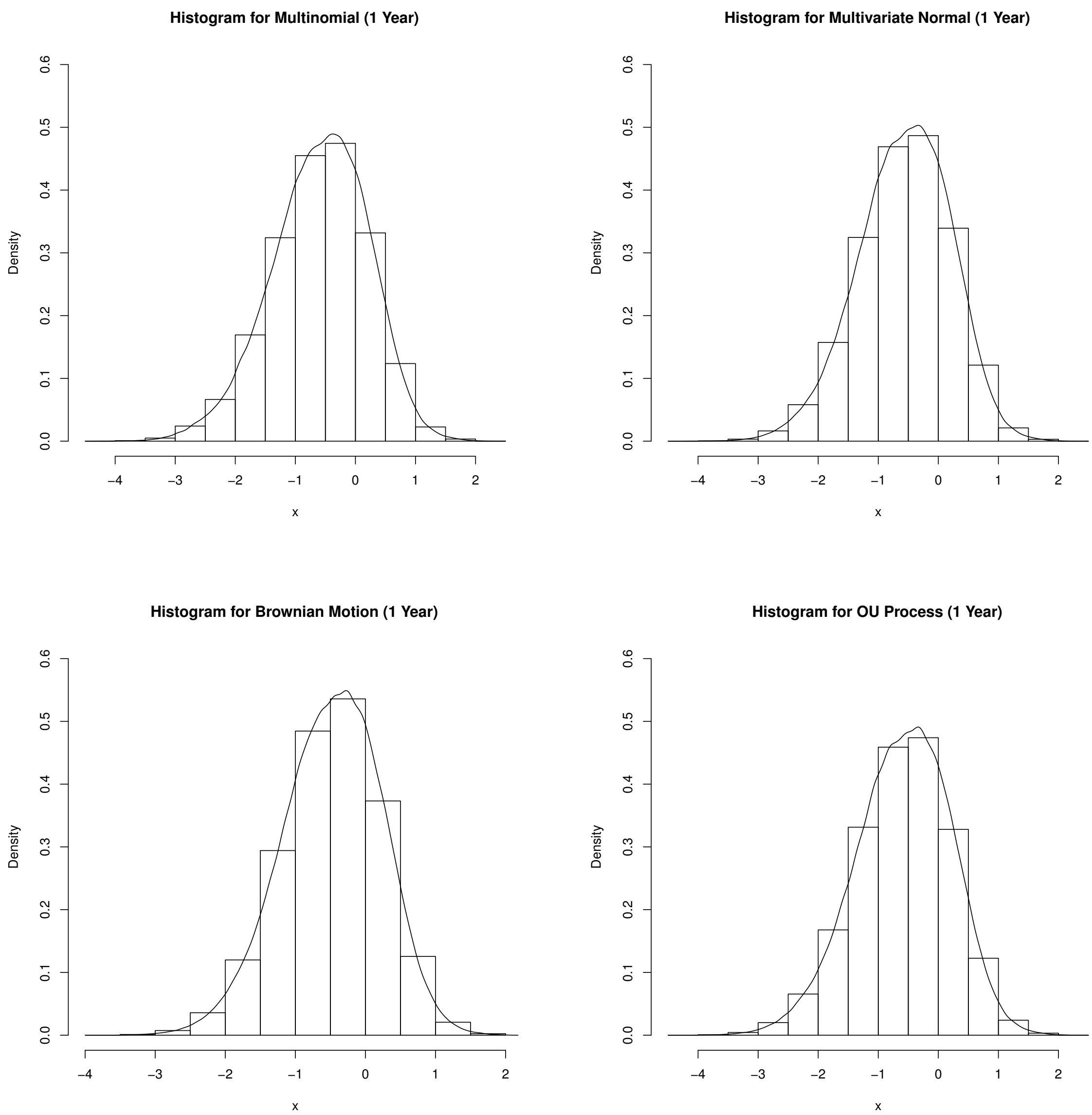


Figure 6.8: Short Term Period Data Analysis (1 Year Period) with Redefined the Price Relative

From the Figure 6.8, the four graphs are skew to the left, these concluded that the four universal portfolio strategies are not performed good when compare to CRP strategy. There are only around 25% of the results showed the four universal portfolio strategies can beat the CRP strategy in short term investment.

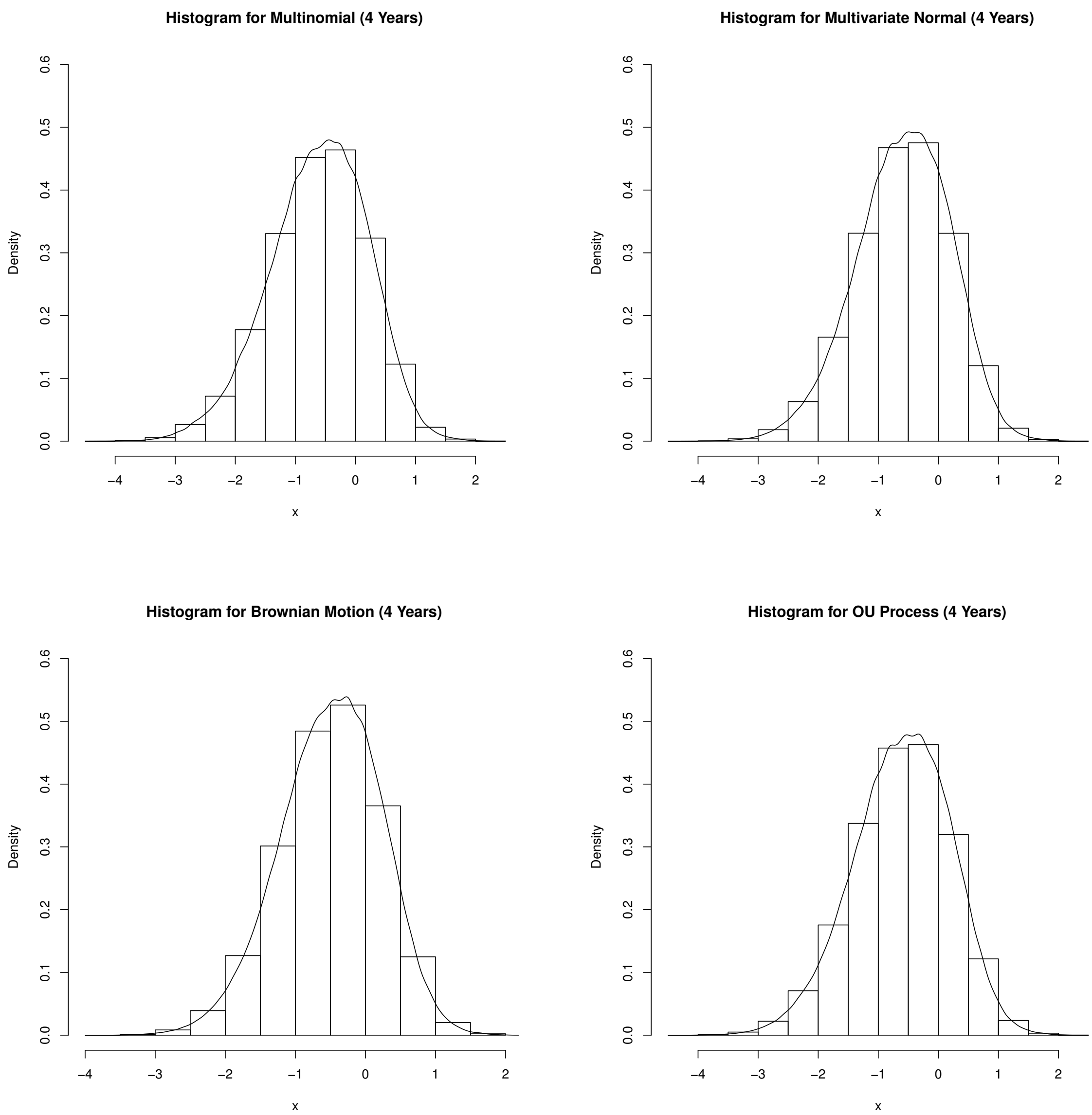


Figure 6.9: Middle Term Period Data Analysis (4 Years Period) with Redefined the Price Relative

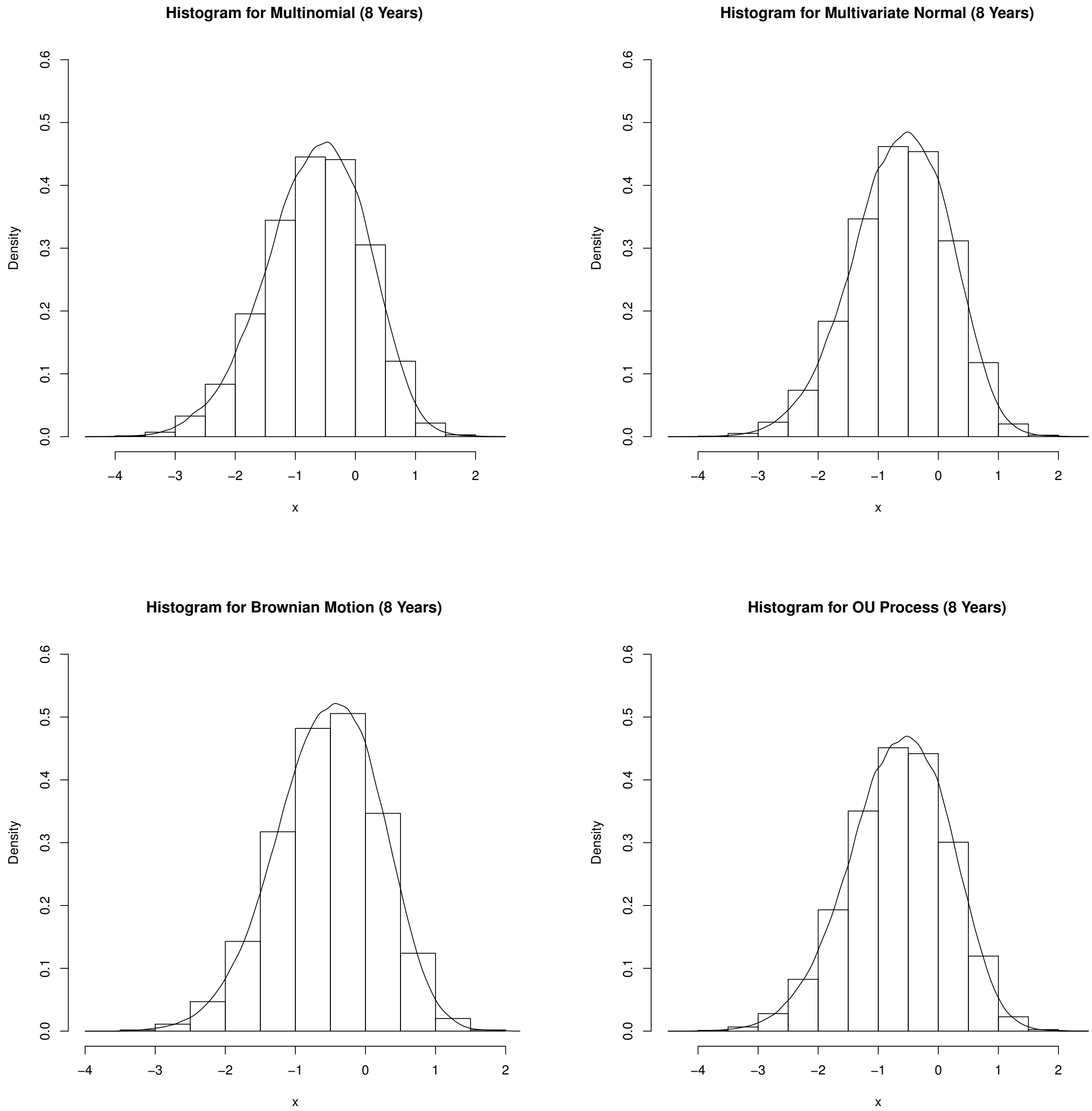


Figure 6.10: Middle Term Period Analysis (8 Years Period) with Redefined the Price Relative

For the middle term analysis, Figure 6.9 and Figure 6.10 allow us to make the following conclusion, all the histograms are skew to the left even they are close in the shape. The CRP strategy is consistently beating the four propose universal portfolio strategies. The histograms also show that around 25% of the results indicated the four universal portfolio strategies can perform better than CRP strategy in middle term investment.

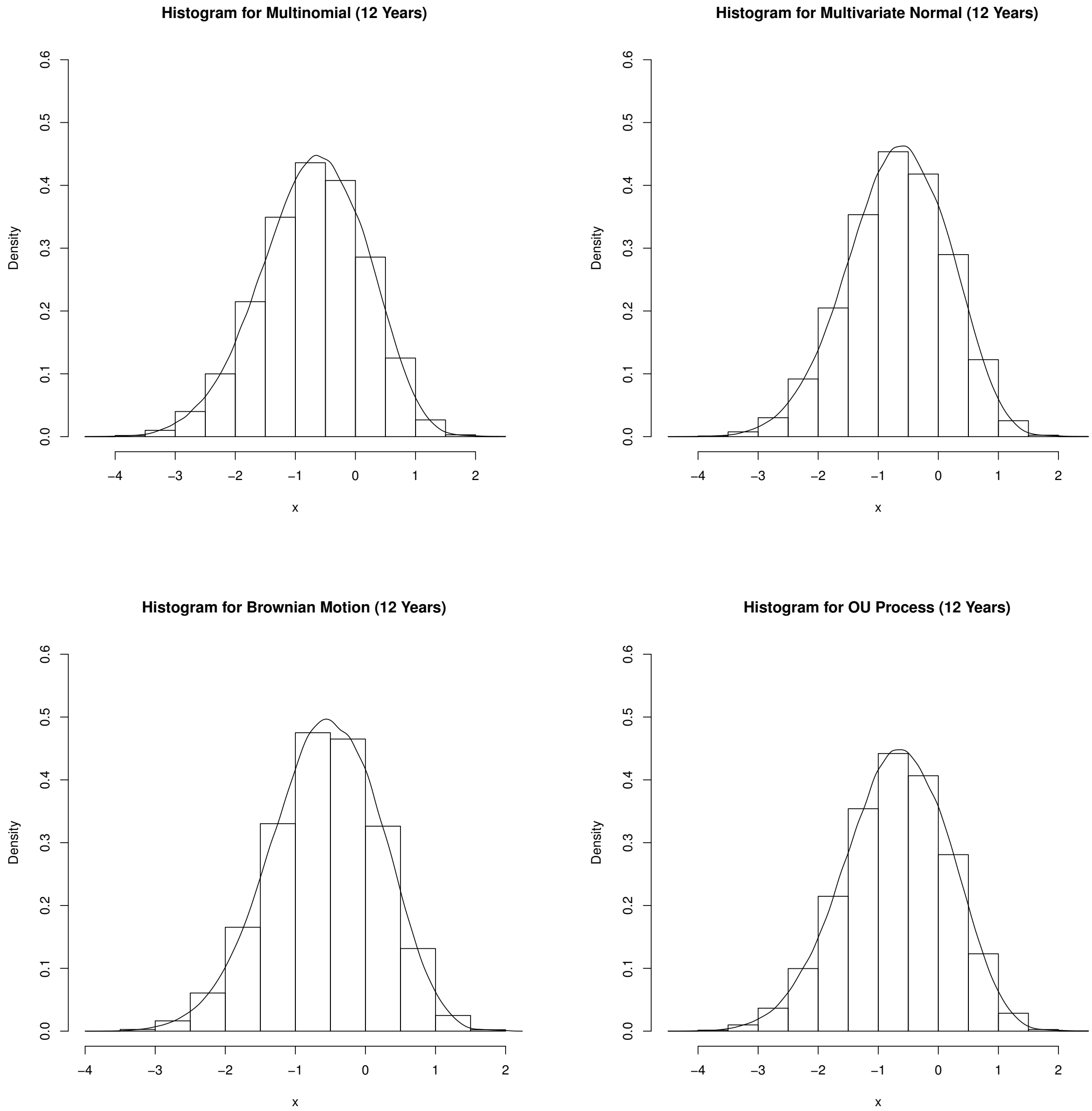


Figure 6.11: Long Term Period Data Analysis (12 Years Period) with Redefined the Price Relative

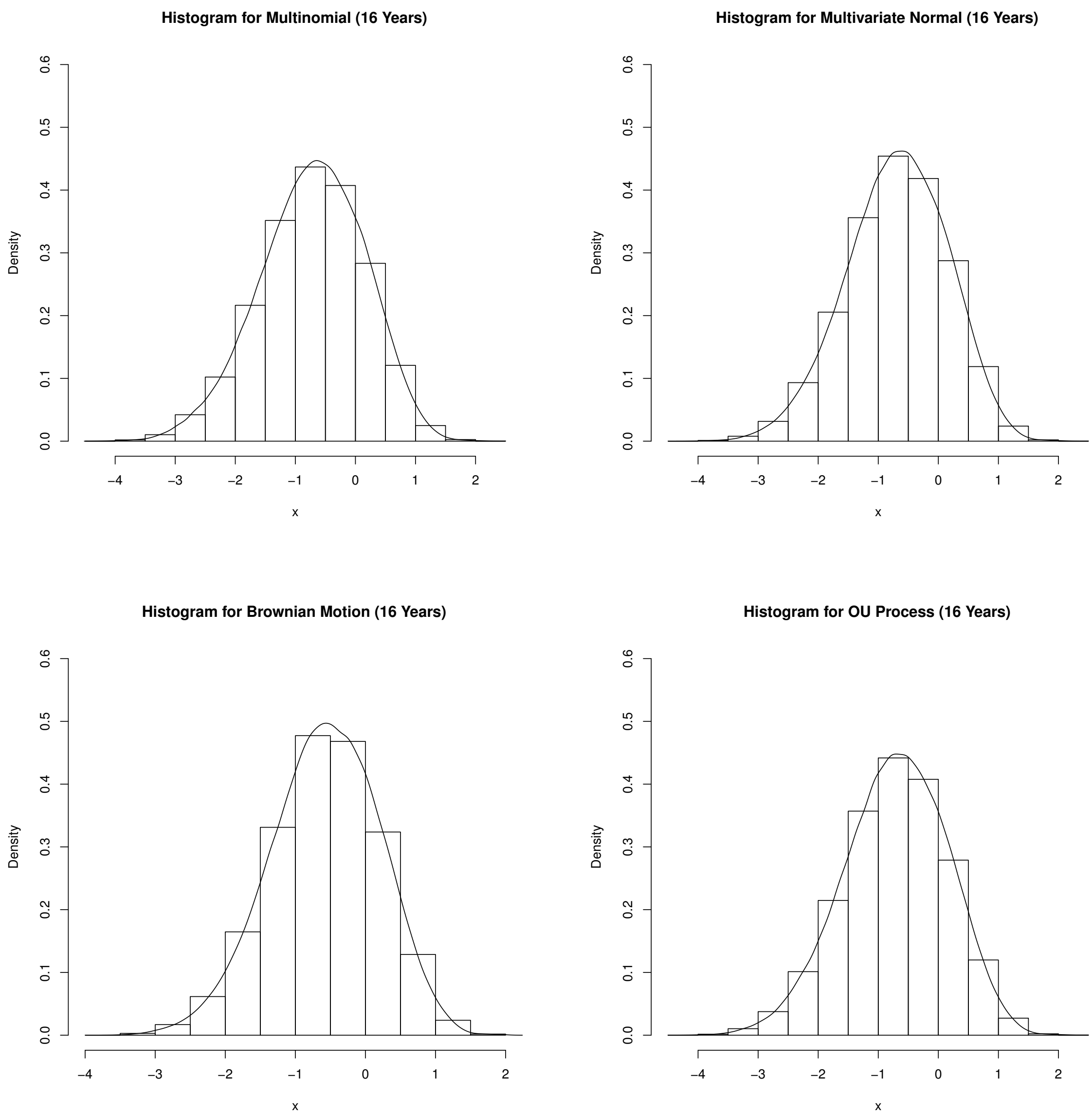


Figure 6.12: Long Term Period Data Analysis (16 Years Period) with Redefined the Price Relative

For the long term investment, From Figure 6.11 to Figure 6.12, again the histograms showed the results obtained still skew to the left. This led us to conclude that the four universal portfolio strategies are still not perform better than CRP strategy in long term period.

In summary, in order to compare to KLCI, when the price relatives for a given stock changed to ratio of the closing price on the current trading day to the closing price on the day before, the analysis of the results obtained conclude that CRP strategies are outperform the four propose universal portfolio strategies in short term, middle term and long term period investment.

6.4 Summary and Thoughts on Performance Analysis

The numerical experiment comparison of the achieved wealth by four different universal portfolio strategies against the constant rebalanced portfolio are studied. The empirical performances of the four portfolio strategies are investigated through a set of numerical experiments concerning 95 stocks data selected from Kuala Lumpur Stock Exchange.

So far, the research of universal portfolios in Malaysia was focused on determining the best performing parameters for m stocks. In the first section of this chapter, The result obtained emphasise the proposed investment strategy performed better than the constant rebalanced portfolio for short term. Also, the performance of the four proposed universal portfolio strategies performed more or less the same as the CRP. We believe that most parameters of these four order one universal portfolio strategies for 3 stocks computation will perform more or less the same. This may due to the nature of the joint distribution of the random variables Y_1, Y_2, Y_3. The four universal portfolio strategies are performed well comparing to CRP for short term. However, they are found performed terribly when trading in middle to long term.

In second section, in order to compare with benchmark KLCI, noticed that the four universal portfolio strategies did not perform well for short term, middle term and long term period. However, CRP is consistently performed good throughout the three terms.

As a conclusion, CRP is better and more stable strategy. In contrast, four universal portfolios may be able to achieve better return in wealth some of the time at the larger risks.

CHAPTER 7

CONCLUSION

The goal of our research is to study how the finite order universal portfolio strategies perform for a fund which invest on 3 out of 95 Malaysian stocks. Since universal portfolio strategies are comparable to BCRP, we will use CRP as a benchmark for comparison. The study was limited to four proposed universal portfolio strategies, the finite order Multinomial Universal Portfolio, the finite order Multivariate Normal universal portfolio, universal portfolio generated by Brownian-motion and universal portfolio generated by Ornstein-Uhlenbeck processes.

After collecting and compiling the 95 Malaysian stock data, I proceed to propose a unified framework for the research carried out by Tan (2004b), Tan et al. (2012), Tan & Lim (2011) and Tan (2013) research group. Using the basic probability theory, I have formalised the finite order universal portfolio. This type of universal portfolio depends only on the positive moments of the generating probability distribution. The moments of the proposed universal portfolio are computed for low order, i.e, order one, two and three. The price relatives used for a given stock is the ratio of closing price to the opening price on the same trading day.

Based on the formulas derived from the finite order portfolios, we have implemented them in Python programming language to test how well they perform. Cover have shown both mathematically and numerically that universal portfolio performed well. To show that the finite order universal portfolio perform well, they are compared against Cover's Dirichlet $(\alpha_1, \alpha_2, \alpha_3)$ universal portfolio strategy. I used the modified algorithm of Diriclet $(\alpha_1, \alpha_2, \alpha_3)$universal portfolio obtained by Tan (2004b) to perform computation on three stock data selected from Kuala Lumpur Stock Exchange. The finite order universal portfolio performed as good as Cover's universal portfolio, but finite order universal portfolio is computationally better in speed and memory.

An experimental study of the above four universal portfolios strategies was conducted for the 10 most active stocks data among the 95 selected stocks from kuala Lumpur Stock Exchange. The trading period is from 1 Jan 2000 to 31 Dec 2015. From the first section in Chapter 5, the proposed four finite order universal portfolios strategies are found can perform better than the Cover Ordentlich Dirichlet universal portfolio for same data sets. Noticed that the four universal portfolios with order one performed as well as order two and three. Therefore, this research will be limited to order one universal portfolio.

The above four order one universal portfolios are run on 3 stock-price data sets generated by the combination of the 10 selected most active stocks from the 95 selected stocks. Selection of good parameters for each proposed universal portfolios are carry out. We use CRP as a benchmark to evaluate and to compare with these four universal portfolio strategies. A comparison can be made with the wealth obtained by CRP and the wealths achieved by the four proposed universal portfolios. Every one year period starting from year 2000 to year 2015 of the available stock data are used for study. The best performing parameter of all four classes of universal portfolio strategies were identified from the box-plot plotted.

Next, we studied the performances of four proposed universal portfolio strategies with their best performance parameters identified in previous study. Numerical experiment for the four classes of portfolio strategies are studied for short term (1 year period), middle term (4 years and 8 years periods) and long term (12 years and 16 years periods) data. We employed the four portfolio strategies on the above five groups of data to learn the performance of these methods and used CRP as a benchmark. Every 3 stocks data generated by combination of the selected of 95 stocks are chosen for study. The empirical results showed that the performances of the proposed four strategies are better than CRP in 1 year and 4 years period. However, the four strategies did poorly in 8 years, 12 years and 16 years period. Therefore, these four universal portfolio strategies are confirmed to be an extremely competitive investment strategies in short term period.

Lastly, we try to compare the performances of four proposed strategies to KLCI. We changed the price relatives for a given stock as the ratio of closing price on the current trading day to the closing price on the day before. After the investigation and analysis of the results obtained, we found that the four proposed universal portfolio strategies are not perform good compared to benchmark CRP and KLCI in short term, middle term and long term period. Furthermore, CRP is consistently performed good when compare to four proposed universal portfolio strategies and KLCI in short, middle and long term period.

In conclusion, this research has simplified the formulation of μ-weighted universal portfolio Cover & Ordentlich (1996) and order ν universal portfolio Tan (2013) using probability theory and further extend to probability generated order ν universal portfolio to stochastic process based universal portfolio. The order ν universal portfolio has much better speed and memory performance as well as easy to implement in a high level programming language as illustrated in this research. The four order one universal portfolio strategies are found to perform better than CRP in the short term when used the price relatives mentioned in first section of Chapter 6. Therefore, these four universal portfolio strategies are confirmed to be an extremely competitive investment strategies in short term period.

Stochastic processes based universal portfolio is a good generalisation order ν universal portfolio which is believed be able to perform well with the right stochastic processes. However, the derivation of the algorithm will be more complicated. Malaysian stock data are used in this research and we believe that the order ν universal portfolio may help to promote active online trading for Malaysia stock market, as well as to have more investment opportunities for Malaysian but with the right combination of stocks.

7.1 Further Research

So far, the research of universal portfolios in Malaysia was focused on determining the best performing parameters for $m \leq 10$ stocks. In analysing the performance of the four strategies, we find that they perform more or less the same as the CRP. We believe that most of the parameters for these four order one strategies with 3 stocks study will perform more or less the same. This may due to the nature of the joint distribution of the random variables Y_1, Y_2, Y_3.

The four universal portfolio strategies are found to perform better than CRP for short term in first section of Chapter 6. However, they are found perform poorly when trading in middle to long term. Therefore, further study for the causes of the poor performance of the proposed four strategies in middle and long term are suggest to carry out for future.

Furthermore, the four universal portfolio strategies are found to perform poorly in short term, middle term and long term when comparing to KLCI. In this situation, suggest further study to investigate the causes of the poor performance of the proposed four strategies in the future.

The result also showed that the influences of parameters of the strategies to the wealth obtained. Therefore, it is important to understand whether different values of the parameters of the universal portfolio strategies will demonstrate similar pattern. It is also important to try the order one universal portfolios on the United State and European Stocks to see if the performance with respect to CRP are similar.

The proposed four order one universal portfolio strategies did not perform well with some possible combinations of Malaysian stocks. I believe if we choose the right combination of stocks will lead to the four strategies perform as well as CRP strategy. Also, it is interesting to see if higher finite order universal portfolio strategies of mixing strategies approach will reduce the occurrence of outliers in long term.

REFERENCES

Agarwal, A. & Hazan, E. (2005), New algorithms for repeated play and universal portfolio management. Princeton University Technical Report TR-740-05.

Agarwal, A., Hazan, E., Kale, S. & Schapire, E. R. (2006), Algorithms for portfolio management based on the newton method, *in* 'The 23rd International Conference on Machine learning, Pittsburgh,PA'.

Borodin, A., El-Yaniv, R. & Gogan, V. (2004), 'Can we learn to beat the best stock', *Journal of Artificial intelligence Research* **21**(80), pp. 579–594.

Capital, H. (n.d.), 'Harimau capital'.
URL: *www.harimaucapital.com*

Chan, C. T. (2002), Universal portfolios and relative entropy bounds in investment, Master's thesis, University of Malaya, Kuala Lumpur.

Cover, T. M. (1991), 'Universal portfolios', *Math. Finance* **1**, pp.1–29.

Cover, T. M. & Ordentlich, E. (1996), 'Universal portfolios with side information', *IEEE Trans. Inform. Theory* **42**, pp. 348–363.

Goodaire, E. G. (2003), *Linear Algebra A Pure and Applied First Course*, Pearson Education.Inc, Prentice Hall. chapter 3.

Gupta, G. S., Khoon, H. C. & Suhaimi, S. (2001), 'How many securities make a diversified portfolio in klse stocks?', *Asian Academy of Management Journal* **6**(1), pp. 63–79.

Hazan, E., Kalai, A., Kale, S. & Agarwal, A. (2006), Logarithmic regret algorithms for online convex optimization, *in* '19th Annual Conference on Learning Theory (COLT)'.

Helmbold, D. P., Shapire, R. E., Singer, Y. & Warmuth, M. K. (1998), 'On-line portfolio selection using multiplicative updates', *Math. Finance* **8**(4), pp. 325–347.

Ishijima, H. (2001), Numerical methods for universal portfolios. Research Gate.

Kalai, A. & Vempala (2002), 'Efficient algorithms for universal portfolios', *Machine Learning Research* **3**, pp. 423–440.

Kelly, J. J. (1956), 'A new interpretation of information rate', *Journal of Bell Systems Technical* **35**, pp.917–926.

Kivinen, J. & Warmuth, M. K. (1997), 'Exponentiated gradient versus gradient descent for linear predictors', *Information and Computation* **132**, pp. 1–63.

Li, B. & Hoi, S. C. H. (2012), 'Online portfolio selection:a survey'.

Markowitz, H. (1952), 'Portfolio selection', *The Journal of Finance* pp. 77–91.

Nkomo, R. & Kabundi, A. (2013), Kalman filtering and online learning algorithms for portfolio selection. ERSA working paper 394.

Ross, S. M. (2007), *Introduction to Probability Models*, Elsevier, San Diego. pp. 625–657.

Sharpe, W. F. (1963), 'A simplified model for portfolio analysis', *Management Science* **9**(2), pp. 277–293.

Sinakalai & Suppayah (2011), An empirical study on volatility in the malaysia stock market, Master's thesis, University of Malaya, Kuala Lumpur.

Statman, M. (1987), 'How many stocks make a diversified portfolio?', *Journal of Financial and Quantitative Analysis* **22**(3), pp. 353–363.

Tan, C. P. (2002), Performance bounds for dirichlet-weighted universal portfolios, *in* 'The 2002 International Symposium on Information Theory and Its Applications,Xi'an,China', The Society of Information Theory and Its Applications,Tokyo, pp. 587–590.

Tan, C. P. (2004b), Some developments in universal portfolios, *in* B. M. de Silva & N. Mulhopadhyay, eds, 'Proceedings of the Sri Lanka International Statistical Conference', University of Paredeniya and RMIT (Australia) Press, Kandy, Sri Lanka, pp. 495–505.

Tan, C. P. (2013), Performance bounds for the distribution generated universal portfolio, *in* 'Proceedings of the 59th World Statistics Congress of the International Statistical Institute', pp. 5327–5332.

Tan, C. P. & Chu, S. Y. (2012), 'Universal portfolios generated by a matrix and its inverse', *Canadian Journal on Computing in Mathematics, Natural Sciences, Engineering and Medicine* **3**(4), pp. 80–87.

Tan, C. P., Chu, S. Y. & Lim, W. X. (2012), Dirichlet universal portfolios of finite order, *in* 'The 16th International Congress on Insurance: Mathematics and Economic', Vol. 3, Hong Kong, p. pp. 113.

Tan, C. P., Kuang, K. S. & Phoon, S. W. (2015), 'Universal portfolios generated by reciprocal functions of price relatives', *Journal of Mathematics* **11**, pp. 54–61.

Tan, C. P. & Lim, W. X. (2010), Improving the performance of the helmbold universal portfolio with an unbound learning parameter, *in* 'Proceeding of the Int. Conf. on Industrial and Appl. Math.', Indonesia.

Tan, C. P. & Lim, W. X. (2011), Universal portfolios generated by the mahalanobis squared divergence, *in* 'Proceedings of the International Conference in Mathematics and Applications', Bangkok,Thailand, p. pp. 395.

Tan, C. P. & Lim, W. X. (2013), 'Universal portfolio using the best current-run parameter', *Mathematika, Department of Mathematical Sciences, UTM* **29**(1b), pp. 89–94.

Tan, C. P. & Pang, S. T. (2011), On the asymptotic behaviour of the ratio of capitals of the dirichlet universal portfolios, *in* 'Proceedings of the International Conference in Mathematics and Applications', Bangkok,Thailand, p. 399.

Tan, C. P. & Pang, S. T. (2012), A parameter-varying universal portfolio using cyclic constrained-search algorithm, *in* 'IEEE Student Conference, 6–9 October'.

Tuttle, B. & Butcher, D. (2016), 'The most in-demand programming languages on wall street', `/http://news.efinancialcareers.com/us-en/197544/demand-programming-languages-wall-street`. [Online; accessed 12-Oct-2016].

Wackerly, D. D., III, W. M. & Scheaffer, R. L. (2000), *Mathematical Statistics with Application*, Brooks/Cole, Canada. pp. 279-285.

Wikipedia (2016*a*), 'FTSE 100 Index', `http://en.wikipedia.org/wiki/FTSE_100_Index`. [Online; accessed 3-August-2016].

Wikipedia (2016*b*), 'FTSE Bursa Malaysia KLCI', `http://en.wikipedia.org/wiki/FTSE_Bursa_Malaysia_KLCI`. [Online; accessed 3-August-2016].

Yahoo (n.d.), 'Yahoo! finance'.
URL: *finance.yahoo.com*

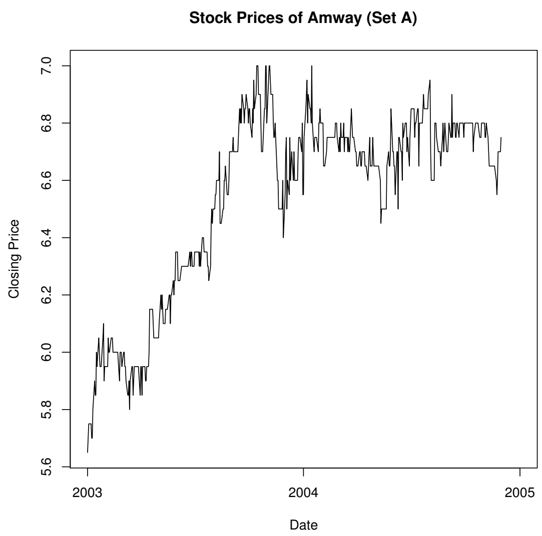
Yeoh, B.-K., Hooy, C.-W. & Arsad, Z. (2010), 'Time varying world integration of the malaysian stock market: A kalman filter approach', *Asian Academy of Management Journal of Accounting and Finance* **6**(2), pp. 1–17.

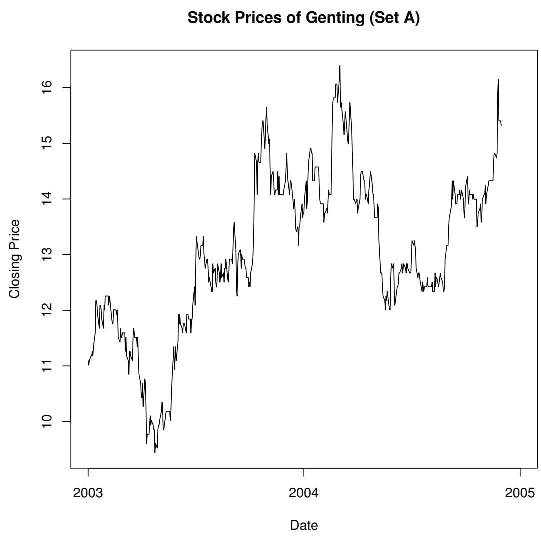
APPENDIX A

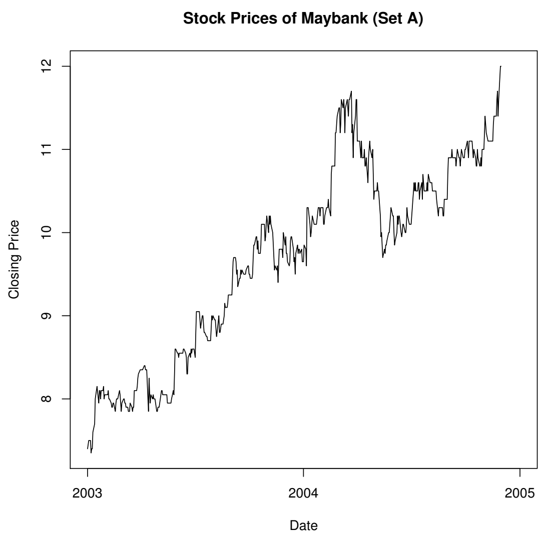
SET A, SET B AND SET C DATASETS

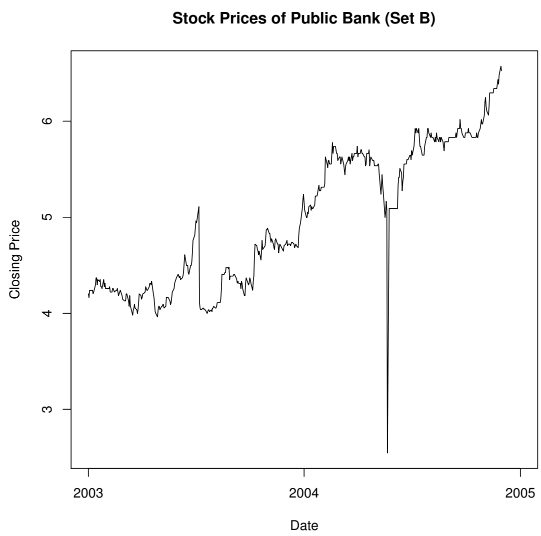
Three sets of stocks data shown in the following figures were selected randomly from KLSE (Kuala Lumpur Stock Exchange) and Yahoo Finance Yahoo (n.d.) for the period from 1 January, 2003 to 30 November, 2004, consisting of 500 trading days. Each set consists of 3 company stocks. Set A consists of the stocks of Malayan Banking, Genting and Amway(M) Holdings. Set B consists of the stocks of Public Bank, Sunrise and YTL Corporation. Lastly, set C consists of the stocks of Hong Leong Bank, RHB Capital and YTL Corporation.

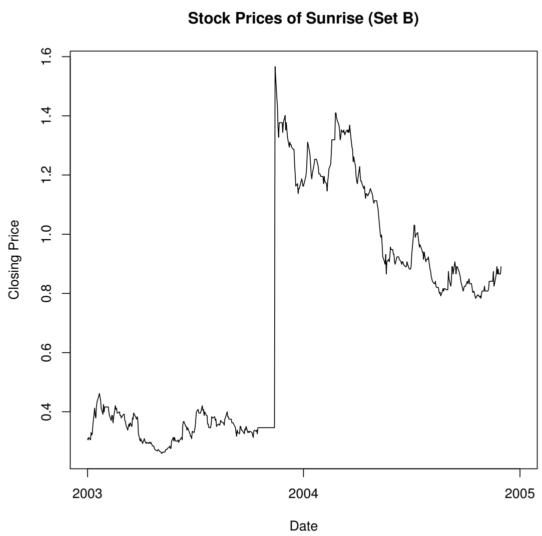
There is a gap shown in the graph of Public Bank and Sunrise. This may cause by the stock splits or bonus issue was announced by the respective company. When a gap in a stock occurs, it means the stock's price has either increased or decreased sharply with no trading occurring in between. On a chart, this abrupt price movement will form an empty space or break between the prices.

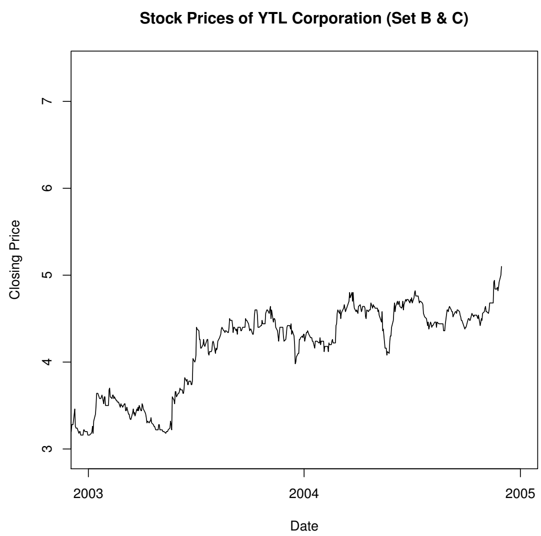


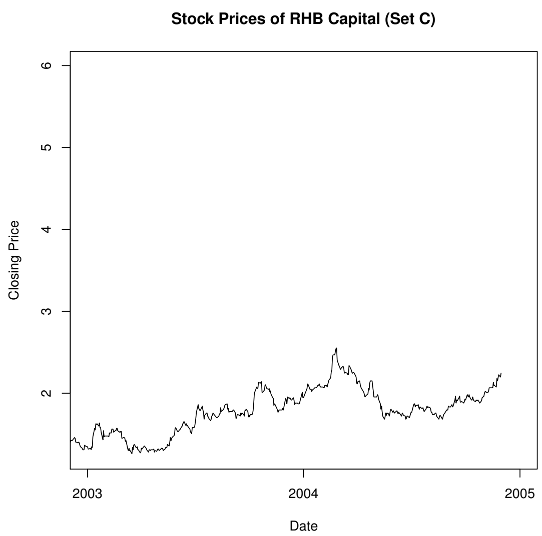


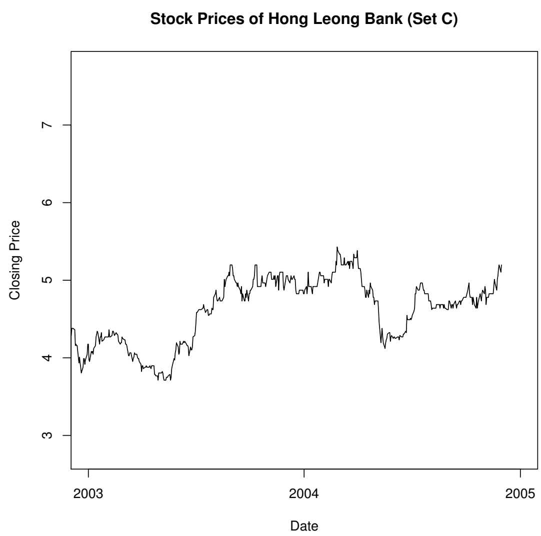

Stock Prices of Maybank (Set A)
Closing Price
Date
8
9
10
11
12
2003
2004
2005


Stock Prices of Public Bank (Set B)
Closing Price
Date
3
4
5
6
2003
2004
2005


Stock Prices of Sunrise (Set B)
Closing Price
Date
0.4
0.6
0.8
1.0
1.2
1.4
1.6
2003
2004
2005


Stock Prices of YTL Corporation (Set B & C)
Closing Price
Date
3
4
5
6
7
2003
2004
2005


Stock Prices of RHB Capital (Set C)
Closing Price
Date
2
3
4
5
6
2003
2004
2005


Stock Prices of Hong Leong Bank (Set C)
Closing Price
Date
3
4
5
6
7
2003
2004
2005

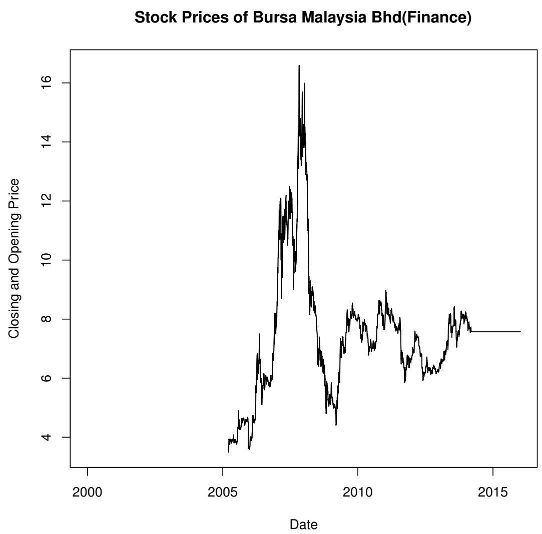
APPENDIX B

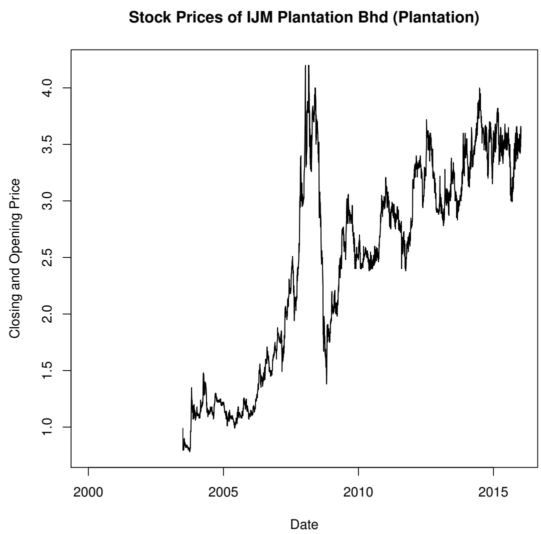
TEN MOST ACTIVE STOCKS

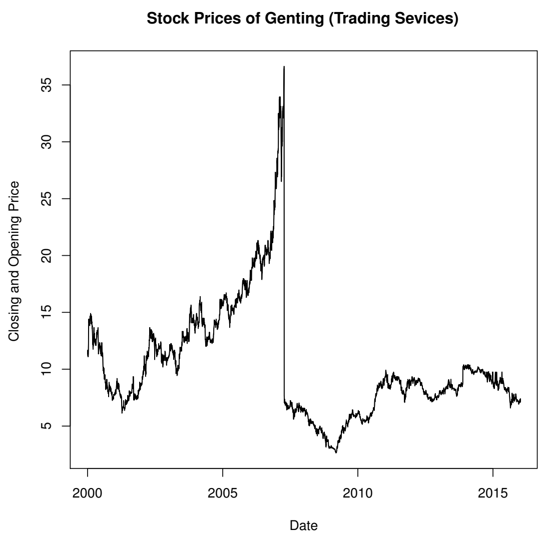
The following figures showed the 10 most active and long-term (having at least 10 years being listed in the stock exchange) stocks from each categories among the 95 selected stocks.

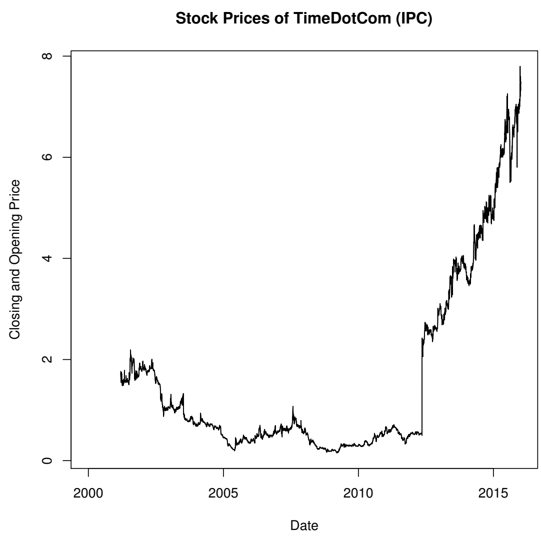
There is a gap shown in the graph of Bursa Malaysia Bhd, Genting and Time Dotcom. This may cause by the stock splits or bonus issue was announced by the respective company. When a gap in a stock occurs, it means the stock's price has either increased or decreased sharply with no trading occurring in between. On a chart, this abrupt price movement will form an empty space or break between the prices.

.

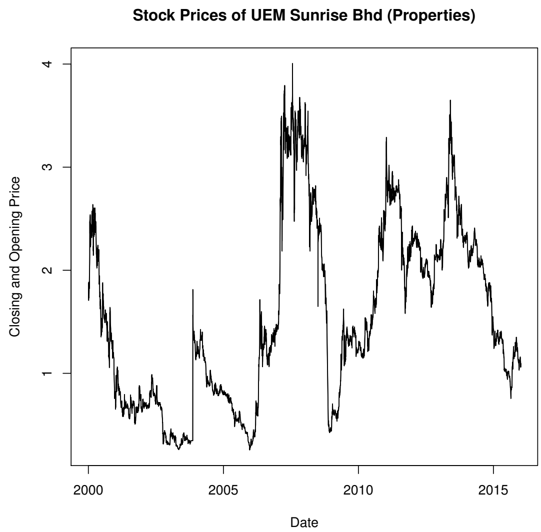


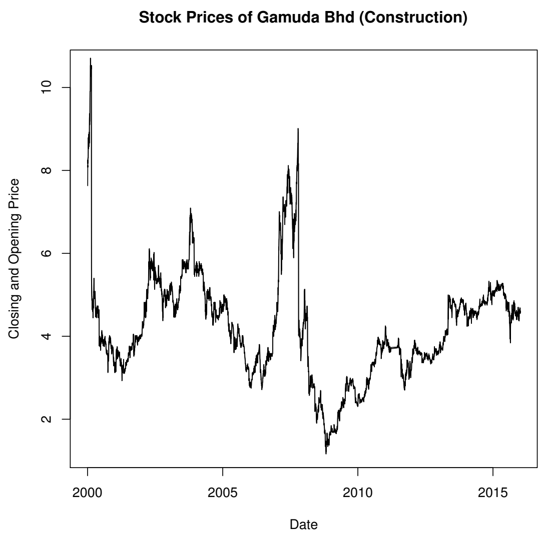


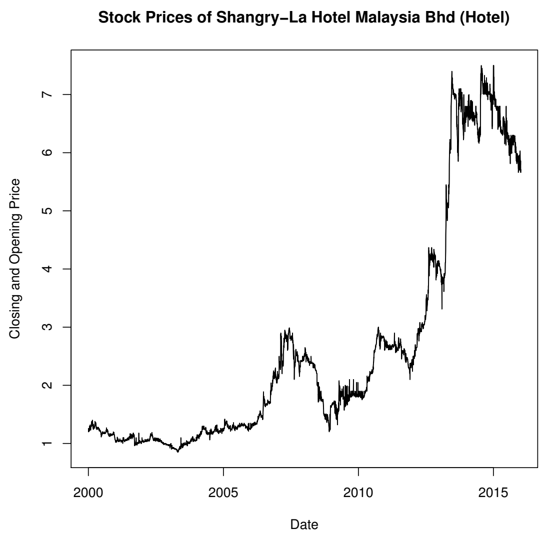

Stock Prices of Genting (Trading Sevices)
Closing and Opening Price
Date

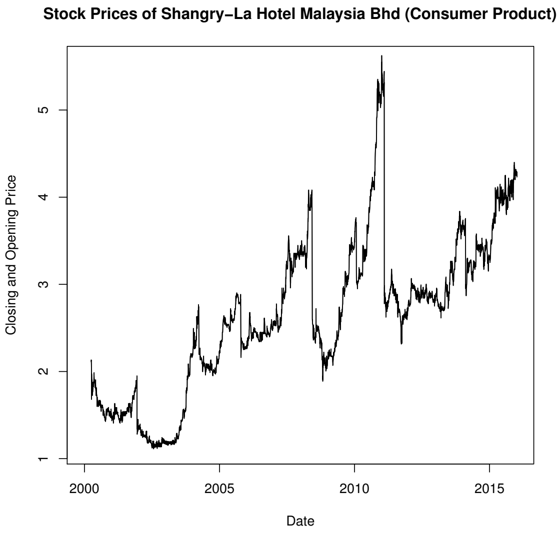

Stock Prices of TimeDotCom (IPC)
Closing and Opening Price
Date

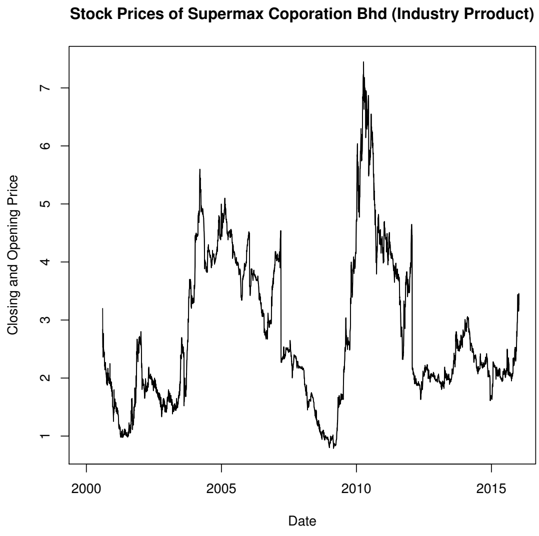

Stock Prices of Air Asia Bhd (Trading Sevices)
Closing and Opening Price
Date


Stock Prices of UEM Sunrise Bhd (Properties)
Closing and Opening Price
Date


Stock Prices of Gamuda Bhd (Construction)
Closing and Opening Price
Date


Stock Prices of Shangry−La Hotel Malaysia Bhd (Hotel)
Closing and Opening Price
Date


Stock Prices of Shangry−La Hotel Malaysia Bhd (Consumer Product)
Closing and Opening Price
1
2
3
4
5
2000
2005
2010
2015
Date


Stock Prices of Supermax Coporation Bhd (Industry Prroduct)
Closing and Opening Price
1
2
3
4
5
6
7
2000
2005
2010
2015
Date

APPENDIX C

WELL PERFORMING PARAMETERS

For selection of well performing parameters for each universal portfolios, we use CRP as a benchmark to evaluate and to compare with these four universal portfolio strategies. A comparison can be made with the wealth obtained by CRP and the wealth achieved by the four proposed universal portfolios. Every one year period starting from year 2000 to year 2015 of the available stock data listed in Table 2. At least 20% of the wealth achieved by proposed universal portfolio performed better than wealth obtained by CRP are analysed. The good performance of the parameters of each universal portfolio with average and standard deviation is observed and listed in the following table.

Strategies	Duration	Average	Standard Deviation
Multinomial [0.8, 0.001, 50]	1 Jan 2015 to 31 Dec 2015	1.872272	0.144625964
Multinomial [0.8, 0.001, 600]	1Jan 2015 to 31 Dec 2015	1.872129	0.1446316
Multinomial [0.9, 0.004, 10]	1 Jan 2015 to 31 Dec 2015	2.0260995	0.083495876
Multinomial [0.9, 0.1, 600]	1 Jan 2015 to 31 Dec 2015	2.021586	0.078432284
Multinomial [0.95, 0.05, 99]	1 Jan 2015 to 31Dec 2015	2.105049	0.047998408
Multinomial [0.9, 0.004, 10]	1 Jan 2014 to 31 Dec 2015	7.3133625	1.162690731
Multinomial [0.9, 0.1, 600]	1 Jan 2014 to 31 Dec 2015	7.3025065	1.205954352
Multinomial [0.95, 0.05, 99]	1 Jan 2014 to 31 Dec 2015	7.9577425	0.708231788
Multinomial [0.9, 0.004, 10]	1 Jan 2013 to 31 Dec 2015	30.06651151	8.336130634
Multinomial [0.9, 0.1, 600]	1 Jan 2013 to 31 Dec 2015	30.0945315	8.537531616
Multinomial [0.95, 0.05, 99]	1 Jan 2013 to 31 Dec 2015	33.7422785	4.745774739

Strategies	Duration	Average	Standard Deviation
Multinomial [0.8, 0.001, 50]	1 Jan 2008 to 31 Dec 2015	158.612327	97.71612695
Multinomial [0.8, 0.001, 600]	1 Jan 2008 to 31 Dec 2015	158.587814	97.72132277
Multinomial [0.9, 0.004, 10]	1 Jan 2008 to 31 Dec 2015	237.065062	74.75428805
Multinomial [0.9, 0.1, 600]	1 Jan 2008 to 31 Dec 2015	238.6583255	77.68982841
Multinomial [0.95, 0.05, 99]	1 Jan 2008 to 31 Dec 2015	296.497761	48.91055484
Multinomial [0.8, 0.001, 50]	1 Jan 2007 to 31 Dec 2015	132.2665145	75.33927112
Multinomial [0.8, 0.001, 600]	1 Jan 2007 to 31 Dec 2015	132.2201705	75.31942405
Multinomial [0.9, 0.004, 10]	1 Jan 2007 to 31 Dec 2015	204.8146685	61.20016333
Multinomial [0.9, 0.1, 600]	1 Jan 2007 to 31 Dec 2015	204.964026	62.10088514
Multinomial [0.95, 0.05, 99]	1 Jan 2007 to 31 Dec 2015	258.1299895	41.37791389
Multinomial [0.8, 0.001, 50]	1 Jan 2006 to 31 Dec 2015	178.6115835	113.2260044
Multinomial [0.8, 0.001, 600]	1 Jan 2006 to 31 Dec 2015	178.537211	113.2070136
Multinomial [0.9, 0.004, 10]	1 Jan 2006 to 31 Dec 2015	277.5363855	92.07861702
Multinomial [0.9, 0.1, 600]	1 Jan 2006 to 31 Dec 2015	274.7625725	91.26704653
Multinomial [0.95, 0.05, 99]	1 Jan 2006 to 31 Dec 2015	347.7052635	58.89452853

Strategies	Duration	Average	Standard Deviation
Multinomial [0.8, 0.001, 50]	1 Jan 2005 to 31 Dec 2015	239.105569	167.9310497
Multinomial [0.8, 0.001, 600]	1 Jan 2005 to 31 Dec 2015	239.00653	167.8962275
Multinomial [0.9, 0.004, 10]	1 Jan 2005 to 31 Dec 2015	380.6836375	141.8727569
Multinomial [0.9, 0.1, 600]	1 Jan 2005 to 31 Dec 2015	379.5841465	142.4877631
Multinomial [0.95, 0.05, 99]	1 Jan 2005 to 31 Dec 2015	487.8792055	95.32278617
Multinomial [0.8, 0.001, 50]	1 Jan 2004 to 31 Dec 2015	400.2781995	212.1060772
Multinomial [0.8, 0.001, 600]	1 Jan 2004 to 31 Dec 2015	400.128036	212.0530265
Multinomial [0.9, 0.004, 10]	1 Jan 2004 to 31 Dec 2015	604.2257875	157.7730287
Multinomial [0.9, 0.1, 600]	1 Jan 2004 to 31 Dec 2015	602.6234155	157.7730287
Multinomial [0.95, 0.05, 99]	1 Jan 2004 to 31 Dec 2015	748.3181105	102.8328358
Multinomial [0.001, 0.8, 50]	1 Jan 2003 to 31 Dec 2015	32.215131	11.89520483
Multinomial [0.001, 0.8, 600]	1 Jan 2003 to 31 Dec 2015	32.187138	11.91306069
Multinomial [0.001, 0.8, 50]	1 Jan 2001 to 31 Dec 2015	24.7398855	0.16061577
Multinomial [0.001, 0.8, 600]	1 Jan 2001 to 31 Dec 2015	24.730969	0.160571222

Strategies	Duration	Average	Standard Deviation
Brownian motion (100, 1, 10, 1.4, 2.5, 3.6)	1 Jan 2015 to 31 Dec 2015	2.0265425	0.079012819
Brownian-motion (100, 10, 1, 1.4, 2.5, 3.6)	1Jan 2015 to 31 Dec 2015	2.023856	0.070359953
Brownian-motion (100, 1, 10, 1.4, 2.5, 3.6)	1 Jan 2014 to 31 Dec 2015	7.3190495	1.146490914
Brownian-motion(100, 10, 1, 1.4, 2.5, 3.6)	1 Jan 2014 to 31 Dec 2015	7.30123	1.118149367
Brownian-motion (100, 1, 10, 1.4, 2.5, 3.6)	1 Jan 2013 to 31 Dec 2015	30.064781	8.165413136
Brownian-motion (100, 10, 1, 1.4, 2.5, 3.6)	1 Jan 2013 to 31 Dec 2015	29.811969	7.895890902
Brownian-motion [100, 1, 10, 1.4, 2.5, 3.6]	1 Jan 2008 to 31 Dec 2015	235.709762	70.87220535
Brownian-motion [100, 10, 1, 1.4, 2.5, 3.6]	1 Jan 2008 to 31 Dec 2015	238.609657	75.33569953

Strategies	Duration	Average	Standard Deviation
Brownian-motion [100, 1, 10, 1.4, 2.5, 3.6]	1 Jan 2007 to 31 Dec 2015	205.250947	60.58070031
Brownian-motion [100, 10, 1, 1.4, 2.5, 3.6]	1 Jan 2007 to 31 Dec 2015	201.434666	55.46979897
Brownian-motion [100, 1, 10, 1.4, 2.5, 3.6]	1 Jan 2006 to 31 Dec 2015	278.414544	89.7088738
Brownian-motion [100, 10, 1, 1.4, 2.5, 3.6]	1 Jan 2006 to 31 Dec 2015	271.118674	82.66402834
Brownian-motion [100, 1, 10, 1.4, 2.5, 3.6]	1 Jan 2005 to 31 Dec 2015	384.4106285	137.6970185
Brownian-motion [100, 10, 1, 1.4, 2.5, 3.6]	1 Jan 2005 to 31 Dec 2015	382.468823	137.9797162
Brownian-motion [100, 1, 10, 1.4, 2.5, 3.6]	1 Jan 2004 to 31 Dec 2015	604.4072875	154.8509637
Brownian-motion [100, 10, 1, 1.4, 2.5, 3.6]	1 Jan 2004 to 31 Dec 2015	595.479407	148.4204222
Brownian-motion [1, 10, 100, 1.4, 2.5, 3.6]	1 Jan 2003 to 31 Dec 2015	50.535135	13.37465041
Brownian-motion [1, 10, 100, 1.4, 3.6, 2.5]	1 Jan 2003 to 31 Dec 2015	50.534848	13.37537307
Brownian-motion [1, 10, 100, 2.5, 1.4, 3.6]	1 Jan 2003 to 31 Dec 2015	50.53511	13.37392067
Brownian-motion [1, 10, 100, 2.5, 3.6, 1.4]	1 Jan 2003 to 31 Dec 2015	50.5346405	13.37510508
Brownian-motion [1, 10, 100, 3.6, 1.4, 2.5]	1 Jan 2003 to 31 Dec 2015	50.5347845	13.37350136
Brownian-motion [1, 10, 100, 3.6, 2.5, 1.4]	1 Jan 2003 to 31 Dec 2015	50.5346015	13.37396381
Brownian-motion [10, 1, 100, 1.4, 2.5, 3.6]	1 Jan 2003 to 31 Dec 2015	45.0898545	5.490426062

Strategies	Duration	Average	Standard Deviation
Brownian-motion[1, 100, 10, 1.4, 2.5, 3.6]	1 Jan 2001 to 31 Dec 2015	22.2905245	0.497273551
Brownian-motiom [1, 10, 100, 1.4, 2.5, 3.6]	1 Jan 2000 to 31 Dec 2015	19.3637865	6.39364608
Brownian-motion [1, 10, 100, 1.4, 3.6, 2.5]	1 Jan 2000 to 31 Dec 2015	18.075308	4.570732577
Brownian-motion [1, 10, 100, 2.5, 1.4, 3.6]	1 Jan 2000 to 31 Dec 2015	19.3638495	6.393660223
Brownian-motion [1, 10, 100, 2.5, 3.6, 1.4]	1 Jan 2000 to 31 Dec 2015	19.364966	6.394042767
Brownian-motion [1, 10, 100, 3.6, 1.4, 2.5]	1 Jan 2000 to 31 Dec 2015	19.36463	6.393915488
Brownian-motion [1, 10, 100, 3.6, 2.5, 1.4]	1Jan 2000 to 31 Dec 2015	19.365065	6.394063981
Brownian-motion [10, 1, 100, 1.4, 2.5, 3.6]	1 Jan 2000 to 31 Dec 2015	18.3479715	4.638068234

Strategies	Duration	Average	Standard Deviation
MultivariateNormal [7.1, 0.2, 0.8, 1.0]	1 Jan 2015 to 31 Dec 2015	1.955066	0.09180933
Multivariate Normal [7.1, 0.8, 0.2, 1.0]	1Jan 2015 to 31 Dec 2015	1.956286	0.081379505
Multivariate Normal [9.0, 0.1, 0.1, 2.0]	1 Jan 2015 to 31 Dec 2015	2.0877165	0.041910926
Multivariate Normal [7.1, 0.2, 0.8, 1.0]	1 Jan 2014 to 31 Dec 2015	6.742861	1.501959857
Multivariate Normal [7.1, 0.8, 0.2, 1.0]	1 Jan 2014 to 31 Dec 2015	6.69516	1.395113194
Multivariate Normal [9.0, 0.1, 0.1, 2.0]	1 Jan 2014 to 31 Dec 2015	7.770994	0.763040342
Multivariate Normal [7.1, 0.2, 0.8, 1.0]	1 Jan 2013 to 31 Dec 2015	185.063506	72.81175258
multivariate Normal [7.1, 0.8, 0.2, 1.0]	1 Jan 2013 to 31 Dec 2015	193.523848	85.71626608
Multivariate Normal[9.0, 0.1, 0.1, 2.0]	1 Jan 2013 to 31 Dec 2015	276.3518625	51.95913896
Multivariate Normal [7.1, 0.2, 0.8, 1.0]	1 Jan 2008 to 31 Dec 2015	27.2618865	9.966086718
multivariate Normal [7.1, 0.8, 0.2, 1.0]	1 Jan 2008 to 31 Dec 2015	26.586557	8.681444109
Multivariate Normal [9.0, 0.1, 0.1, 2.0]	1 Jan 2008 to 31 Dec 2015	32.4116065	5.123314606

Strategies	Duration	Average	Standard Deviation
Multivariate Normal [7.1, 0.2, 0.8, 1.0]	1 Jan 2007 to 31 Dec 2015	164.975529	69.01928153
multivariate Normal [7.1, 0.8, 0.2, 1.0]	1 Jan 2007 to 31 Dec 2015	161.813746	64.6768412
Multivariate Normal [9.0, 0.1, 0.1, 2.0]	1 Jan 2007 to 31 Dec 2015	240.5946675	45.23218466
Multivariate Normal [7.1, 0.2, 0.8, 1.0]	1 Jan 2006 to 31 Dec 2015	225.052084	98.65878373
multivariate Normal [7.1, 0.8, 0.2, 1.0]	1 Jan 2006 to 31 Dec 2015	214.1665545	89.89713179
Multivariate Normal [9.0, 0.1, 0.1, 2.0]	1 Jan 2006 to 31 Dec 2015	327.6703685	65.94573513
Multivariate Normal [7.1, 0.2, 0.8, 1.0]	1 Jan 2005 to 31 Dec 2015	313.443567	154.3113924
multivariate Normal [7.1, 0.8, 0.2, 1.0]	1 Jan 2005 to 31 Dec 2015	310.560472	154.978443
Multivariate Normal [9.0, 0.1, 0.1, 2.0]	1 Jan 2005 to 31 Dec 2015	467.2951875	101.9507625

Strategies	Duration	Average	Standard Deviation
Multivariate Normal [7.1, 0.2, 0.8, 1.0]	1 Jan 2004 to 31 Dec 2015	490.154204	183.1714692
multivariate Normal [7.1, 0.8, 0.2, 1.0]	1 Jan 2004 to 31 Dec 2015	471.429849	159.3962751
Multivariate Normal [9.0, 0.1, 0.1, 2.0]	1 Jan 2004 to 31 Dec 2015	701.0537485	110.2752124
Multivariate Normal [0.2, 0.8, 7.1, 1.0]	1 Jan 2003 to 31 Dec 2015	44.052089	15.0078507
multivariate Normal [0.8, 0.2, 7.1, 1.0]	1 Jan 2003 to 31 Dec 2015	39.6614195	8.585701144
Multivariate Normal [0.2, 8.1, 9.1, 2.0]	1 Jan 2001 to 31 Dec 2015	25.8827735	1.301132339
Multivariate Normal [0.2, 0.8, 7.1, 1.0]	1 Jan 2000 to 31 Dec 2015	17.940328	7.129771817
Multivariate Normal [0.2, 8.1, 9.1, 2.0]	1 Jan 2000 to 31 Dec 2015	14.691917	4.566167495
Multivariate Normal [0.8, 0.2, 7.1, 1.0]	1 Jan 2000 to 31 Dec 2015	17.200072	6.209487897

Strategies	Duration	Average	Standard Deviation
Ornstein [16, 0.1, 0.5,0.2,1.2,2]	1 Jan 2015 to 31 Dec 2015	2.121387	0.04412912
Ornstein [16, 0.5, 0.1,0.2,1.2,2]	1Jan 2015 to 31 Dec 2015	2.1186715	0.035085931
Ornstein [50, 0.5, 2,0.2,1.2,2]	1 Jan 2015 to 31 Dec 2015	2.111905	0.047320293
Ornstein [50, 2, 0.5,0.2,1.2,2]	1 Jan 2015 to 31 Dec 2015	2.109227	0.038255891
Ornstein [16, 0.1, 0.5,0.2,1.2,2]	1 Jan 2014 to 31 Dec 2015	8.067012	0.570505065
Ornstein[16, 0.5, 0.1,0.2,1.2,2]	1Jan 2014 to 31 Dec 2015	8.04818	0.536612023
Ornstein [50, 0.5, 2,0.2,1.2,2]	1 Jan 2014 to 31 Dec 2015	7.9887175	0.630782382
Ornstein [50, 2, 0.5,0.2,1.2,2]	1 Jan 2014 to 31 Dec 2015	7.9701295	0.594828831
Ornstein [16, 0.1, 0.5,0.2,1.2,2]	1 Jan 2013 to 31 Dec 2015	34.254638	3.75991303
Ornstein[16, 0.5, 0.1,0.2,1.2,2]	1Jan 2013 to 31 Dec 2015	33.9694685	3.468970409
Ornstein [50, 0.5, 2,0.2,1.2,2]	1 Jan 2013 to 31 Dec 2015	33.795792	4.234245915
Ornstein [50, 2, 0.5,0.2,1.2,2]	1 Jan 2013 to 31 Dec 2015	33.5046435	3.927663507

Strategies	Duration	Average	Standard Deviation
Ornstein [50, 0.5, 2,0.2,1.2,2]	1 Jan 2009 to 31 Dec 2015	189.172132	101.8969
Ornstein [50, 2, 0.5,0.2,1.2,2]	1 Jan 2009 to 31 Dec 2015	200.759345	117.3114418
Ornstein [16, 0.1, 0.5,0.2,1.2,2]	1 Jan 2008 to 31 Dec 2015	301.439745	35.3349362
Ornstein [16, 0.5, 0.1]	1 Jan 2008 to 31 Dec 2015	305.113613	41.01533286
Ornstein [50, 0.5, 2]	1 Jan 2008 to 31 Dec 2015	293.96517	39.45733479
Ornstein [50, 2, 0.5]	1 Jan 2008 to 31 Dec 2015	297.5198085	45.04923068
Ornstein [16, 0.1, 0.5,0.2,1.2,2]	1 Jan 2007 to 31 Dec 2015	266.2532905	35.79009871
Ornstein [16, 0.5, 0.1,0.2,1.2,2]	1 Jan 2007 to 31 Dec 2015	264.900757	34.06162922
Ornstein [50, 0.5, 2,0.2,1.2,2]	1 Jan 2007 to 31 Dec 2015	259.4439375	38.96697109
Ornstein [50, 2, 0.5,0.2,1.2,2]	1 Jan 2007 to 31 Dec 2015	257.844673	36.90749077
Ornstein [16, 0.1, 0.5,0.2,1.2,2]	1 Jan 2006 to 31 Dec 2015	363.7716645	50.53144935
Ornstein [16, 0.5, 0.1,0.2,1.2,2]	1 Jan 2006 to 31 Dec 2015	358.3183775	47.82847994
Ornstein [50, 0.5, 2,0.2,1.2,2]	1 Jan 2006 to 31 Dec 2015	353.992774	54.97640672
Ornstein [50, 2, 0.5,0.2,1.2,2]	1 Jan 2006 to 31 Dec 2015	347.856213	51.40405377

Strategies	Duration	Average	Standard Deviation
Ornstein [16, 0.1, 0.5,0.2,1.2,2]	1 Jan 2005 to 31 Dec 2015	511.4671185	78.54948217
Ornstein [16, 0.5, 0.1,0.2,1.2,2]	1 Jan 2005 to 31 Dec 2015	509.6434985	80.83019004
Ornstein [50, 0.5, 2,0.2,1.2,2]	1 Jan 2005 to 31 Dec 2015	497.090707	85.22536709
Ornstein [50, 2, 0.5,0.2,1.2,2]	1 Jan 2005 to 31 Dec 2015	494.9842985	87.83733822
Ornstein [16, 0.1, 0.5,0.2,1.2,2]	1 Jan 2004 to 31 Dec 2015	771.5952125	80.67172882
Ornstein [16, 0.5, 0.1,0.2,1.2,2]	1 Jan 2004 to 31 Dec 2015	762.823391	71.85391563
Ornstein [50, 0.5, 2,0.2,1.2,2]	1 Jan 2004 to 31 Dec 2015	752.528413	88.88544513
Ornstein [50, 2, 0.5,0.2,1.2,2]	1 Jan 2004 to 31 Dec 2015	742.4438335	78.66768072
Ornstein[0.1, 0.5, 16,0.2,1.2,2]	1 Jan 2003 to 31 Dec 2015	57.5212825	5.81661442
Ornstein[0.5, 0.1, 16,0.2,1.2,2]	1 Jan 2003 to 31 Dec 2015	55.8499615	3.217462427
Ornstein [0.5, 16, 0.1,0.2,1.2,2]	1 Jan 2003 to 31 Dec 2015	56.7503485	6.569207966
Ornstein [0.1, 0.5, 16,0.2,1.2,2]	1 Jan 2002 to 31 Dec 2015	27.1964845	1.721150938
Ornstein[0.5, 0.1, 16],0.2,1.2,2	1 Jan 2002 to 31 Dec 2015	26.3251285	0.779508152
Ornstein [0.5, 16, 0.1,0.2,1.2,2]	1 Jan 2002 to 31 Dec 2015	26.976849	1.914191797

APPENDIX D

PUBLICATIONS RELATED TO THIS RESEARCH

1. S. T. Pang and C. P. Tan, (2013a) Empirical performance of multivariate normal universal portfolio. *In Proceedings of the 3rd International Conference on Mathematical Sciences*, volume AIP Conference Proceedings, Melville N. Y., American Institute of Physics.

2. C. P. Tan and S. T. Pang, (2013b),The finite and moving order multinomial universal portfolio. *In 2012 iCAST Contemporary Mathematics, Mathematical Physics and Their Application*. Journal of Physics., Conference Series 435. Institute of Physics, UK, Bristol, 012039.

3. C. P. Tan and S. T. Pang, (2014a), Universal portfolios generated by weakly stationary. *In Proceedings of the International Conference on Quantitative Sciences and its Application*, volume AIP Conference Proceedings, Melville N. Y.,. American Institute of Physics.

4. C. P. Tan and S. T. Pang,(2014b), Performance of brownian-motion generated universal portfolios. *In Proceedings of the 3rd International Conference on Mathematical Sciences*, volume AIP Conference Proceedings, Melville N. Y.,. American Institute of Physics.

5. S. T. Pang and H. H. Liew and Y. F. Chang,(2017), Performance of Finite order Distribution-Generated Universal Portfolios. *In Proceedings of the 4th International Conference on Mathematical Sciences*, volume AIP Conference Proceedings, Melville N. Y.,. American Institute of Physics.