

Order Book Based Decision Support Model for Traders

Matharage Dona Piyumi Radeeshani

13000952

University of Colombo

School of Computing

2018



Order Book Based Decision Support Model for Traders

M.D.P Radeeshani

Index No: 13000952

Supervisors: Dr F.H.A.M Premachandra

Dr K.D Sandaruwan

December 2017

Submitted in partial fulfillment of the requirements of the

B.Sc in Computer Science Final Year Project (SCS4124)



Declaration

I certify that this dissertation does not incorporate, without acknowledgement, any material previously submitted for a degree or diploma in any university and to the best of my knowledge and belief, it does not contain any material previously published or written by another person or myself except where due reference is made in the text. I also hereby give consent for my dissertation, if accepted, be made available for photocopying and for interlibrary loans, and for the title and abstract to be made available to outside organizations.

Candidate Name: M.D Piyumi Radeeshani

.....

Signature of Candidate

Date:

This is to certify that this dissertation is based on the work of

Ms. M.D Piyumi Radeeshani

under my supervision. The thesis has been prepared according to the format stipulated and is of acceptable standard.

Supervisor Name: Dr F.H.A.M Premachandra

.....

Signature of Supervisor

Date:

Supervisor Name: Dr K.D Sandaruwan.

.....

Signature of Supervisor

Date:

Abstract

Stock market trading is a complicated task when considering the profit earning because stock market behavior is dynamic and complex. Kelly criterion is an optimal strategy which is used for betting systems and also for stock market investing or trading. There are several models which have modeled Kelly Criterion to the stock market. However, those models are only theoretical models with lack of practical validity. Practical behavior, dynamicity of the stock market and giving clear decisions to traders are not concerned in the existing models implemented using Kelly criterion to stock market. Also validation of models with real stock market have not being conducted.

In this thesis a dynamic computational model of Kelly criterion for stock traders is built. Selecting one suitable model from the above models and order book and trades data are used in order to solve the above stipulated problems. Four cases are introduced on how short term returns behave like normal. Then model considered normal distribution of the short term returns.

After investigating the trades with chosen Kelly criterion to stock market model, the methodology and equations on how to handle trades with the existing Kelly model of stock market is introduced. Introduction of short term traders or day traders to the Kelly model of stock market is carried out. Also, after investigating trades with the chosen Kelly criterion to stock market model, introduction of methodology and equations on how to use order book with the existing Kelly model of stock market is conducted. Then, validation of the model with the real stock market behavior and comparison of the accuracy of the model without modifications to the model with order book is conducted. Developed dynamic model provides stock buy or sell decisions to a day trader.

Preface

The basic proofs and derivations of equations of section 1.1.2 and section 2.3 are based on the research group who developed the theoretical model I selected.

Investigation of trades with used Kelly criterion to stock market model and introducing the methodology and equations about how to handle trades with the existing Kelly model of stock market in section 3.1.1 comprises entirely of my own work.

Also, investigation of order book with used Kelly criterion to stock market model and introducing the methodology and equations about how to handle trades with the existing Kelly model of stock market in section 3.1.2 comprises entirely of my own work.

Order book and trades gathering, preprocessing analyzing and integration implementation with python programming in chapter 4 comprises entirely of my own work.

In section 5.1 I introduced the four cases introduced on how short term returns behave like normal. Also the calculations using real data in section 5.1 with introduced equations for trades, calculations using real data in section 5.2 with introduced equations for order book, the decision giving scenario from the output in section 5.4, model validation scenario in section 5.5 and the accuracy comparison without order book are entirely my work.

Acknowledgement

I would like to express my deepest gratitude to my supervisor, Dr. F.H. Ann Mindika Premachandra whose expertise, support and generous guidance motivated me to make this research a success. Without her mentorship and the wisdom, this thesis would not have been possible.

My sincere thanks also goes to my co-supervisor Dr. K.D Sandaruwan for his insightful comments, kind support and encouragement given.

I am highly grateful to Dr. Chamath Keppetiyagama for giving me valuables and important ideas to continuing the research successfully. Without his concepts it would be more difficult to continue the research successfully.

I would like to thank Dr. Kasun Gunawardena for his valuable feedbacks given.

Also, I am thankful to the DirectFN for providing me the access to their data feeds.

I would like to pay my heart full grateful attitude to University Of Colombo School of Computing along with faculty of computer science for providing me with the valuable opportunity to study in a world class university and providing us with units like final year researches to improve and enhance our instinct in field of computer science

Last but not the least, I would like to thank my family and my friends and specially to Kasun Ranga for supporting me throughout the research work and my life in general.

Table of Contents

- Declaration iv
- Abstract v
- Preface..... vi
- Acknowledgement..... vii
- List of Figures..... x
- List of Tables..... xii
- Chapter 1 – Introduction 1
 - 1.1 Background to the Research 1
 - 1.1.1 The Kelly Criterion 1
 - 1.1.2 Motivation of Kelly Criterion to Gambling Games 1
 - 1.1.3 Main idea of Kelly Criterion 2
 - 1.1.4 Stock Market 3
 - 1.1.5 Stock Market Trading 3
 - 1.1.6 Kelly to the Stock Market 4
 - 1.1.7 Order Book, Trades and Kelly Criterion 4
 - 1.2 Research Problem and Questions 5
 - 1.2.1 Research Problem..... 5
 - 1.2.2 Research Question..... 5
 - 1.2.3 Aims and Objectives 5
 - 1.3 Justification for the research..... 5
 - 1.4 Methodology 7
 - 1.5 Outline of the Dissertation 8
 - 1.6 Definitions 9
 - 1.7 Delimitations of Scope 11
 - 1.8 Conclusion 11
- Chapter 2 – Literature Review..... 12
 - 2.1 Order book and trades data from trader’s perspective 12
 - 2.2 Kelly Criterion 14
 - 2.3 Kelly Criterion model to stock market..... 14
 - 2.4 Conclusion 18
- Chapter 3 – Design 19
 - 3.1 Kelly computational model to Stock Market..... 20
 - 3.1.1 Trade data Investigation with the Kelly model 22
 - 3.1.2 Order Book Investigation with the Kelly model 24

Chapter 4 – Implementation	33
4.1 Order book and Trades data Gathering	33
4.2 Order book and trades data processing	34
4.3 Integration implementation process for output	35
Chapter 5 – Result and Evaluation	39
5.1 Trades data towards results	40
5.2 Order book towards results.....	47
5.3 Output from the model	48
5.4 Decisions to the trader from the model.....	53
5.5 Model validation.....	53
5.6 Accuracy without order book.....	57
Chapter 6 – Conclusions	58
6.1 conclusions	58
6.2 Limitations	59
6.3 Implications for further research	59
References.....	60
Appendix A: Code Listing.....	62

List of Figures

Figure 1.1 Research Methodology	7
Figure 2.1 Fraction of current wealth verses exponential increase per trial.....	15
Figure 2.2 Altered normal distribution of annual returns.....	16
Figure 3.1 overall design of the research.	19
Figure 3.2 PDF of normal distribution	25
Figure 3.3 Normal distribution of unit returns	26
Figure 3.4 An order book update during one second	27
Figure 3.5 Analyzed order book to calculate returns	28
Figure 3.6 Cut off points of distribution	30
Figure 3.7 Distribution cutting	30
Figure 3.8 Adding h value to the distribution	31
Figure 4.1 Section of an order book log file	33
Figure 4.2 Section of a trades data file.....	34
Figure 4.3 Processed order book	34
Figure 4.4 Processed trades	34
Figure 5.1 Implemented model	39
Figure 5.2 An analyzed trade data set to calculate q values.....	40
Figure 5.3 Sample p-value check for 4 weeks	43
Figure 5.4 Distribution for 4 weeks	43
Figure 5.5 Sample p-value check for 3 weeks	44
Figure 5.6 Distribution for 3 weeks	44
Figure 5.7 Sample p-value check for 2 weeks	45
Figure 5.8 Distribution for 2 weeks	45
Figure 5.9 Sample p-value check for 1 weeks	46

Figure 5.10 Distribution for 1 weeks	46
Figure 5.11 Cut off points from order book	47
Figure 5.12 Output with $0 < f_{\max} < 1$	49
Figure 5.13 Output with $f_{\max} < 1$	50
Figure 5.14 Output with $f_{\max} > 1$	51
Figure 5.15 Output with very large $f > 1$	52
Figure 5.16 Model validation 1	55
Figure 5.17 Model validation 2	56

List of Tables

Table 5.1 Decision to day trader.....	55
Table 5.2 Day trader decision evaluation.....	56

Chapter 1 – Introduction

Stock Market plays a vital role within the financial domain. Trading in the stock market has become common due to the fact that investors have involved trading in stock markets for decades. Traders cannot predict the future outcome of their investments because the behavior of the stock market is a very complex and dynamic. However, traders with different beliefs use different models in order to maximize the utility of their trading. Kelly criterion also such a model can model to stock market as an optimization problem.

1.1 Background to the Research

1.1.1 The Kelly Criterion

John Kelly, who worked for AT&T's Bell Laboratory, originally developed the Kelly Criterion to assist AT&T with its long distance telephone signal noise issues [16]. Soon after, the method was published as "A New Interpretation of Information Rate" (1956) [12]. However, the gambling community got wind of it and identified the potential of the application of Kelly's findings as optimal betting system.

1.1.2 Motivation of Kelly Criterion to Gambling Games

Imagine that someone is playing a coin tossing game against an infinitely wealthy opponent. The coin is a biased coin and bets are even money bets made on repeatedly independent trials. Further suppose that on each trial, win probability is $p > 1/2$ and probability of losing is $q = 1 - p$. Capital is W_0 and the problem is to decide what is the amount B_i to bet on each i^{th} trial. Considering $E(W_n)$ as the expected wealth after n trials, Then equation can be write as

$$E(W_n) = W_0 + \sum_{i=1}^n (p - q) E(B_i) \quad (1)$$

$p > 1/2$. So we have that $p - q > 0$. Thus, we can maximize the expected amount of money ($E(W_n)$) by maximizing $E(B_i)$ at each trial. Thus, to maximize the expected value of wealth we should

bet our all our money at each trial. However, the probability of ruin after n trials is given by $1 - p^n$ and $\frac{1}{2} < p < 1$ and $\lim_{n \rightarrow \infty} [1 - p^n] = 1$. So the ruin is almost sure. Thus, betting to maximize the expected wealth is an undesirable strategy.

Also if we play to minimize the probability of eventual ruin ($x_i=0$ on the i^{th} trial) then gambler's ruin formula [16] can be used to show that we can only minimize the ruin by making the minimum bet on each trial. However it also minimizes the expected average wealth. Thus it is also not suitable.

Some intermediate strategy is required which is between maximizing $E(X_n)$ and minimizing the probability of ruin. So Kelly Criterion enters the picture as an optimal strategy.

1.1.3 Main idea of Kelly Criterion

In Kelly's original paper [12], he proposed a different criterion for gamblers. The classic gambler thought to maximize the expected value of wealth. As explained in section 1.1.2 maximizing the expected value of wealth occurs a problem. Rather than maximizing the expected value of wealth, Kelly maximizes the expected value of the logarithm of gambler's wealth. So the Kelly criterion optimal strategy solves the problems of maximizing the expected value of gambler's wealth.

Logarithm of wealth can be identified as a utility function. Thus, similarly Kelly Criterion can be called as maximization of expected value of the utility function which gives us a way to measure gambler's preferences for wealth and the amount of risk they are willing to undertake in the hope of attaining greater wealth [22]. Log function is a concave function. So Kelly described about a risk-averse trader or investor who prefer a certain outcome to an uncertain one, even though they have same expectation. Thus rather than a strategy, Kelly model the nature and the behavior of humans. Then, Kelly illustrates a decision for a trader considering risky and comparatively risk free investment which I further explain in chapter 3.

1.1.4 Stock Market

The stock market is a collection of market and exchanges where issuing and trading of stocks of publicly held companies, bond and other securities. The stock market also considered as an aggregation of buyers and sellers of stocks.

The stock market can be divided into two main parts as primary market and the secondary market. Primary market is where new securities or shares are first sold through the initial public offerings. Initial public offerings means the first time that the stocks of a private company offered to the public. Secondary market is where both individual and company wise investors are involved.

Stock of larger companies are usually traded through exchanges. Those companies have their stock listed on a stock exchange. Such stock exchanges have located in major cities all over the world.

1.1.5 Stock Market Trading

Trading in stock market means buying and selling stocks or shares as a buyer or seller. Traders in the stock market range from small individual stock investors to larger trader investors. Traders can be in anywhere in the world and may include banks, companies and some funds.

To happening a trade, both buyers and sellers have to agree on a price. A buyer bids a specific price for a stock and a seller asks a specific price for the same stock. When the bid and ask prices match, a trade happens on a first come first served manner if there are multiple askers and bidders at a same price.

In the past, trading took place at the physical stock exchanges. But nowadays it has totally varied with the electronic trading. Electronic trading is a method of trading securities or shares electronically. An electronic trading platform is used to connect buyers and sellers together. Also, with the improvement of the technology electronic order book has introduced for the security trading in the stock market.

1.1.5.1 Order Book

An order book is a list of electronic list of buy and sell orders for a specific security organized by the price level. Order book lists the number of shares being bid or offered at each price point and it is dynamic and constantly updated in real time throughout the day. The highest bid and the lowest ask

are referred as the top of the book. The book depth refers simply to the number of price levels available at a particular time in the book. When a bid or ask is placed, if it matches to the top of the order book it executes. That bid or ask does not include to the order book. Otherwise that bid or ask adds to the order book. Then, orders are added to the order book is called limit orders. As in order book contains the list of interested buyers, sellers and their prefer stock price, stock volume(the number of shares) preferences we can identified order book as a place which contains the utility of stock traders.

1.1.5.2 Trades

To happen a trade there should be a stock buyer and a stock seller. Trades contains the details of trades happened in the stock market. Information about the price of the current trade, number of share a particular trader bought and a particular trader sold, open price, close price, high price, low prices etc.

1.1.6 Kelly to the Stock Market

In the stock market money is invested in stocks that have highly expected return. Each investing or trading in stock market can be considered as a continuous gambling game [7]. Using this consideration there are several researches that model Kelly criterion to the stock market more theoretically [7], [9], [10]. Thus considering the game concept discussed in section 1.1.2 and the Kelly criterion discussed in section 1.1.3, Kelly Criterion model to the stock market will discuss in section 2.2 and section 2.3.

1.1.7 Order Book, Trades and Kelly Criterion

As stated in section 1.1.5.1, order book which concerns the utility of traders and trade data set which have all the trade details will be able to use as powerful sources when proposing dynamic computational model of Kelly criterion to the stock traders. However researchers have not concerned the order book and trades data when modelling the Kelly criterion to stock market. Thus proposing dynamic Kelly computational model for stock traders with order book and trade data, will give a better decision to stock trader.

1.2 Research Problem and Questions

1.2.1 Research Problem

When stock traders trading in the stock market they have to make decisions. But looking at the stock market they cannot obtain decisions at once because stock market behavior prediction is a very difficult task. There are various models that provides decisions to traders. However, what is the accuracy or the trustworthiness of that models? Have all the models proven their correctness with the actual behavior of stock market? Can a trader trust the results of the model?

1.2.2 Research Question

How to build a dynamic Kelly computational model for stock traders using existing Kelly model to stock market, order book and trades?

1.2.3 Aims and Objectives

Aim

Aim of the research is to develop a dynamic computational model Kelly model for stock traders using existing Kelly criterion model to stock market, trades and order book.

Objective

Investigating order book and trades with existing Kelly Criterion model to stock market, in order to build a dynamic Kelly computational model to stock traders.

1.3 Justification for the research

As described in above sections Kelly criterion is an optimal and well known strategy for a long time which can model to stock market. As in section 2.3, there are researches which model Kelly criterion to the stock market. However in those researches, researchers have only concerned the theoretical view of Kelly criterion when modeling it to stock market. They have not considered the dynamicity, practical situation of the stock market. Thus, they could not able to give clear decision to stock traders.

As described above order book is an important source of stock market because it contains traders' ideas, predictions and utility etc. Trades also very important data when make predictions and when analyzing both stock market and traders. Thus, a dynamic computational model for stock traders will be developed using these important resources with one of the existing Kelly criterion to stock market model.

1.4 Methodology

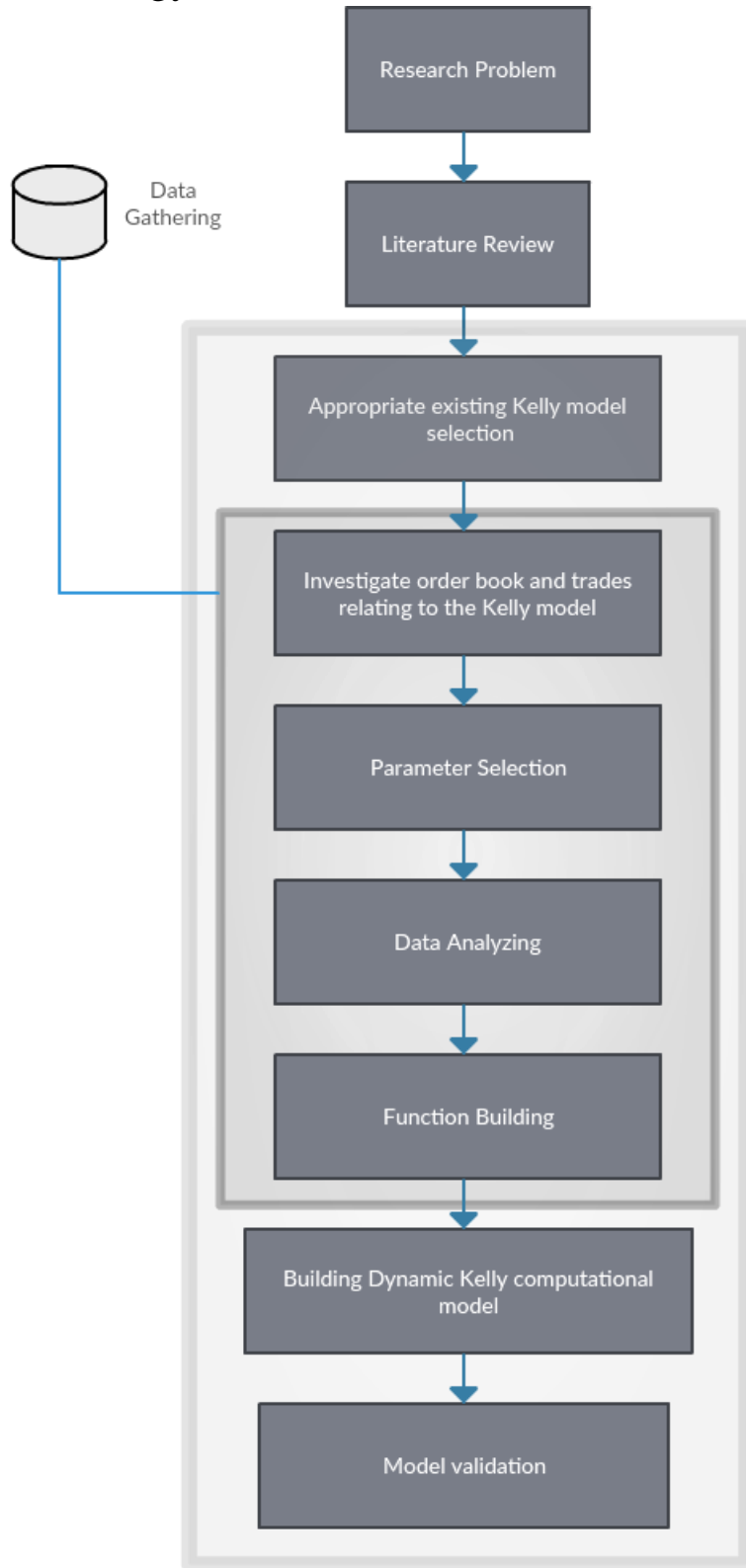


Figure 1.1 Research Methodology

As the initial step of the research methodology, order book and trades data gathering was conducted. Followed by the problem identification widely incorporation with the literature review. Also the existing approaches of modeling the Kelly criterion to stock market has been analyzed. Then an appropriate Kelly model to stock market was taken into consideration. However, that Kelly model has drawbacks such as not considering dynamicity of the stock market which mainly effect to the results of the model. Those are discussed in section 2.3. Thus, this research focuses on order book and trades to solve the problems by building improved dynamic version of the Kelly model for stock traders.

Thus, investigating the order book with trades has been done. That process involves suitable parameter selection, data analyzing in suitable way and function building of the model. With that, in next step the dynamic computational model of Kelly is developed with order book and trades, to give better decisions to stock traders. Finally, evaluation of the output of computational model is done to check whether order book and trades improve the Kelly model to give better decisions to stock traders by checking the accuracy of given decision from the model.

Here the high level picture of the methodology is described. Further details are described in section 2, mainly in section 3 and section 5.

1.5 Outline of the Dissertation

Firstly, in chapter 2 the existing works relevant to this research are considered. Section 2.1 focuses the existing work on understanding the nature of the order book and trades in the stock market from the trader's perspective. Section 2.2 concerns the works related to Kelly Criterion. Section 2.3 discusses about the Kelly Criterion models that has been modeled with stock market. Section 2.4 concerns overall conclusion of the literature review.

Chapter 3 provides the overall design of the dynamic Kelly model of stock traders which this research introduces. Section 3.1 begins with basic derivations of the computational model. Section 3.1.1 describes the methodology and equations which are introduced by investigating trades with existing Kelly model to stock market. Also section 3.1.2 describes the methodology and equations which are introduced by investigating order book with existing Kelly model to stock market.

Chapter 4 contains the details about the implementation process. Section 4.1 provides the trades and order book gathering implementation details. Section 4.2 contains the implantation details of order book and trades data processing in order to generate the values for introduced equations in section 3.1.1 and section 3.1.2. The implementation details of dynamic computational model building with integration is describes in section 4.3.

Chapter 5 the results and the evaluation process of the developed model. In section 5.1, the methodology and equations about how to use trades which was originally introduced in chapter 3 is illustrated with real trades. Also, in section 5.2 the methodology and equations about how to use order book which I originally introduced in chapter 3 is illustrated with real trades. Section 5.3 describes about the output from the model. Section 5.4 describes the way, how model suggest traders' decision. Section 5.5 describes the validation process using real stock market behavior and accuracy of the developed model. Section 5.6 explains about the accuracy of model without using the order book.

Finally, Chapter 6 comprises of the conclusions of the research. Section 6.2 contains the limitation of the research and section 6.3 contains the future work.

1.6 Definitions

- **Stock Trader**

A stock trader is a person or company involved in trading equity securities. Stock traders may be an agent, hedger, arbitrageur, speculator or stockbroker.

- **Utility**

Utility refers to the total satisfaction received from consuming a good or service.

- **Optimal wager fraction**

The amount of fraction of money trader can invest in order to get the highest expected return.

- **Treasury bills**

A short-term debt obligation backed by the treasury department of the U.S. government with a maturity of less than one year, sold in denominations of \$1,000 up to a maximum purchase of \$5 million.

- **Open price**

The opening price is the price at which a security first trades upon the opening of an exchange on a given trading day. Here the Middle East stock market is concerned. It opens at the 12.30 PM from Sri Lankan time. The price of the first trade for any listed stock is its daily opening price. A stock's opening price is an important marker for that day's trading activity.

- **Close price**

The closing price is the final price at which a security is traded on a given trading day.

- **High price**

The high price is the highest price at which a security is traded on a given trading day.

- **Low price**

The low price is the lowest price at which a security is traded on a given trading day.

- **Stock volume**

Stock volume refers the number of shares.

- **Excess return**

Excess returns are investment returns from a security or portfolio that exceed the riskless rate on a security generally perceived to be risk free, such as a certificate of deposit or a government-issued bond.

- **Limit order**

A limit order is an order to buy or sell a stock at a specific price or better. A limit order is not guaranteed to execute. A buy limit order can only be executed at the limit price or lower, and a sell limit order can only be executed at the limit price or higher

- **Market order**

A market order is a buy or sell order to be executed immediately at current market prices. As long as there are willing sellers and buyers, market orders are filled.

1.7 Delimitations of Scope

A dynamic Kelly computational model with order book and trades is concerned. Kelly criterion is modeled to the entire market data. Not for individual traders. Only Kelly Criterion is used to investigate the order book and trades dataset. Cost for placing an order and other expenses of trader are not concerned. Consider unique data of the Middle East stock market. Only one symbol is selected from the order book and trades for the investigation and analyzing purpose.

1.8 Conclusion

This chapter explains the foundations for the dissertation. Firstly, it introduced the back ground details and concepts which need to understand the research. Then, it introduced the research problem and research questions and hypotheses. Then the research was justified, definitions were presented, the methodology was briefly described and justified, the dissertation was outlined, and the limitations were given. On these foundations, the dissertation can proceed with a detailed description of the research.

Chapter 2 – Literature Review

In this section the existing works relevant to my research are considered. Section 2.1 focuses the existing work on understanding the nature of the order book and trades in the stock market from the trader's perspective. Section 2.2 focuses on the works related to Kelly Criterion. Section 2.3 discusses about the Kelly Criterion models that has been modeled with stock market. Section 2.4 concludes the literature.

2.1 Order book and trades data from trader's perspective

In the recent past, almost all major stock exchanges have introduced electronic limit order books. Thus, researchers have paid significant attention towards the order book information. T Foucault et al. [1] considered a model of price formation in a limited order market. However, they had utilized only a few publicly available order book rows. Using numerical examples, they have shown that the dynamics of the book are different in the cases where traders are homogeneous or heterogeneous. Furthermore, their model suggests that a limit order market will be quite illiquid when the proportion of impatient traders are large. Then, they have tried to map the trader's behavior with the order book data. A Ranaldo [3] also examined the information content of a limit order book in a purely order-driven market and analyzed how the state of the limit order book affects a trader's strategy. This study investigated a trader's decision to submit orders.

As explained above Foucault et al. [1] and A Ranaldo [3] have analyzed the behavior of traders with order book. Thus the fact that order book shows the trader's preference or behaviors can identify by considering above researches.

Order book effects to the traders when they are taking decisions on trading. Then order book can be used to get decisions about trading. M Hoschler et al. [4] proposed a mathematical model for a dynamic, continuous time limit order book and tested the model with real-world high-frequency order book data and showed that important properties are well reproduced by the model. They found the solution to determine the optimal time when the limit order should be converted to a market order if it has not been filled yet.

V Robu et al. [2] proposed a trading strategy, combining few existing trading strategies. Static order book imbalance (SOBI) strategy, volume average weighed prices (VWAP) strategy, trend following (TF) strategy, and reverse strategy have been used for the comparison and combination.

In particular, they had examined from an order book based strategy whether supporting the buy or sell decision from a price based strategy could lead to better experimental performance. Profit and sharp ratio values of all the strategies had being incorporated to compare the above strategies. The simple reverse strategy was the overall winner in terms of sharpe ratio, although their combined strategy using two from SOBI, trend following and reverse strategy is the overall winner in terms of raw profit. However, they could have incorporated better ways to combine and investigate strategies. Raw profit over the 15 days is not sufficient criteria since in real-life a trade-off between risk and return of a strategy must also be taken into account.

Carteaa et al. [6] have used volume of both buy and sell side of the order book of Nasdaq exchange to build a measure of volume imbalance in the limit order book. They illustrated that their measure is a good predictor of the sign of the next market order which bettor is buy or sell decision, and also helps to predict price changes immediately after the arrival of a market order.

Thus as mentioned above, researchers have used order book or its models to provide trader decisions with various strategies and to predict how trading strategies depend on the order book.

There are researches that have used trades data such as volume and prices, to provide some decisions to traders. Q.A.Al Radaideh et al. [19] have tried to help the investors in the stock market to decide the ideal timing for buying or selling stocks based on the knowledge extracted from the historical pricing of stocks. The decision obtained is based on decision tree classifier which is one of the data mining techniques. However, the model is not reliable due to the lack of accuracy inherent in the technique. Hsuan-Ming Feng et al. [20] have developed a fuzzy stock prediction system for price prediction using trade data. The developed evolutionary algorithm is capable of generating bettor good decisions in the short-term and middle-term stock markets.

The above discussed researches concerns with order book and trade data with trader's point of view. Most of them provide some decision to traders about stock trading. Also as mentioned above some of

the researches have pointed out that order book provides traders decisions or preferences while directly effecting the traders' strategies.

2.2 Kelly Criterion

In 1956 J.L. Kelly [12] originally proposed this Kelly Criterion as an asymptotically optimal strategy meaning that in the limit over many gambles, a Kelly bettor will grow wealthier than an otherwise identical non-Kelly bettor with probability 1.

The result “maximum exponential rate of growth of gambler’s capital is equal to the rate of transmission of information over the channel” has generalized to include the case of arbitrary odds. The channel of the theory might correspond to a real communication channel or simply to the totality of information available to investor. He showed that the same math C.E Shannon [13], [14] used in his theory of noisy communications channels applies to the gambler. According to the Kelly’s conclusion the gambler's optimal policy is to maximize the expected value of logarithm of wealth.

Researches have formed different concepts of the Kelly Criterion. Alina Beygelzimer et al. [11] and E.O Thorp et al. [8] have shown the fractional Kelly Criterion which says to invest a small fraction of wealth. Alina Beygelzimer et al. [11] has applied fractional Kelly Criterion to prediction markets as a risk reduction strategy.

2.3 Kelly Criterion model to stock market

E.O Thorp et al. [7] have described the Kelly Criterion with respect to binomial games, continuous gambling games and stock market. At the beginning, the need of an optimal strategy for favorable gambling games is shown, considering the coin tossing example which is explained in section 1.1.2 and Kelly Criterion is used to solve the problem.

The below equations are explained combining with the section 1.1.2. If the gambler bet for the coin tossing according to $B_i = fW_{i-1}$, where $0 \leq f \leq 1$, f is the percentage of the current wealth which

always uses for bets. f is a fixed fraction. Thus, this can be called as fixed fractional betting. Consider S and F are number of successes and failures respectively. After n trials ($S+F=n$) our capital is given by

$$W_n = W_0(1 + f)^S(1 - f)^F \quad (2)$$

Thus, with f in the interval $0 < f < 1$ the above described (in Section 1.1.2) gambler's ruin problem cannot ever occur. Then the exponential rate of increase per trial ($G(f) = \log\left[\frac{W_n}{W_0}\right]^{\frac{1}{n}}$) which is equivalent to expected value of logarithm of wealth ($E \log W_n$) is calculated with respect to the equation (2).

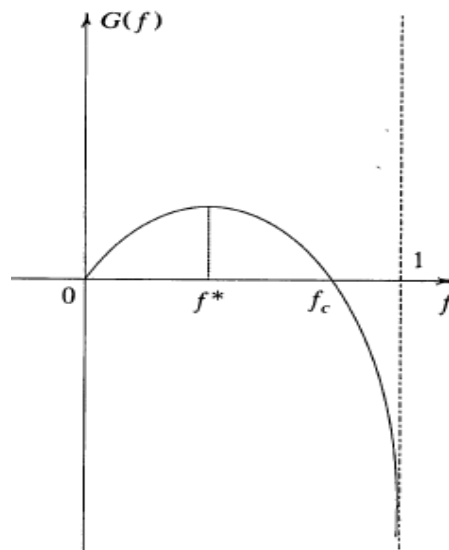


Figure 2.1 Fraction of current wealth verses exponential increase per trial

As in the figure 2.1 they have proven the behavior of $G(f)$ with respect to the f value. There is a unique maximum of $G(f)$ when f reaches to some value (f^*). Thus, they have proven that the maximization of expected value of logarithm of wealth or the Kelly Criterion to this binomial game.

Furthermore, considering the investment in the stock market as continuous gambling game with a positive one year expected return, they have worked to maximize the exponential rate of increase per trial ($G(f) = \log(1 + fs) dN(s)$) with a theoretical application to the stock market. Thus, a theoretical model of Kelly Criterion is developed with above ($G(f)$) concept and theoretically proven that, developed model can be used to stock market.

After using quasi normal distribution (figure 2.2) of the historical annual excess returns of U.S stock market as an input to the developed Kelly model, optimal wager fraction is given as the output of the computational model.

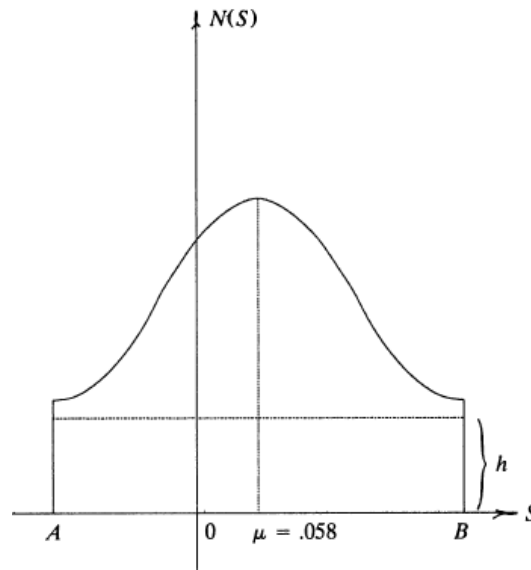


Figure 2.2 Altered normal distribution of annual returns

Calculated mean and standard deviation of the annual excess total returns of S&P blue chips for 59 years being used for the distribution. According to their experiment optimal wager fraction was 117%. Thus, according to their experiment, a long term investor should be invest fully plus borrow an additional of 17% from available resources in order to achieve maximal growth rate relatively to the T bills. However, there is no any practical validation for their result. So an investor or trader would not invest based on the result.

Also, they just retrieved one value per year for the distribution. Stock market behavior is dynamic and the expanded stock market volatility caused by program trading and internationalization of finance market. Thus, there is a high probability of changing the distribution used for the model time to time. In this scenario the applied model is not taken into consideration. The annual excess returns and just two cut off points for the distribution of the model are taken into consideration, to provide a theoretical decision to long term investors. Thus, the quasi probability distribution artificial distribution.

As mentioned above there are two cut off points marked as A and B (as in figure 2.2). A and B are finite values as the return of the stock market trading cannot reach to infinity. To get more data points, they have calculated

A as mean-3*standard deviation ($A = \mu - 3\sigma$)

and

B as mean+3*standard deviation ($B = \mu + 3\sigma$).

However, this leads to incorrect results from the model as the actual return range for a particular time period has not be taken into consideration by these two cut off points. Thus, this altered distribution is not valid if the actual behavior of the stock market concerns.

Modifying the model with dynamic behavior while optimizing on better information source to calculate the cut off points are the solutions for the above problems. Also a proper validation is required to prove the reliability and accuracy of the model.

As described in section 2.1 researchers have proven that order book model the decisions or the behavior of traders. Also trade data or the historical data can be used to predict many information such as stock prices, volumes and returns. Thus, using trade data and specially order book data for modifying the above Kelly model, will solve the above problems.

Both J hung [9] and N Khanna [10] have modified the equation (2) of E.O Thorp et al. [7] in a similar manner. A return rate of alternative investment is included as a variable of their model. An assumption is made to simplifying the outcomes of investing in the stock market to two separate equally likely outcomes. Either the investment is successful in which case the return per unit is a standard deviation above the mean, otherwise it is unsuccessful in which case the investment is a standard deviation below the mean. Likewise, they have made several assumptions for simplifying their model. Finally, a simple model of Kelly criterion is derived.

J hung [9] has used closed prices of the stock market and rate return of the other investment as 0 to apply the developed model. N Khanna [10] has used the excess return data used by E.O Thorp et al.

[7] and treasury bonds as alternative investment. However any validation has not been carried out with the model results.

J hung [9] and N Khanna [10] have just derived the Kelly Criterion to a simplified model. The assumptions of this model not suitable for the practical situation of the stock market because those assumptions do not practically compile the stock market behavior. N Khanna [10] has also mentioned that the assumptions that he made will be a problem with real world or with the stock market.

As a theoretical model of Kelly criterion to stock market, from the above 3 models, the theoretical model introduced by E.O. Thorp et al. [7] is comparatively reliable and accurate as the other two models do not meet the stock market requirements even theoretically..

2.4 Conclusion

According to the researches and description in section 2.1, order book is an important resource which models the stock traders' decisions and behaviors. Historical trade data can be used to obtain price, volume and return information of the stock market trading. Also both resources are crucial when predicting and analyzing stock market behavior.

According to the section 2.2 researchers have implemented original Kelly s' strategy in different ways to the betting systems, predictions market etc.

According to the section 2.3 E.O Thorp et al. [7] have introduced a theoretical model of Kelly criterion to the stock market. This research is conducted modifying this Kelly model to stock market. As described in section 2.3 there are numerous problems with that model when considering the actual stock market. Problem of dynamicity of the model, the problem of cut off points, problem of no proper validation and problem of not generating clear, realistic decision to stock investor are solved through this research by the Kelly criterion that model.

Chapter 3 – Design

According to the literature review, there are three possible approaches could model Kelly to stock market. As pointed out in section 2.3, the basic Kelly model to stock market of E.O Thorp et al. [7] is concerned. Also in section 2.3 and section 2.4, described the main problems of that Kelly model. This section illustrates the design used to solve those problems, by modifying the computational model of Kelly to generate better decision to traders with order book and trades.

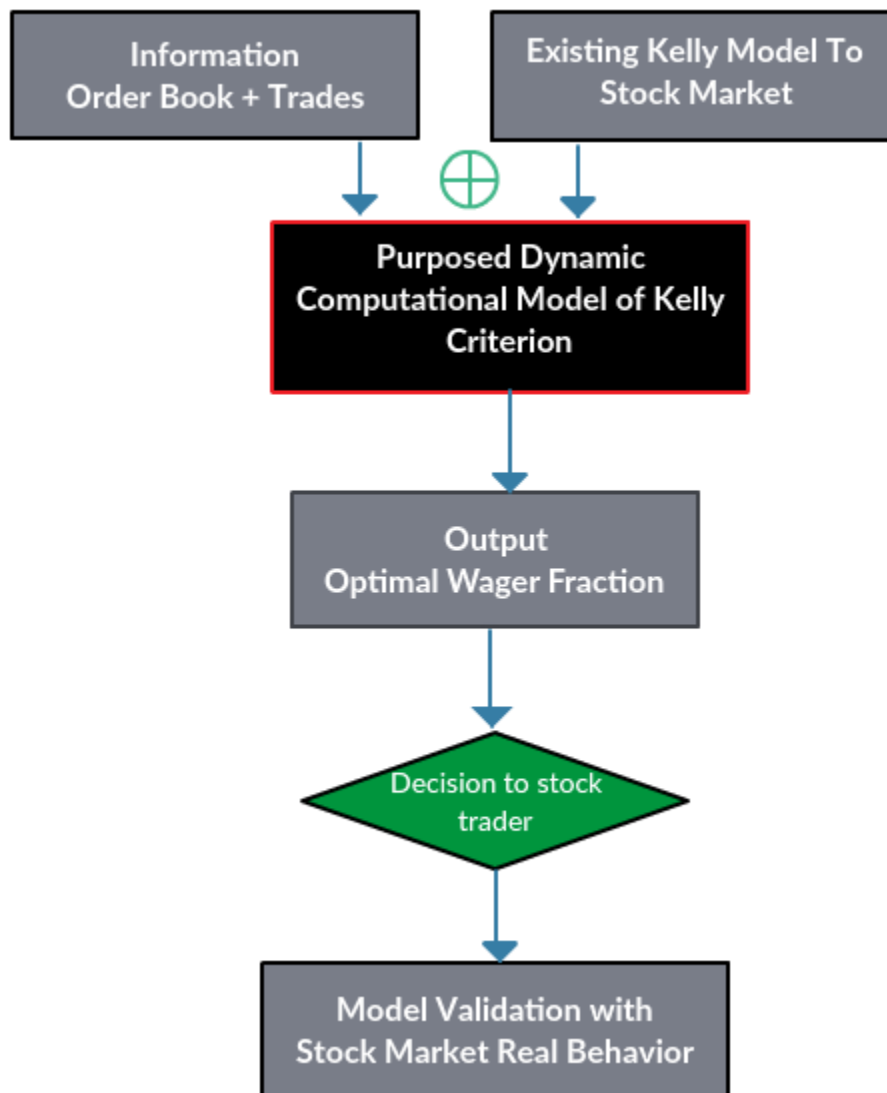


Figure 3.1 overall design of the research.

3.1 Kelly computational model to Stock Market

Key assumption: Whole market is considered as a Kelly bettor as entire market is modeled with Kelly Criterion.

This section explain the implementation method of the Kelly criterion model to stock market with modifications to the model of E.O Thorp et al. [7]. All the modifications are the contributions of this research.

Consider stock shares buying and selling situation of a particular trader.

Let,

W_0 : The trader's initial capital (money).

f : Fraction of money which trader wants or likes to spend to buy stock shares.

q : The outcome per unit of the money.

q can be positive or negative.

Let the trader purchases stock shares Which are sold later. Then, trader's total amount of money after selling those shares is given by the below equation.

$$W_1 = W_0 - W_0f + (W_0f + W_0fq) \quad (3)$$

W_0f : The amount of money trader spent to buy shares.

$W_0f + W_0fq$: The amount of money trader gain by selling the shares.

Above (3) equation can be consider as

$$W_1 = W_0(1 + fq) \quad (4)$$

Assuming trader uses same fraction of money (f) in each buying and selling situation. After n number of buying and selling traders wealth (amount of money he has) is given by (using equation (4))

$$W_n = W_0(1 + fq)^n \quad (5)$$

Kelly Criterion maximizes the expected value of logarithm of wealth (money). According to equation (5) $E \log W_n$ where E = expected value is given by below equation.

$$E \log W_n = E \log(W_0(1 + fq)^n) \quad (6)$$

W_0 should remove from the right hand because every traders' initial wealth is not required when modeling the stock market. Then the equation for $E \log\left(\frac{W_n}{W_0}\right)^{\frac{1}{n}}$ can be derived using equation (6).

$$E \log W_n = E (\log W_0 + \log(1 + fq)^n)$$

$$E \log W_n - E \log W_0 = n E \log(1 + fq)$$

$$E \log\left(\frac{W_n}{W_0}\right)^{\frac{1}{n}} = E \log(1 + fq) \quad (7)$$

Here $E \log\left(\frac{W_n}{W_0}\right)^{\frac{1}{n}}$ where,

$$E \log\left(\frac{W_n}{W_0}\right)^{\frac{1}{n}} = \frac{1}{n} E \log W_n - \frac{1}{n} E \log W_0$$

is equal to maximization of logarithm of wealth because for fixed n, $E \log\left(\frac{W_n}{W_0}\right)^{\frac{1}{n}} = E \log W_n$ because $\frac{1}{n} E \log W_0$ is a constant. Now W_0 does not need to know for the considering scenario.

Each investment or trading of stock market has finite number of outcomes. However, according to the E.O Thorpe et al. [7] it is mathematically convenient to approximate a finite distribution using a continuous model.

This is also a key assumption when modeling Kelly to stock market because in section 1.1.2 and in section 2.3 (equation 1 and equation 2) to prove theories and concepts, finite variables have been taken into consideration.

Consider the continuous distribution of q . Considering m number of q values of a particular scenario, below equation could be derived using equation (7). $P(q)$ is the probability of happening q

$$E \log(1 + fq) = \log(1 + fq_1)P(q_1) + \log(1 + fq_2)P(q_2) + \log(1 + fq_3)P(q_3) + \dots + \log(1 + fq_m)P(q_m)$$

Using above equation, below equation (8) can be derived.

$$E \log(1 + fq) = \int \log(1 + fq) P(q) dq$$

$$E \log\left(\frac{w_n}{w_0}\right)^{\frac{1}{n}} = \int \log(1 + fq) P(q) dq \quad (8)$$

3.1.1 Trade data Investigation with the Kelly model

For this research Kelly model was investigated in many ways with the trades data. Out of the many way the below stated manner was found to be the ideal method. Short term trading to the model of Kelly criterion to the stock market was introduced. Dynamic behavior of stock market also comes into play with this introduction. This is an original contribution of this research.

Consider a day trading scenario of a trader. Day trading is an active-trading style. Day trading, as its name implies, is the method of buying and selling stocks within the same day. Positions are closed out within the same day they are taken, and no position is held overnight.

As explained at the beginning of the explanation (in section 3.2) q is outcome per unit of money.

In order to calculate q , a variable called r was introduced which can calculate from the parameters of trades and considering the trader buys stocks at opening price of a day and sell it at close price of the day assuming trader is a day trader

$$r = \frac{\text{close price} - \text{open price}}{\text{open price}} \quad (9)$$

Assuming a day trading scenario in the stock market, r is the return or outcome per unit of money from the trade per day.

As mentioned in section 1.1.3 Kelly Criterion gives its decision using both risky and risk free investment because Kelly concept has introduced both wealth maximization and risk reduction. Risk reduction is illustrated with the fraction of money which the trader should spend for the trade (f). It says if $0 < f < 1$, invest f fraction of money in a risky investment and at the same time invest $1 - f$ fraction of the money in a risk free investment. Thus, in this model, stock market investment is used as risky investment and fixed investment (certificate of deposit) in Saudi Arabiya is used as relatively risk free investment.

Then, these two investments should consider with equation 8 in order to include the risk reduction concept of Kelly Criterion to it. q value or the outcome per unit of money is the variable which can include both investments returns like excess returns explained in section 1.6.

Thus, the introduction of the below equation 10 of q as a comparative measure.

$$q = r - \text{unit return per day from the deposit}$$

or

$$q = \frac{(\text{close price} - \text{open price})}{\text{open price}} - \text{unit return per day from the deposit} \quad (10)$$

Thus, q calculates how much excess unit return can get from the riskier stock market trading comparatively the deposit. It can be positive or negative.

By substituting q value as in equation 10 to the equation 8, as mentioned above, the output f value can retrieve as a comparative value between stock market investment and deposit. Thus, risk reduction happens using the q value which is introduced as in equation 10.

3.1.1.1 Behavior of f value

This model indicate different ideas with the output f value.

Consider the maximum of f output value as f_{max} .

- f_{max} can be in between 0 and 1 ($0 < f_{max} < 1$).

Idea is invest f fraction of money in a risky investment and at the same time invest $1 - f$ fraction of the money in a risk free investment.

- f_{max} can more than or equal to 1 ($f_{max} \geq 1$).

Idea is trader can invest total amount of money in the stock market or in another word, trader can buy shares with the total amount of money.

- f_{max} can be less than 0 ($f_{max} \leq 0$).

Idea is trader does not invest money on stocks, in other word it is better to sell shares and invest total amount of money in deposits

Now, a continuous distribution of q can consider based on the equation 8.

For that, q values of consecutive 30 days of a particular company in stock market has being considered. In one of 4 the cases, q value set behaved more likely the normal distribution (refer section 5.1 for more details about this). Thus, the normal distribution could be used as the continuous distribution for the equation 8.

3.1.2 Order Book Investigation with the Kelly model

This section contains the findings after the investigation of the order book with Kelly Criterion. Introducing the order book to Kelly criterion to stock market is also a main contribution of this research.

As explained above (in section 3.1.1) the normal distribution is used as the continuous distribution for equation 8. Then, the probability density function (PDF) of normal distribution where,

$$PDF = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(q-\mu)^2}{2\sigma^2}}. \quad (11)$$

q – Unit return per trade (Random variable)

μ – Mean of the data

σ – Standard deviation of data

This is the behavior of Probability density function of a normal distribution. Thus, q can behave in any way like in below figure 3.3.

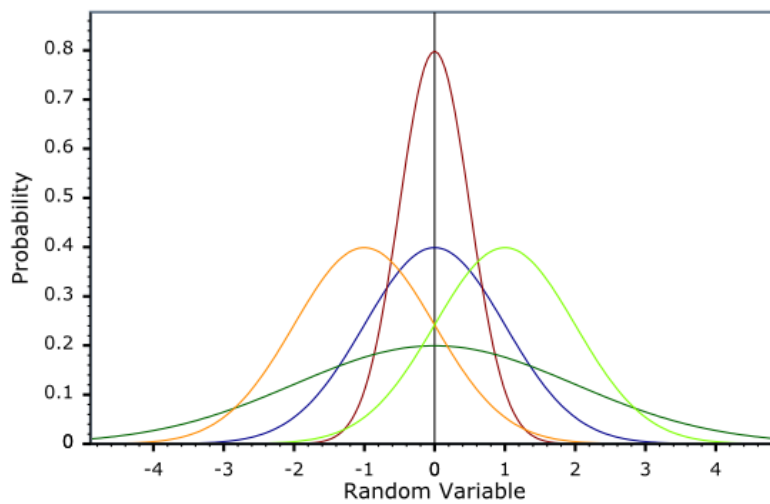


Figure 3.2 PDF of normal distribution

As in figure 3.2 normal distribution goes to both negative and positive infinity. As in figure 3.3 the normal fitting for q values also shows a similar variation

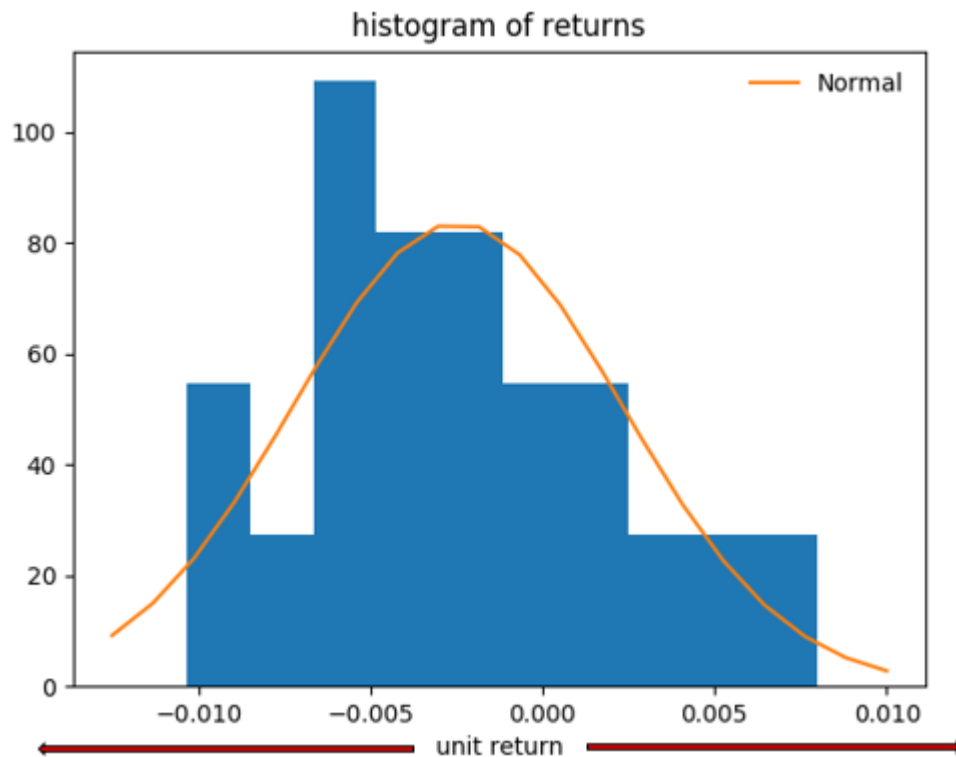


Figure 3.3 Normal distribution of unit returns

Now, this PDF can be substituted into the equation 8. However, the problem with the substitution is as in the figure 3.3, normal distribution allows for unboundedly large unit returns increments or declines. However, in a real stock market, a trader cannot gain infinite returns. Thus, normal distribution cannot be plugged into integral equation (Equation 8) directly.

A change for the normal distribution is required.

As described in section 3.2.1, E.O thorp al. [7] (in the existing model) uses two cut off point for the distribution as A and B where,

$$(A = \mu - 3\sigma)$$

And

$$(B = \mu + 3\sigma).$$

Also as explained there, the main problem with the model is without an acceptable reason only to get more data points, they have calculated A and B. As stated above they have considered only the theoretical validity of the model.

Furthermore, they have not considered the dynamicity of the stock market. They have got 59 data points of past 59 years and have used all the points to say about present and future. This scenario is explained in section 2.3.

Thus, In order to eliminate these draw backs order book is introduced to Kelly criterion stock model as a source to get better information for the two cutoff points.

Reasons to concern about order book:

- i. Order book model the behaviors and some decisions of stock traders. (from the literature review (section 2.1 showed this))
- ii. Kelly is more than a strategy. It also model the traders' behavior. (described in section 1.1.3)
- iii. Some orders which are entered to the order book execute during the day and some orders do not execute during the day. Thus, it gives the stock price and volume measure for today and for a future day (tomorrow). Then, order book includes some prediction for future.

Symbol: 1150

Splits	Bid Quantity	Bid Price	Ask Price	Ask Quantity	Splits
6	1219541	14.380	14.400	101700	2
5	1344050	14.360	14.420	384301	9
3	190050	14.340	14.440	1092965	8
5	349152	14.320	14.460	87697	9
13	242985	14.300	14.480	255629	17
0	0	0.000	0.000	0	0

Symbol: 1150

Splits	Bid Quantity	Bid Price	Ask Price	Ask Quantity	Splits
5	1618526	14.340	14.360	137419	3
7	1099202	14.320	14.380	1789779	4
13	242985	14.300	14.400	107340	5
10	211585	14.280	14.420	255184	8
10	551619	14.260	14.440	1092465	7
0	0	0.000	0.000	0	0

Figure 3.4 An order book update during one second

Thus, as I mentioned above traders add orders to order book according to their decisions. As I stated in section 1.1.5.1 it includes utility of traders.

Let's say particular trader's decision is to buy shares. Trader adds share price and volume as trader thinks and as matching with the whole market. Then that order is added to the order book as a row of the order book if the entered price is not equal to current market price (Order book have all the limit

orders). This situation is same for a trader who is looking to sell the stocks and sell traders' order is added to the sell side of the order book as a row.

Then, by investing order book, a good approximation about what are the stock prices and volume on today and tomorrow because whole order book is better source that represent the market (buyers and sellers). Specially, order book contains the orders which do not execute once (Limit orders), a better approximation can be retrieve about future prices and volumes.

Thus, after obtaining all the buying and selling prices of the order book and considering the fact that, today's order book prices give an ideal approximation prediction about tomorrow prices, the two cut off points are calculated in order to get better judgment for the Kelly's' behavior.

12:32:08						
3	866164	14.78	14.8	150603	2	
2	210000	14.76	14.82	238095	6	
5	324430	14.74	14.84	99701	9	
2	150200	14.72	14.86	96973	16	
10	149060	14.7	14.88	94248	16	
12:32:11						
4	866193	14.78	14.8	150603	2	
2	210000	14.76	14.82	238095	6	
5	324430	14.74	14.84	99701	9	
2	150200	14.72	14.86	96973	16	
10	149060	14.7	14.88	94248	16	
12:32:14						
4	866193	14.78	14.8	147173	2	
2	210000	14.76	14.82	238095	6	
5	324430	14.74	14.84	99701	9	
2	150200	14.72	14.86	96973	16	
10	149060	14.7	14.88	94248	16	

Figure 3.5 Analyzed order book to calculate returns

Red color column – Buy side.

Green Color column – Sell side.

Considering all the price values in both buy side and sell side as in figure 3.5. Then I calculate B_{min} , B_{max} , S_{min} and S_{max} variables where,

B_{min} = minimum price value of order book buy side of a particular day

B_{max} = maximum price value of order book buy side of a particular day

S_{min} = minimum price value of order book sell side of a particular day

S_{max} = maximum price value of order book sell side of a particular day.

Now, I calculate the maximum and minimum unit returns relatively to the risk free deposit where,

$$R_{min} = \frac{(S_{min}-B_{max})}{B_{max}} - \text{unit return per day from the deposit} \quad (12)$$

$$R_{max} = \frac{(S_{max}-B_{min})}{B_{min}} - \text{unit return per day from the deposit} \quad (13)$$

R_{min} – Minimum unit returns relatively to the risk free deposit.

R_{max} – Maximum unit returns relatively to the risk free deposit.

Now I use these R_{min} and R_{max} values as two cut off points to the continuous normal distribution of q .

Figure 3.6 shows the two cut off points and distribution is cut from these two points.

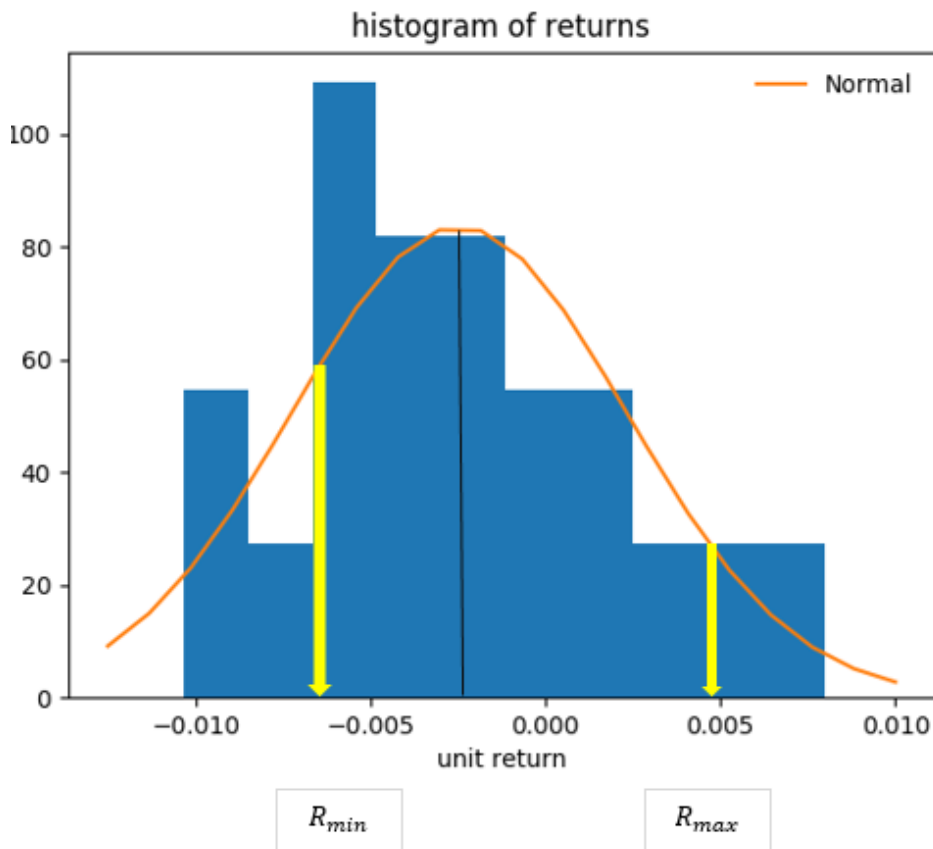


Figure 3.6 Cut off points of distribution

Figure 3.7 shows how it behave after cutting from these two points.

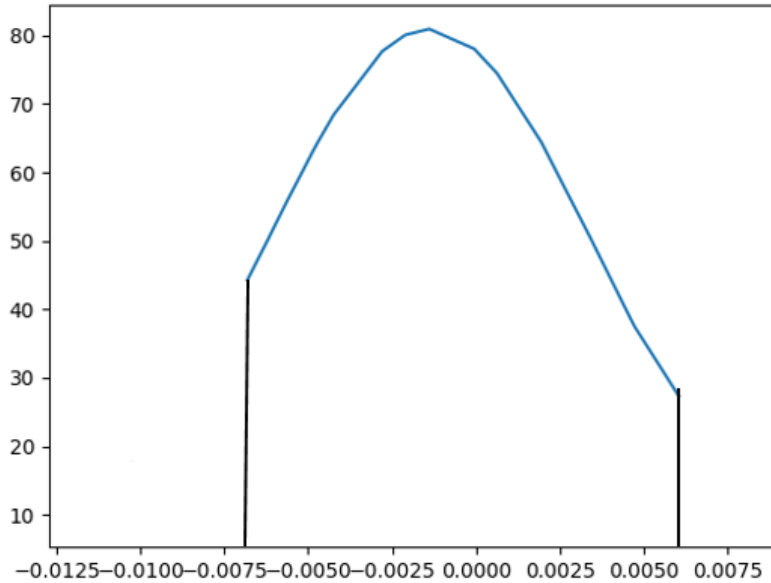


Figure 3.7 Distribution cutting

Now, another problem arises with this distribution cutting which if we cut from these two points area under the distribution is not equal to one. However, area under the distribution should equal to 1 and simultaneously we want the same standard deviation σ as in the original distribution. Thus, as E.O Thorp et al. [7] introduced, in this research also considered h and α values to solve these problems.

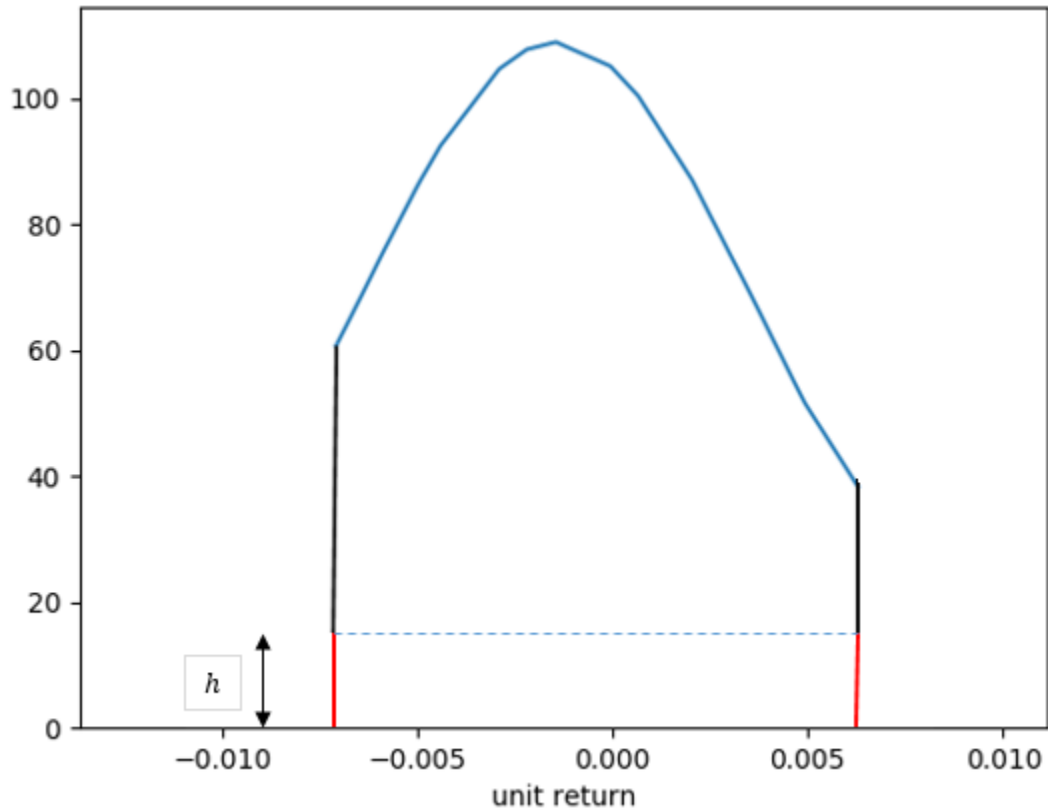


Figure 3.8 Adding h value to the distribution

This (Figure 3.8) is the modified distribution. As described above, using the theory that integration of distribution should equal to 1 where,

$$\int_{R_{min}}^{R_{max}} (h + F(q)) d(q) = 1 \quad (14)$$

$$F(q) = \frac{1}{\sqrt{2\pi\alpha^2}} e^{-\frac{(q-\mu)^2}{2\alpha^2}}$$

and after getting the below equation, considering the relation between standard deviation and mean

$$\int_{R_{min}}^{R_{max}} q^2 N(q) dq = \sigma^2 + \mu^2 \quad (15)$$

calculation for h and α values is done. Further described about this in section 4.3

The final modified equation of the distribution is $N(q)$ where,

$$N(q) = \left\{ \begin{array}{ll} h + \frac{1}{\sqrt{2\pi\alpha^2}} e^{-\frac{(q-\mu)^2}{2\alpha^2}}, & R_{min} \leq q \leq R_{max} \\ 0 & , \quad q < R_{min} \\ 0 & , \quad q > R_{max} \end{array} \right\}. \quad (16)$$

Now, substitution of $N(q)$ to the equation 8 as below.

$$E \log\left(\frac{W_n}{W_0}\right)^{\frac{1}{n}} = \int_{R_{min}}^{R_{max}} \log(1 + fq) dN(q) \quad (17)$$

This is the overall design of the model. As explained in section 3.1.1 day trading scenario is concerned for this model. Also as stated above these R_{min} and R_{max} values are calculated considering order book for a one day. Thus, obtaining the distribution of q until up to day (say until x day) and calculate R_{min} and R_{max} values considering order book of that last day (x day). Then, gives the trader decision for next day ($x+1$ day).

Then with the function 16, $E \log\left(\frac{W_n}{W_0}\right)^{\frac{1}{n}}$ or the exponential rate of increase per trade with the fraction of money which the trader should spend for the trade (f) is calculated. Finally, considering the f_{max} value and assuming an anti-Kelly bettor. This model yields a decision to the trader.

Further described about this in chapter 5.

Chapter 4 – Implementation

This chapter contains the details about the implementation process. Section 4.1 provides the trades and order book gathering implementation details. Section 4.2 contains the implantation details of order book and trades data processing in order to generate the values for introduced equations in section 3.1.1 and section 3.1.2. The implementation details of dynamic computational model building with integration is describes in section 4.3.

4.1 Order book and Trades data Gathering

Order book is dynamically updating source even in than a second. To gather order book, a connection was setup to a data feed server of Middle East stock exchange. Scheduler in an Ubuntu server was run at 12.30 PM (starting time of the Middle East stock market) and stopped at 5.30 PM (Ending time of the Middle East stock market) automatically from Sunday to Friday.

Code for gathering data was written using java [15] programming. Threads are used for the code. In Figure 4.1 is illustrate the view of a log file of an order book.

```
22/03/17 13:51:18|1=1004|101=T|102=4230|103=0|104=1|108=1|109=11|300=1;301=0;302=24.15;303=1650|300=1;
22/03/17 13:51:18|1=1004|101=T|102=1810|103=0|104=1|108=2|109=6|300=1;301=0;302=30.2;303=13683;304=11|
22/03/17 13:51:18|1=1004|101=T|102=1810|103=0|104=1|108=1|109=11|300=1;301=4;302=30.2;303=2000|300=1;3
22/03/17 13:51:18|1=1004|101=T|102=1150|103=0|104=1|108=2|109=6|300=2;301=2;302=14.75;303=1171905;304=
22/03/17 13:51:18|1=1004|101=T|102=3030|103=0|104=1|108=2|109=6|300=1;301=0;302=59.5;303=7099;304=11|3
22/03/17 13:51:19|1=1004|101=T|102=3030|103=0|104=1|108=1|109=11|300=1;301=0;302=59.5;303=318|300=2;30
22/03/17 13:51:19|1=1004|101=T|102=3002|103=0|104=1|108=2|109=6|300=1;301=0;302=10.75;303=44035;304=14
22/03/17 13:51:19|1=1004|101=T|102=6020|103=0|104=1|108=2|109=6|300=2;301=0;302=14.05;303=700;304=1|30
22/03/17 13:51:19|1=1004|101=T|102=6020|103=0|104=1|108=1|109=11|300=2;301=0;302=14.05;303=700|300=1;3
22/03/17 13:51:19|1=1004|101=T|102=8310|103=0|104=1|108=2|109=6|300=2;301=0;302=18.45;303=2352;304=2|3
```

Figure 4.1 Section of an order book log file

For the research, trades was used data which gives information for a particular trading day. Firstly trade data sets was used contains all the trades of a particular day. Although, that data set was investigated, could not build suitable way to find a distribution for the Kelly model. Then, a data set contains calculated stock prices of Middle East stock exchange such as open prices, high prices, low prices and close was used which is defined in section 1.6. A JSON formatted data files have been gathered.

```
[1490486400,15.000,15.050,14.850,14.880,693377322],
[1490572800,14.900,14.950,14.800,14.830,693377323],
[1490659200,14.850,14.950,14.850,14.880,693377324],
[1490745600,14.900,14.950,14.850,14.900,693377325],
[1490832000,14.900,15.100,14.850,15.020,693377326],
[1491091200,15.100,15.150,15.000,15.010,693377327],
```

Figure 4.2 Section of a trades data file

4.2 Order book and trades data processing

Using the illustrated java [15] code in listing 1 in appendix A order book is preprocessed in order to do calculations. Then using the equation 11 and equation 12 which was introduced in this research is used to calculate the R_{min} and R_{max} values. Figure 4.3 illustrate the processed order book data.

17:29:32	60	1570739	14.45	14.5	574980	10
	45	839668	14.4	14.55	749388	19
	21	262034	14.35	14.6	685309	37
	31	237574	14.3	14.65	2634943	88
	17	159204	14.25	14.7	2416097	138
17:29:36	60	1570739	14.45	14.5	574980	10
	44	837175	14.4	14.55	749388	19
	21	262034	14.35	14.6	685309	37
	31	237574	14.3	14.65	2634943	88
	17	159204	14.25	14.7	2416097	138

Figure 4.3 Processed order book

Also, using the java [15] code in listing 2 in Appendix A, trades are processed. Then using the equation 10 which was used when introducing short term trading (day trading) to Kelly criterion for stock market, q values were calculated. Figure 4.4 illustrates the processed order book data.

1.49E+09	14.6	14.7	14.55	14.61
1.49E+09	14.6	14.65	14.55	14.57
1.49E+09	14.55	14.6	14.4	14.48
1.49E+09	14.55	14.6	14.35	14.38
1.49E+09	14.4	14.45	14.3	14.32
1.49E+09	14.35	14.4	14.2	14.29
1.49E+09	14.35	14.55	14.3	14.46

Figure 4.4 Processed trades

After calculating μ and σ in normal distribution of q values and R_{min} and R_{max} values of order book, these values were feed to the implemented model using python [17].

4.3 Integration implementation process for output

This section describes the python [17] programming implementation for the integrations of model. Python [17] programming was used for the integration implementations because its’ “scipy” library is an ideal one for integration calculations and optimization (when finding an optimal solution for f value). Integration implementations with MATLAB [18] was carried out. However, optimization could not done with MATLAB [18].

As mentioned in section 3.1.2, normal distribution is altered with α and h values. Thus α and h values should find according to the given 4 input variables. These values are calculated using the equation 14, equation 15 and equation 16 which is mentioned in section 3.1.2.

$$\int_{R_{min}}^{R_{max}} (h + F(q)) d(q) = 1 \quad (14)$$

$$F(q) = \frac{1}{\sqrt{2\pi\alpha^2}} e^{-\frac{(q-\mu)^2}{2\alpha^2}}$$

$$\int_{R_{min}}^{R_{max}} q^2 N(q) dq = \sigma^2 + \mu^2 \quad (15)$$

From equation 13, h value can calculate like below.

$$(R_{max} - R_{min})h + \int_{R_{min}}^{R_{max}} F(q)d(q) = 1$$

$$h = \frac{1 - \int_{R_{min}}^{R_{max}} F(q)d(q)}{(R_{max} - R_{min})} \quad (18)$$

Also, substituting the $N(q)$ from equation 16 to the equation 15, h value can calculate like below.

$$h = \{(\sigma^2 + \mu^2) - \int_{R_{min}}^{R_{max}} q^2 F(q)d(q)\} \frac{1}{\int_{R_{min}}^{R_{max}} q^2 dq} \quad (19)$$

Now equaling the equation (18) and equation (19), α value can be find as below.

$$(R_{max} - R_{min})\{(\sigma^2 + \mu^2) - (R_{max} - R_{min}) \int_{R_{min}}^{R_{max}} q^2 F(q) d(q) - \int_{R_{min}}^{R_{max}} q^2 d(q) - \int_{R_{min}}^{R_{max}} q^2 d(q) \int_{R_{min}}^{R_{max}} F(q) d(q)\} = 0 \quad (20)$$

The explanation of the python [17] implementation to find the α value using above equation.

Function which calculate q^2 .

1. **def** integrand1(q):
2. **return** q**2

Function which calculate $F(q)$. 'exp' in numpy library is used to calculate exponential value and 'sqrt' in numpy library used to calculate square root of a value.

1. **def** integrand2(q,alp):
2. **return** ((np.exp(-1*((q-m)**2)/(2*(alp**2))))/(np.sqrt(2*p*(alp**2))))

Function which calculate $q^2 * F(q)$.

1. **def** integrand3(q,alp):
2. **return** ((q**2)*
3. ((np.exp(-1*((q-m)**2)/(2*(alp**2))))/(np.sqrt(2*p*(alp**2))))

Using above 3 functions, I got all the integrations in equation (20). For calculating the integrations I have used 'quad' function in python. In below code segment R_{min} is referred as 'a' and R_{max} is referred as 'b'.

1. **def** func(alp):
2. [y1,error1]=quad(integrand1, a, b)
3. [y2,error2]=quad(integrand2, a, b, args=(alp,))
4. [y3,error3]=quad(integrand3, a, b, args=(alp,))
5. **return** y1-(y1*y2)-((b-a)*((m**2) + (v**2)))+(b-a)*y3

'scipy.optimize.fsolve' in python [17] is used to find the roots (values of α) of equation (20) with the above calculated integrations.


```

1. def alp_finder():
2.     sol = fsolve(func,0.2)
3.     print "Solution for alpha : ", sol[0]
4.     return sol[0]

```

Now α value is calculated. Substituting this α value to equation (18), h value is calculated using python programming as below.

```

1. def h_finder(alp):
2.     def integrand(q):
3.         return ((np.exp(-1*((q-m)**2)/(2*(alp**2))))/(np.sqrt(2*p*(alp**2))))
4.     [val,error]=quad(integrand, a, b)
5.     h=(1-val)/(b-a)
6.     return h

```

Finally, the overall model which is given by equation (17) is built. The maximization of the equation 16 should be done here because the optimal f value need to determine. The 'python.scipy.optimize.fmin' was used which minimizes a function using the downhill simplex algorithm. To obtain the maximum f value, the equation (17) was negated in the implementation. The python code segment is illustrated below.

```

1. def model_implement(alp,h):
2.
3.     def integrand(q,f):
4.         return ((np.log(1+f*q))*(h+((np.exp(-1*((q-
5.             m)**2)/(2*(alp**2))))/(np.sqrt(2*p*(alp**2))))))
6.     def g(f):
7.         return quad(integrand, a, b, args=(f,))[0]
8.
9.     def t(f):
10.        return -1*quad(integrand, a, b, args=(f,))[0]
11.
12.     x = np.arange(-20, 100, 0.01)

```

```
12. vec_expint = np.vectorize(g)
13. plt.plot(x,vec_expint(x))
14.
15. d=fmin(t,0)
16. y=vec_expint(d)
```

Then, the implementation for the output graph of $E \log\left(\frac{w_n}{w_0}\right)^{\frac{1}{n}}$ or the exponential rate of increase per trade with the fraction of money which the trader should spend for the trade (f) is plotted with the maximum f value. The complete code for this process is in appendix A listing 3. Also the implementation for the model without order book (without calculating cutoff points from order book) is in the Appendix A listing 4.

Chapter 5 – Result and Evaluation

This chapter explains the results and the evaluation process of the developed model. In section 5.1, the methodology and equations about how to use trades which was originally introduced in chapter 3 is illustrated with real trades. Also, in section 5.2 the methodology and equations about how to use order book which was originally introduced in chapter 3 is illustrated with real trades. Section 5.3 describes about the output from the model. Section 5.4 describes the way, how model suggest traders' decision. Section 5.5 describes the validation process using real stock market behavior and accuracy of the developed model. Finally, section 5.6 explains about the accuracy of model without using the order book.

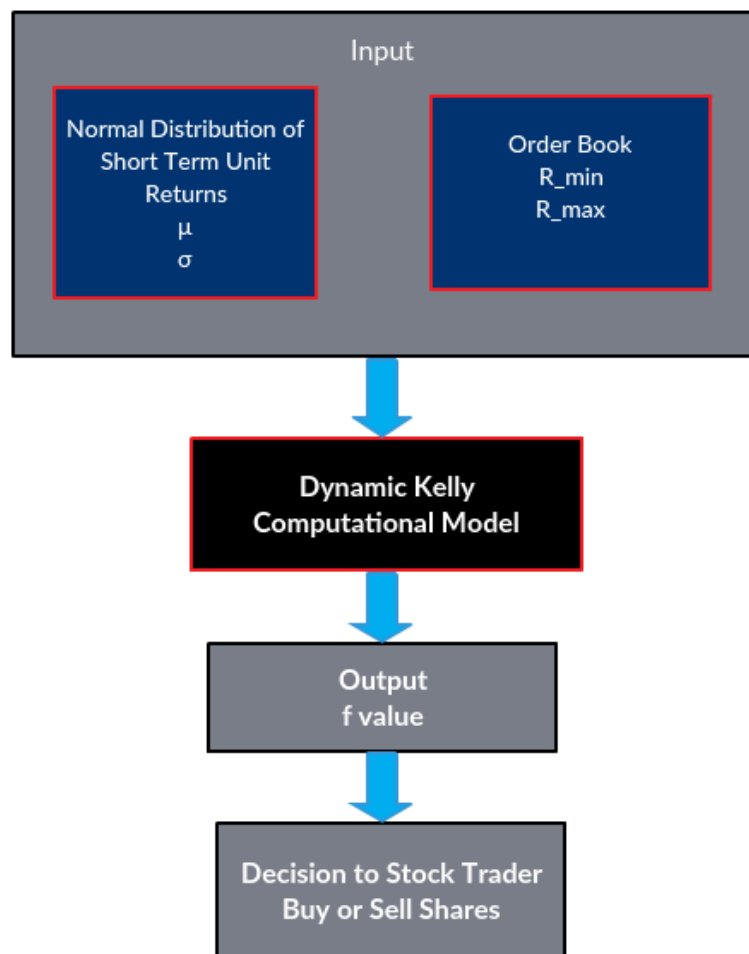


Figure 5.1 Implemented model

5.1 Trades data towards results

Section 3.1.1 describes about q value which calculates how much excess unit return could generate from the riskier stock market trading comparatively the deposit and how it is calculated in this model. Furthermore, as explained in section 3, final model receives altered normal distributions which was originally introduced using short term returns, as an input to the model. Section 5.1 explains how the behavior of q value is more likely the normal distribution.

Figure 5.2 shows part of a data set and I use it to show the calculation of q value.

Time	Open	High	Low	Close	r	q
1.494E+09	14.05	14.2	14.05	14.06	0.000711744	0.000656949
1.494E+09	14.05	14.2	14.05	14.14	0.006405694	0.006350899
1.494E+09	14.15	14.35	14.1	14.29	0.009893993	0.009839198
1.494E+09	14.3	14.4	14.25	14.37	0.004895105	0.00484031
1.494E+09	14.35	14.4	14.25	14.33	-0.001393728	-0.00144852
1.495E+09	14.35	14.35	14.25	14.32	-0.002090592	-0.00214539
1.495E+09	14.3	14.35	14.2	14.23	-0.004895105	-0.0049499
1.495E+09	14.25	14.5	14.2	14.42	0.011929825	0.01187503
1.495E+09	14.4	14.65	14.35	14.58	0.0125	0.012445206
1.495E+09	14.55	14.7	14.55	14.59	0.002749141	0.002694346
1.495E+09	14.65	14.9	14.65	14.72	0.004778157	0.004723362
1.495E+09	14.8	14.8	14.6	14.68	-0.008108108	-0.0081629
1.495E+09	14.6	14.75	14.4	14.74	0.009589041	0.009534247
1.496E+09	14.7	14.75	14.45	14.53	-0.011564626	-0.01161942
1.496E+09	14.55	14.65	14.15	14.19	-0.024742268	-0.02479706
1.496E+09	14.2	14.25	14.05	14.18	-0.001408451	-0.00146325
1.496E+09	14.2	14.25	14.05	14.07	-0.00915493	-0.00920972
1.496E+09	14.1	14.3	14.1	14.2	0.007092199	0.007037404
1.496E+09	14.2	14.3	14.2	14.21	0.000704225	0.000649431
1.496E+09	14.2	14.25	14.1	14.12	-0.005633803	-0.0056886
1.497E+09	14.1	14.48	14.1	14.4	0.021276596	0.021221801

Figure 5.2 An analyzed trade data set to calculate q values

As explained in section 3.1.1 I assuming a day trading scenario and r value is calculated as in equation 9.

Consider the highlighted row.

$$r = \frac{(\text{close price} - \text{open price})}{\text{open price}}$$

As explained in the design, considering the trader buys a stock to \$14.3 (opening price) and sell it at \$14.37 (close price). Then unit return,

$$r = \frac{14.37 - 14.3}{14.3}$$

$$r = 0.0048951.$$

Now calculation of q value can be done with equation 10.

$$q = \frac{(\text{close price} - \text{open price})}{\text{open price}} - \text{unit return per day from the deposit}$$

Annual fixed deposit rate in Saudi Arabia is 2% in 2017. Thus, unit return per day from the fixed deposit is equal to \$ 0.0000547945.

Thus,

$$q = 0.0048951 - 0.0000547945$$

$$q = 0.00484031$$

Like above q values for all days are calculated.

Now consider the distributions of these data sets.

Consider, the model provides a trader decision today. For that q values until yesterday could be obtained. The q values for 30 days up till yesterday are obtained as this model considered a short term trader (day trader). However, 30 day q values always do not behave like normal. However, the normality of q values happens in one of the following cases.

- i. q values of consecutive 30 days (4 weeks)
- ii. q values of consecutive 21 days (3 weeks)
- iii. q values of consecutive 14 days (2 weeks)
- iv. q values of consecutive 7 days (1 weeks)

Hypothesis testing with the P-value approach was used to check whether the distributions are normal. Minitab [21] programming was used for this.

The hypothesis testing for normality with p-values.

Null hypothesis

H0: Data are normally distributed

Alternative hypothesis

H1: Data are not normally distributed

Alpha=0.05

If p-value<alpha:

Reject H0

For all the testing I checked the normality of the data set because it is directly affect to the results of the model and it is include in the developed model.

Below I have included p-value testing and the normal distribution graphs with histograms of some used data for the above stated cases.

i. q values of consecutive 30 days (4 weeks)

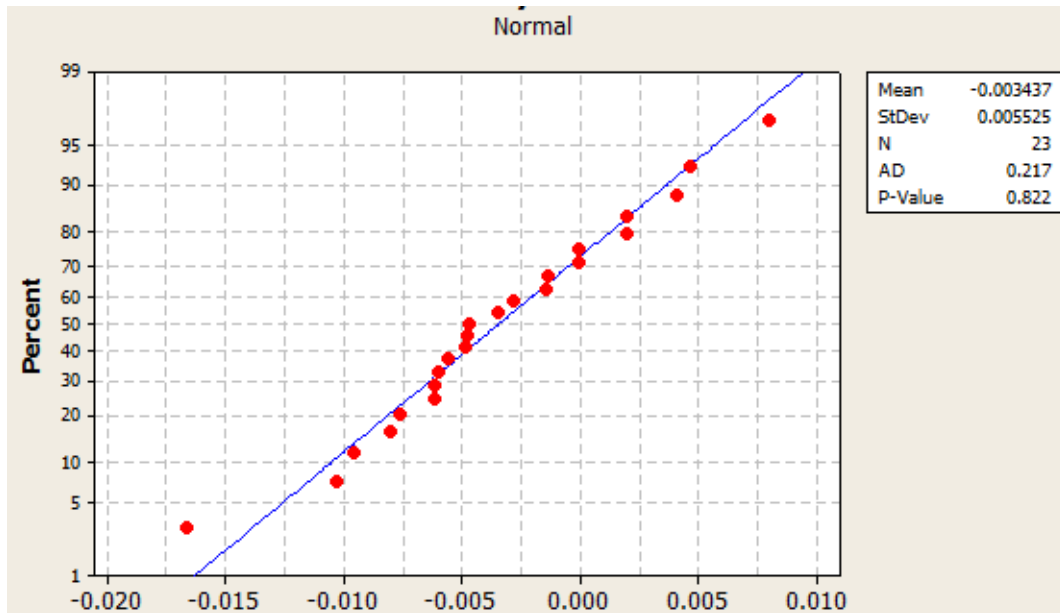


Figure 5.3 sample p-value check for 4 weeks

Since $p\text{-value} > 0.05$, There is not enough evidence to reject H_0 at 5% level of significance.

Data are normally distributed.

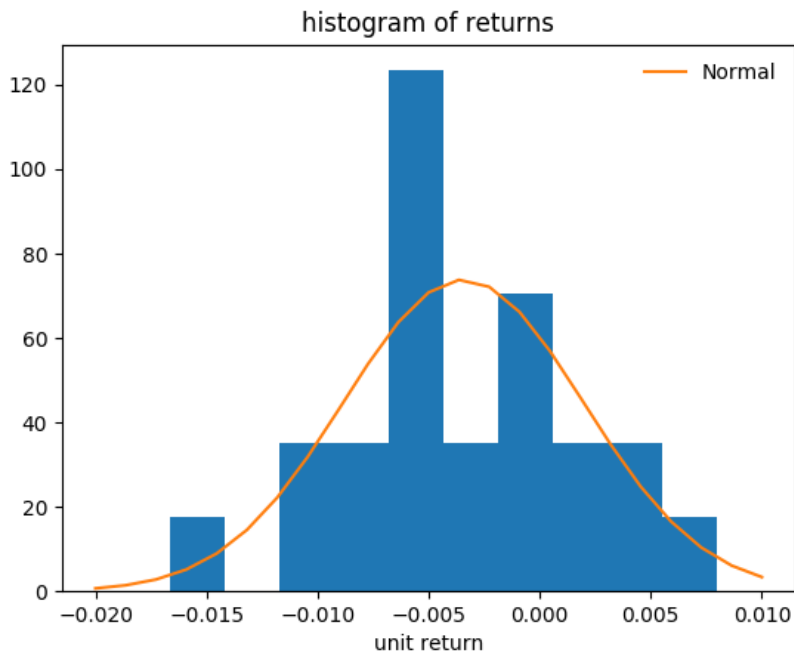


Figure 5.4 Distribution for 4 weeks

ii. q values of consecutive 21 days (3 weeks)

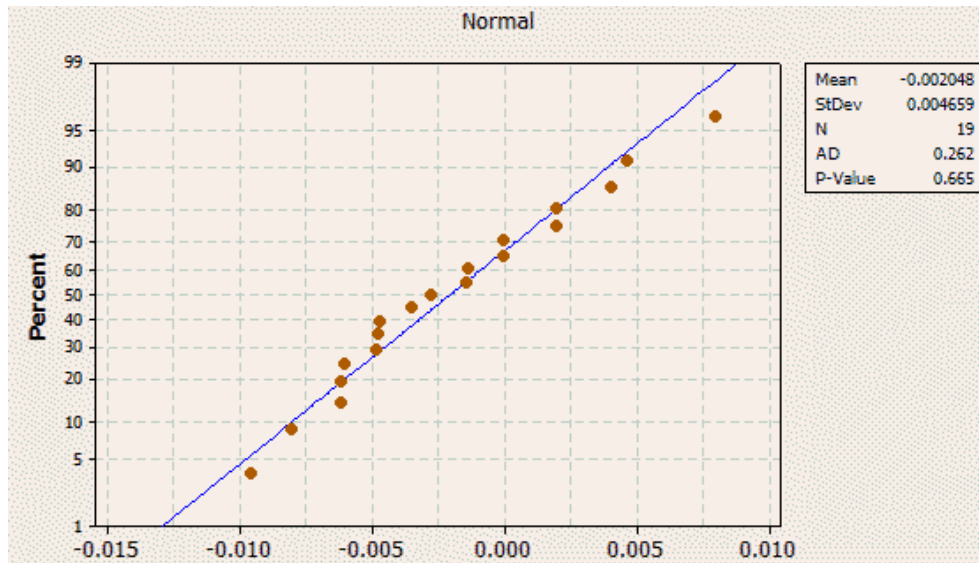


Figure 5.5 sample p-value check for 3 weeks

Since p-value (0.822) $>$ 0.05, There is not enough evidence to reject H_0 at 5% level of significance.

Data are normally distributed.

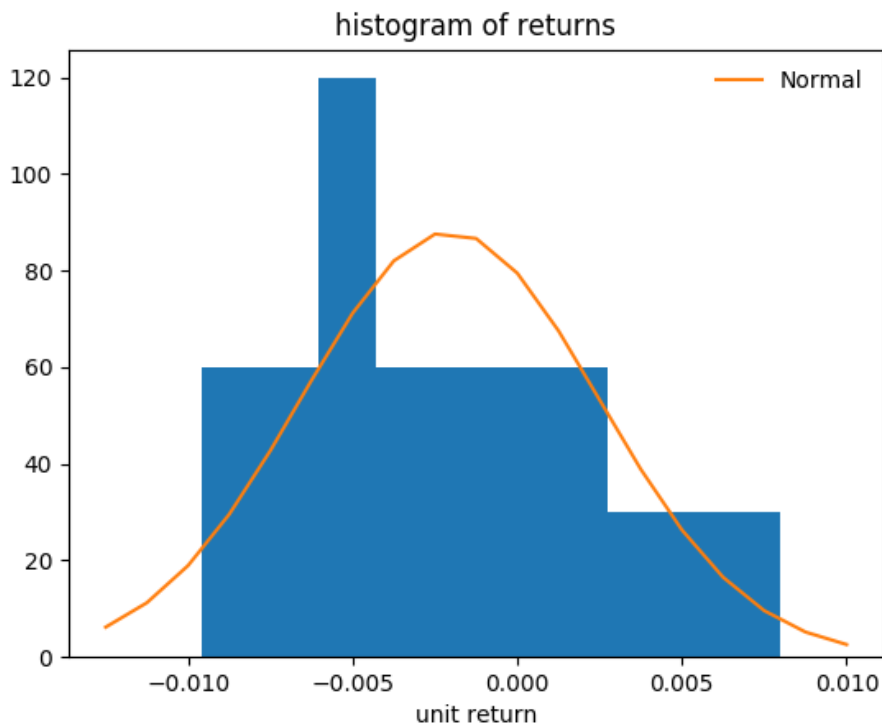


Figure 5.6 Distribution for 3 weeks

iii. q values of consecutive 14 days (2 weeks)

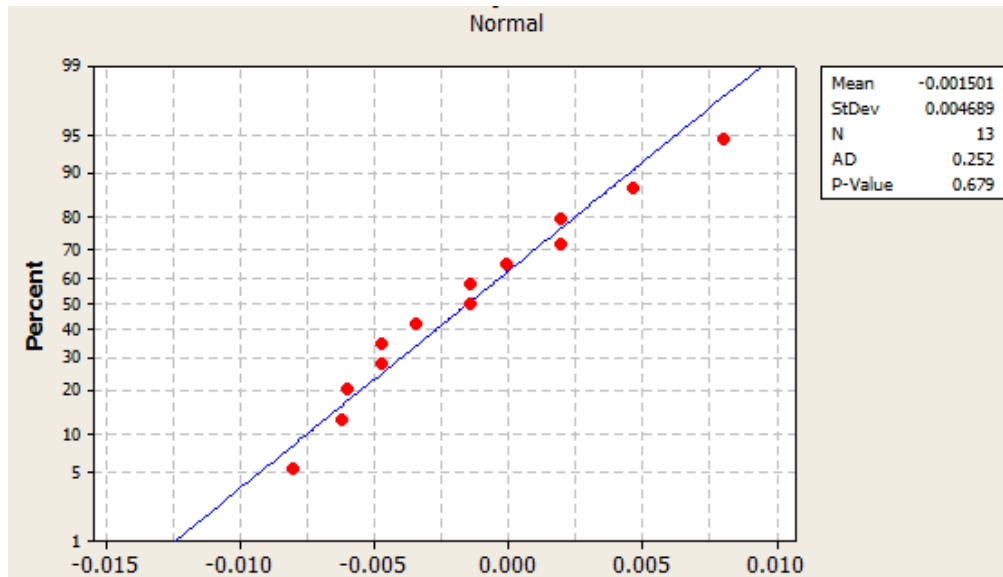


Figure 5.7 Sample p-value check for 2 weeks

Since $p\text{-value}(0.679) > 0.05$, There is not enough evidence to reject H_0 at 5% level of significance.

Data are normally distributed.

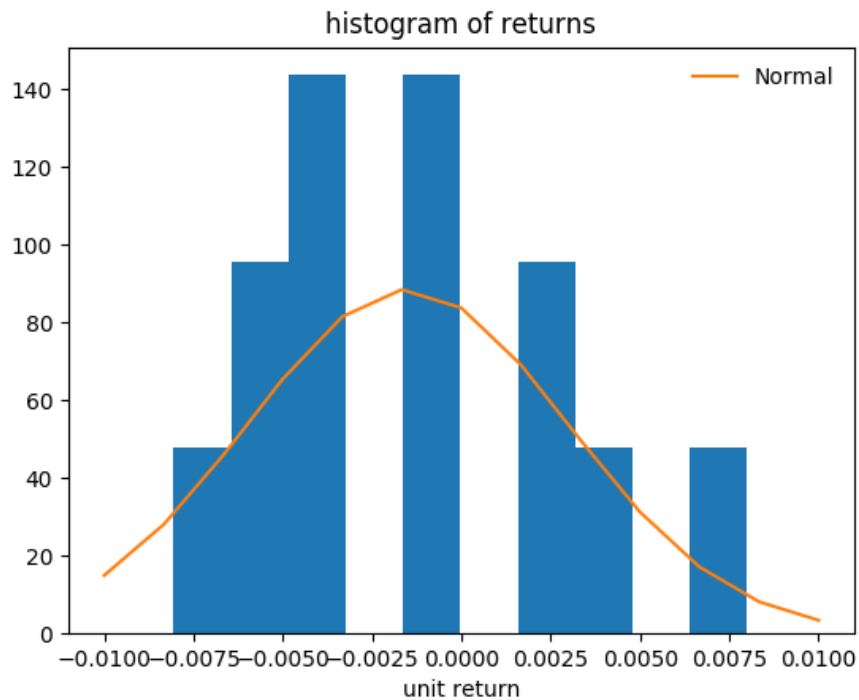


Figure 5.8 Distribution for 2 weeks

iv. q values of consecutive 7 days (1 weeks)

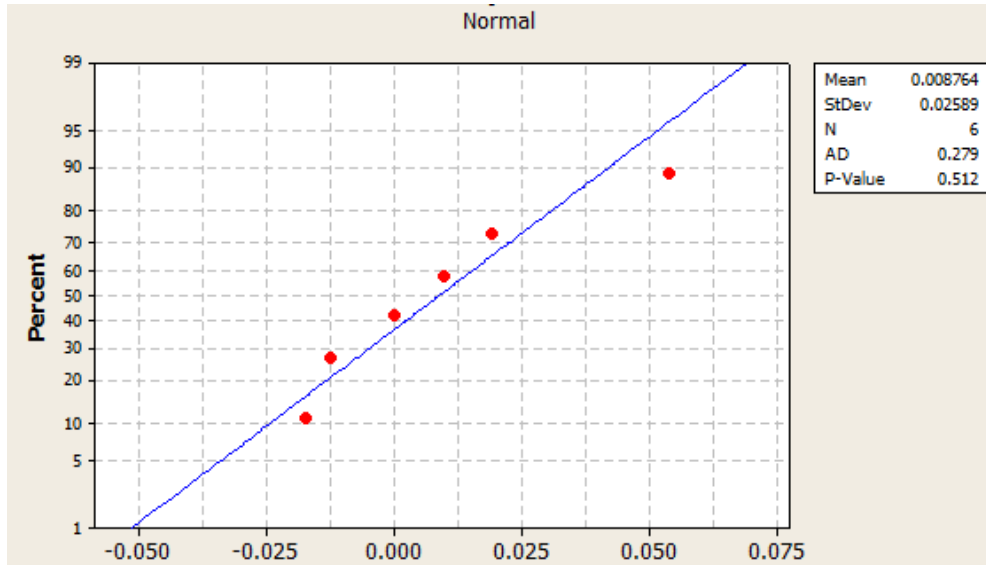


Figure 5.9 Sample p-value check for 1 weeks

Since $p\text{-value}(0.839) > 0.05$, There is not enough evidence to reject H_0 at 5% level of significance.

Data are normally distributed.

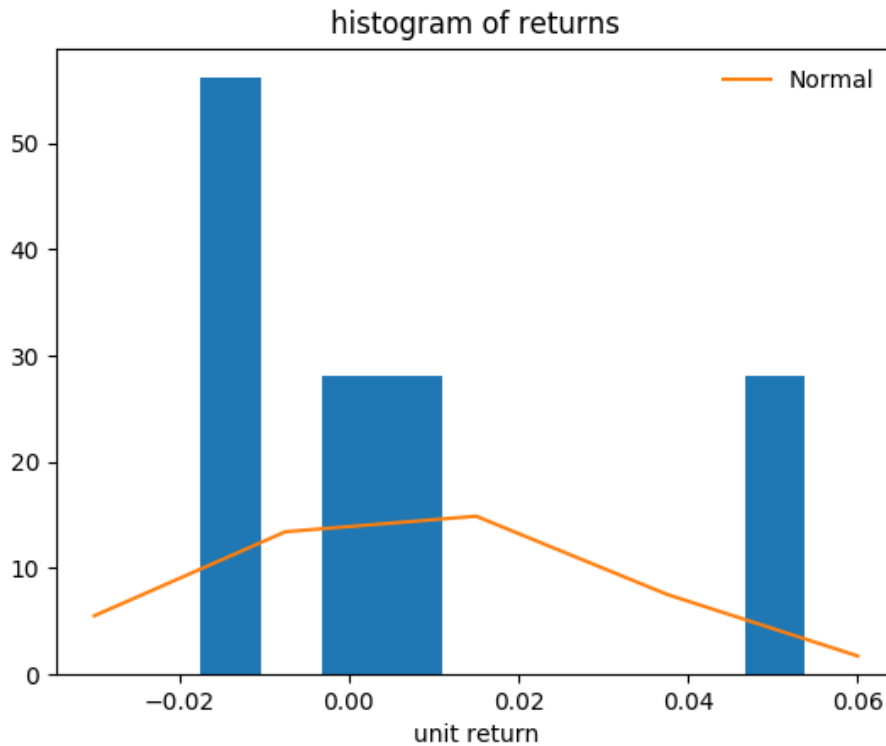


Figure 5.10 Distribution for 1 weeks

When selecting the distribution of q , the normality and appropriate case was selected as shown in the above figures.

5.2 Order book towards results

Accordance to the description in [section 3.1.2](#), cut off points were calculated from order book which was originally introduces for the model.

The calculation of B_{min} , B_{max} , S_{max} , S_{min} values are illustrated in figure 5.11.

Splits	Volume	Buy	Sell	Volume	splits
1	150681	14.78	14.8	260722	15
21	647662	14.76	14.82	129157	16
32	308113	14.74	14.84	113434	19
13	126169	14.72	14.86	228100	32
39	207271	14.7	14.88	180724	29
17:37:16					
1	150681	14.78	14.8	260722	15
21	647662	14.76	14.82	129157	16
32	308113	14.74	14.84	113434	19
13	126169	14.72	14.86	228100	32
39	207271	14.7	14.88	180724	29
17:38:16					
1	150681	14.78	14.8	260722	15
21	647662	14.76	14.82	129157	16
32	308113	14.74	14.84	113434	19
13	126169	14.72	14.86	228100	32
39	207271	14.7	14.88	180724	29
		14.68	14.94	0.017656	
		14.84	14.78	-0.0041	

Figure 5.11 Cut off points from order book

As calculated here,

$$B_{min} = 14.68$$

$$B_{max} = 14.86$$

$$S_{min} = 14.78$$

$$S_{max} = 14.94$$

Now, R_{min} and R_{max} are calculated using the equation 12 and equation 13 which was originally introduced after investigating the order book with Kelly criterion.

$$R_{min} = \frac{(B_{max} - S_{min})}{B_{max}} - \text{unit return per day from the deposit}$$

$$R_{min} = \frac{(14.78 - 14.86)}{14.86} - 0.0000547945$$

$$R_{min} = -0.004$$

$$R_{max} = \frac{(S_{max} - B_{min})}{B_{min}} - \text{unit return per day from the deposit}$$

$$R_{max} = \frac{(14.94 - 14.68)}{14.68} - 0.0000547945$$

$$R_{max} = 0.017656$$

5.3 Output from the model

As I described in section 4 and as in figure(5.1) following values are feed as the input for the model (equation 17).

- Mean (μ) of normally distributed q values
- standard deviation (σ) of normally distributed q values
- R_{max}
- R_{min}

Model calculates h and α values and other calculations. Then, generate the results with a graph of f verses $E \log \left(\frac{w_n}{w_0} \right)^{\frac{1}{n}}$ or f verses $E \left(\left(\frac{w_n}{w_0} \right)^{\frac{1}{n}} \right)$.

f – Fraction of money, trader can invest

$E \left(\left(\frac{w_n}{w_0} \right)^{\frac{1}{n}} \right)$ – Expected rate (this means if trader invest x amount of money, then he can earn $\left(\left(\frac{w_n}{w_0} \right)^{\frac{1}{n}} \right)$ including x .)

Below, is the illustration and description about the behavior of output using different output types.

Accordance to the section 3.1.1 the behavior of the output could be described and also as mentioned in section 3, model is built assuming market as a Kelly trader because all market data considering all traders have used to build the modification of the model.

- Output f_{max} is in between 0 and 1 ($0 < f_{max} < 1$)



Figure 5.12 Output with $0 < f_{max} < 1$

Input variables for the model

μ	0.002907
σ	0.16614
R_{max}	0.50133
R_{min}	-0.496

Based on the output results of this case, the Kelly trader could make the decision of investing 0.1053 in the stock and invest the remaining 0.8947 in deposits; while maximizing on the expected rate. Output f is less than or equal to 1 ($f_{max} \geq 1$).

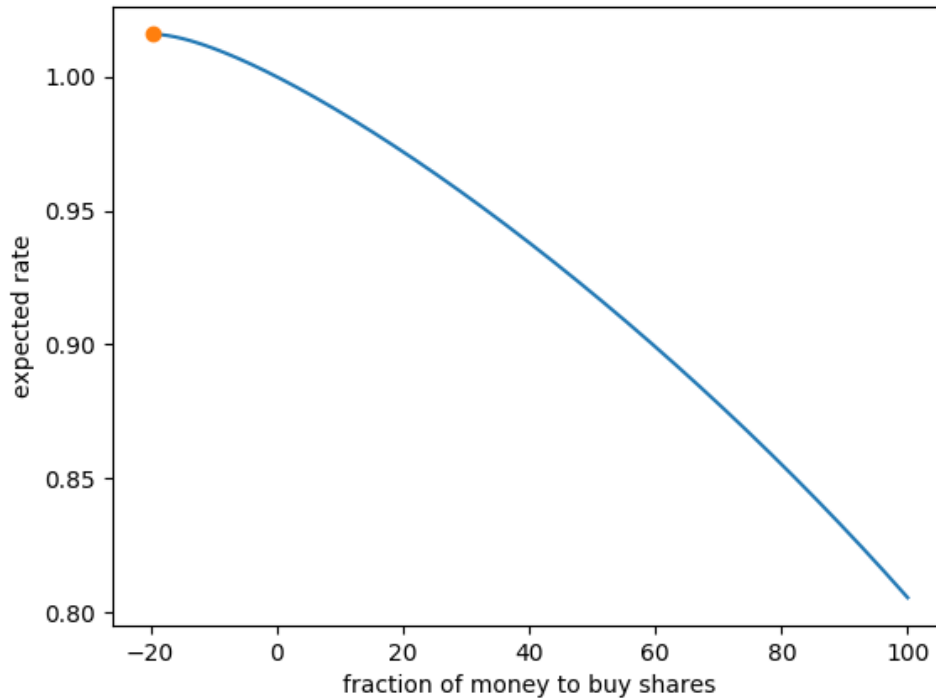


Figure 5.13 Output with $f_{max} < 1$

Input variables for the model

μ	-0.00081
σ	0.007563
R_{max}	0.03194
R_{min}	-0.0021

Based on the output results of this particular case, the ideal suggestion for Kelly trader is to sell the shares in the stock market.

- Output f is more than or equal to 1 ($f_{max} \geq 1$).

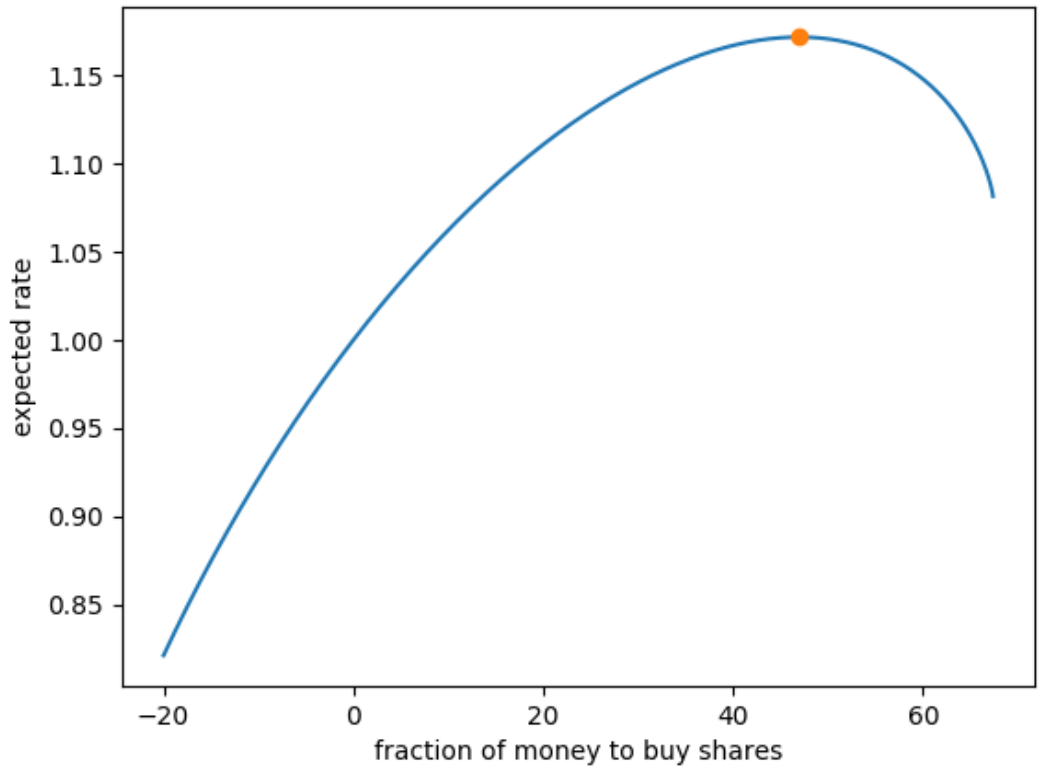


Figure 5.14 Output with $f_{max} > 1$

Input variables for the model

μ	0.008764
σ	0.02589
R_{max}	0.029084
R_{min}	-0.0156

Based on the output results, the ideal solution for the Kelly trader is to invest or buy shares in the stock market using all the money the trader has in possession and also if he is capable of borrowing money to invest up to 43 multiple of money the trader possesses, then the trader has the potential of obtaining the maximum expected rate (trader will be able to earn 1.15 multiple of the total investment).

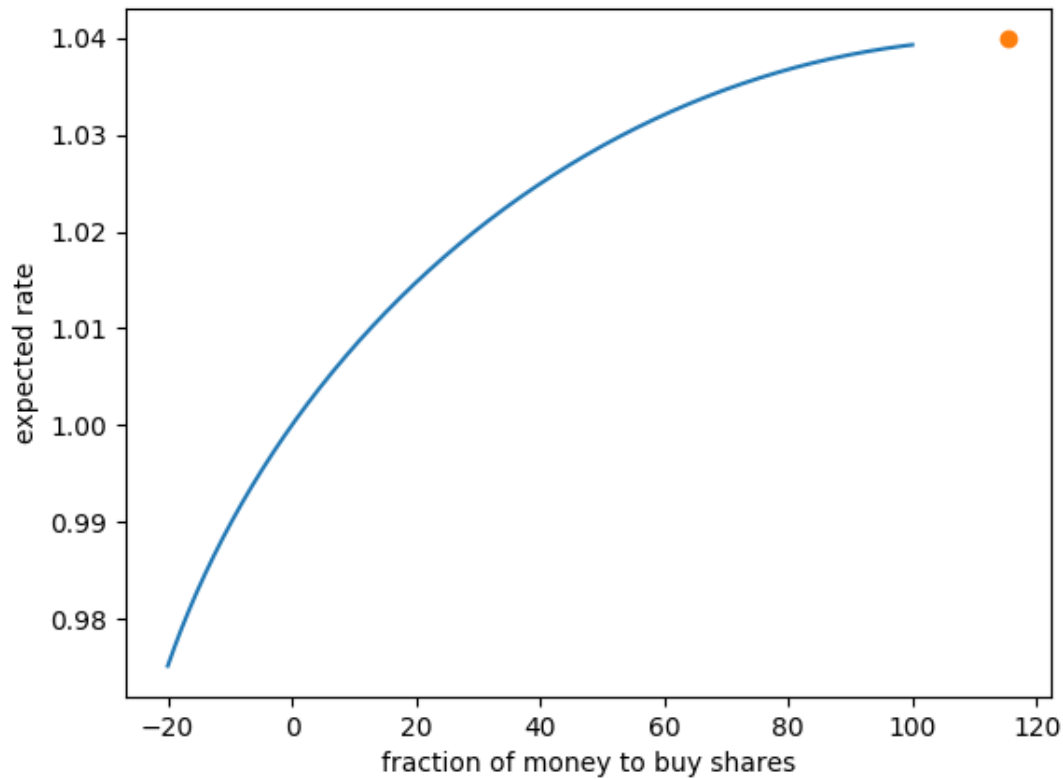


Figure 5.15 Output with very large $f > 1$

Input variables for the model

μ	0.0000444
σ	0.004691
R_{max}	0.03711
R_{min}	-0.0034

Based on the output results, the ideal solution for the Kelly trader is to invest or buy shares in the stock market using all the money the trader has in possession and also if he is capable of borrowing money to invest up to 120 multiple of money the trader possesses; then the trader has the potential of obtaining the maximum expected rate (trader will be able to earn 1.04 multiple of the total investment).

5.4 Decisions to the trader from the model

Decision is given by the output (f verses $E((\frac{w_n}{w_0})^{\frac{1}{n}})$) from the model (with or without order book).

As described in chapter 3, this model is developed (modifying the E.O thorp et al.[7] Kelly model) for day traders (short term traders). Say a day trader requires to obtain a decision about trading for day y . Then, use atmost 30 day historical trade data (as I described in section 5.1) up to day $y-1$ and obtain the normal distribution of q values. Then, calculate R_{min} and R_{max} values from the order book of day y . Feed μ , σ , R_{max} and R_{min} values to the model then output (f verses $E((\frac{w_n}{w_0})^{\frac{1}{n}})$) is retrived. In above section 5.2, described what the model says to Kelly trader (here for the whole market as the Kelly trader) looking at the output.

Basically, model says to Kelly trader to,

- Buy shares in the stock market if $f_{max} > 0$
- Sell shares in the stock market if $f_{max} < 0$

As described in above section this model assume whole market as the Kelly trader. Thus, in order to obtain a profit the trader who obtains a decision from the model need to buy shares if model says Kelly trader to sell shares. Also the trader who obtain a decision from the model need to sell shares if model says Kelly trader to buy shares. Then, the trader who obtain decision from the model need to behave like anti Kelly bettor.

Table 5.1 Decision to day trader

Output f value	Day Trader decision
$f_{max} > 0$	Sell shares at the beginning of the market
$f_{max} < 0$	Buy shares at the beginning of the market

5.5 Model validation

To validate the decision given to the trader by the model, the open and close price was used (from trades) for the day $y+1$ (The day model give decision to the trader). When the time, model generating

the buy or sell decision both the prices are unknown. However, based on the past data an evaluation could be carried out.

Table 5.2 Day trader decision evaluation.

Day trader decision by model		Correctness of decision
Buy shares	$Open\ Price - Close\ Price > 0$	Incorrect
Sell shares		Correct
Buy shares	$Open\ Price - Close\ Price < 0$	Correct
Sell shares		Incorrect

If $(open\ price - close\ price) > 0$, it indicate the beginning of the market share price is high than the share price of market end time.

If $(open\ price - close\ price) < 0$, it indicate the beginning of the market share price is low than the share price of market end time.

- If trader decision by the model is to sell shares and at the end of the day if $(open\ price - close\ price) > 0$, then the given trader decision is correct.
- If trader decision by the model is to buy shares and at the end of the day if $(open\ price - close\ price) > 0$, then the given trader decision is incorrect.
- If trader decision by the model is to sell shares and at the end of the day if $(open\ price - close\ price) < 0$, then the given trader decision is incorrect.
- If trader decision by the model is to buy shares and at the end of the day if $(open\ price - close\ price) < 0$, then the given trader decision is correct.

With this concept, the model has being evaluated for 30 days providing trader decision for 30 days.

The practical scenario is illustrated from the model until evaluation for 2 days buy and decisions.

- Validation 01:

Day trader requires to obtain a decision for tomorrow (consider tomorrow is July 6)

Then considering 2 weeks until July 5 to get the q value distribution (because these 2 weeks q values behave like normal is obtained).

Calculate the R_{min} and R_{max} values from order book of July 5.

The output f value is determined to be 27.6.

Thus, trader decision from the model is to sell shares at the beginning of the market on July 6.

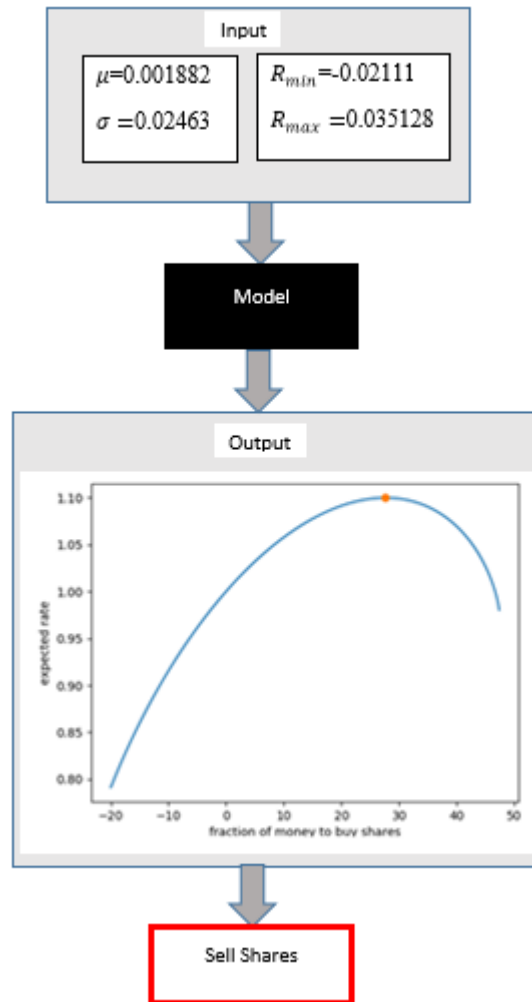


Figure 5.16 Model validation 1

Now after giving decision to sell shares on July 6, the open price and close price is concerned.

Open price-close price=0.34

The sign of the obtained value is plus thus, the decision is to sell shares. Thus the decision is correct.

However, if this value was a minus value then the decision obtained is incorrect.

- Validation 02:

Day trader wants to obtain a decision for tomorrow (consider tomorrow is April 20)

Then considering 3 weeks until April 19 to get the q value distribution (because these 3 weeks q values behave like normal).

Calculate the R_{min} and R_{max} values from order book of April 19.

The output f value is determined to be -19.15

Thus, trader decision from the model is to sell shares at the beginning of the market on July 20.

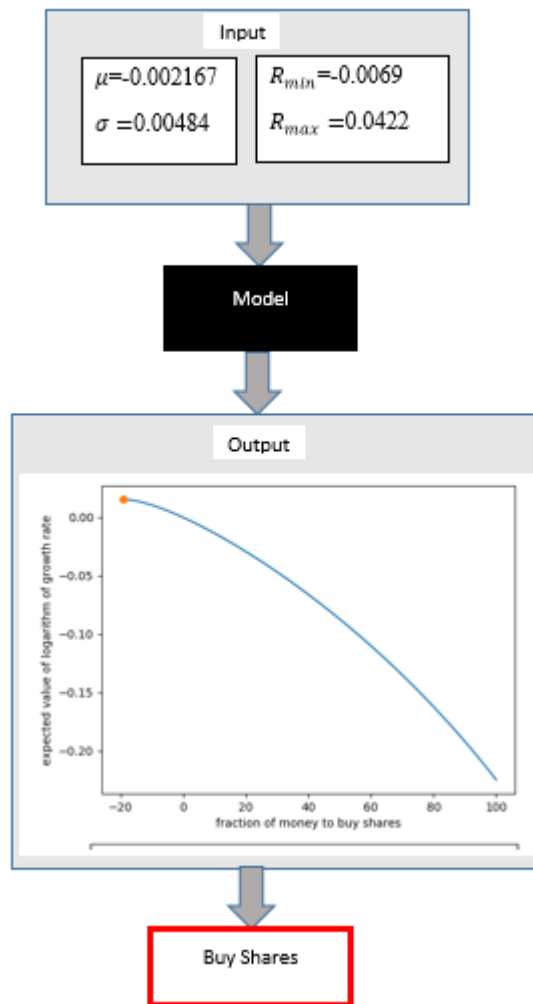


Figure 5.17 Model validation 2

Now After giving decision to sell shares on April 20, the open price and close price is concerned.

Open price-close price= -0.01

The sign of the obtained value is plus thus, decision is to buy shares. Thus the decision obtained is correct.

However, if this value was a minus value then the decision obtain is incorrect.

This process has being conducted for 30 days to obtain 30 decisions.

Accuracy of the developed model 70%

5.6 Accuracy without order book

Here also the above process was conducted for 30 days to obtain the decisions. However, for this evaluation R_{min} and R_{max} values was not used from order book. Like E.O Thorp et al. [7] used, cut off points of the distribution calculated as $(\mu - 3\sigma)$ and $(\mu + 3\sigma)$.

Accuracy is of the developed model 53%

Thus, there is a proper improvement of the Kelly model by using the order book for the model.

Chapter 6 – Conclusions

6.1 conclusions

This thesis aimed to building a dynamic computational model of Kelly Criterion for stock traders investigating trades and order book. As described in chapter 2 and chapter 3, the Kelly stock theoretical model of E.O Thorp et al. [7] was used. Modifying that model by investigating order book and trades as in chapter 3, I introduced dynamic Kelly computational model for stock traders as in chapter 5.

As described in section 5.5, validation of the model is processed considering the real behavior of the stock market with the implemented model described in chapter 4. The accuracy of introduced dynamic Kelly stock model to stock traders is 70%. A comparison cannot be made because there are not any existing a Kelly model to stock market which uses real behavior of market for the evaluation. The considered Kelly model to stock market of E.O Thorp et al. [7] is also a theoretical model.

As mentioned in chapter 3, before investigating the order book and trades with Kelly criterion, the entire market behave as a Kelly bettor was assumed. This developed dynamic model is based on this key assumption. Also as in section 5.4, this assumption is directly used to generate final decision to stock trader considering that stock trader should behave as an anti-Kelly bettor.

According to the E.O Thorp et al. [7], it is mathematically convenient to approximate finite distribution to a continuous model. This assumption too was taken into consideration. Thus, as stated section 5.1, I introduced 4 cases how short term returns behave like normal distribution as the model considers normal distribution as an input.

As in section 3.1.1, after investigating trades with used Kelly criterion to stock market model I introduced the methodology and equations about how to use trades with the existing Kelly model of stock market. There short term traders or day traders were introduced to the Kelly model of stock market which is a main contribution of this thesis. The model is modified with these methodology and equations.

As in chapter 5, with this modifications model has 70% accuracy. Thus, the accuracy verifies the modification with trades.

Assuming order book will give better information for the Kelly model of stock market, investigation of order book with Kelly model of stock market is processed. The reasons for the assumption is described in section 3.1.2 and in section 2.1. There also a methodology and equations were introduced about how to use order book with the existing Kelly model of stock market as a major contribution of this thesis.

As in section 5.6, to verify whether the assumption is correct or wrong a validation process without order book modifications with implemented model described in chapter 4 was implemented. The accuracy of the model without modifications with order book is 53%. This is less than the accuracy of the model with the modification of order book. Thus, it verifies order book generate better information when providing trader decisions from the Kelly model of stock market.

6.2 Limitations

If a particular person buy or sell Considerable amount of shares at once it will effect to these results. If the stock market crashes, then it will effect to the results of the model. if some malicious enters to the order book it also directly effect to the results of the developed model. Natural or man-made disasters that are beyond the reach of avoidably has a direct impact on the results.

6.3 Implications for further research

The developed model is evaluated with using one symbol in the Middle East stock market. Thus, another extension can be conducted considering more symbols of the stock market or considering various stock markets. Also, this model was developed for short term traders or day traders. Furthermore, an extension can be implemented considering the long term traders.

This research only provides the traders with the buy or sell decision of stocks. However, further improvements could be conducted in order to provide traders with the extent of the buy or sell decision of the stocks which would yield higher rate of return on investments.

References

- [1.]T. Foucault, O. Kadan and E. Kandel, "Limit Order Book as a Market for Liquidity", SSRN Electronic Journal
- [2]G. Silaghi and V. Robu, "An Agent Strategy for Automated Stock Market Trading Combining Price and Order Book Information", 2005 ICSC Congress on Computational Intelligence Methods and Applications.
- [3]A. Rinaldo, "Order aggressiveness in limit order book markets", Journal of Financial Markets, vol. 7, no. 1, pp. 53-74, 2004.
- [4]M. Höschler, Limit order book models and optimal trading strategies, 1st ed. 2011.
- [6]I. Cartea, R. Donnelly and S. Jaimungal, "Enhancing Trading Strategies with Order Book Signals", SSRN Electronic Journal.
- [7]L. Rotando and E. Thorp, "The Kelly Criterion and the Stock Market", The American Mathematical Monthly, vol. 99, no. 10, p. 922, 1992.
- [8]E. Thorp, The Kelly criterion in blackjack sports betting, and the stock market, 1st ed. 2006.
- [9]J. hung, Betting with the Kelly Criterion, 1st ed. 2010.
- [10]N.khanna, The Kelly Criterion and the Stock Market, 1st ed. 2016
- [11]Learning Performance of Prediction Markets with Kelly Bettors, 1st ed. 2012.
- [12]J. Kelly, "A New Interpretation of Information Rate", Bell System Technical Journal, vol. 35, no. 4, pp. 917-926, 1956.
- [13]Shannon, C. (1984). Communication in the presence of noise. Proceedings of the IEEE, 72(9), pp.1192-1201.
- [14]Shannon, C. (1948). A Mathematical Theory of Communication. Bell System Technical Journal, 27(4), pp.623-656.
- [15]Oracle.com. (2017). Java Software | Oracle. [online] Available at: <https://www.oracle.com/java/index.html> [Accessed 15 Dec. 2017].
- [16]O. Bunke, "Feller, F.: An Introduction to Probability Theory and its Applications, Vol. II, John Wiley & Sons, Inc., New York-London-Sydney, 1966. XVIII + 626 S., 3 Abb., 2 Tab., Preis \$ 12,00", Biometrische Zeitschrift, vol. 11, no. 6, pp. 428-428, 1969.
- [17]Python.org. (2017). Welcome to Python.org. [online] Available at: <https://www.python.org> [Accessed 17 Dec. 2017].

- [18]In.mathworks.com. (2017). MathWorks - Makers of MATLAB and Simulink. [online] Available at: <https://in.mathworks.com/> [Accessed 17 Dec. 2017].
- [19]Q. A and E. Al, "Using Data Mining Techniques to Build a Classification Model for Predicting Employees Performance", International Journal of Advanced Computer Science and Applications, vol. 3, no. 2, 2012.
- [20]H. Feng and H. Chou, "Evolutional RBFNs prediction systems generation in the applications of financial time series data", Expert Systems with Applications, vol. 38, no. 7, pp. 8285-8292, 2011.
- [21]Minitab.com. (2017). Minitab. [online] Available at: <http://www.minitab.com/en-us/> [Accessed 17 Dec. 2017].
- [22] Sinn, H. (1989). Economic decisions under uncertainty. Heidelberg: Physica-Verl.

Appendix A: Code Listing

Listing 1 Order book gathering code

```
1. package researchdata;
2. public class OrderBook extends javax.swing.JFrame {
3.
4.     /**
5.      * Creates new form OrderBook
6.      */
7.     public OrderBook() {
8.         initComponents();
9.     }
10.
11.     /**
12.      * This method is called from within the constructor to initialize the form.
13.      * WARNING: Do NOT modify this code. The content of this method is always
14.      * regenerated by the Form Editor.
15.      */
16.     @SuppressWarnings("unchecked")
17.     // <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN: initComponents
18.     private void initComponents() {
19.
20.         jScrollPane1 = new javax.swing.JScrollPane();
21.         orderBook = new javax.swing.JTable();
22.         jLabel1 = new javax.swing.JLabel();
23.         label = new javax.swing.JLabel();
24.
25.         setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
26.
27.         orderBook.setModel(new javax.swing.table.DefaultTableModel(
28.             new Object [][] {
29.                 { null, null, null, null, null, null },
30.                 { null, null, null, null, null, null },
31.                 { null, null, null, null, null, null },
```

```

32.         { null, null, null, null, null, null },
33.         { null, null, null, null, null, null },
34.         { null, null, null, null, null, null },
35.         { null, null, null, null, null, null },
36.         { null, null, null, null, null, null },
37.         { null, null, null, null, null, null },
38.         { null, null, null, null, null, null },
39.         { null, null, null, null, null, null },
40.         { null, null, null, null, null, null },
41.         { null, null, null, null, null, null },
42.         { null, null, null, null, null, null },
43.         { null, null, null, null, null, null },
44.         { null, null, null, null, null, null },
45.         { null, null, null, null, null, null },
46.         { null, null, null, null, null, null },
47.         { null, null, null, null, null, null },
48.         { null, null, null, null, null, null },
49.         { null, null, null, null, null, null },
50.         { null, null, null, null, null, null },
51.         { null, null, null, null, null, null },
52.         { null, null, null, null, null, null },
53.         { null, null, null, null, null, null }
54.     },
55.     new String [] {
56.         "Splits", "Bid Quantity", "Bid Price", "Ask Price", "Ask Quantity", "Splits"
57.     }
58. ));
59. jScrollPane1.setViewportViewView(orderBook);
60.
61. jLabel1.setText("Symbol : ");
62.
63. label.setText("jLabel2");
64.
65. javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
66. getContentPane().setLayout(layout);

```

```

67.     layout.setHorizontalGroup(
68.         layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
69.         .addGroup(layout.createSequentialGroup())
70.         .addGap(36, 36, 36)
71.         .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
72.             .addGroup(layout.createSequentialGroup())
73.             .addComponent(jLabel1)
74.             .addGap(18, 18, 18)
75.             .addComponent(label))
76.         .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 452, java
x.swing.GroupLayout.PREFERRED_SIZE))
77.         .addContainerGap(58, Short.MAX_VALUE))
78.     );
79.     layout.setVerticalGroup(
80.         layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
81.         .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, layout.createSequentialGroup())
82.         .addGap(22, 22, 22)
83.         .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
84.             .addComponent(jLabel1)
85.             .addComponent(label))
86.         .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 29, Short.M
AX_VALUE)
87.         .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 309, javax.s
wing.GroupLayout.PREFERRED_SIZE)
88.         .addContainerGap())
89.     );
90.
91.     pack();
92. }// </editor-fold>//GEN-END:initComponents
93.
94. /**
95.  * @param args the command line arguments
96.  */
97. public static void main(String args[]) {

```

```

98.     /* Set the Nimbus look and feel */
99.     //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
100.         /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and fe
           el.
101.         * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.ht
           ml
102.         */
103.         try {
104.             for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInst
           alledLookAndFeels()) {
105.                 if ("Nimbus".equals(info.getName())) {
106.                     javax.swing.UIManager.setLookAndFeel(info.getClassName());
107.                     break;
108.                 }
109.             }
110.         } catch (ClassNotFoundException ex) {
111.             java.util.logging.Logger.getLogger(OrderBook.class.getName()).log(java.util.logging
           .Level.SEVERE, null, ex);
112.         } catch (InstantiationException ex) {
113.             java.util.logging.Logger.getLogger(OrderBook.class.getName()).log(java.util.logging
           .Level.SEVERE, null, ex);
114.         } catch (IllegalAccessException ex) {
115.             java.util.logging.Logger.getLogger(OrderBook.class.getName()).log(java.util.logging
           .Level.SEVERE, null, ex);
116.         } catch (javax.swing.UnsupportedLookAndFeelException ex) {
117.             java.util.logging.Logger.getLogger(OrderBook.class.getName()).log(java.util.logging
           .Level.SEVERE, null, ex);
118.         }
119.     //</editor-fold>
120.
121.     /* Create and display the form */
122.     java.awt.EventQueue.invokeLater(new Runnable() {
123.         public void run() {
124.             new OrderBook().setVisible(true);
125.         }

```

```

126.         });
127.     }
128.
129.     // Variables declaration - do not modify//GEN-BEGIN:variables
130.     private javax.swing.JLabel jLabel1;
131.     private javax.swing.JScrollPane jScrollPane1;
132.     public javax.swing.JLabel label;
133.     public javax.swing.JTable orderBook;
134.     // End of variables declaration//GEN-END:variables
135. }

```

```

1. package researchdata;
2. import java.io.*;
3. import java.text.DecimalFormat;
4. import java.util.HashMap;
5. import java.util.Map;
6. import java.util.logging.Level;
7. import java.util.logging.Logger;
8. import javax.swing.table.DefaultTableModel;
9.
10. public class OrderbookViewFromLogFile extends Thread{
11.     Map<String, String> headerDictionary = new HashMap<String,String>();
12.     Map<String, String> tagDictionary1 = new HashMap<String,String>();
13.
14.     OrderBook ob = new OrderBook();
15.     DefaultTableModel model = (DefaultTableModel) ob.orderBook.getModel();
16.     int position;
17.     private static String FILENAME = "";
18.     PrintStream fileOut = null;
19.
20.     public OrderbookViewFromLogFile(){
21.         ob.setVisible(true);
22.         FILENAME = "F:\\DataLogs\\OrderBook\\log20170606123202Processed.txt";
23.         try {

```

```

24.     fileOut = new PrintStream(new FileOutputStream(FILENAME,true));
25.   } catch (FileNotFoundException ex) {
26.   }
27. }
28.
29. public void run(){
30.     try {
31.         BufferedReader br = new BufferedReader(new FileReader("F:\\DataLogs\\OrderBook\\log20
170606123202.txt"));
32.         String frame;
33.         while ((frame = br.readLine()) != null) {
34.             updateOBView(frame);
35.             sleep(10);
36.         }
37.     } catch (IOException ex) {
38.         Logger.getLogger(OrderbookViewFromLogFile.class.getName()).log(Level.SEVERE, null, e
x);
39.     } catch (InterruptedException ex) {
40.         Logger.getLogger(OrderbookViewFromLogFile.class.getName()).log(Level.SEVERE, null, e
x);
41.     }
42. }
43.
44. public static void main(String[]args){
45.     OrderbookViewFromLogFile viewForm = new OrderbookViewFromLogFile();
46.     viewForm.start();
47. }
48.
49. public void updateOBView(String frame){
50.     String headers[]= frame.split("\\|");
51.     String dateTime[] = headers[0].split(" ");
52.     String time = dateTime[1];
53.     for(int i=1;i<headers.length-1;i++){
54.         String temp[]=headers[i].split("=");
55.         if(temp[0].contains("300")){

```

```

56.         position = i;
57.         break;
58.     }else{
59.         headerDictionary.put(temp[0], temp[1]);
60.     }
61. }
62.
63. if (headerDictionary.get("102").contains("1150")){
64.     ob.label.setText(headerDictionary.get("102"));
65.     int decimalFactor = Integer.parseInt(headerDictionary.get("104"));
66.     if(headerDictionary.get("108").contains("2")){
67.         for (int i = position; i<headers.length;i++){
68.             String temp1[]=headers[i].split(";");
69.             for(int j=0;j<temp1.length;j++){
70.                 String x[] = temp1[j].split("=");
71.                 tagDictionary1.put(x[0],x[1]);
72.             }
73.             double Price = Double.parseDouble(tagDictionary1.get("302"))/decimalFactor;
74.             DecimalFormat df = new DecimalFormat("#####0.000");
75.             String realPrice = df.format(Price);
76.
77.             if(tagDictionary1.get("300").contains("1")){ //Bid
78.                 model.setValueAt(tagDictionary1.get("304"), Integer.parseInt(tagDictionary1.get("30
79.                 1")), 0);
80.                 model.fireTableDataChanged();
81.                 model.setValueAt(tagDictionary1.get("303"), Integer.parseInt(tagDictionary1.get("30
82.                 1")), 1);
83.                 model.fireTableDataChanged();
84.                 model.fireTableDataChanged();
85.                 else if(tagDictionary1.get("300").contains("2")){ //Ask
86.                     model.setValueAt(realPrice, Integer.parseInt(tagDictionary1.get("301")), 2);
87.                     model.fireTableDataChanged();
88.                     model.fireTableDataChanged();
89.                     model.setValueAt(realPrice, Integer.parseInt(tagDictionary1.get("301")), 3);
90.                     model.fireTableDataChanged();
91.                     model.setValueAt(tagDictionary1.get("303"), Integer.parseInt(tagDictionary1.get("30
92.                     1")), 4);

```



```

88.         model.fireTableDataChanged();
89.         model.setValueAt(tagDictionary1.get("304"), Integer.parseInt(tagDictionary1.get("30
1")), 5);
90.         model.fireTableDataChanged();
91.     }
92. }
93.     System.out.println(time);
94.     for(int j=0; j<5;j++){
95.         System.out.println(model.getValueAt(j,0)+","+model.getValueAt(j,1)+","+model.getVal
ueAt(j,2)+","+model.getValueAt(j,3)+","+model.getValueAt(j,4)+","+model.getValueAt(j,5));
96.         System.setOut(fileOut);
97.     }
98.     System.out.println("");
99.     System.setOut(fileOut);
100.    }
101.    }
102.    }

```

Listing 2 Trades processing

```

1. package researchdata;
2.
3. import java.io.BufferedReader;
4. import java.io.FileNotFoundException;
5. import java.io.FileReader;
6. import java.io.FileWriter;
7. import java.io.IOException;
8. import java.io.Writer;
9.
10. public class ReadTradeOnline {
11.
12.     * @param args the command line arguments
13.     */
14.     public static void main(String[] args) throws FileNotFoundException {
15.         // TODO code application logic here'
16.
17.         String symbol=null;
18.         String time=null;
19.         BufferedReader br = new BufferedReader(new FileReader("F:\\DataLogs\\Trade\\newdata\\Ne
w folder\\2016 april 6 to 2017 April 6.txt"));
20.         String row;

```

```

21.
22.     try {
23.         while ((row = br.readLine()) != null) {
24.
25.             String data[]= row.split(",");
26.             //System.out.print(row);
27.
28.             for (int i=0;i<data.length;i++){
29.                 // System.out.print(data[i].length()+"....");
30.
31.                 String r[]=data[i].split(",");
32.
33.                 try(Writer writer = new FileWriter("F:\\DataLogs\\Trade\\newdata\\New folder\\2016 ap
ril 6 to 2017 April 6 Made.txt",true)){
34.
35.                     writer.append(r[0]+" "+r[1]+" "+r[2]+" "+r[3]+" "+r[4]);
36.                     writer.append("\n");
37.                 }catch(IOException ex){
38.                     ex.printStackTrace(System.err);
39.                 }
40.
41.
42.                 System.out.print("\n");
43.             }
44.         }
45.
46.
47.         }catch(IOException ex){
48.             ex.printStackTrace(System.err);
49.         }
50.     }
51.
52. }

```

Listing 3 Model implementation after giving inputs

```

1. __author__ = 'PiyumiR'
2.
3. from scipy.optimize import fsolve
4. import numpy as np
5. import math
6. from scipy.integrate import quad,simps,fixed_quad
7. from scipy.optimize import fmin
8. from matplotlib import pyplot as plt
9.
10. m=0.001126

```

```

11. v=0.01965
12. a=-0.0041
13. b= 0.017656
14.
15. #pi value
16. p=3.1415926535
17.
18. #equation for q**2
19. def integrand1(q):
20.     return q**2
21.
22. #equation for f(q)
23. def integrand2(q,alp):
24.
25.     return ((np.exp(-1*((q-m)**2)/(2*(alp**2))))/(np.sqrt(2*p*(alp**2))))
26.
27. #equation for q**2 * f(q)
28. def integrand3(q,alp):
29.
30.     return ((q**2)*((np.exp(-1*((q-m)**2)/(2*(alp**2))))/(np.sqrt(2*p*(alp**2)))))
31.
32. #get the integrals and return the function that equal to 0
33. def func(alp):
34.     [y1,error1]=quad(integrand1, a, b)
35.     [y2,error2]=quad(integrand2, a, b, args=(alp,))
36.     [y3,error3]=quad(integrand3, a, b, args=(alp,))
37.     return y1-(y1*y2)-((b-a)*((m**2) + (v**2)))+((b-a)*y3)
38.
39. def h_finder(alp):
40.     def integrand(q):
41.         return ((np.exp(-1*((q-m)**2)/(2*(alp**2))))/(np.sqrt(2*p*(alp**2))))
42.     [val,error]=quad(integrand, a, b)
43.     h=(1-val)/(b-a)
44.     print "h value : ",h
45.     return h
46.
47. def model_implement(alp,h):
48.
49.     def integrand(q,f):
50.         return ((np.log(1+f*q))*(h+((np.exp(-1*((q-m)**2)/(2*(alp**2))))/(np.sqrt(2*p*(alp**2)))))
51.
52.     def g(f):
53.         return quad(integrand, a, b, args=(f,))[0]
54.
55.     def t(f):
56.         return -1*quad(integrand, a, b, args=(f,))[0]

```

```

57. x = np.arange(-20, 100, 0.01)
58.
59. vec_expint = np.vectorize(g)
60. plt.plot(x,vec_expint(x))
61.
62. d=fmin(t,0)
63. y=vec_expint(d)
64. print("f = ")
65. print(d)
66. print("\n")
67. print("E(log(w/w0)) = ")
68. print(y)
69. max = y.argmax()
70. plt.plot(d,y,'o')
71. plt.xlabel("fraction of money to buy shares")
72. plt.ylabel("expected value of logarithm of growth rate")
73. plt.show()
74. plt.plot(x,np.exp(vec_expint(x)))
75. ynew=np.exp(y)
76. plt.plot(d,ynew,'o')
77. plt.xlabel("fraction of money to buy shares")
78. plt.ylabel("expected rate")
79. print(ynew)
80. plt.show()
81.
82. def alp_finder():
83.     sol = fsolve(func,0.2)
84.     print "Solution for alpha : ", sol[0]
85.     return sol[0]
86.
87. alp=alp_finder()
88. h=h_finder(alp)
89. model_implementation(alp,h)

```

Listing 4 Model implementation without order book after giving inputs

```

1. from scipy.optimize import fsolve
2. import numpy as np
3. import math
4. from scipy.integrate import quad,simps,fixed_quad
5. from scipy.optimize import fmin
6. from matplotlib import pyplot as plt

```

```

7.
8. m=0.001126
9. v=0.01965
10.
11. a=m-3*v
12. b=m+3*v
13.
14. #pi value
15. p=3.1415926535
16.
17. #equation for q**2
18. def integrand1(q):
19.     return q**2
20.
21. #equation for f(q)
22. def integrand2(q,alp):
23.
24.     return ((np.exp(-1*((q-m)**2)/(2*(alp**2))))/(np.sqrt(2*p*(alp**2))))
25.
26. #equation for q**2 * f(q)
27. def integrand3(q,alp):
28.
29.     return ((q**2)*((np.exp(-1*((q-m)**2)/(2*(alp**2))))/(np.sqrt(2*p*(alp**2))))))
30.
31. #get the integrals and return the function that equal to 0
32. def func(alp):
33.     [y1,error1]=quad(integrand1, a, b)
34.     [y2,error2]=quad(integrand2, a, b, args=(alp,))
35.     [y3,error3]=quad(integrand3, a, b, args=(alp,))
36.     return y1-(y1*y2)-((b-a)*((m**2) + (v**2)))+((b-a)*y3)
37.
38. def h_finder(alp):
39.     def integrand(q):
40.         return ((np.exp(-1*((q-m)**2)/(2*(alp**2))))/(np.sqrt(2*p*(alp**2))))
41.     [val,error]=quad(integrand, a, b)

```

```

42. h=(1-val)/(b-a)
43. print "h value : ",h
44. return h
45.
46. def model_implement(alp,h):
47.
48.     def integrand(q,f):
49.         return ((np.log(1+f*q))*(h+((np.exp(-1*((q-m)**2)/(2*(alp**2))))/(np.sqrt(2*p*(alp**2))))))
50.
51.     def g(f):
52.         return quad(integrand, a, b, args=(f,))[0]
53.
54.     def t(f):
55.         return -1*quad(integrand, a, b, args=(f,))[0]
56.     x = np.arange(-20, 100, 0.01)
57.
58.     vec_expint = np.vectorize(g)
59.     plt.plot(x,vec_expint(x))
60.
61.     d=fmin(t,0)
62.     y=vec_expint(d)
63.     print("f = ")
64.     print(d)
65.     print("\n")
66.     print("E(log(w/w0)) = ")
67.     print(y)
68.     max = y.argmax()
69.     plt.plot(d,y,'o')
70.     plt.xlabel("fraction of money to buy shares")
71.     plt.ylabel("expected value of logarithm of growth rate")
72.     plt.show()
73.     plt.plot(x,np.exp(vec_expint(x)))
74.     ynew=np.exp(y)
75.     plt.plot(d,ynew,'o')
76.     plt.xlabel("fraction of money to buy shares")

```

```
77. plt.ylabel("expected rate")
78. print(ynew)
79. plt.show()
80.
81. def alp_finder():
82.     sol = fsolve(func,0.2)
83.     print "Solution for alpha : ", sol[0]
84.     return sol[0]
85.
86. alp=alp_finder()
87. h=h_finder(alp)
88. model_implement(alp,h)
```