공학박사학위논문

# Financial Market Forecasting and Trading Strategy based on Network Fractality

## 네트워크 프랙탈 성질에 기반한 금융시장 예측 및 거래전략

2020 년 8 월

서울대학교 대학원

산업공학과

구 승 모

# Financial Market Forecasting and Trading Strategy based on Network Fractality

## 네트워크 프랙탈 성질에 기반한 금융시장 예측 및 거래전략

지도교수 장 우 진

이 논문을 공학박사 학위논문으로 제출함

2020 년 6 월

서울대학교 대학원

산업공학과

구 승 모

구승모의 공학박사 학위논문을 인준함

2020 년 7 월

| | | |
|---|---|---|
| 위 원 장 | 조 성 준 | (인) |
| 부위원장 | 장 우 진 | (인) |
| 위 원 | 박 건 수 | (인) |
| 위 원 | 이 경 식 | (인) |
| 위 원 | 장 희 수 | (인) |

## Abstract

# Financial Market Forecasting and Trading Strategy based on Network Fractality

Seungmo Ku

Department of Industrial Engineering

The Graduate School

Seoul National University

Extensive academic research was performed for the financial market as it is closely connected to practical economy. Research in traditional financial economics resulted in economic indicators and the econometrics was instrumental for quantitative research in financial market. However, it proved to be difficult to predict the market behavior as it is a result of complex interaction among many agents with their own agenda. An effective tool to predict a change in market would be beneficial for policy makers and market participants to assist them with rational and consistent decision making. On the other hand, inconsistent prediction would lead to a suboptimal and inconsistent market activity which sometimes result in sudden collapse in the market as it did in 2008 Financial crisis and 1997 Asian financial crisis. The purpose of this dissertation is to develop approach based on econophysics and machine learning to systematically analyze the financial market.

The main focus of this dissertation involves the network structure of stock

market. To predict the change in market behavior, it is critical to understand the relationship or correlation among the market participants beforehand, and complex network analysis is one of the most prominent methods for such study. The fractal theory was employed as the primary approach to analyze the network structure of financial market. The empirical study shows that the network of financial market exhibits fractal properties. Also, analysis of fractal dimension and network topology led to two key discoveries. First, the fractal dimension and the Strong effective repulsion between distinct network nodes known as the hub are closely related. Second, the fractal dimension reveals the shortcut of network structure. Through further analysis, these two properties were proved to be useful for risk management in financial market. Three fractal measures were proposed to specify network structure for ease of implementation in future studies.

In the second step, the fractal measures were implemented in a financial market to assess its ability to predict the market movement. Recently, studies were conducted to determine if a new measure or index improves the prediction accuracy for financial time series. These studies are advantageous for future studies as it proposes new indices for other implementation and further analysis rather than studying the precision of their own method. In this paper, machine learning algorithms were employed to assess the predictive properties of fractal measures. Empirical experiments were performed to predict direction of market movement, which is effectively a classification task, and prediction for returns, a regression task. The studies concludes that the fractal measure proposed were effective in prediction for long-term stock returns of more than three months period.

Finally, a model to improve trading strategy based on learning-to-rank algorithm and the fractal measures was introduced. Previous studies are often based on the modern portfolio theory(MPT), but it is insufficient for real-world application as it doesn't provide any implication for rebalancing period of portfolio. The optimal rebalancing model proposed in this study allows its application with traditional portfolio methods. The experiments were carried out in two steps. The model learns to predict the better time period to perform rebalancing between two time periods in the future, followed by the empirical simulation to apply the model in real world trading scenario. Two traditional portfolio methods, equal weighted and maximized Sharpe ratio, were taken for experiment. The result affirms that the optimal rebalancing model was able to capture the better time period of rebalancing portfolio. In addition, the model outperformed a simple rebalancing method of fixed time period. When the fractal measures were employed as an input variable, the model performance was further improved. The primary contribution achieved through this model is that it allows application and expansion into all traditional portfolio models. Also, the fractal measures observed in the network structure grants insight regarding the market behavior and empirically proved that the measure provides benefit in prediction for the real-world stock market.

**Keywords**: Econophysics, Risk management, Complex network analysis, Fractal theory, Network topology, Market index prediction, Optimal rebalancing model, Trading strategy

**Student Number**: 2015-21126

# Contents

# List of Tables

# List of Figures

# Chapter 1

# Introduction

## 1.1 Research Motivation and Purpose

Since the relationship between stock market and investment planning is highly related it has been studied widely in both academic and practical field. Especially, several studies that showed the quantitative analysis of stocks based on prices and returns have been proven theoretically and empirically. Stocks represent a variety of interactions and have complex characteristics, which is why investigating investment strategies based on efficient forecasting of stock prices and markets is essential for investors to make optimal decisions.

Complex network analysis is a representative methodology for analyzing the relationships in stock market. Complex network analysis explains the complex non-linear phenomena in the stock market in a variety of ways. In order to build a network of stock markets, it is necessary to define a distance metric that indicates the relationship between stocks. The most widely used distance metric is a metric based on Pearson correlation. It is known that correlation plays an important role in the time series of stock prices, and there are previous research results that a net-

work analysis methodology based on correlation metrics can explain various characteristics in financial markets citeplongin1995correlation, tumminello2010correlation, kwapien2012physical, patro2013simple, marti2017review. Especially, stocks that are high-correlated in the network have successfully derived useful insights related to the structure of the stock market, using various methodologies derived from financial analysis(Coletti, 2016; Tumminello et al., 2005; Boginski et al., 2005).

The stock market network has a characteristic that the structure of the network is complicated because it affects market participants of various behaviors, and thus it is difficult to analyze. As a complex network is representative methodology for analyzing network, various characteristics of structure network are analyzed after the concept of Small-world networks(Watts and Strogatz, 1998) was announced, analysis using various coefficients, including a methodology for scale-free network structure analysis using scale-free gamma(Barabási and Bonabeau, 2003) indicators, has also been studied(Costenbader and Valente, 2003; Saramäki et al., 2007; Perra and Fortunato, 2008). Recently, there are many studies are being conducted to observe self-similarity properties in various real-world networks: biological, material, social, and logistics networks(Song et al., 2005, 2007; Wei et al., 2013; Guida and Maria, 2007; Ivanovici and Richard, 2010; Bunde and Havlin, 2013; Santos and Santos, 2018). The self-similarity characteristic of the network, also called fractality, is particularly suitable for observing the pattern of a growing network and robustness of the network structure(Song et al., 2006; Goh et al., 2006; Kim et al., 2007; Yook et al., 2005; Li et al., 2017). We will observe the structure of the stock market network through the measurement of fractality, and defines fractal measures that include these network characteristics. We also analyze whether proposed fractal measures

can be used to analyze the stock market.

Although various market participants artificially constitute a financial network, we assume that the network exhibits some natural properties of a fractal structure. In addition, we believe that those properties can be further utilized in the management of financial markets and products. For instance, it is possible to identify and even predict the relevance of changes in market structure through a comparative analysis of fractal dimension. In this research, we propose fractal measures that quantify network fractality similar to the means of scale-free gamma and clustering coefficients. Furthermore, we examine the predictive power of the proposed measures for the $N$ weeks future return series of S&P500 as attempted in previous literature(Lee et al., 2019; Caraiani, 2014; Darbellay and Wuertz, 2000; Maasoumi and Racine, 2002). Before predicting the future return series, this study first verifies whether the proposed fractal measures have the predictive power in the future price direction. The algorithms used for analysis are logistic regression and random forest, which are simple models used in previous studies(Ballings et al., 2015). If there is a predictive power for the direction of the S&P500 index, proposed fractal measures can be used for the predicting the actual price. We utilize the neural network to successfully capture the non-linearity of the prediction function (Qiu et al., 2016; Lahmiri and Bekiros, 2019; Chai and Lim, 2016). Note that we only used simple machine learning algorithms since the main purpose of this research is to assert the relations of the proposed indicators in the stock market.

On the other hand, various algorithms have been used to develop stock market trading strategies, and these efforts are mainly based on return series. Starting from

modern portfolio theory(Markowitz, 1952) various portfolio composition studies have been continued with considering additional variables such as short sales, transaction cost, cornet solution, etc (Clarke et al., 2006; Blitz and Van Vliet, 2007; Doganoglu et al., 2007; Strub and Baumann, 2018; Yu et al., 2020). In addition, a study combining complex theory and traditional portfolio theory is being conducted(Yu et al., 2014).

Recently, as the availability of data has increased, the constructing a portfolio based on profit prediction using machine learning is also actively researched.Dempster and Leemans (2006); Lu et al. (2009); Almahdi and Yang (2017). Among various machine learning approaches, the work of Song et al. (2017) proposed a portfolio composition and trading strategy using learning-to-rank algorithm as news sentiment data. Learning-to-rank is a machine learning algorithm that learns the ranks used in the ranking system. In that study, an decision making process of investor is following methodology that first ranking all available assets and then buying the top assets and selling the bottom assets.

In this dissertation, we are planning to apply the rank algorithm of the ranking system with a different perspective. In the previous study above, the ranking of different assets is ranked at each time point. That is, the learning-to-rank algorithm is used for the portfolio construction method. However, our research goal is to fix the portfolio selection method at a different time point. When the portfolio selection methodology is decided, it is to propose an appropriate portfolio selection timing and rebalancing strategy. In order to experiment with our strategy, we need to define some portfolio selection methodologies that are most commonly used for

portfolio construction methodologies, These two are the portfolio that is evenly distributed across all available assets, and portfolio selection methodology proposed by Markowitz (1952) to maximize Sharpe ratio(Sharpe, 1966). The reason this strategy is highly useful is that once the portfolio selection methodology and ranking is determined, the model finds better rebalancing timing through training. Therefore, there are advantages of expanding the various portfolio selection methodologies and satisfies various investor with different invest sentiments.

In summary, the contribution of this dissertation is proposing fractal measures to extract the structural characteristics of the stock market network. Through the network structure analysis, this dissertation observed that the proposed fractal measures extract two important structural characteristics that could help the risk management of the stock market. The two important structural characteristics extracted from the stock market network are "hub repulsion" and "shortcut pattern". To evaluate these structural characteristics, two aspects of application are studied. First, we verified that these structural characteristics help to improve the predictive power of stock market using machine learning. Second, we developed a trading strategy using the proposed fractal measures. Through simulations of trading strategies, it was shown that the proposed fractal measures are practical indicators that can also be used in actual investment.

## 1.2 Organization of the Research

This dissertation is organized in six chapters: Chapter 1 introduces research motivation and purpose. Chapter 2 reviews prior studies on measuring the fractality

of the network, machine learning algorithms related to prediction in financial markets, portfolio construction using them, and trading strategies. Chapter 3 describes how to configure the threshold network to be used in our research , how to measure network fractality(fractal dimension), and three fractal measures are defined. Then, we analyze how the proposed fractal measures are related to the structural characteristics of the stock market network. Chapter4 demonstrates the effect of the proposed fractal measures on improving the predictive power of the stock market. The experiment proceeds in two steps. First, we examine the effect of improving the predictive power of the proposed fractal measures as a model that classifies the future direction of the stock market index. Next, we will examine the effect of improving the predictive power of the proposed fractal measures as a model for predicting the stock market index, and analyze the impact of each fractal measure. Chapter5 proposes a RankNet-based optimal rebalancing model that reflects the network structure characteristics indicated by fractal measures. Then, we develop a trading strategy using an optimal rebalancing model, and show that the proposed model can be used for actual investment through simulation analysis. Chapter 6 is about conclusions and future works.

# Chapter 2

# Literature Review

## 2.1 Complex Network

Since the concept of the 'small-world' network(Watts and Strogatz, 1998) is known, complex network analysis has been actively studied. Various types of networks have been studied, the scale-free network(Barabási and Bonabeau, 2003) is well known. This study claims that degree of multiple network nodes created in real-world network follows the power-law distribution. This presents that the scale-free network is robust to accidental failures, and suggests that it can be applied to many fields with these characteristics. Since then, a methodology for analyzing the network structure as an indicator was studied. These methodologies are using centrality measure and clustering coefficient. The work of Perra and Fortunato (2008) compared the conceptual basis of various centrality measures to calculate the important nodes of the network, and the work of Saramäki et al. (2007) is a summary and comparison of clustering coefficients and indicators representing the structure of several complex networks. The work of Mao and Xiao (2019) proposed a methodology that can increase the predictive power by combining link prediction with general time series prediction.

In this dissertation, the main objective is to effectively observe changes in the structure of the stock market. To observe changes in the stock market structure, not only information about individual stocks, but also quantitative indicators that can efficiently represent the structural characteristics of the entire market are essential. In this regard, a scale-free gamma representing a scale-free property is typical. Subsequently, Song et al. (2005) propose indicators with more fundamental length-scale invariant properties. This index is a fractal dimension value that represents the network's fractality. Further studies(Song et al., 2006) found that the key principle for generating fractal structures in a network is strong effective repulsion phenomenon between most connected nodes at all length scales, and this characteristic is related to the stability of the network. Another study(Kim et al., 2007) discovered that the fractal network consists of skeletons and shortcuts. Also, by defining local shortcuts and global shortcuts, we discovered another structural feature of networks with fractality. That is, the distribution of the network, especially the distributed form of the most connected nodes and the pattern of the growing network, can be determined by calculating the fractal dimension. Therefore, we use the fractal dimension measurement methodology to efficiently represent the overall network structure of stocks that are highly correlated in the stock market.

## 2.2 Market Prediction with Machine Learning

Machine learning algorithms are widely used in stock market forecasting and trading strategies. The most fundamental approach is a classification algorithm that predicts the future cumulative return direction. The work of Ballings et al. (2015)

studied the prediction of performance of European stock prices direction with various machine learning algorithms such as logistic regression, support vector machines, K-nearest neighbor, random forest, adaBoost, and neural network. In the work of Fischer and Krauss (2018), long short-term memory (LSTM) network was applied to the direction prediction of the S&P500 market. This approach generally has high prediction accuracy, but has the limitation that good accuracy does not necessarily lead to high profitability. For example, the model can predict correctly at small gains, but not at large losses, which increases the risk of downside. Another approach is methodologies for predicting the price or return of a stock. In the financial time series forecasting, a variety of machine learning algorithms have been studied, from a linear regression algorithm(Lu et al., 2009) with a simple structure but strong interpretation performance to a neural network algorithm(Kaastra and Boyd, 1996) with strong predictive power compared to existing algorithms. The limitation of this research is that it is difficult to accurately predict future asset values based on confusing stock market data, resulting in somewhat less predictive power.

In the prediction experiment part of this dissertation, we used logistic regression, random forest, and artificial neural networks(ANN) since those are commonly used machine learning algorithms. We select those three machine learning algorithms to claim the effect of improving the predictive power of by using proposed fractal measures. The description of each specific machine learning model is as follows. The logistic regression(Berkson, 1944; Cramer, 2002) is used to model the probability of a certain binary class. Several extension models exist, but we use a basic logistic regression model because we aim to investigate the predictive power of fractal measures in binary classifiers (stock market direction classification). For the purposes of this

study, we used a regularized approach to logistic regression. This technique improves overall prediction performance by sacrificing some bias to reduce the variance of the predicted values. We implemented this model using a *scikit-learn 0.23(Python)* package. The random forest(Ho, 1995) is an ensemble learning method for classification that operate by constructing a multitude of decision trees. Through the ensemble technique, the limitations of robustness or suboptimal performance of the single decision tree are solved. Same as logistic regression, this model is used for the stock market direction classification problem. We implemented this model using a *scikit-learn 0.23(Python)* package. The artificial neural networks(McCulloch and Pitts, 1943) is based on a collection of connected nodes called artificial neurons. We used an artificial neural network with 2-hidden layers. The reason for using 2-hidden layer structure in the model is as follows. By universal approximation theorem(Cybenko, 1989), simple neural networks can in principle be applied to nearly any problem, as they can approximate essentially any function of interest. However, in general prediction problems, the convergence of the model is often difficult with 1-hidden layer. Conversely, this dissertation does not require a complex structure beyond the 3-hidden layer, and the neural network of the 2-hidden layer is sufficient because it aims to improve the predictability of fractal measures. We implemented this model using a *tensorflow 2.0(Python)* package.

Recently, the range of available data has been widened, and the disadvantages of low predictive power are being overcome as data. The work of Zhang et al. (2018) developed a model that can improve the prediction of stock market index movement by extracting indicators such as emotions from the web, and proposed a consistent method of organizing between various data sources. As such, due to the rapidly

evolving machine learning methodologies and the increase in the amount of data that can be accessed and processed, smarter decision-making by learning is possible. Now, modeling using machine learning is becoming increasingly essential for stock market forecasting.

## 2.3 Trading Strategies

The modern portfolio theory(Markowitz, 1952) is a methodology for evaluating portfolios with expected return and variance. Various studies that developed this have been conducted, Clarke et al. (2006) showed empirically that low-volatility portfolios can be constructed with return-based covariance matrix estimation methodologies, Blitz and Van Vliet (2007) conducted an in-depth study of these stocks with high risk-adjusted returns, which is low volatility stocks. The work of Strub and Baumann (2018) proposed a mixed-integer programming formulation that composes a portfolio that minimizes tracking errors in index funds (such as S%P500 funds). In addition, they applied real-world data to the proposed portfolio composition method and showed better performance in index fund than other studies. On the other hand, the work of Chou et al. (2017) pointed out the limitation that all coefficients of variation cannot be calculated when calculating the relationship between stocks in MPT. Therefore, in order to find the optimal portfolio, they proposed a genetic algorithm and a methodology to standardize funds. The work of Yu et al. (2020) proposed an idea to develop a portfolio model by integrating return projection, considering transaction cost and short-selling. The effectiveness of the study was verified by comparing the performance of the proposed model with other

portfolio models.

In this dissertation, we propose a strategy that focuses on the timing of rebalancing rather than the portfolio selection perspective, and the optimal rebalancing timing is calculated through a machine learning technique based on various portfolio selection methods previously studied. In other words, our model is to develop an existing model that finds the appropriate timing to apply the proposed portfolio selection algorithm. As in the performance evaluation methodology(Yu et al., 2020), we evaluated the proposed rebalancing strategy by comparing the existing portfolio selection model with rebalancing at regular periods. Therefore, we insist that the proposed strategy performs better by following the existing portfolio selection method.

# Chapter 3

# Fractal Structure in Stock Market

This chapter contains content published by Ku et al. (2020).

## 3.1 Network Fractality

### 3.1.1 Threshold Network

Let $N$ be the total number of considered stocks in the S&P500, then the Pearson correlation matrix can be defined as,

$$\rho_{ij} = \frac{\langle r_i(t) r_j(t) \rangle - \langle r_i(t) \rangle \langle r_j(t) \rangle}{\sqrt{\langle r_i(t)^2 - \langle r_i(t) \rangle^2 \rangle \langle r_j(t)^2 - \langle r_j(t) \rangle^2 \rangle}} \tag{3.1}$$

where $\langle r_i(t) \rangle$ represents the mean of $r_i(t)$ over a given period, which is set to be 48 weeks (roughly 1 year) in this experiment.

There are two main approaches to construct the threshold network. The first approach is sequential elimination of edges from the entire correlation matrix with $\rho$ to some specified threshold, whereas the second approach is the addition of edges from the skeleton to some specified threshold. Note that both approaches must

hold the condition of the connected graph to compute the fractal dimension. In this regard, the structures of both approaches are the same unless the same $\rho_{ij}$ exists simultaneously. Note that we choose the second approach to proceed with the experiment similar to the definition of fractal shape.

Specifically, we modify the planar maximally filtered graph-based threshold network (PTN) method proposed in the work of Nie and Song (2018) and construct a threshold network based on the minimum spanning tree (MTN), which can be considered as the skeleton of the financial network. Then, we grow the threshold network from the most straightforward shape, MST, to observe the changes in the fractal dimension concerning time. Our proposed method can be summarized in the following steps.

- **Step 1. Calculate the correlation coefficient matrix** $\rho = [\rho_{ij}]$.
- **Step 2. Construct the MST based on** $\rho$**.** Note that we utilize the distance between the stocks as $d_{ij} = \sqrt{2(1-\rho_{ij})}$. Therefore, the MST, $G_t^{MST} = (V, E_t^{MST})$, can be obtained by applying the Kruskal algorithm (Kruskal, 1956), where $V = \{1, ...N\}$ and $E_t^{MST}$ are the set of stocks and the $(N-1)$ non-direction edges calculated by the MST algorithm at time $t$, respectively. Then, the chosen $\rho_{i,j}$-weighted edges are replaced by un-weighted edges (0 or 1) since the fractal dimension analysis only considers the connections of vertices.
- **Step 3. Define the parameter** $\alpha$**, the threshold of the network.** Note that $\alpha$ is,

$$\alpha = \frac{|\rho_{ij} \geq \rho_\alpha|}{|\rho_{ij}|} \times 100, \quad (j > i) \tag{3.2}$$

where $|\cdot|$ and $\rho_\alpha$ are the cardinality of a given set and the correlation coefficient threshold value of the network corresponding to threshold $\alpha$, respectively. $\alpha$ refers to the growth rate of the network at each time $t$ in percentage, that is, the ratio of chosen edges to the total edges. Therefore, as $\alpha$ increases, the number of edges in the network increases.

- **Step 4. Construct the threshold network $G_{t,\alpha}$ based on $G_t^{MST}$, $\alpha$ and $\rho_\alpha$.** Given that $\alpha \in \{0.55, 0.6, ..., 1.5\}$, we construct the threshold network $G_{t,\alpha} = (V, E_{t,\alpha})$, where $V = \{1, ...N\}$ is the set of stocks, $E_{t,\alpha}$, such that,

$$E_{\substack{t,\alpha \\ j>i}}(i,j) = \begin{cases} 1 & \text{if } E_t^{MST}(i,j) = 1 \quad or \quad \rho_{ij} \geq \rho_\alpha \\ 0 & \text{otherwise} \end{cases} \tag{3.3}$$

In summary, the threshold network, $G_{t,\alpha}$, is an unweighted undirected graph connecting stock nodes, $i, j \in V, j > i$, with a correlation coefficient greater than $\rho_\alpha$ whose skeleton is its MST.

### 3.1.2 Fractal Dimension

Let us recall the definition of the box covering algorithm proposed in the work of Bunde and Havlin (2013). For a given network, $G$, and box size, $l_B$, a box is a set of nodes where all distances, $l_{ij}$, between any two nodes $i$ and $j$ in the box are smaller than $l_B$. The minimum number of boxes required to cover network $G$ is denoted by $N_B$. Once the $N_B$ values corresponding to all $l_B$ are obtained, the fractal dimension, $d_B$, of network $G$ can be computed as $N_B \sim l_B^{-d_B}$.

Then, the maximum-excluded-mass-burning (MEMB) algorithm(Song et al., 2007) is used to measure the minimum number of boxes $N_B$. We define the excluded mass of a node as the number of uncovered nodes within a distance less than $r_B$. Note that the uncovered node refers to nodes that are not included in the box while the algorithm is in progress. The step-by-step procedure of the MEMB algorithm can be summarized in five steps as follows:

- **Step 1. Initially, all the nodes are marked as uncovered and non-centers.**
- **Step 2. Calculate excluded mass for all non-center nodes.** If the covered node is a non-center node, then the node also computes the excluded mass.
- **Step 3. Select the node with the maximum excluded mass and mark it as a central node.** This node is called C (if there is a node with the same excluded mass, select any one node as C).
- **Step 4. Mark all nodes with distance less than $r_B$ from C node as covered nodes.**
- **Step 5. Repeat Steps 2 to 4 until all nodes are covered or centers.**

For more details, we provide an example of MEMB algorithm for a case of $r_B = 1$ that yields $N_B = 2$ in Figure 3.1. Even though the MEMB algorithm has a subsequent coloring process, it is sufficient to only measure the minimum number of boxes, $N_B$, since the purpose of this research is to measure the fractal dimension, $d_B$. Specifically, the burning radius, $r_B$, is used instead of the $l_B$ where $l_B = 2r_B + 1$. Then, the fractal dimension can be calculated by applying the ordinary least squares

(OLS) regression for $d_B$ in $log(N_B) \sim -d_B log(l_B)$ where $N_B$ exists for each $l_B$.

To evaluate the goodness of fit of the estimated $d_B$, we calculate the OLS mean squared error (OLS-MSE) of the relation $log(N_B) \sim -d_B log(l_B)$ for all constructed networks. The equation for calculating OLS-MSE is as follows.

$$FD_{t,\alpha}^{Fit} = \frac{1}{|All\ of\ l_B\ in\ G_{t,\alpha}|} \sum_{l_B \in \{3,5,7...max(l_B)\}}^{G_{t,\alpha}} (log(N_B) - (-d_B)log(l_B))^2 \quad (3.4)$$

The goodness of fit in S&P500 network is evaluated based on the OLS-MSEs of the random network. Specifically, the random network is constructed by adding random edges with the same number of edges as $G_{t,\alpha}$, based on the MST, $G_t^{MST}$ with a randomly shuffled correlation matrix, corresponding to each time point $t$. If the OLS-MSE of the financial network is significantly smaller than that of the random network based on paired $t$-test, it is reasonable to state that the high-correlation network of the S&P500 market exhibits a fractal structure.

### 3.1.3 Fractal Measures

In Section 3.1.1, we suggest to construct a threshold network based on its correlation-based MST, $G_{t,\alpha}$, for the threshold $\alpha \in \{0.55, 0.6, ..., 1.5\}$. By calculating the minimum number of boxes, $N_B$, covering $G_{t,\alpha}$ for each $l_B$ using the MEMB algorithm for the configured network $G_{t,\alpha}$, we can compute all of $d_B$ in relation of $N_B \sim l_B^{-d_B}$ for parameters $t$ and $\alpha$. Note that we let $FD_{t,\alpha}$ denotes $d_B$ for parameters $t$ and $\alpha$. Using the computed fractal dimensions, $FD_{t,\alpha}$, we define three measures that can represent the following network structures.

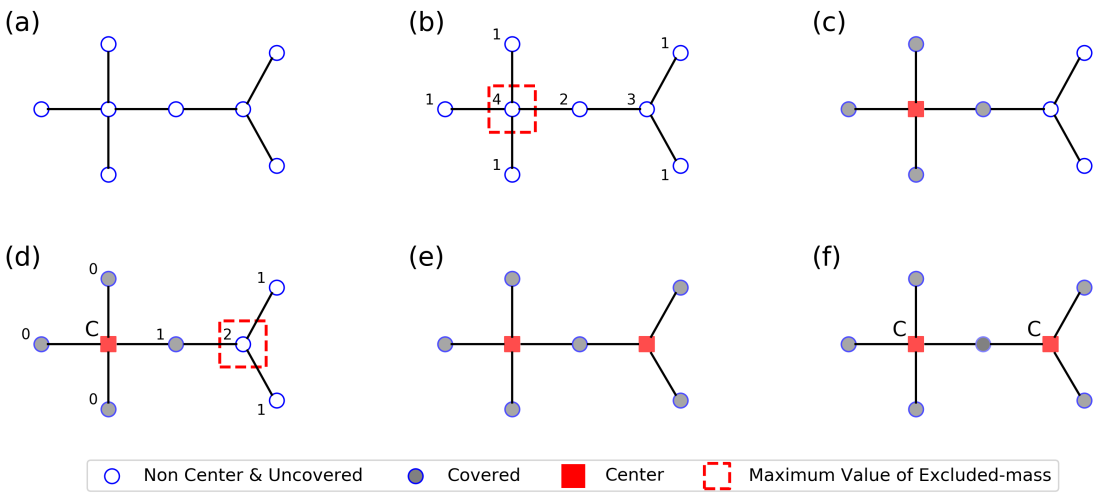


Figure 3.1: **Example of MEMB algorithm** ($r_B = 1$)

(a) Realize the network, (b) Calculate the excluded mass for all nodes, (c) Set the node with maximum mass (max = 4) as center node C and mark the nodes within $r_B = 1$ as covered nodes, (d) Re-calculate the excluded mass for non-center nodes, (e) Set the node with maximum mass (max = 2) as center node C and mark the nodes within $r_B = 1$ as covered nodes, (f) Terminate the algorithm if all nodes are covered.

At first, the core principle that leads to the fractal structure of the network is a strong effective repulsion (disassortativity) phenomenon between the hubs on all length scales where the hub is defined as the most connected nodes(Song et al., 2006). In this regard, we calculate the mean value of the fractal dimension of the threshold network at each point to measure the degree of repulsion. Note that the mean value of the fractal dimension, $FD_t^{Mean}$, can be defined as,

$$FD_t^{Mean} = \frac{1}{|\alpha|} \sum_{\alpha} FD_{t,\alpha} \tag{3.5}$$

where the repulsion phenomenon can be observed by comparing $FD_t^{Mean}$ with the average distance of the top 10% nodes among all nodes constituting the S&P500 network. We attempt to ensure robust results by analyzing five different cases of hubs.

Secondly, a fractal network consists of a skeleton and shortcuts(Li et al., 2017). Since the threshold network defined in this study possesses the MST as a skeleton with added edges, it is possible to analyze whether the growth pattern of edges is a global- or local-shortcut.

In this context, we calculate the standard deviation of the fractal dimensions to measure the degree of variations for different thresholds at a fixed point. Note that the standard deviation of fractal dimension can be defined as,

$$FD_t^{Std} = \sqrt{\frac{1}{|\alpha|} \sum_{\alpha} (FD_{t,\alpha} - FD_t^{Mean})^2}. \tag{3.6}$$

Lastly, it is well-known that a robust network containing functional modules

entails a fractal topology(Song et al., 2006). Therefore, a break of the fractal topology can cause a significant change in the financial network. In this sense, we define a measure for the goodness of fit of the threshold network at a fixed point based on the average of Eq. (3.4) such that,

$$FD_t^{Fit} = \frac{1}{|\alpha|} \sum_{\alpha} FD_{t,\alpha}^{Fit}. \tag{3.7}$$

## 3.2 Fractal Analysis on Stock Market

### 3.2.1 Data Description

In this study, we use weekly closing prices of Standard & Poor's 500 (S&P500) to measure the fractality of the U.S. financial network. The price series are obtained from Thomson Reuters Datastream. The experimental period is from 1999-01-01 to 2019-12-31, a total of 1056 weeks, which yields 373 stocks whose time-series is fully available within the period (see Table 3.1). For modelling, the price series of stock $i$ at time $t$, $P_i(t)$, is transformed to the logarithmic return series, $r_i(t) = \ln(P_i(t)) - \ln(P_i(t-1))$. Furthermore, we utilize weekly log-return series of S&P500 index and associated stocks for all experiments, including the construction of the threshold network, computation of fractal dimension, and prediction of the cumulative returns of the S&P500 index. We choose weekly rather than daily log-returns, which has more volatile movements, to capture major structural changes of the U.S. financial network. Note that the utilization of weekly log-return series provides an advantage in incorporating the overall trend by smoothing the extreme daily returns of individual stocks.

Table 3.1: **List of selected stocks in alphabetical order**

| Symbol | Company | Symbol | Company |
|---|---|---|---|
| AAPL | Apple Inc. | KLAC | KLA Corporation |
| ABC | AmerisourceBergen Corp | KMB | Kimberly-Clark |
| ABMD | ABIOMED Inc | KMX | Carmax Inc |
| ABT | Abbott Laboratories | KO | Coca-Cola Company |
| ADBE | Adobe Inc. | KR | Kroger Co. |
| ADI | Analog Devices, Inc. | KSS | Kohl's Corp. |
| ADM | Archer-Daniels-Midland Co | KSU | Kansas City Southern |
| ADP | Automatic Data Processing | L | Loews Corp. |
| ADSK | Autodesk Inc. | LB | L Brands Inc. |
| AEE | Ameren Corp | LEG | Leggett Platt |

| | |
|---|---|
| AEP | American Electric Power |
| AES | AES Corp |
| AFL | AFLAC Inc |
| AGN | Allergan |
| AIG | American International Group |
| AIV | Apartment Investment Management |
| AJG | Arthur J. Gallagher Co. |
| ALB | Albemarle Corp |
| ALK | Alaska Air Group Inc |
| ALL | Allstate Corp |
| ALXN | Alexion Pharmaceuticals |
| AMAT | Applied Materials Inc. |
| AMD | Advanced Micro Devices Inc |
| AME | AMETEK Inc. |
| AMGN | Amgen Inc. |
| AMT | American Tower Corp. |
| AMZN | Amazon.com Inc. |
| ANSS | ANSYS |
| AON | Aon plc |
| AOS | A.O. Smith Corp |
| APA | Apache Corporation |
| APD | Air Products Chemicals Inc |
| APH | Amphenol Corp |
| ARE | Alexandria Real Estate Equities |
| ARNC | Arconic Corporation |
| ATO | Atmos Energy |
| ATVI | Activision Blizzard |
| AVB | AvalonBay Communities, Inc. |
| AVY | Avery Dennison Corp |
| AXP | American Express Co |
| AZO | AutoZone Inc |
| BA | Boeing Company |
| BAC | Bank of America Corp |
| BAX | Baxter International Inc. |
| BBY | Best Buy Co. Inc. |
| BDX | Becton Dickinson |
| BEN | Franklin Resources |
| BF.B | Brown-Forman Corp. |
| BIIB | Biogen Inc. |
| BK | The Bank of New York Mellon Corp. |
| BKR | Baker Hughes Co |
| BLL | Ball Corp |
| BMY | Bristol-Myers Squibb |
| BRK.B | Berkshire Hathaway |
| BSX | Boston Scientific |
| BWA | BorgWarner |
| BXP | Boston Properties |
| C | Citigroup Inc. |
| CAG | Conagra Brands |
| CAH | Cardinal Health Inc. |
| CAT | Caterpillar Inc. |
| CB | Chubb Limited |
| CCI | Crown Castle International Corp. |
| CCL | Carnival Corp. |
| CDNS | Cadence Design Systems |
| CERN | Cerner |
| CHD | Church Dwight |
| CHRW | C. H. Robinson Worldwide |
| LEN | Lennar Corp. |
| LH | Laboratory Corp. of America Holding |
| LHX | L3Harris Technologies |
| LIN | Linde plc |
| LLY | Lilly (Eli) Co. |
| LMT | Lockheed Martin Corp. |
| LNC | Lincoln National |
| LNT | Alliant Energy Corp |
| LOW | Lowe's Cos. |
| LRCX | Lam Research |
| LUV | Southwest Airlines |
| M | Macy's, Inc. |
| MAA | Mid-America Apartments |
| MAR | Marriott Int'l. |
| MAS | Masco Corp. |
| MCD | McDonald's Corp. |
| MCHP | Microchip Technology |
| MCK | McKesson Corp. |
| MCO | Moody's Corp. |
| MDT | Medtronic plc |
| MGM | MGM Resorts International |
| MHK | Mohawk Industries |
| MKC | McCormick Co. |
| MLM | Martin Marietta Materials |
| MMC | Marsh McLennan |
| MMM | 3M Company |
| MNST | Monster Beverage |
| MO | Altria Group Inc |
| MOS | The Mosaic Company |
| MRK | Merck Co. |
| MRO | Marathon Oil Corp. |
| MS | Morgan Stanley |
| MSFT | Microsoft Corp. |
| MSI | Motorola Solutions Inc. |
| MTB | MT Bank Corp. |
| MTD | Mettler Toledo |
| MU | Micron Technology |
| MXIM | Maxim Integrated Products Inc |
| MYL | Mylan N.V. |
| NBL | Noble Energy Inc |
| NEE | NextEra Energy |
| NEM | Newmont Corporation |
| NI | NiSource Inc. |
| NKE | Nike |
| NLOK | NortonLifeLock |
| NOC | Northrop Grumman |
| NOV | National Oilwell Varco Inc. |
| NSC | Norfolk Southern Corp. |
| NTAP | NetApp |
| NTRS | Northern Trust Corp. |
| NUE | Nucor Corp. |
| NVR | NVR Inc |
| NWL | Newell Brands |
| O | Realty Income Corporation |
| ODFL | Old Dominion Freight Line |
| OKE | ONEOK |
| OMC | Omnicom Group |
| ORCL | Oracle Corp. |

| | |
|---|---|
| CI | CIGNA Corp. |
| CINF | Cincinnati Financial |
| CL | Colgate-Palmolive |
| CLX | The Clorox Company |
| CMA | Comerica Inc. |
| CMCSA | Comcast Corp. |
| CMI | Cummins Inc. |
| CMS | CMS Energy |
| CNP | CenterPoint Energy |
| COF | Capital One Financial |
| COG | Cabot Oil Gas |
| COO | The Cooper Companies |
| COP | ConocoPhillips |
| COST | Costco Wholesale Corp. |
| CPB | Campbell Soup |
| CPRT | Copart Inc |
| CSCO | Cisco Systems |
| CSX | CSX Corp. |
| CTAS | Cintas Corporation |
| CTL | CenturyLink Inc |
| CTSH | Cognizant Technology Solutions |
| CTXS | Citrix Systems |
| CVS | CVS Health |
| CVX | Chevron Corp. |
| D | Dominion Energy |
| DE | Deere Co. |
| DGX | Quest Diagnostics |
| DHI | D. R. Horton |
| DHR | Danaher Corp. |
| DIS | The Walt Disney Company |
| DISH | Dish Network |
| DLTR | Dollar Tree |
| DOV | Dover Corporation |
| DRE | Duke Realty Corp |
| DRI | Darden Restaurants |
| DTE | DTE Energy Co. |
| DUK | Duke Energy |
| DVA | DaVita Inc. |
| DVN | Devon Energy |
| DXC | DXC Technology |
| EA | Electronic Arts |
| EBAY | eBay Inc. |
| ECL | Ecolab Inc. |
| ED | Consolidated Edison |
| EFX | Equifax Inc. |
| EIX | Edison Int'l |
| EL | Estée Lauder Companies |
| EMN | Eastman Chemical |
| EMR | Emerson Electric Company |
| EOG | EOG Resources |
| EQR | Equity Residential |
| ES | Eversource Energy |
| ESS | Essex Property Trust, Inc. |
| ETFC | E*Trade |
| ETN | Eaton Corporation |
| ETR | Entergy Corp. |
| EVRG | Evergy |
| EXC | Exelon Corp. |
| ORLY | O'Reilly Automotive |
| OXY | Occidental Petroleum |
| PAYX | Paychex Inc. |
| PBCT | People's United Financial |
| PCAR | PACCAR Inc. |
| PEAK | Healthpeak Properties |
| PEG | Public Serv. Enterprise Inc. |
| PEP | PepsiCo Inc. |
| PFE | Pfizer Inc. |
| PG | Procter Gamble |
| PGR | Progressive Corp. |
| PH | Parker-Hannifin |
| PHM | PulteGroup |
| PKI | PerkinElmer |
| PLD | Prologis |
| PNC | PNC Financial Services |
| PNR | Pentair plc |
| PNW | Pinnacle West Capital |
| PPG | PPG Industries |
| PPL | PPL Corp. |
| PRGO | Perrigo |
| PSA | Public Storage |
| PVH | PVH Corp. |
| PWR | Quanta Services Inc. |
| PXD | Pioneer Natural Resources |
| QCOM | QUALCOMM Inc. |
| QRVO | Qorvo |
| RCL | Royal Caribbean Cruises Ltd |
| RE | Everest Re Group Ltd. |
| REG | Regency Centers Corporation |
| REGN | Regeneron Pharmaceuticals |
| RF | Regions Financial Corp. |
| RHI | Robert Half International |
| RJF | Raymond James Financial Inc. |
| RL | Ralph Lauren Corporation |
| RMD | ResMed |
| ROK | Rockwell Automation Inc. |
| ROL | Rollins Inc. |
| ROP | Roper Technologies |
| ROST | Ross Stores |
| RSG | Republic Services Inc |
| RTX | Raytheon Technologies |
| SBUX | Starbucks Corp. |
| SCHW | Charles Schwab Corporation |
| SEE | Sealed Air |
| SHW | Sherwin-Williams |
| SIVB | SVB Financial |
| SJM | JM Smucker |
| SLB | Schlumberger Ltd. |
| SLG | SL Green Realty |
| SNA | Snap-on |
| SNPS | Synopsys Inc. |
| SO | Southern Company |
| SPG | Simon Property Group Inc |
| SPGI | SP Global, Inc. |
| SRE | Sempra Energy |
| STE | STERIS plc |
| STT | State Street Corp. |

| | |
|---|---|
| EXPD | Expeditors |
| F | Ford Motor Company |
| FAST | Fastenal Co |
| FCX | Freeport-McMoRan Inc. |
| FDX | FedEx Corporation |
| FE | FirstEnergy Corp |
| FISV | Fiserv Inc |
| FITB | Fifth Third Bancorp |
| FLIR | FLIR Systems |
| FLS | Flowserve Corporation |
| FMC | FMC Corporation |
| FRT | Federal Realty Investment Trust |
| GD | General Dynamics |
| GE | General Electric |
| GILD | Gilead Sciences |
| GIS | General Mills |
| GL | Globe Life Inc. |
| GLW | Corning Inc. |
| GPC | Genuine Parts |
| GPS | Gap Inc. |
| GWW | Grainger (W.W.) Inc. |
| HAL | Halliburton Co. |
| HAS | Hasbro Inc. |
| HBAN | Huntington Bancshares |
| HD | Home Depot |
| HES | Hess Corporation |
| HFC | HollyFrontier Corp |
| HIG | Hartford Financial Svc.Gp. |
| HOG | Harley-Davidson |
| HOLX | Hologic |
| HON | Honeywell Int'l Inc. |
| HP | Helmerich Payne |
| HPQ | HP Inc. |
| HRB | HR Block |
| HRL | Hormel Foods Corp. |
| HSIC | Henry Schein |
| HST | Host Hotels Resorts |
| HSY | The Hershey Company |
| HUM | Humana Inc. |
| IBM | International Business Machines |
| IDXX | IDEXX Laboratories |
| IEX | IDEX Corporation |
| IFF | Intl Flavors Fragrances |
| INCY | Incyte |
| INTC | Intel Corp. |
| INTU | Intuit Inc. |
| IP | International Paper |
| IPG | Interpublic Group |
| IRM | Iron Mountain Incorporated |
| IT | Gartner Inc |
| ITW | Illinois Tool Works |
| J | Jacobs Engineering Group |
| JBHT | J. B. Hunt Transport Services |
| JCI | Johnson Controls International |
| JKHY | Jack Henry Associates |
| JNJ | Johnson Johnson |
| JPM | JPMorgan Chase Co. |
| JWN | Nordstrom |
| STZ | Constellation Brands |
| SWK | Stanley Black Decker |
| SWKS | Skyworks Solutions |
| SYK | Stryker Corp. |
| SYY | Sysco Corp. |
| T | ATT Inc. |
| TAP | Molson Coors Brewing Company |
| TFC | Truist Financial |
| TFX | Teleflex |
| TGT | Target Corp. |
| TIF | Tiffany Co. |
| TJX | TJX Companies Inc. |
| TMO | Thermo Fisher Scientific |
| TROW | T. Rowe Price Group |
| TRV | The Travelers Companies Inc. |
| TSCO | Tractor Supply Company |
| TSN | Tyson Foods |
| TT | Trane Technologies plc |
| TTWO | Take-Two Interactive |
| TXN | Texas Instruments |
| TXT | Textron Inc. |
| UDR | UDR, Inc. |
| UHS | Universal Health Services, Inc. |
| UNH | United Health Group Inc. |
| UNM | Unum Group |
| UNP | Union Pacific Corp |
| URI | United Rentals, Inc. |
| USB | U.S. Bancorp |
| UTX | United Technologies |
| VAR | Varian Medical Systems |
| VFC | V.F. Corp. |
| VIAC | ViacomCBS |
| VLO | Valero Energy |
| VMC | Vulcan Materials |
| VNO | Vornado Realty Trust |
| VRSN | Verisign Inc. |
| VRTX | Vertex Pharmaceuticals Inc |
| VTR | Ventas Inc |
| VZ | Verizon Communications |
| WAB | Wabtec Corporation |
| WAT | Waters Corporation |
| WBA | Walgreens Boots Alliance |
| WDC | Western Digital |
| WEC | WEC Energy Group |
| WELL | Welltower Inc. |
| WFC | Wells Fargo |
| WHR | Whirlpool Corp. |
| WM | Waste Management Inc. |
| WMB | Williams Cos. |
| WMT | Walmart |
| WRB | W. R. Berkley Corporation |
| WY | Weyerhaeuser |
| XEL | Xcel Energy Inc |
| XLNX | Xilinx |
| XOM | Exxon Mobil Corp. |
| XRAY | Dentsply Sirona |
| XRX | Xerox |
| YUM | Yum! Brands Inc |

| | | | |
|---|---|---|---|
| K | Kellogg Co. | ZBRA | Zebra Technologies |
| KEY | KeyCorp | ZION | Zions Bancorp |
| KIM | Kimco Realty | | |

### 3.2.2 Fractality of S&P500 Network

Table 3.2 summarizes the mean and standard deviation of OLS-MSE based on 1008 data points of $d_B$ in $N_B \sim l_B^{-d_B}$ for different S&P500 threshold networks. Note that the random networks based on $\alpha$ in Eq. (3.2) are provided. At first, the mean and standard deviation of MST are 0.1844 and 0.0585, respectively, whose values are the smallest among the threshold networks. Interestingly, the threshold networks whose OLS-MSE values are less than 1.35 show statistical significances in 1% in paired t-tests against the random network simultaneously. That is, the S&P500 networks whose thresholds are less than 1.35 exhibit a strong network fractality.

In contrast, the S&P500 networks whose thresholds are larger than 1.35 show similar or even less-fitted OLS-MSE than those of corresponding random networks. A possible cause is the shrink of the network for added edges. That is, $N_B$ corresponding to $l_B$ rapidly decays during the box covering algorithm due to its relatively small diameter. Therefore, we focus on the structure of S&P500 networks whose thresholds are between the 0.55 to 1.35.

Fig. 3.2-(a) plots the evolution of fractal dimension, $d_B$, for different S&P500 networks whose thresholds are less than or equal to 1.35. Specifically, the black line and gray region in Fig. 3.2-(b) refer to the mean of Fractal dimension $d_B$ and

Table 3.2: **Mean and standard deviation of the OLS-MSEs of $d_B$**

| OLS-MSE | S&P500 | | Random Network | |
|---|---|---|---|---|
| Threshold | mean | stdev | mean | stdev |
| MST(0.53) | 0.1844 | 0.0585 | - | - |
| 0.55 | 0.1707 | 0.0542 | 0.2512*** | 0.0535 |
| 0.60 | 0.1729 | 0.0572 | 0.3699*** | 0.0542 |
| 0.65 | 0.1750 | 0.0582 | 0.4078*** | 0.0560 |
| 0.70 | 0.1746 | 0.0598 | 0.4204*** | 0.0564 |
| 0.75 | 0.1744 | 0.0608 | 0.4132*** | 0.0569 |
| 0.80 | 0.1741 | 0.0612 | 0.4056*** | 0.0557 |
| 0.85 | 0.1739 | 0.0620 | 0.3883*** | 0.0603 |
| 0.90 | 0.1732 | 0.0623 | 0.3721*** | 0.0610 |
| 0.95 | 0.1743 | 0.0642 | 0.3822*** | 0.0760 |
| 1.00 | 0.1722 | 0.0648 | 0.3738*** | 0.0693 |
| 1.05 | 0.1695 | 0.0626 | 0.3306*** | 0.0481 |
| 1.10 | 0.1677 | 0.0630 | 0.2924*** | 0.0508 |
| 1.15 | 0.1665 | 0.0646 | 0.3074*** | 0.0956 |
| 1.20 | 0.1664 | 0.0654 | 0.3200*** | 0.0913 |
| 1.25 | 0.1657 | 0.0668 | 0.3289*** | 0.0771 |
| 1.30 | 0.1627 | 0.0650 | 0.2821*** | 0.0566 |
| 1.35 | 0.1616 | 0.0655 | 0.2378*** | 0.0532 |
| 1.40 | 0.1609 | 0.0661 | 0.1906 | 0.0506 |
| 1.45 | 0.1595 | 0.0674 | 0.1651 | 0.0447 |
| 1.50 | 0.1585 | 0.0687 | 0.1449 | 0.0426 |

Note: the star superscript *** indicates 1% significance level in paired t-test

time-varying ranges of min and max values of $d_B$, respectively.

Table 3.3 summarizes the mean and standard deviation of the size of the fractal dimension in each threshold. Obviously, the mean of fractal dimension increases as the threshold increases with mild increments of corresponding standard deviation.

Table 3.3: **Mean and standard deviation of fractal dimension in S&P500 networks for different thresholds**

| Threshold | Mean | Stdev |
|---|---|---|
| 0.55 | 1.7313 | 0.3703 |
| 0.60 | 1.7859 | 0.3944 |
| 0.65 | 1.8344 | 0.4179 |
| 0.70 | 1.8732 | 0.4354 |
| 0.75 | 1.9082 | 0.4494 |
| 0.80 | 1.9461 | 0.4639 |
| 0.85 | 1.9822 | 0.4769 |
| 0.90 | 2.0199 | 0.4883 |
| 0.95 | 2.0559 | 0.4961 |
| 1.00 | 2.0898 | 0.5032 |
| 1.05 | 2.1230 | 0.5074 |
| 1.10 | 2.1528 | 0.5134 |
| 1.15 | 2.1893 | 0.5185 |
| 1.20 | 2.2199 | 0.5236 |
| 1.25 | 2.2479 | 0.5274 |
| 1.30 | 2.2794 | 0.5295 |
| 1.35 | 2.3015 | 0.5327 |

### 3.2.3 Network Topology and Fractal Measures

In addition to the simple descriptive statistics on the fractal dimension in Table 3.3, we focus on the amount of change in the size of fractal dimension according to the increased threshold (i.e., number of edges) at each time point. Note that such

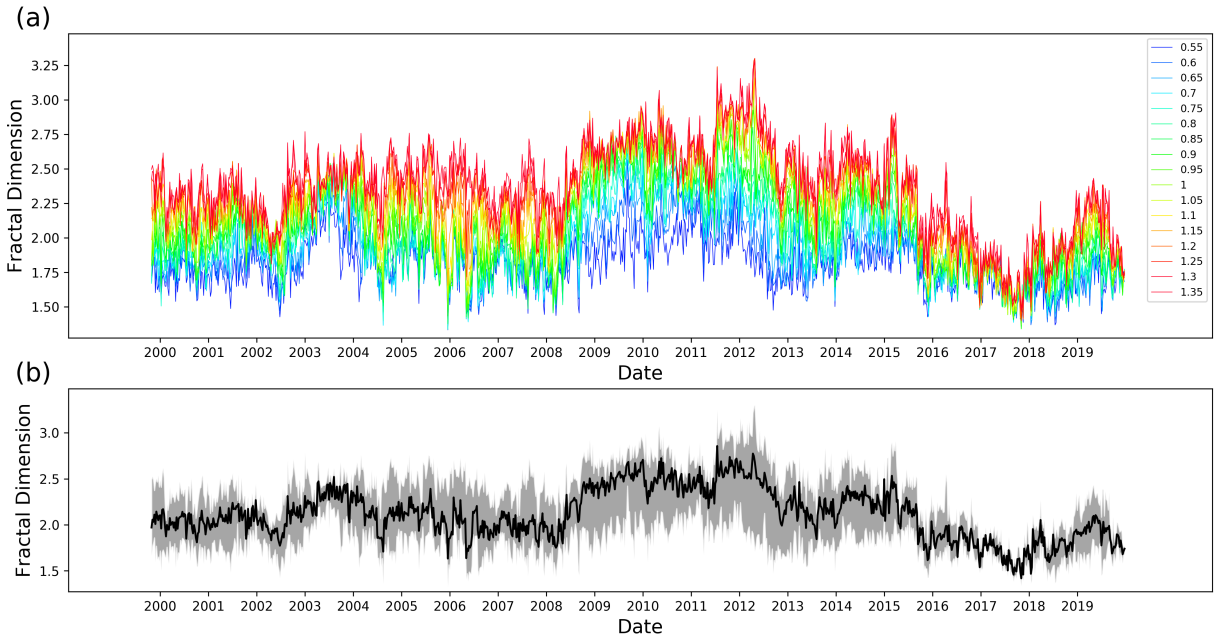


Figure 3.2: **Evolution of fractal dimension**

(a) Each threshold network, (b) Average (Black line) and min-max band (Grey region)

a change can be intuitively observed in the gray region in Fig. 3.2-(b). Therefore, in this section, we further analyze the fractal dimension based on the measures defined in Section 3.1.3.

Fig. 3.3 depicts the time-varying evolutions of $FD_t^{Mean}$ in Eq. (3.5), $FD_t^{Std}$ in Eq. (3.6), and $FD_t^{Fit}$ in Eq. (3.7). In each graph, the black and gray dotted lines indicate the mean and the range of one standard deviation, respectively. Then, the regions whose values are above or below the range of one standard deviation are filled with red and blue colors, respectively. Furthermore, we consider the values of $FD_t^{Mean}$ and $FD_t^{Std}$ simultaneously to spot four distinctive points. The green dotted lines refer to the cases of mixed sets based on large and small $FD_t^{Mean}$ and $FD_t^{Std}$ where the green circles indicate their values. Note that each point is obtained by comparing the sums and differences based on the standardized values (i.e. z-scores) of $FD_t^{Mean}$ and $FD_t^{Std}$. For instance, the green dotted line in Fig. 3.3 for the case of large mean and large standard deviation indicates the location of the largest value of standardized $FD_t^{Mean}$ plus standardized $FD_t^{Std}$, whereas that for the case of large mean and small standard deviation indicates the location of the largest value of standardized $FD_t^{Mean}$ minus standardized $FD_t^{Std}$.

Fig. 3.3-(a) shows that the mean and standard deviation of $FD_t^{Mean}$ are 2.1165 and 0.2679, respectively. In most of the intervals, $FD_t^{Mean}$ is located within one standard deviation around the mean, while some intervals show a consistent large or small values. Interestingly, Table 3.3 shows that the fractal dimension tends to increase as the threshold increases. Furthermore, Fig 3.2-(a) shows that the fractal dimension at each $t$ tends to increase as threshold increases. In this context, a

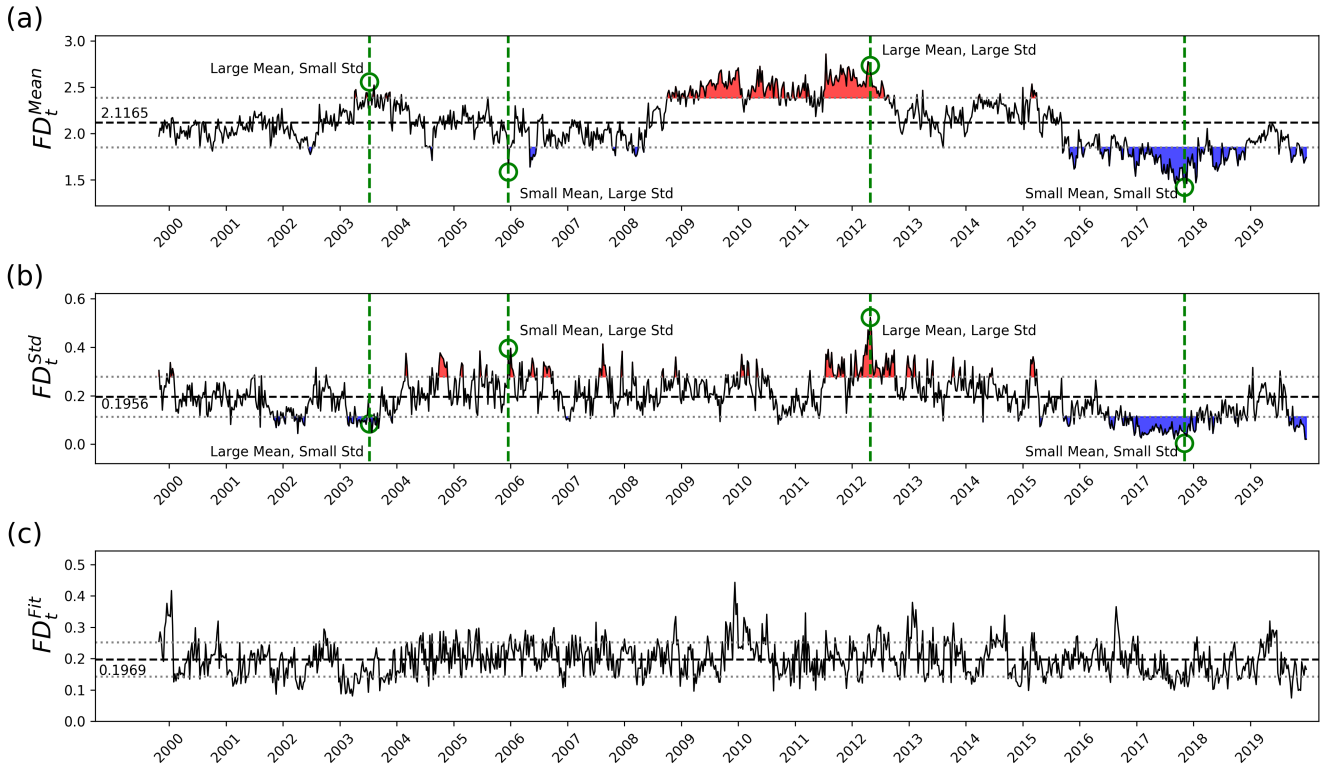


Figure 3.3: **Evolution of fractal measures**
(a) $FD_t^{Mean}$, (b) $FD_t^{Std}$, (c) $FD_t^{Fit}$

small value of $FD_t^{Mean}$ indicates that the fractal dimensions for various thresholds at time $t$ are generally small, which also implies a small absolute value of the slope of the relation $N_B \sim l_B^{-d_B}$. That is, the decrease of $N_B$ is relatively slower than the increase of $l_B$. Considering that the box's central node is composed of few hubs, we can interpret that the hubs are located far from one another. In case of the stock market, such network is difficult to transfer the shock to another sector or affect the entire network when a positive or negative shock occurs in one sector, which implies a strong effective repulsion phenomenon(Song et al., 2006). The corresponding intervals are December-2005, April-2006 ,from mid-2016 to early-2018 and late-2019.

In contrast, the large values of the $FD_t^{Mean}$ indicate a network that can reach from a hub to another hub in relatively close distance. It implies that all stocks in the network are closely located. Therefore, such a network can easily propagate a positive or negative shock from one sector to the entire network. That is, such a network is either vulnerable to a crisis or solid enough to lead a strong bull market. The corresponding intervals are July-2003 and from early-2009 to mid-2012.

Fig. 3.4 shows that sample network configurations for small values of $FD_t^{Mean}$, whereas Fig. 3.5 shows that for large values of $FD_t^{Mean}$ in different thresholds. The networks are grown as the threshold increases from 0.53(MST) to 1.3, where the size of the node is set in proportion to its degrees. Note that the green nodes indicate hubs (nodes with top 10% degrees). As previously mentioned, the apart sets of hubs are observed in the small values of $FD_t^{Mean}$, whereas the hubs are relatively evenly distributed in the network in the large values of $FD_t^{Mean}$. Note that the interpretation of the fractal dimension is not directly related to the simultaneous

condition of the market; the fractal structure can be considered whether the network is structurally robust or vulnerable to shocks, but shocks may not occur at that time. However, measuring these intrinsic properties can be helpful in diagnosing and managing the risks in the stock market.

Fig. 3.3-(b) shows that the mean and standard deviation of $FD_t^{Std}$ are 0.1956 and 0.0827, respectively. At first, the fractal network is composed of the skeleton and shortcuts Li et al. (2017). To examine such property in proposed threshold networks of S&P500 market, we closely examine the network structures of the intervals filled with a consistent small (blue) and large (red) values of $FD_t^{Std}$ in Fig. 3.6, 3.7. Note that the visualizations of networks are analogous to that of Fig. 3.4, 3.5. The black edges are the edges of MST, whereas blue edges are the added edges for increased thresholds.

Fig. 3.6 shows that sample network configurations for small values of $FD_t^{Std}$ in different thresholds. The trend of the added edges shows a local growth pattern (local shortcut) where the connectedness becomes stronger only among the related nodes in clusters. Fig. 3.7 shows that sample network configurations for large values of $FD_t^{Std}$ in different thresholds. The trend of the added edges shows a dispersed growth pattern (global shortcut) where the clusters become connected much faster than those of Fig. 3.6. Note that observing the shortcuts of a network aims to identify the source of the change in its diameter or the connection between clusters, which refers to sectors in the financial market. Thus, these features can be utilized to measure the degree of dispersion of stocks.

$FD_t^{Mean}$ and $FD_t^{Std}$ can be further analyzed in Fig. 3.8, which displays the

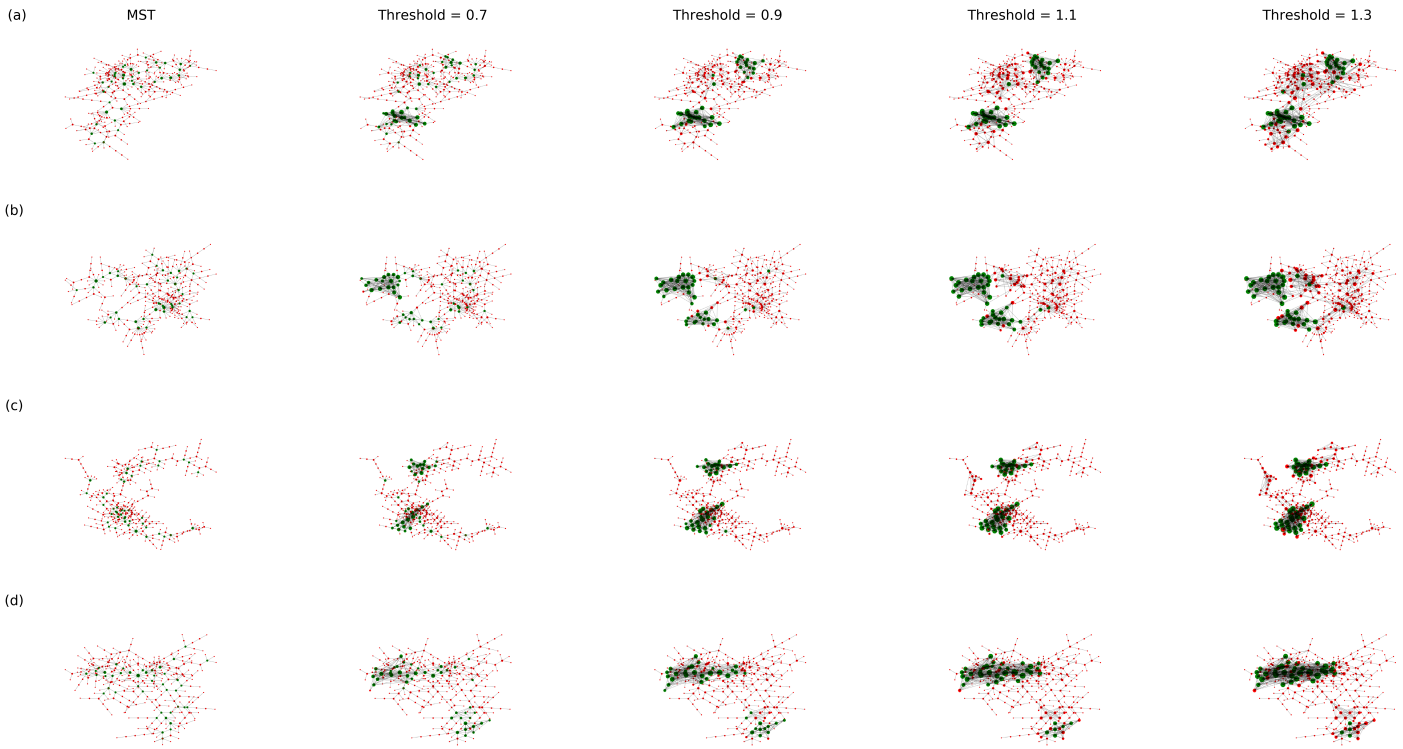


Figure 3.4: **Snapshots of network configuration for small $FD_t^{Mean}$**

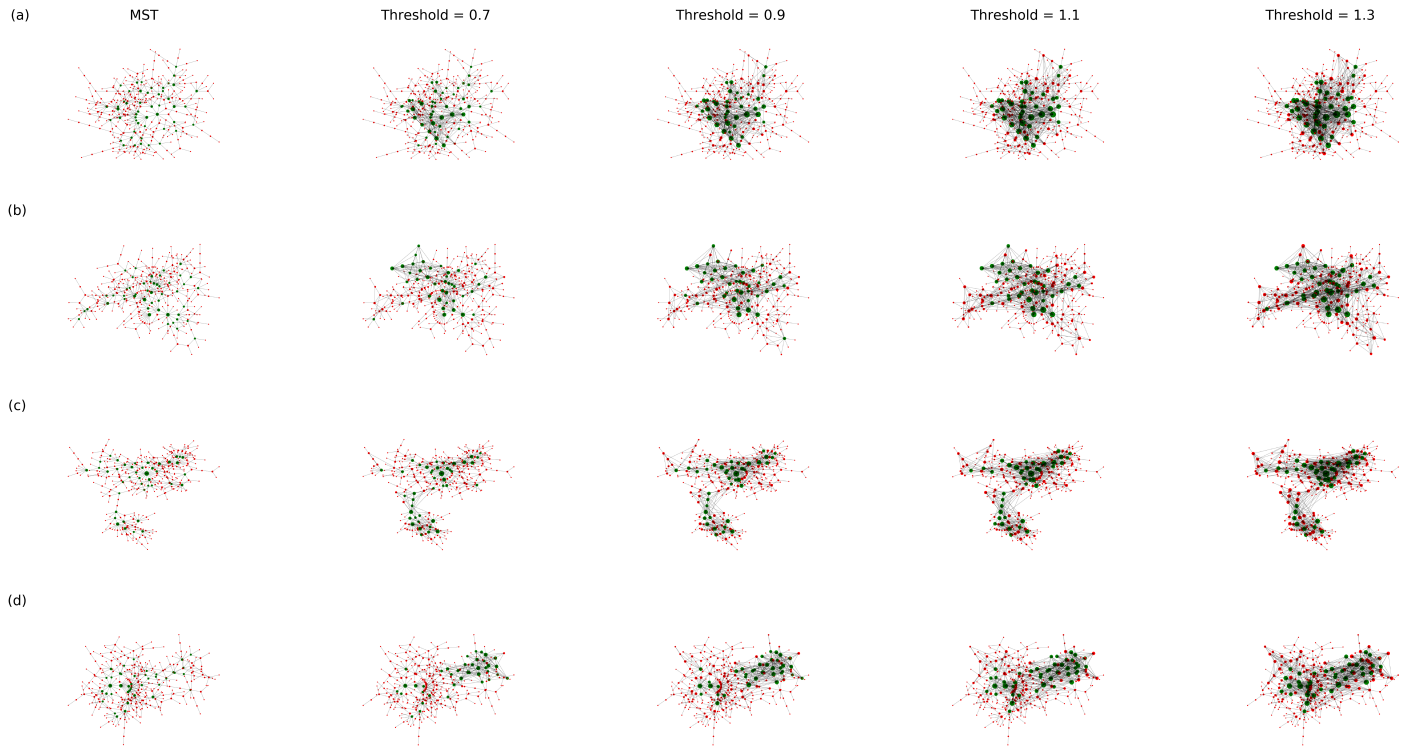


Figure 3.5: **Snapshots of network configuration for large** $FD_t^{Mean}$

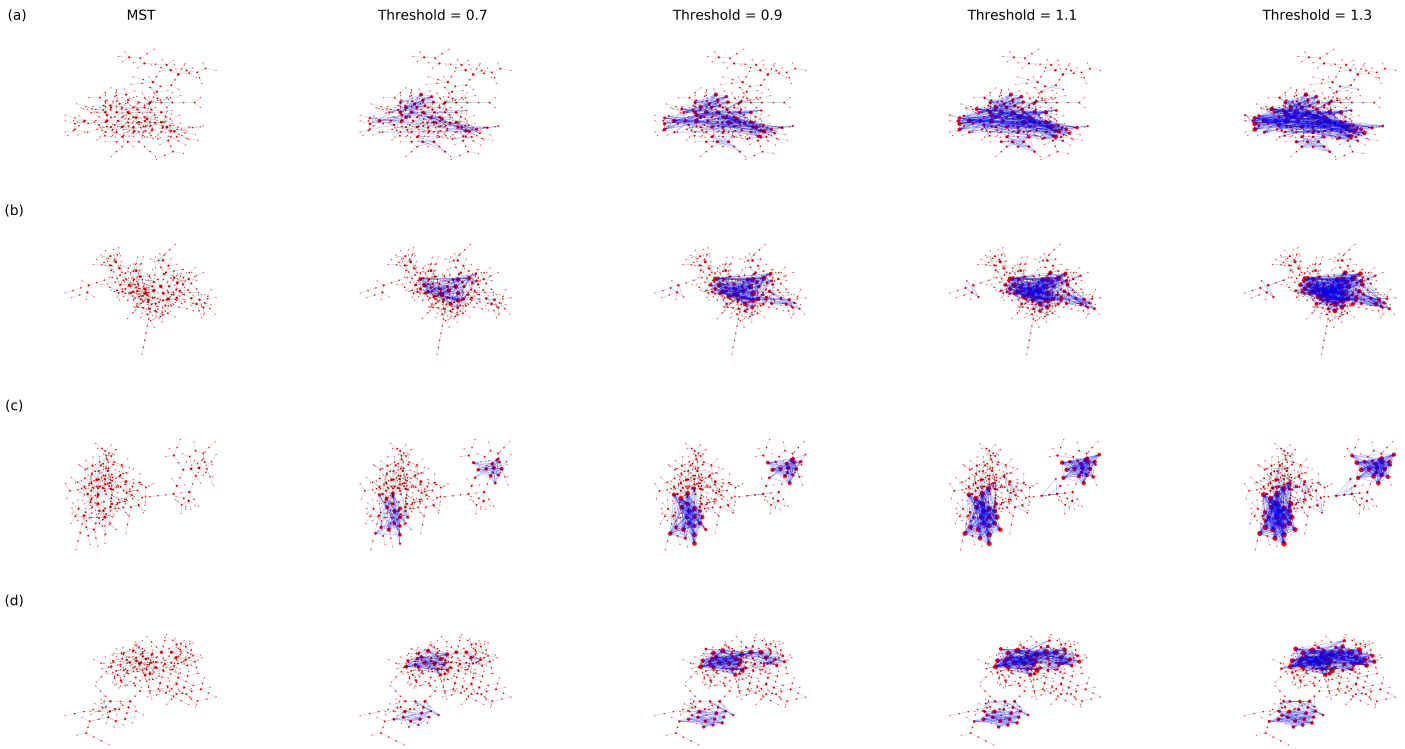


Figure 3.6: **Snapshots of network configuration for small $FD_t^{Std}$**

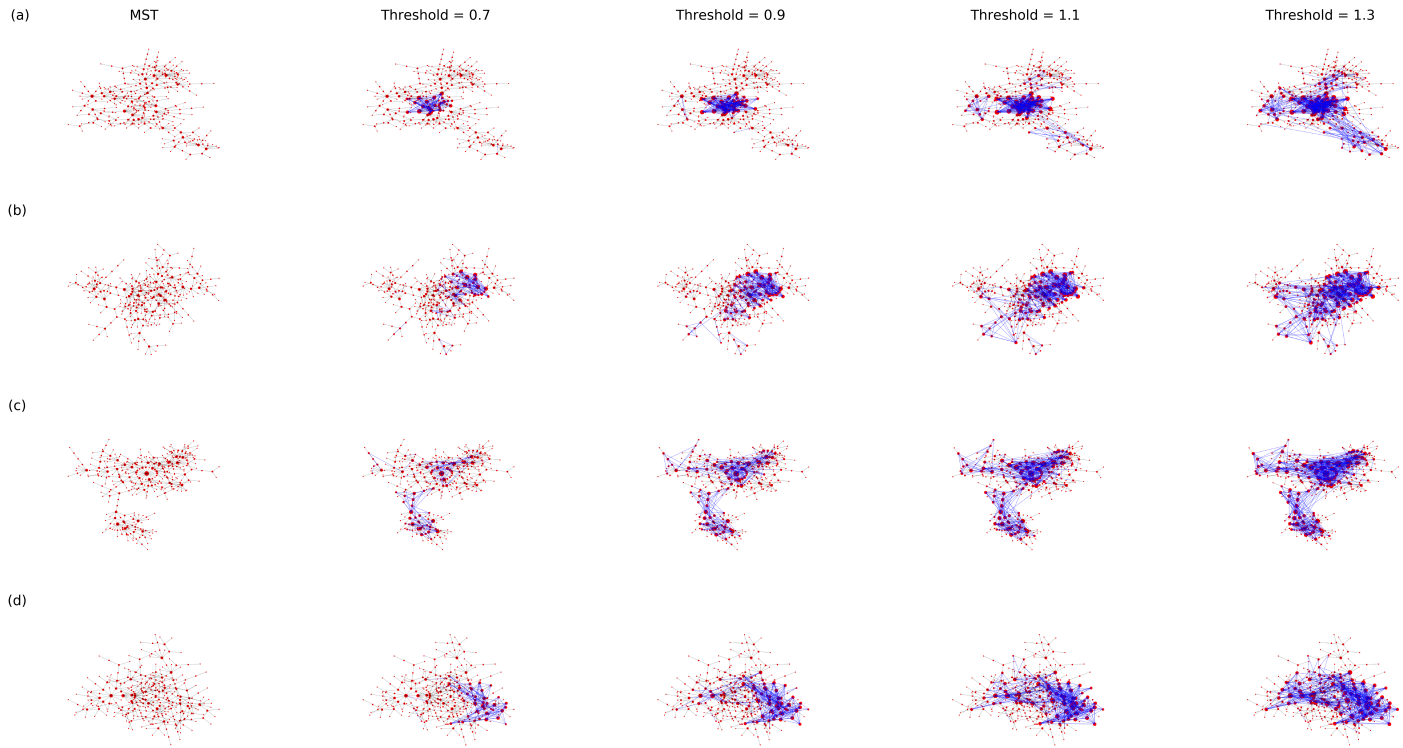


Figure 3.7: **Snapshots of network configuration for large** $FD_t^{Std}$

network with 1.35 threshold at green dotted lines in Fig. 3.3. Note that the network with 1.35 threshold is the maximum growth of the S&P500 network with a significant fractal structure, as stated in Table 3.2. At first, Fig. 3.8-(a), the network of large $FD_t^{Mean}$ and small $FD_t^{Std}$, indicates that the hubs represented in green are spread intensively at the center of the network, but growth patterns represented by blue edges show growth of local shortcuts in one cluster at the center. Secondly, Fig. 3.8-(b), the network of large $FD_t^{Mean}$ and large $FD_t^{Std}$, demonstrates the evenly distributed hubs across the entire network with the growth of global shortcuts. Thirdly, Fig. 3.8-(c), the network of small $FD_t^{Mean}$ and small $FD_t^{Std}$. The result indicates the two apart clusters with hubs with the growth of local shortcuts. Lastly, Fig. 3.8-(d), the network of small $FD_t^{Mean}$ and large $FD_t^{Std}$, exhibits more than two large clusters with hubs with the growth of global shortcuts. In summary, the results show the same insights, as stated in Fig. 3.4, 3.5, 3.6 and 3.7. Also, in general, the large value of $FD_t^{Std}$ is observed when the value of $FD_t^{Mean}$ is large.

Now, in order to verify the insights obtained from the snapshots of network configuration results for the previous large and small values of $FD_t^{Mean}$ and $FD_t^{Std}$, two analyzes are performed. First, the work of Song et al. (2006) argued that the fractality of the network is related to the phenomenon of repulsion (or, disassortativity) between most connected nodes (i.e. hubs). According to the observations of Fig. 3.4 and 3.5, it can be inferred that the repulsion phenomenon among hubs is related to the value of $FD_t^{Mean}$. To verify this, consider the average distance and relationship of the top 10% nodes of $FD_t^{Mean}$ and degree. In other words, it calculates the relationship between the fractal dimension value and the distribution on the network of hub nodes. Second, the work of Kim et al. (2007) argued that the

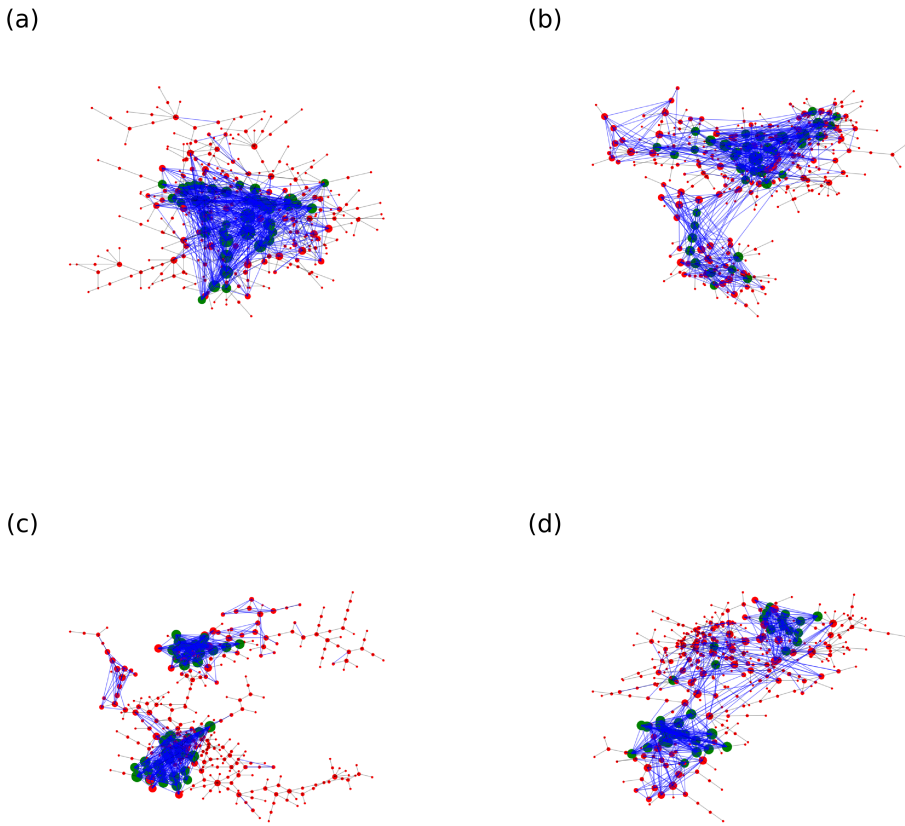


Figure 3.8: **Snapshots of network configuration for different $FD_t^{Mean}$ and $FD_t^{Std}$ at $Threshold = 1.35$**

(a) Large $FD_t^{Mean}$ & Small $FD_t^{Std}$, (b) Large $FD_t^{Mean}$ & Large $FD_t^{Std}$
(c) Small $FD_t^{Mean}$ & Small $FD_t^{Std}$, (d) Small $FD_t^{Mean}$ & Large $FD_t^{Std}$

fractality of networks is related to the shortcut form of the generated edges. According to the observations of Fig. 3.6 and 3.7, it can be inferred that this phenomenon is related to the value of $FD_t^{Std}$. To verify this, observe the relationship between the value of $FD_t^{Std}$ and the average distance between all nodes in the two threshold networks based on the minimum spanning tree (MTN) of $\alpha = 0.6, 1.35$. That is, the new edge created at $\alpha = 1.35$ is a local shortcut, the average distance of the entire network will be slightly reduced, and if it is a global shortcut, the average distance of the entire network will be relatively largely reduced.

Fig. 3.9-(a) shows a strong effective repulsion phenomenon based on the scatter plot of the values of average shortest distance among the hubs(nodes with top 10% degrees) versus the corresponding values of $FD_t^{Mean}$. Note that the blue and red dots indicate the small and large values of $FD_t^{Mean}$ in 3.3-(a). The result shows the negative correlation between the average hub distance and $FD_t^{Mean}$ where the Pearson correlation coefficients are $-0.6418$. Fig. 3.9-(b) shows the difference between the mean values of the distances between all nodes of the MTN for $\alpha = 0.6, 1.35$ and scatter plot of $FD_t^{Std}$. Same as Fig. 3.9-(a), note that the blue and red dots indicate the small and large values of $FD_t^{Std}$ in 3.3-(b). The result shows the negative correlation between the average hub distance and $FD_t^{Mean}$ where the Pearson correlation coefficients are $-0.4936$. In other words, if $FD_t^{Std}$ is large, it means that the edges added to the MTN of $\alpha = 1.35$ based on the MTN of $\alpha = 0.6$ significantly reduce the average distance of the network. This indicates that the newly created edge has the characteristics of a global shortcut. Conversely, if $FD_t^{Std}$ is small, the edges added to the MTN of $\alpha = 1.35$ based on the MTN of $\alpha = 0.6$ decrease the average distance of the network relatively small. This indicates that the newly created edge has the

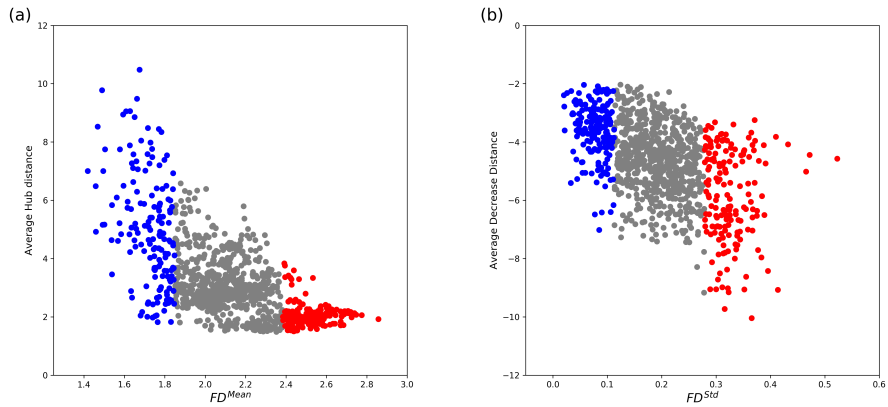


Figure 3.9: **Scatter plots of network properties and fractal measures**
(a) Average shortest distance among hubs for different $FD_t^{Mean}$
(b) Difference between average distance of the MTNs($\alpha = 0.6, 1.35$) for different $FD_t^{Std}$

characteristics of a local shortcut.

Fig. 3.10 plots the evolution of Fig. 3.9. In Fig. 3.10-(a),(b), the following points should be considered. In intervals with successively high or low fractal measures, the average distance between the hubs of the network and the reductions in the average distance between two MTNs also have low or high values, respectively. This shows that the network topology can be known consistently at high and low values among the proposed fractal measures. That is, the proposed fractal measures are suitable for use in the analysis using structural changes in the stock market network.

Fig. 3.3-(c) shows that the mean and standard deviation of $FD_t^{Fit}$ are 0.1969 and 0.0545, respectively. The work of Song et al. (2006) stated that the robust network composed of functional modules (e.g., a cellular or communications network) exhibits the fractal topology. A drastic change in the stability of estimation could imply the change in the network structure. Throughout the experiment, the S&P500 network

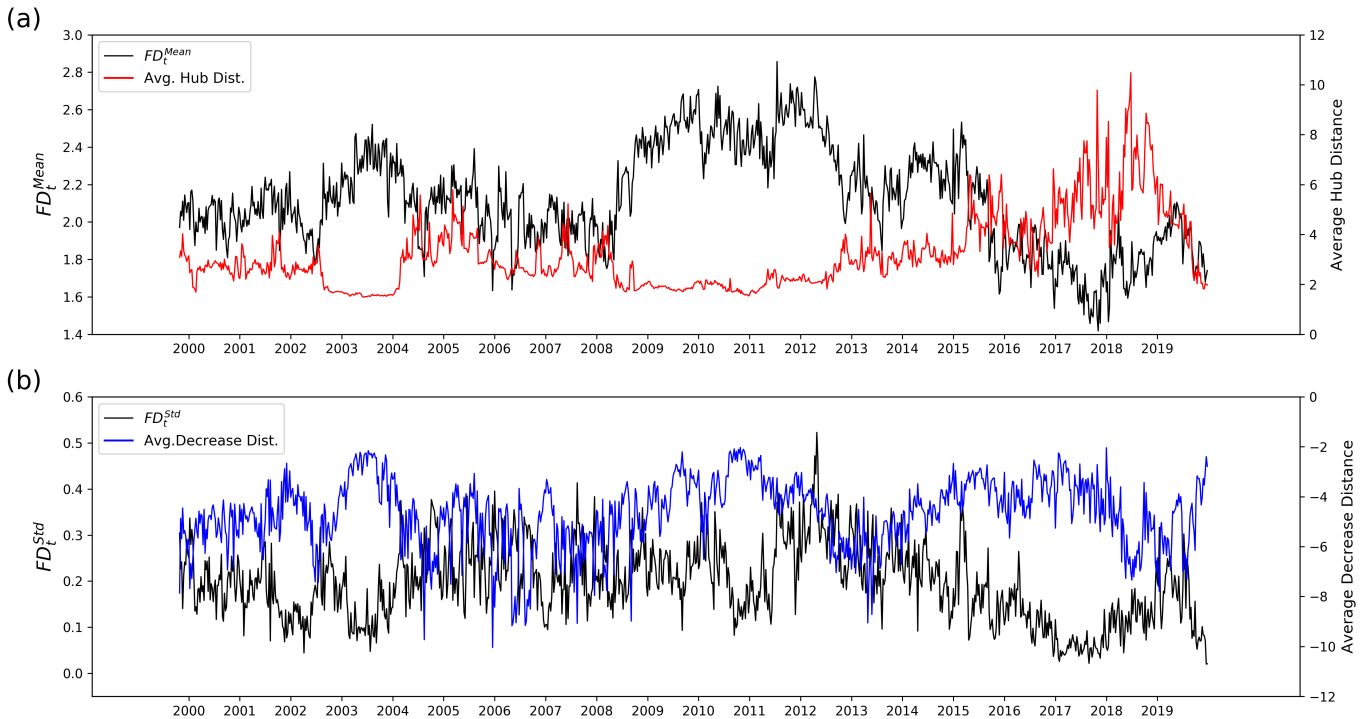


Figure 3.10: **Evolution of network properties and fractal measures**

(a) $FD_t^{Mean}$ (Black line), Average shortest distance among hubs for different (Red line)

(b) $FD_t^{Std}$ (Black line) , Difference between average distance of the MTNs($\alpha = 0.6, 1.35$) (Blue line)

shows consistently low estimation errors, which impose difficulty in analyzing the $FD_t^{Fit}$. We believe that the implication of $FD_t^{Fit}$ can be further examined in more volatile financial markets.

## 3.3 Summary and Discussion

In this chapter, we analyzed the S&P500 network based on the measures of fractal structure. We proposed a threshold network based on MST to properly exploit the concept of fractal dimensions, which measure the increase in unit metric with respect to the scale in the geometry. In our best knowledge, our approach is the first attempt to utilize the concept of fractal structure in a correlation-based financial network. Based on the results, we reached several conclusions as follows. First, we discovered the fractality of S&P500 network. Given that the network with fractal structure implies the robust modular system, we suggested that the financial network possesses the robust structure comprising the set of hubs. Second, we detected the phenomenons related to the fractal structure in S&P500 network. Based on $FD_t^{Mean}$, we observe the strong effective repulsion phenomenon. The reason why the repulsion phenomenon can be useful for the structural characteristics of the network in stock market analysis is as follows. When a strong shock (positive or negative) comes to a node in the network, the impact of the shock propagates to nearby nodes with high correlation. If the hub nodes are located in a position relatively close to the shocked node, the impact of the shock will more easily propagate through the network as a strong effect. In opposite case, if there are relatively far hub nodes near the shocked nodes, the impact of the shock will gradually decrease before reaching other hub nodes, so it will be difficult to have a large impact on the entire network. This characteristics of the network, such as the repulsion phenomenon, are important in terms of risk management in the stock market. This is because the robust stock market network topology must be structured to be shock resistant. If the structure of

the stock market network is vulnerable to shock, then this characteristic that should be considered first in the various decisions made in the stock market. Finally, we verified that $FD_t^{Std}$ reveals the growth behaviors of local and global shortcuts in the stock market network. The type of shortcut is closely related to the distance between network nodes. If a global shortcut is created on the network, the distance between the two sets of nodes directly connected by the shortcut will be greatly reduced. In other words, it is a numerical measure of the relationship between stock sets that has different characteristics in the stock market. Since many stock market studies are conducted within the category of sector, measuring the relationship between stocks due to the shortcut structure is important because it not only changes the relevance between sectors, but also changes the composition of sectors. Using the combinations of large and small values of $FD_t^{Mean}$ and $FD_t^{Std}$, we confirmed that the growth pattern of threshold networks are well represented in measures. In addition, $FD_t^{Fit}$ explains the stability of the S&P500 network structure with its stable evolutions.

# Chapter 4

# Stock Market Prediction with Fractality

## 4.1 Classification of Stock Market

### 4.1.1 Classification Model

In this section, we demonstrate the effect of the fractal measures developed in Chapter 3 to predict the cumulative return direction of the S&P500 index. Therefore, in this section, we first create a benchmark model that consists of only basic variables and analyze whether the performance of the direction prediction improves compared to the benchmark model when the proposed fractal measures are included.

The data used for the direction prediction model is S&P500 index price data of 1056 weeks from 1999-01-01 to 2019-12-31 period, which is the same as Chapter 3. As in the previous chapter, the S&P500 index price data is transformed to the weekly logarithmic return series. Let $P^{Index}(t)$ denotes the closing price of S&P500 index at time $t$, then the cumulative log-return of the index from $t-\tau$ to $t$ can be defined as follows.

$$R^{Index}(t-\tau,t) = \ln(P^{Index}(t)) - \ln(P^{Index}(t-\tau)) \tag{4.1}$$

where $\tau(= 1, 4, 12, 24, 36, 48, 60)$ indicates the lag for cumulative log-return. Let a set of past cumulative log-return series for input variable can possess $m$(=4,12,24) weeks number of series; then, the set can be defined as $\{R^{Index}(t-m+1-\tau, t-m+1), R^{Index}(t-m+2-\tau, t-m+2), ..., R^{Index}(t-1-\tau, t-1), R^{Index}(t-\tau, t)\}$.

There are two steps to the analysis. First, we make a benchmark model for direction prediction using only past cumulative log-return series for each algorithm and evaluate the performance. The next step is to build a model that adds the fractal measures suggested in Chapter 3 for the same prediction algorithm. By comparing the performance of this model with the benchmark, we evaluate the predictive power improvement of the three proposed fractal measures. The output variable is composed of categorical variables that become 1 if $R^{Index}(t, t+\tau)$ is greater than 0, otherwise 0. In other words, when the output variable is 1, the future price direction is an upward trend. we employ sampling to split the dataset for prediction into the training (70%) and test (30%) sets. Note that we use 20% of the training set for the validation.

Researches to predict the future direction of stock prices have been extensively studied, and there are various types of classification algorithms. In this section, the main purpose of predicting the direction of the S&P500 index is to verify whether the proposed fractal measures contribute to improving predictive power. Therefore, rather than considering the structure of the classification algorithm, we intend to analyze it as the most frequently used model in related research. As a representative machine learning model with these characteristics, we will describe the two machine learning techniques: logistic regression (LR) and Random Forest (RF)

**Logistic Regression :** Logistic regression(Cox, 1958) is a multivariate model, a simple model for classification problems. For this reason, logistic regression is used in various studies in the financial field as a baseline model to predict the direction of stock prices. In this study, in order to predict the future direction of the S&P500 index price, we will use logistic regression as a model to observe the degree of improvement in the predictive power of fractal measures. In general, a properly learned machine learning model should have the same or better performance as the number of input variables increases. In experiments, l2 regularized (also called penalized) is used because the number of input variables varies depending on whether the number of past cumulative return series or the fractal dimension is included.

**Random Forest :** Random forest is a model built with ensemble of trees and is used to cope with the robustness or suboptimal performance problems of decision tree models. Random forest creates several individual decision trees through bagging, and learns by randomly selecting variables for each tree. Therefore, it is a robust model for noise and less overfitting than a single decision tree model. If the number of trees in the random forest increases, the learning time increases, but the model stability is improved because relatively more consistent results are obtained. There are several parameters in the random forest, and we use a large number of trees (100) and set the number of variables to the square root of the total number of predictors recommended in the previous study(Breiman, 2001).

The data for direction prediction is the price of the Standard&Poor's 500 (S%P500) index. Fig. 4.1 shows the price series of the S%P500 index from 1999-01-01 to 2019-12-31 period. As shown in Fig. 4.1, the price series of the S&P500

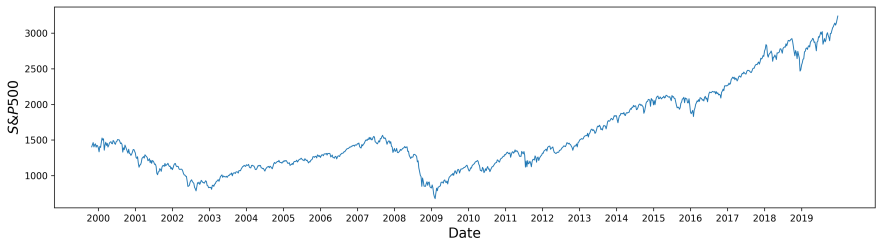


Figure 4.1: **Standard & Poor's 500 index**

Table 4.1: **Relationship between $\tau$ and stock market direction**

| $\tau$ (weeks) | 1 | 4 | 12 | 24 | 36 | 48 | 60 |
|---|---|---|---|---|---|---|---|
| UP | 0.5569 | 0.6238 | 0.6538 | 0.6781 | 0.7168 | 0.7335 | 0.7343 |
| DOWN | 0.4431 | 0.3762 | 0.3462 | 0.3219 | 0.2832 | 0.2665 | 0.2657 |

index has a long-term upward trend. In particular, there are several periods with a strong long-term upward trend, the corresponding intervals are from early-2003 to early-2008 and from early-2009 to mid-2018.

In this section, we predict the direction of the future $\tau(= 1, 4, 12, 24, 36, 48, 60)$ weeks of the S&P500 index. Table 4.1 shows the ratio of the future S&P500 index direction by $\tau$. As shown in Fig. 4.1, it is observed that the upward trend of the S&P500 index direction increases as the value of $\tau$ increases. In the classification problem, an imbalanced dataset leads to large distortions in the results, so appropriate performance measures should be considered. Therefore, we use accuracy and Area Under the CurveHuang and Ling (2005) as performance measures in the S&P500 index direction prediction experiment.

Table 4.2 shows the confusion matrix for classification. When classifying an

Table 4.2: **Confusion matrix of stock market direction**

| | | **Predicted** | |
|---|---|---|---|
| | | UP | DOWN |
| **Observed** | UP | True Positive(TP) | False Negative(FN) |
| | DOWN | False Positive(FP) | True Negative(TN) |

imbalanced dataset, if only accuracy is used as a performance measure, the performance of the model cannot be properly evaluated. This is because high accuracy can be obtained even if the minority class is not properly classified. As the long-term future direction prediction as shown in Table 4.1 becomes more imbalanced, we also use AUC as a performance measure. In addition, as shown in Table 4.2, the minority class is in the downward direction. Since downside risk in the stock market can lead to serious crises such as market collapse, the use of AUC as a performance measure is even more important in interpreting the stock market. For the Confusion matrix of Table 4.2, accuracy and AUC can be defined as follows.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{4.2}$$

$$AUC = \frac{1 + TruePositiveRate - FalsePositiveRate}{2} \tag{4.3}$$

where $TruePositiveRate$ and $FalsePositiveRate$ denote the the ratio of positive correctly classified and that of negative misclassified, respectively. In other words,

by evaluating the performance of the classification model with AUC, it is possible to ensure that the majority class is correctly classified as well as the minority data.

### 4.1.2 Classification Results

This section demonstrates that the proposed fractal measures are meaningful in predicting the future S&P500 index direction based on logistic regression and random forest. To prove this, we compare the performance of the proposed model with fractal measures and the benchmark model predicted with only the S&P500 index. Note that, since this experiment is only to confirm the effect of improving the predictive power of fractal measures, observe the results of logistic regression and random forest model in general parameters.

Table 4.3 summarizes the accuracy of the logistic regression model for each input variable with or without fractal measures. In general, in the binary classification problem, the performance is evaluated as a baseline of 0.5. However, in this case, accuracy greater than 0.5 cannot be considered as good performance. According to Table 4.1, if the model has been learned meaningfully, it should be better than the performance of the model that is classified only in the upward direction. From this point of view, the accuracy is higher than the model that classifies only the upward direction for predictions after $\tau = 48$ weeks, but the difference is very small. And, for the results after $\tau = 24$ weeks, we can observe that the model including fractal measures improves slightly compared to the accuracy of without fractal measures. From

Table 4.3: **Accuracy rate of logistic regression model with fractal measures**

| $\tau$ | **Without Fractal Measures** | | | **With Fractal Measures** | | |
|---|---|---|---|---|---|---|
| | past 1M | past 3M | past 6M | past 1M | past 3M | past 6M |
| 1 | 54.80 | 53.01 | 52.49 | 53.01 | 52.34 | 52.05 |
| | (±2.13) | (±1.87) | (±1.64) | (±2.41) | (±2.03) | (±1.93) |
| 4 | 62.64 | 62.33 | 60.56 | 62.61 | 61.81 | 60.54 |
| | (±2.02) | (±2.08) | (±2.42) | (±2.11) | (±2.14) | (±2.08) |
| 12 | 65.67 | 64.97 | 62.91 | 65.12 | 64.43 | 62.65 |
| | (±2.71) | (±2.01) | (±2.66) | (±2.43) | (±1.89) | (±2.47) |
| 24 | 67.41 | 65.70 | 63.46 | 69.64 | 66.93 | 63.49 |
| | (±2.25) | (±1.94) | (±2.54) | (±2.24) | (±1.97) | (±2.59) |
| 36 | 70.31 | 68.06 | 68.75 | 69.91 | 67.49 | 69.00 |
| | (±2.28) | (±2.91) | (±2.89) | (±2.46) | (±2.74) | (±2.68) |
| 48 | 71.95 | 72.30 | 74.04 | 72.80 | 72.70 | 74.82 |
| | (±2.70) | (±2.81) | (±2.70) | (±3.31) | (±2.95) | (±2.67) |
| 60 | 72.36 | 73.56 | 75.13 | 76.36 | 75.02 | 75.20 |
| | (±2.26) | (±2.98) | (±2.78) | (±2.47) | (±2.44) | (±2.48) |

the results of the accuracy, it can be seen that the models including the proposed fractal measures show a slight improvement in predictive power for prediction in the long-term future ($\tau = 24, 36, 48, 60$) than the benchmark models. However, considering the ratio of upward direction for each $\tau$ in Table 4.1, there is no significant predictive improvement effect except for $\tau = 48, 60$.

Table 4.4 summarizes the accuracy of the random forest model for each input variable with or without fractal measures. Similarly, a model with accuracy higher than the ratio of the majority class of Table 4.1 is learned meaningfully. From this point of view, the accuracy is higher than the model that classifies only the upward direction for predictions after $\tau = 12$ weeks. And, for the results after $\tau = 12$ weeks, we can observe that the model including fractal measures improves slightly

Table 4.4: **Accuracy rate of random forest model with fractal measures**

| $\tau$ | **Without Fractal Measures** | | | **With Fractal Measures** | | |
|---|---|---|---|---|---|---|
| | past 1M | past 3M | past 6M | past 1M | past 3M | past 6M |
| 1 | 51.93 | 48.96 | 50.56 | 49.39 | 51.49 | 53.95 |
| | (±3.10) | (±2.44) | (±1.92) | (±2.81) | (±2.45) | (±2.04) |
| 4 | 62.06 | 57.01 | 59.80 | 59.92 | 61.41 | 63.42 |
| | (±1.96) | (±2.46) | (±2.29) | (±2.38) | (±2.22) | (±2.19) |
| 12 | 65.29 | 62.26 | 63.63 | 63.69 | 66.67 | 69.85 |
| | (±2.56) | (±2.48) | (±2.35) | (±2.11) | (±2.34) | (±2.28) |
| 24 | 66.51 | 66.06 | 67.68 | 70.24 | 73.03 | 74.58 |
| | (±2.12) | (±2.09) | (±2.69) | (±2.18) | (±1.67) | (±2.48) |
| 36 | 69.88 | 67.99 | 70.21 | 74.04 | 74.74 | 76.37 |
| | (±2.46) | (±2.63) | (±2.73) | (±1.87) | (±2.67) | (±2.58) |
| 48 | 71.41 | 72.07 | 73.72 | 76.86 | 77.19 | 79.02 |
| | (±2.59) | (±2.32) | (±2.36) | (±2.35) | (±2.33) | (±2.42) |
| 60 | 71.74 | 72.79 | 76.22 | 78.13 | 77.90 | 80.90 |
| | (±2.17) | (±2.97) | (±2.27) | (±1.53) | (±2.53) | (±1.98) |

compared to the accuracy of without fractal measures. The increase in the accuracy of the random forest model shows a greater predictive power improvement than the results of logistic regression. That is, if the model considers sufficient non-linearity, the model shows significant performance in predictions after $\tau = 12$ weeks, and consistent predictive power improvement in models with fractal measures. This result is the same as the hypothesis of most time series theories that there is as much information as possible in the latest data.

Table 4.5 summarizes the AUC of the logistic regression model for each input variable with or without fractal measures. By Eq. (4.3), the AUC result is a significant performance when it is greater than 0.5. Similar to Table 4.3, AUC results show significant performance at $\tau = 48, 60$. Note that, in the case of the accuracy result of

Table 4.5: **AUC of logistic regression model with fractal measures**

| $\tau$ | **Without Fractal Measures** | | | **With Fractal Measures** | | |
|---|---|---|---|---|---|---|
| | past 1M | past 3M | past 6M | past 1M | past 3M | past 6M |
| 1 | 49.51 | 49.16 | 50.14 | 48.65 | 48.78 | 49.84 |
| | (±1.03) | (±1.82) | (±1.56) | (±2.15) | (±1.73) | (±1.99) |
| 4 | 49.96 | 49.68 | 50.95 | 49.97 | 49.31 | 51.12 |
| | (±0.17) | (±0.79) | (±1.86) | (±0.22) | (±0.95) | (±1.66) |
| 12 | 50.00 | 49.92 | 49.73 | 49.59 | 49.65 | 50.82 |
| | (±0.00) | (±0.59) | (±1.23) | (±0.52) | (±0.60) | (±1.87) |
| 24 | 50.75 | 49.62 | 48.81 | 56.74 | 53.70 | 50.14 |
| | (±2.34) | (±1.64) | (±1.34) | (±2.34) | (±1.81) | (±1.37) |
| 36 | 49.88 | 48.99 | 49.05 | 50.89 | 49.20 | 49.95 |
| | (±0.25) | (±0.87) | (±1.07) | (±1.26) | (±1.35) | (±1.07) |
| 48 | 50.73 | 50.18 | 53.84 | 56.18 | 53.86 | 56.51 |
| | (±2.34) | (±0.92) | (±1.81) | (±2.97) | (±1.99) | (±2.33) |
| 60 | 50.80 | 50.36 | 52.46 | 61.17 | 57.30 | 56.29 |
| | (±3.10) | (±0.92) | (±1.56) | (±3.31) | (±2.20) | (±1.95) |

Table 4.3, the fractal measures had only a slight improvement in predictive power, but in the case of the AUC result of 4.5, it had a relatively large prediction power improvement effect. It can be thought that the model including the fractal measures showed a significant improvement in predictive power in the downward direction.

Table 4.6 summarizes the AUC of the random forest model for each input variable with or without fractal measures. The AUC results showed significant performance for the prediction of direction after $\tau = 4$ weeks. In particular, in the prediction of the long-term future after $\tau = 24$ weeks, the model including fractal measures showed over 60% AUC for all parameters, and AUC improved about 5% 10% compared to the benchmark model. This is a significant improvement in predictive power compared to Table 4.5. As with the previous accuracy conclusions,

Table 4.6: **AUC of random forest model with fractal measures**

| $\tau$ | **Without Fractal Measures** | | | **With Fractal Measures** | | |
|---|---|---|---|---|---|---|
| | past 1M | past 3M | past 6M | past 1M | past 3M | past 6M |
| 1 | 48.33 | 47.47 | 48.43 | 48.04 | 49.21 | 51.56 |
| | (±2.19) | (±2.37) | (±1.81) | (±2.58) | (±2.40) | (±1.76) |
| 4 | 50.64 | 50.34 | 52.21 | 52.63 | 52.50 | 54.58 |
| | (±0.81) | (±2.48) | (±2.24) | (±2.10) | (±1.42) | (±1.63) |
| 12 | 51.03 | 50.79 | 54.13 | 53.33 | 56.96 | 61.94 |
| | (±0.99) | (±1.52) | (±2.15) | (±1.94) | (±2.30) | (±2.36) |
| 24 | 57.27 | 57.62 | 58.21 | 60.94 | 64.04 | 66.17 |
| | (±4.00) | (±2.21) | (±2.52) | (±2.62) | (±1.95) | (±1.94) |
| 36 | 50.05 | 51.04 | 53.19 | 62.47 | 62.61 | 63.37 |
| | (±0.88) | (±2.26) | (±1.25) | (±2.05) | (±2.79) | (±1.95) |
| 48 | 50.43 | 51.77 | 54.22 | 65.34 | 63.44 | 64.08 |
| | (±2.13) | (±1.55) | (±1.79) | (±2.86) | (±2.85) | (±2.00) |
| 60 | 50.90 | 50.73 | 56.57 | 66.49 | 64.94 | 66.30 |
| | (±3.55) | (±1.01) | (±1.74) | (±2.22) | (±2.31) | (±2.99) |

fractal measures can make a significant contribution to improving predictive power by using a model that is sufficiently considered for non-linearity.

## 4.2 Fractal Measures and Predictive Power

This section contains content published by Ku et al. (2020).

### 4.2.1 Prediction of Stock Market Return

Based on three measures defined in Section 3.1.3, we attempt to predict the S&P500 index for different time in the future. Specifically, we utilize a simple artificial neural network (ANN) model based on the AdamOptimizer(Kingma and Ba, 2014) to claim the predictability of the fractal measures. Selecting a simple neural network as the prediction model is due to the results discussed in the direction prediction results in the previous section. Fractal measures are due to their contribution to future predictions on models where non-linearity is considered. Therefore, this section uses the ANN model, which is the simplest model that can consider non-linearity. The step-by-step procedure of market prediction is summarized in Fig. 4.2.

The input layer is composed of a set of past cumulative log-return series and combination sets of three fractal measures. Let $P^{Index}(t)$ denotes the closing price of S&P500 index at time $t$, then the cumulative log-return of the index from $t-\tau$ to $t$ can be defined as Eq. (4.1) where $\tau(=1,4,12,24,36,48,60)$ indicates the lag for cumulative log-return. Fig. 4.3 shows the cumulative log-return series of S&P500 index from $t-\tau$ to $t$ for each $\tau(=1,4,12,24,36,48,60)$. As shown in the figure, the length of the data depends on $\tau$. Based on the cumulative log-return series, input and output variables are configured.

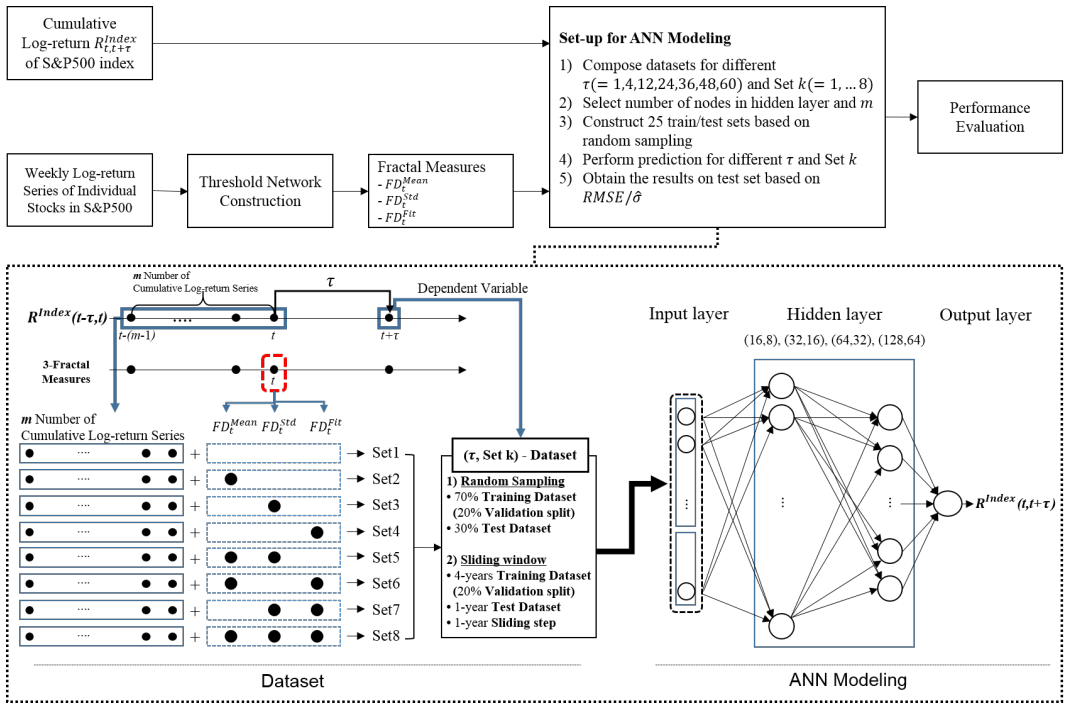


Figure 4.2: **Step-by-step procedure of market prediction**

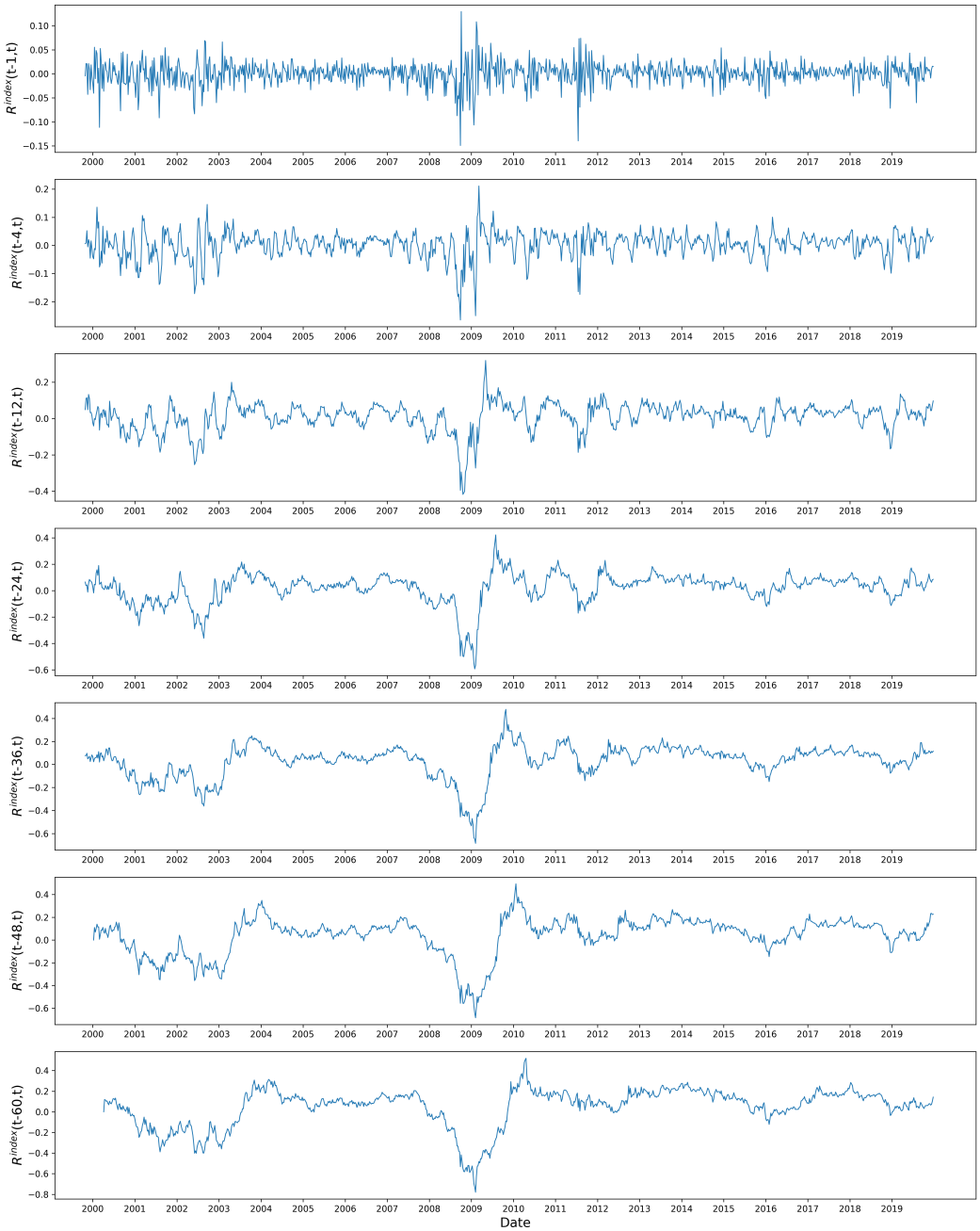


Figure 4.3: **Evolution of $R^{Index}(t-\tau,t)\}$ by $\tau$**

Let a set of past cumulative log-return series for input variable can possess $m(= 1, 2, ..., 10)$ number of series; then, the set can be defined as $\{R^{Index}(t - m + 1-\tau, t-m+1), R^{Index}(t-m+2-\tau, t-m+2), ..., R^{Index}(t-1-\tau, t-1), R^{Index}(t-\tau, t)\}$. Note that we define this set as Set 1. Based on Set 1, we extend the set of input variable by adding the combinations of corresponding fractal measures including $FD_t^{Mean}$, $FD_t^{Std}$, and $FD_t^{Fit}$. Hence, we obtain eight $(k = 8)$ sets as input variable. Furthermore, the output layer consists of the cumulative log-return of S&P500 index from $t$ to $t + \tau$, which can be written as $R^{Index}(t, t + \tau)$. In this regard, the dataset for prediction is obtained by applying the above equations for all $t$.

From the perspective of ANN modeling, we decide to use the two-hidden-layer neural network, which successfully provides the convergence on the error function. Then, the model requires two parameters to perform the prediction. The first parameter is the number of layers in each hidden layer, whereas the second one is the number of past cumulative log-returns for the input layer, $m$. To analyze the results based on various cases of ANN models, we perform the prediction with different parameters, the number of nodes $\in \{(16, 8), (32, 16), (64, 32), (128, 64)\}$ and $m \in \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$. For a prediction with $\{\tau$, Set $k\}$, we employ two types of sampling method. The first sampling method is random sampling to split the dataset for prediction into the training (70%) and test (30%) sets. Note that we use 20% of the training set for the validation. Random sampling is primarily aimed at finding a suitable parameters to evaluate the predictive power of fractal measures. We expect the consistent prediction performance through the random sampling since the model learns the entire $t$ defined for each combination of $\{\tau$, Set $k\}$ regardless of the aspects of the financial market over time such as financial crisis, recession,

bullish and bearish market. The second sampling method is sliding window method. In each window, training period is 4 years, the testing period is 1 year, and the sliding step size is 1 year. Similarly, 20% of the training set is used for validation. As a result of this method, we consider the predictive power of fractal measures from the perspective of forecasting problems. Note that the model training is performed 50 times for each $\{\tau$, Set $k\}$ to ensure the robustness to different values of the training set.

The performance evaluation is based on the root mean squared error (RMSE) of the test sets for 50 learned models, which is widely employed in prediction studies. Particularly, we adjust the scale for each $\tau$ by using $\sqrt{MSE}/\hat{\sigma}_\tau$ where $\hat{\sigma}_\tau$ is the standard deviation of aggregated logarithmic return series of the S&P 500 index for $\tau$ weeks. Note that the concept of scaled RMSE, $\sqrt{MSE}/\hat{\sigma}_\tau$, is similar to the Sharpe ratio, the variability-adjusted measure for investment performance, in finance Sharpe (1966). Throughout the prediction, we mainly focus on examining if the fractal measures($FD_t^{Mean}, FD_t^{Std}, FD_t^{Fit}$) can improve the prediction performance on the $\tau$-lagged cumulative returns of S&P500 in the future. That is, the performances of Set1, which contains the plain past cumulative return series, must be enhanced by adding the fractal measures. The improved prediction performance implies the possible utilization of the fractal measures in real-world practice.

### 4.2.2 Parameter Analysis

The evolution of $FD_t^{Mean}$, $FD_t^{Std}$, and $FD_t^{Fit}$ involve the implicit properties regarding the fractal topology, which affects the structural changes of the S&P500

network. Therefore, we examine the utilities of three measures via a simple market prediction algorithm. Specifically, we integrate $FD_t^{Mean}$, $FD_t^{Std}$ and $FD_t^{Fit}$ as added input variables in two-hidden-layer ANN models with four different node sets($\in \{(16,8),(32,16),(64,32),(128,64)\}$) to predict the cumulative log-returns of S&P500 in $\tau(= 1,4,12,24,36,48,60)$ weeks in future based on the past $m(= 1,2,3,4,5,6,7,8,9,10)$ weeks of cumulative log-return series of S&P500.

Table 4.7 summarizes the average of top five prediction performances in $\sqrt{MSE}/\hat{\sigma}_\tau$ for different number of nodes in hidden layers and number of past cumulative log-returns, $m$. In other words, each value in Table 4.7 is the average of the top five performances among the results of the model for all ($\tau$, Set $k$) when the number of nodes and $m$ are fixed. This process is repeated 50 times with random sampling to obtain the top five average values. Note that the similar approach is used in Lee et al. (2019). The result shows that the prediction errors are not considerably different for the number of nodes in hidden layers whose values are $0.8101, 0.8097, 0.8121$, and $0.8097$ for $(16,8),(32,16),(64,32),(128,64)$, respectively.

### 4.2.3 Predictive Power Results

We further investigate the result of the best performance for each number of nodes in hidden layers. Note that the cases are $m = 5$ for (16,8), $m = 5$ for (32,16), $m = 5$ for (64,32), and $m = 5$ for (128,64). Specifically, the model is evaluated based on 50 simulations by fixing the number of nodes in hidden layer and $m$ for each parameter set($\tau$, Set $k$)

Again, the standard deviation of $\tau$ weeks cumulative log-return series, $\hat{\sigma}_\tau$, is

Table 4.7: **Average of top five prediction performances ($\sqrt{MSE}/\hat{\sigma}_\tau$) for different number of nodes in hidden layers and $m$**

| $m$ | (16, 8) | (32, 16) | (64, 32) | (128, 64) |
|---|---|---|---|---|
| 1 | 0.8119 | 0.8037 | 0.8109 | 0.8123 |
| 2 | 0.8111 | 0.8006 | 0.8050 | 0.8115 |
| 3 | 0.8102 | 0.8038 | 0.8033 | 0.8057 |
| 4 | 0.8055 | 0.8088 | 0.8063 | 0.7966 |
| 5 | 0.7935 | 0.7980 | 0.7980 | 0.7947 |
| 6 | 0.8165 | 0.8211 | 0.8150 | 0.8084 |
| 7 | 0.8140 | 0.8111 | 0.8092 | 0.8091 |
| 8 | 0.8158 | 0.8132 | 0.8214 | 0.8154 |
| 9 | 0.8135 | 0.8157 | 0.8233 | 0.8210 |
| 10 | 0.8088 | 0.8213 | 0.8286 | 0.8222 |

used to standardized the prediction performance of models with different $\tau$. The results of $\sqrt{MSE}/\hat{\sigma}_\tau$ for each parameter set are summarized in Table 4.8.

The prediction performance in Table 4.8 can be conveniently explained by plots as illustrated in Figure 4.4. The left plots in Fig. 4.4 depicts the results in Table 4.8 for different $\tau$. Note that the smaller $\sqrt{MSE}/\hat{\sigma}_\tau$ on y-axis indicates better prediction performance. For most cases of Set $k$, $\sqrt{MSE}/\hat{\sigma}_\tau$ decreases as $\tau$ increases until $\tau = 36$, whereas $\sqrt{MSE}/\hat{\sigma}_\tau$ slightly increases thereafter. Hence, the result implies that the structural changes of S&P500 network are useful in the prediction of long-term cumulative return rather than short-term. Furthermore, the right plots in Fig. 4.4-(a) shows the results in Table 4.8 for different input variables, Set $k$. For most cases of $\tau$, the sets comprising $FD_t^{Mean}$ (Set2, Set5, and Set6) show better prediction performances than the others. Especially, the best performance among different $\tau$ is achieved most by Set5(with $FD_t^{Mean}$ and $FD_t^{Std}$).

Table 4.8: **Summary of $\sqrt{MSE}/\hat{\sigma}$ for different $\tau$ and input variable sets**

| $\tau$ | Set1 | Set2 | Set3 | Set4 | Set5 | Set6 | Set7 | Set8 |
|---|---|---|---|---|---|---|---|---|
| 1 | 1.3219 | 1.2592 | 1.2942 | 1.2564 | 1.2756 | 1.3296 | 1.3909 | 1.4430 |
| 4 | 1.1823 | 1.1987 | 1.1805 | 1.2385 | 1.2266 | 1.2094 | 1.2147 | 1.1860 |
| 12 | 1.0524 | 1.0413 | 1.0446 | 1.0552 | 1.0462 | 1.0209 | 1.1018 | 1.0666 |
| 24 | 0.9680 | 0.9320 | 0.9829 | 0.9900 | 0.9677 | 0.9178 | 0.9757 | 0.9622 |
| 36 | 0.8637 | 0.8250 | 0.8710 | 0.8725 | 0.7980 | 0.8162 | 0.8643 | 0.8539 |
| 48 | 0.8749 | 0.8487 | 0.8811 | 0.8805 | 0.8570 | 0.8528 | 0.8876 | 0.8485 |
| 60 | 0.9186 | 0.8580 | 0.9217 | 0.9473 | 0.8608 | 0.8734 | 0.9287 | 0.8657 |

(a) Nodes in hidden layer = (16, 8), $m = 5$

| $\tau$ | Set1 | Set2 | Set3 | Set4 | Set5 | Set6 | Set7 | Set8 |
|---|---|---|---|---|---|---|---|---|
| 1 | 1.4764 | 1.5191 | 1.3919 | 1.5060 | 1.4623 | 1.4590 | 1.5663 | 1.4891 |
| 4 | 1.2666 | 1.2914 | 1.2684 | 1.2834 | 1.3216 | 1.2904 | 1.3708 | 1.4068 |
| 12 | 1.0648 | 1.0417 | 1.0870 | 1.0921 | 1.1259 | 1.0490 | 1.1214 | 1.0649 |
| 24 | 1.0016 | 0.9464 | 0.9840 | 1.0299 | 0.9473 | 0.9374 | 1.0225 | 1.0008 |
| 36 | 0.8714 | 0.8405 | 0.8511 | 0.9052 | 0.8008 | 0.8346 | 0.8811 | 0.8290 |
| 48 | 0.8771 | 0.8245 | 0.8673 | 0.8900 | 0.8264 | 0.8890 | 0.9220 | 0.8677 |
| 60 | 0.9317 | 0.8824 | 0.9104 | 0.9449 | 0.8160 | 0.8751 | 0.9433 | 0.8760 |

(b) Nodes in hidden layer = (32, 16), $m = 5$

| $\tau$ | Set1 | Set2 | Set3 | Set4 | Set5 | Set6 | Set7 | Set8 |
|---|---|---|---|---|---|---|---|---|
| 1 | 1.3648 | 1.4210 | 1.4592 | 1.4353 | 1.4785 | 1.4865 | 1.4564 | 1.5668 |
| 4 | 1.2252 | 1.3355 | 1.2733 | 1.3021 | 1.3648 | 1.3663 | 1.3171 | 1.4301 |
| 12 | 1.0874 | 1.1343 | 1.1003 | 1.1286 | 1.1068 | 1.0961 | 1.1087 | 1.1012 |
| 24 | 1.0494 | 0.9844 | 1.0170 | 1.0779 | 1.0000 | 0.9950 | 1.0917 | 1.0360 |
| 36 | 0.9057 | 0.8548 | 0.9169 | 0.9360 | 0.8189 | 0.8759 | 0.9285 | 0.8671 |
| 48 | 0.9600 | 0.9090 | 0.9735 | 0.9983 | 0.8900 | 0.9094 | 0.9745 | 0.9427 |
| 60 | 0.9854 | 0.9564 | 0.9621 | 0.9981 | 0.8699 | 0.9201 | 0.9636 | 0.9019 |

(c) Nodes in hidden layer = (64, 32), $m = 5$

| $\tau$ | Set1 | Set2 | Set3 | Set4 | Set5 | Set6 | Set7 | Set8 |
|---|---|---|---|---|---|---|---|---|
| 1 | 1.3308 | 1.3954 | 1.3358 | 1.3921 | 1.3803 | 1.3353 | 1.3966 | 1.3442 |
| 4 | 1.2594 | 1.2763 | 1.3122 | 1.2633 | 1.2853 | 1.2606 | 1.3664 | 1.3728 |
| 12 | 1.1308 | 1.1172 | 1.1378 | 1.2761 | 1.1360 | 1.0733 | 1.1454 | 1.1254 |
| 24 | 1.0804 | 1.0232 | 1.0722 | 1.1125 | 1.0092 | 1.0253 | 1.1191 | 1.0579 |
| 36 | 0.9145 | 0.8536 | 0.9382 | 0.9298 | 0.8201 | 0.8808 | 0.9951 | 0.8491 |
| 48 | 0.9651 | 0.9022 | 0.9509 | 0.9865 | 0.9359 | 0.9265 | 0.9820 | 0.9497 |
| 60 | 1.0145 | 0.9336 | 0.9804 | 1.0133 | 0.8859 | 0.9137 | 0.9838 | 0.9224 |

(d) Nodes in hidden layer = (128, 64), $m = 5$

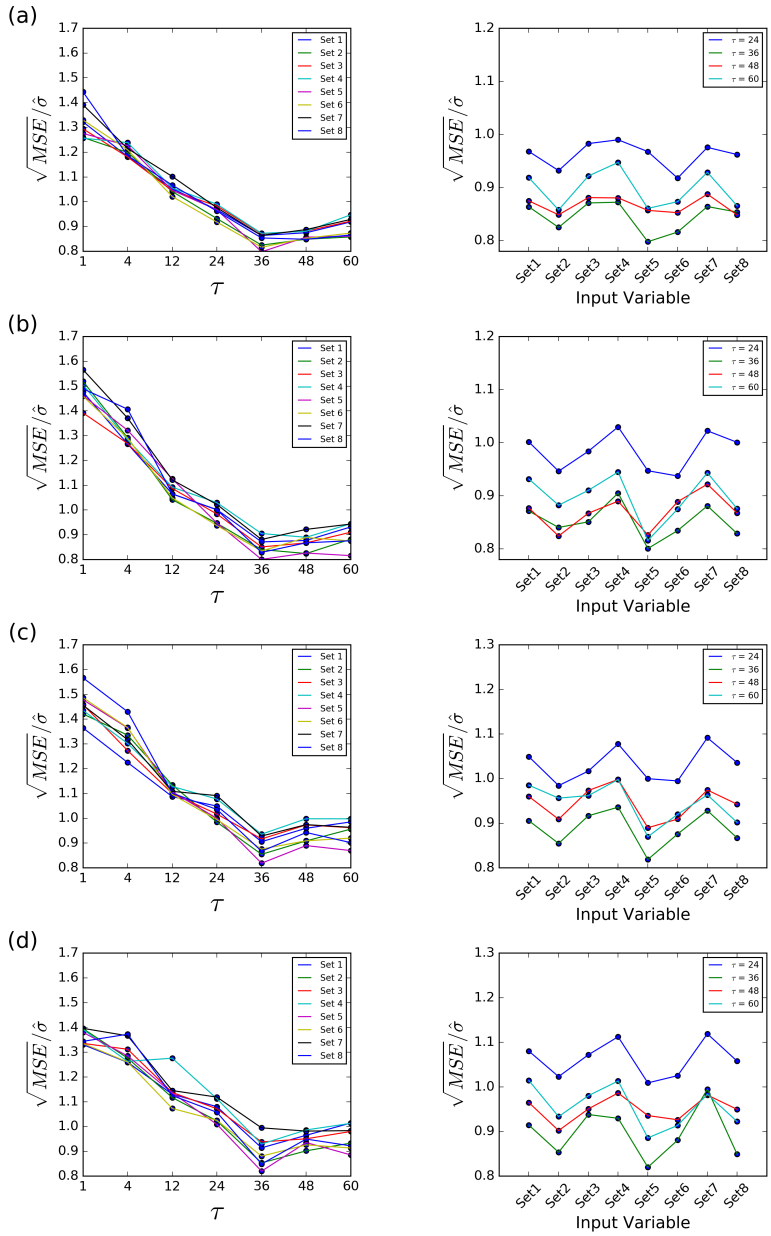


Figure 4.4: **Plots for $\sqrt{MSE}/\hat{\sigma}$ for different $\tau$ and input variable, Set $k$**

(a) $m = 5$ for (16,8), (b) $m = 5$ for (32, 16), (c) $m = 5$ for (64,32), (d) $m = 5$ for (128, 64)

Table 4.9: $\sqrt{MSE}/\hat{\sigma}$ **for different** $\tau$ **and input variable sets**(Sliding window, nodes in hidden layer = (16,8), $m$=5)

| $\tau$ | Set1 | Set2 | Set3 | Set4 | Set5 | Set6 | Set7 | Set8 |
|---|---|---|---|---|---|---|---|---|
| 1 | 3.0696 | 2.0016 | 2.4519 | 3.2109 | 2.0654 | 2.7625 | 2.5683 | 2.6703 |
| 4 | 2.1352 | 1.4119 | 1.6822 | 2.2520 | 1.4263 | 1.9485 | 1.7497 | 1.9286 |
| 12 | 1.8060 | 1.2905 | 1.4933 | 1.8364 | 1.3350 | 1.7060 | 1.4651 | 1.5713 |
| 24 | 1.8874 | 1.4803 | 1.6664 | 1.8871 | 1.4327 | 1.6606 | 1.6469 | 1.7398 |
| 36 | 1.6120 | 1.3403 | 1.4754 | 1.7042 | 1.2737 | 1.4945 | 1.4918 | 1.4737 |
| 48 | 2.0395 | 1.5961 | 1.8113 | 2.0683 | 1.6021 | 1.7894 | 1.8282 | 1.7754 |
| 60 | 2.0632 | 1.6129 | 1.8884 | 2.1863 | 1.5863 | 1.8459 | 1.8879 | 1.8587 |

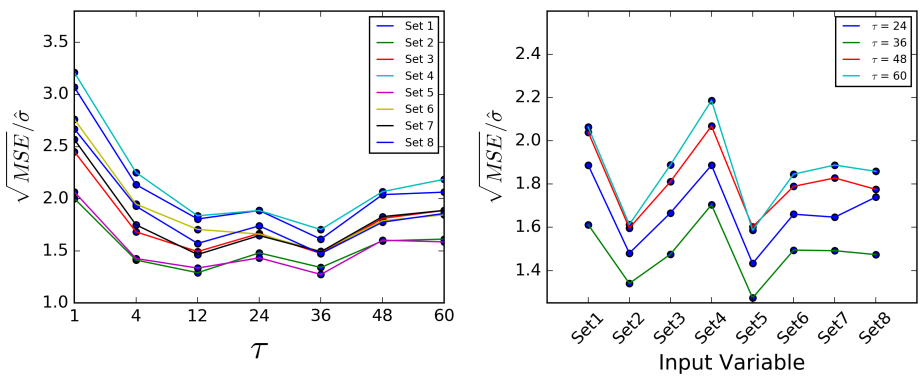


Figure 4.5: **Plots for** $\sqrt{MSE}/\hat{\sigma}$ **for different** $\tau$ **and input variable, Set** $k$(Sliding window, nodes in hidden layer = (16,8), $m$=5)

Table 4.9, Figure 4.5 summarized the results of the $\sqrt{MSE}/\hat{\sigma}_\tau$ using the sliding window method. In the previous results, there was no significant difference in prediction performance for each parameter set. Therefore, the experimental results of the case with m=5 for (16,8) are described. Similar to the previous results, for most cases of $\tau$, the sets comprising $FD_t^{Mean}$ (Set2, Set5, and Set6) show better prediction performances than the others. On the other hand, the performance in the case of $\tau = 12, 24$ are relatively improved compared to the previous results.

In summary, we discover a possible utilization of the fractal measures in market prediction where $FD_t^{Mean}$ is the most useful measure to improve the performance for the cumulative log-returns of S&P500 in long-term future, whereas $FD_t^{Fit}$ is the least useful measure. $FD_t^{Std}$ improves the prediction when it is used with $FD_t^{Mean}$, simultaneously. The predictive power of two measures can be seen as the results of repulsion phenomenon among hubs of the S&P500 network where the growth pattern of the shortcuts in the threshold network implies important characteristics of the stock market in long-term. Meanwhile, a limited predictive power of $FD_t^{Fit}$ is caused by its limited movements in evolutions. However, it is important to keep track of $FD_t^{Fit}$ since the sudden rise or fall of $FD_t^{Fit}$ can indicate a extensive change in its network structure and stability.

## 4.3 Summary and Discussion

In this chapter, we discovered that the proposed fractal measures contribute to the improvement of predictive power of predicting the direction of the future S&P500 index with logistic regression and random forest. To prevent bias caused by model selection, only the two models with the most basic structures were used, and the models that include the fractal measures showed consistently good performance. In addition, the results show that fractal measures contribute to a sufficient predictive power improvement effect in a machine learning model that non-linearity is considered. These results suggest that the proposed fractal measures can be easily used in other studies to predict the stock market. On the other hands, the direction prediction results generally have high prediction accuracy than other prediction problems, but there is a limitation that good prediction accuracy does not necessarily leads to high profitability. For example, if the model predicts correctly at small gains, but not at large losses, the model is helpless at downside risk. To overcome these limitations, we also attempted to predict is cumulative log-returns of S&P500 index with simple neural network that use its own price series as a base and proposed measures as an additional variables. The results show that the predictive power is enhanced when the $FD_t^{Mean}$ and $FD_t^{Std}$ are added in long-term prediction, which also implies that the repulsion phenomenon and the structure of shortcut are well measured and can be applied in real-world practice for risk management. Most of our results have good performance in long-term prediction; therefore, we will find out that reason in future works. These results show that structural changes in the S&P500 network, which can be observed with $FD_t^{Mean}$ and $FD_t^{Std}$ in Chapter 3,

are important properties that explain future changes in the stock market.

Despite many findings, the limitation of this study lies in its implications restricted to the S&P500 market. For instance, even though the change of $FD_t^{Fit}$ is insufficient in S&P500 network, it could be considered as a proxy for substantial changes in the financial market since its sudden elevation or reduction could indicate the structural changes in the S&P500 network. Thus, as future work, we plan to apply the same approach to more volatile financial markets in emerging countries to observe more dynamic changes of the network fractality. Another limitation is that higher prediction performance can be achieved by using more advanced prediction models. However, the main purpose of this chapter is to verify that the proposed fractal measures consistently improve predictive power in the most general situation rather than to find a model with optimal prediction performance. Therefore, the study of developing model structure will be left as future works.

# Chapter 5

# Trading Strategy with Optimal Rebalancing Model

## 5.1 Optimal Rebalancing Model

### 5.1.1 Portfolio Selection Method

The main goal of Chapter 5 is to propose a trading strategy that can be easily applied to prior studies. Most of the studies related to stock market trading strategies are how to organize portfolios. These studies compare the performance of different trading strategies with equal periodically rebalancing method. On the other hands, in this chapter proposes a trading strategy that can calculate the appropriate rebalancing time point for a given portfolio selection methodology. Therefore, the portfolio selection methodologies to be used in this chapter are the following two methodologies that are commonly used in previous studies.

**Equally weighted portfolio:** The most basic portfolio selection methodology is to distribute assets evenly across all available stocks. If there are N stocks available, the investment ratio $w_i$ for stock $i$ at each rebalancing point can be calculated as Eq. (5.1).

$$w_i = \frac{1}{N},\ i = 1...N \tag{5.1}$$

Although there is no special portfolio optimization step, it is known that the idiosyncratic risk can be solved by simply distributing assets evenly. Therefore, an equally weighted portfolio is used as the first portfolio selection method for the optimal rebalancing model we proposed in this chapter.

**Maximized Sharpe ratio portfolio:** Modern portfolio theory(MPT) is the most widely used portfolio theory proposed by Markowitz (1952). MPT has the assumption that investors are risk-averse, which investors prefer a portfolio with a lower risk with same return. In other words, investors will choose a portfolio with higher returns with same risk.

The portfolio selection methodology used in MPT varies depending on the purpose of the investment. The most widely used portfolio is the one that maximizes the Sharpe ratio(Sharpe, 1966). In the maximized Sharpe ratio portfolio, the investment portion $w_i{}_{i=1}^{N}$ of stock $i$ can be obtained by solving Eq. (5.2).

$$\begin{aligned}
\text{Max} \quad & SharpeRatio = \tfrac{E(R_p)-R_f}{\sigma_P^2} \\
\text{s.t.} \quad & \sum_{i=1}^{n} w_i = 1 \\
& \sigma_p = \sum_{i=1}^{n} w_i^2\sigma_i^2 + \sum_{i=1}^{n}\sum_{j=1(j\neq i)}^{n} \sigma_{ij} w_i w_j \\
& E(R_p) = \sum_{i=1}^{n} w_i R_i \\
& w_i \geq 0, i = 1, ..., N
\end{aligned} \tag{5.2}$$

where $N$ is the number of available stocks, $w_i$ is the investment portion of stock $i$, $E(R_p)$ is the expected return of portfolio $p$, $R_f$ is the risk-free rate of return, $\sigma_i^2$ is

the variance of stock $i$, $\sigma_{ij}$ is the corvariance of stock $i$ and $j$, and $\sigma_p$ is the risk of portfolio $p$. It is assumed that $R_f$ used when calculating the sharpe ratio is zero. Even if $R_f$ is zero, there is no significant effect on this chapter because the optimal rebalancing model is a methodology to find a better rebalancing time point given by a portfolio selection method.

### 5.1.2 Learning-to-rank algorithm

Learning-to-rank algorithms are used to compare ranks between items, and are mainly used in areas where ranking is important, such as search engines and recommendation systems. This algorithm learns ranking by machine learning methodology, and is classified into pointwise, pairwise, and listwise approaches. Since the optimal rebalancing model is a methodology for comparing two rebalancing timings, we intend to use a pairwise approach. Among them, we will learn better rebalancing timing with RankNet model(Burges et al., 2005).

RankNet is a model that learns pairwise dataset using neural network algorithm. Let the pair data be $(A, B)$, and the score function values for comparing rankings are $f(A)$ and $f(B)$, respectively. In this case, it is the higher ranking that has a better score function value among A and B. If the higher the score function value is, the better ranking is, the ranking relation for two items $A$ and $B$ is as follows.

$$f(A) > f(B) \Rightarrow A \triangleright B \tag{5.3}$$

RankNet is a neural network model that use the cross-entropy cost function of

Eq. (5.5) for pair data $(A, B)$. The combining probability $P_{AB}$ used for the cross-entropy cost function is a logistic function of the difference between the scores of $A$ and $B$. When the higher score indicates higher rank, it can be calculated as Eq. (5.4). In other words, learning the logistic function of the difference in the output of neural network of each item $A$ and $B$. For example, RankNet model indicates that A is ranked higher than B if $P_{AB}$ is higher than 0.5; otherwise, $B$ is ranked higher than $A$.

$$P_{AB} = P(A \triangleright B) = \frac{e^{f(A)-f(B)}}{1 + e^{f(A)-f(B)}} \tag{5.4}$$

$$C_{AB} = -\overline{P}_{AB} log(P_{AB}) - (1 - \overline{P}_{AB}) log(1 - P_{AB}) \tag{5.5}$$

where $P_{AB}$ is the predicted probability, and $\overline{P}_{AB}$ is the target probability.

Finally, in RankNet, the neural network parameters are determined by gradient descent optimization. If the weight matrix of the neural network in each iteration is $W_n$, the weight matrix $W_{n+1}$ updated in the next iteration is calculated as Eq. (5.6).

$$W_{n+1} = W_n - \eta \times \Delta W_n \tag{5.6}$$

where $\eta$ is the learning rate, and $\Delta W_n$ is the gradient of $W_n$.

### 5.1.3 Proposed Modeling Method

The optimal rebalancing model proposed in this section calculates a better rebalancing time point by using the RankNet model trained with past data. The step-by-step methodology is as follows.

**Construct dataset :** At first, construct the input dataset. To use RankNet as input data, all data in the training set must be converted into pair data. If $X_t$ is a feature vector to be used for RankNet learning at time t, feature vectors at different data point $t_i$, $t_j$ in the learning period are converted into pair data $(X_{t_i}, X_{t_j})$. When the number of data points in the learning section is $N$, $\frac{n(n-1)}{2}$ pair dataset are used as input for training.

This part construct the output of dataset. Let $f(X)$ denotes the score function used when comparing ranks in the pair data. In other words, the score for $X_{t_i}$ is $f(X_{t_i})$. In the constructed input pair dataset, the element with better score is placed in front of the pair and the predicted probability is calculated by Eq. (5.4). The element with better score is placed in front position of the pair. Therefore, if the predicted probability $P_{AB}$ is greater than 0.5, the model is correctly classified. In this way, the RankNet model is learned for all training data and its score function.

**Experiment set-up :** This part describes the experiment set-ups of the RankNet model. For each time point $t$, the experiment proceeds in two portfolio scenarios. As described in Section 5.1.1, the first scenario is an equally weighted portfolio of Eq. (5.1), and the second scenario is a portfolio that maximizes the Sharpe ratio that solves the equation of Eq. (5.2). When constructing the maximize the Sharpe ratio portfolio, the period for obtaining the expected return and standard deviation is 48

weeks (roughly 1 year), which is the same as the period for obtaining the correlation matrix of Chapter 3. The term "portfolio type" is used to describe the portfolio scenario in RankNet model. Then the experimental procedure is performed in the same way for both scenarios. At this time, the feature list to be configured in the input variable $X$ of each time point is as follows.

- Short-term and long-term returns in the market: the log return series of past 1 week, 1 month, 3 months, 6 months and 1 year of S&P500 index.
- Short-term and long-term returns of the top 5 Sharpe ratio stocks: the log return series of the past 1 week, 1 month, 3 months, 6 months, and 1 year of the top 5 stocks of the Sharpe ratio calculated in the past 48 weeks at each time point.
- Fractal Measures: Three fractal measures($FD_t^{Mean}$, $FD_t^{Std}$, and $FD_t^{Fit}$) defined in Chapter 3.

For the above three types of features, the input node consists of four cases of Table 5.1. When the input node is configured as above for all data in a given period, it converts all data into a pair that can be used in RankNet model. For example, when the number of input variables $X$ in the training period is $n$, $\frac{n(n-1)}{2}$ pair form of data is constructed.

The next step is to construct the output variable of RankNet model. To construct the output variable to be used in RankNet model, we need to determine the score function to compare ranks. There are two types of score functions to be used in this chapter: first one is the expected return and sharpe ratio, which is typically used to evaluate portfolio performance(Yu et al., 2014; Almahdi and Yang, 2017;

Table 5.1: **Input variable description for each cases**

| **Variable** | **Input Description** |
|---|---|
| Case1 | Market return series |
| Case2 | Market return series<br>Return series of top 5 Sharpe ratio stocks |
| Case3 | Market return series<br>Fractal measures |
| Case4 | Market return series<br>Return series of top 5 Sharpe ratio stocks<br>Fractal measures |

Chou et al., 2017; Harris et al., 2017; Yu et al., 2020), and the other one is score of the future $m$(= 3 month, 6 month, 1 year) is calculated. Total of six score scenarios are used for this analysis. This chapter refers to the type of score function (Expected Return, Sharpe Ratio) as "Score Type" and the calculation period $m$ of score function as "Score Period". In summary, as described in Section 5.1.2, input is constructed in a pair format, and for each score type and score period, the elements positioned at front are rearranged to have a better score. Then the target probability of the output node, $\overline{P}_{AB}$, will all be one.

As a learning parameter of the RankNet model, the number of training iterations is 200, the learning rate $\eta$ is 0.00001, the number of input nodes is the feature dimension for each case defined in the Table 5.1, and the two hidden layers are structures (16,8) nodes. The number of output nodes is one.

To avoid overfitting problems, the experiment uses a sliding window method. The sliding window method description and the method of dividing training set and

test set are shown in Figure. 5.1. There is a buffer period for each window to consider each training sets includes scores for $m$ future periods. In other words, if data in the buffer period is used for learning, the training set overlaps the test set. For consistency of learning performed for each score period $m$, the length of the buffer period is determined to be 1 year. To ensure the robustness of the model, only 70% of the data that randomly selected from the training set is used. The remaining 30% are used for validation; the average of repeating this process 20 times is used as model performance. The description of each period constituting the sliding window is as follows. The training period is 3 years, the buffer period is 1 year, the testing period is 1 year, and the sliding step size is 1 year.

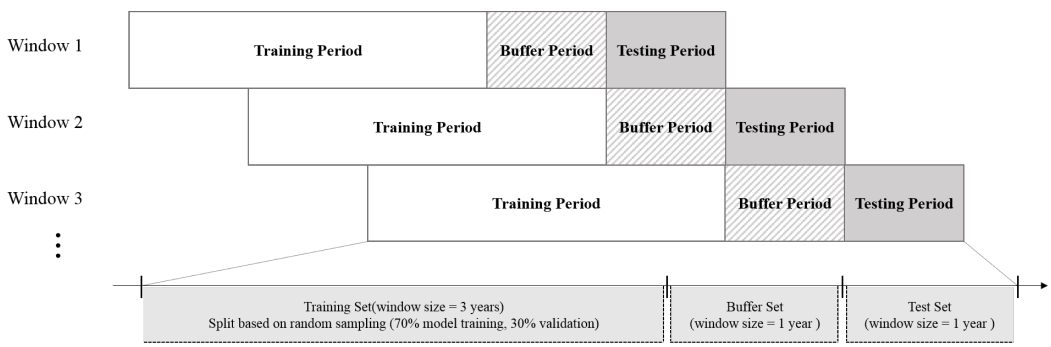


Figure 5.1: **Sliding window structure**

The performance measure of this model is accuracy. A performance measure commonly used in ranking systems is a normalized discounted cumulative gain(Järvelin and Kekäläinen, 2002), which is a particularly effective method for comparing list-wise ranks. On the other hand, this study classifies a time point with a better score among portfolios constructed at two different time points, so it is sufficient to evaluate the model with accuracy. In addition, in order to evaluate the economic effect of

the model, the difference between scores for each pair and the results of the model are compared. The excess score of pair $(A, B)$ is defined as Eq. (5.7).

$$Excess\ Score_{AB} = f(A) - f(B) \tag{5.7}$$

where $f(A)$ and $f(B)$ are scores for $A$ and $B$. For example, suppose that the larger the Score type, the better the function. If the classification is correct ($P_{AB} > 0.5$), $Excess\ Score_{AB}$ will be positive, and if it is incorrect($P_{AB} \leq 0.5$), $Excess\ Score_{AB}$ will be negative. Therefore, $Excess\ Score_{AB}$ is a quantitative measure for evaluating losses in case of incorrect classification.

### 5.1.4 Model Results

In this section, we analyze the average performance of the optimal rebalancing model for the test datasets of each sliding window. We evaluate the performance of the model in two ways. First, we verify that whether the proposed model accurately predicts the better portfolio between future score function of the two portfolios. Second, we analyze the difference in scores of classification results of the proposed model.

The notation used in the results of this section is summarized. The top category of Table 5.2, 5.3, 5.4, and 5.5 is the "score type" to be compared in our model.

RankNet model is a function that compares the ranking. The following categories of score types, represented by Case1, 2, 3, and 4, are input node data used for learning described in Table 5.1. The category (3M, 6M, 1Y) of the row of the table represents the score period. For example, the optimal rebalancing model learned with score type as expected return, score period as 3M, and input node configuration as Case1 is a model with the following characteristics; It uses only market return series as data and classifies the portfolio with higher expected return after 3 months.

**Analyses on equally weighted portfolio :** Table 5.2 summarizes the results of an equally weighted portfolio which the portfolio selection method is calculated as Eq. (5.1). The table results are the average accuracy (%) of the optimal rebalancing models trained 20 times. The baseline performance of the model is 50%. This is because the model learns to compare ranks by combining data from all points in pairs. In the result where the score type is expected return, Case1 shows accuracy rate 50%. In other words, when the model is trained with only market returns as inputs, regardless of the score period, our model cannot classify rebalancing points with better expected returns in the future. In Case2, the model trained for the period of 3 months and 6 months ahead has better performance than both Case1 and baseline performance. This means that the information of stocks with good trend of sharpe ratio contributes to the improvement of predictive power in the model proposed in the future 3 months and 6 months. This result can be noted because an equally weighted portfolio does not consider the weight of each stocks in portfolio. However, since the model results have a certain accuracy improvement, we infer that the trend of individual stock stocks in addition to the market index return series is also meaningful information for classifying rebalancing points. On the other hand, it can

be seen that the model trained with a score period of one year in the future shows a level of performance similar to that of baseline performance. This means that the impact of information on the trends of individual stocks will gradually decrease in future projections after 6 months. In Case3, we observed that the results are similar to Case2. One difference from Case2 is that the accuracy of 6 months is similar to that of 3 months. In addition, it shows significantly higher accuracy rate than Case2. This means that information on individual stock trend is constantly decreased, but the network's fractal measures are well preserved until 6 months. This result is similar to the conclusion of Chapter 4. This is because fractal measures showed the strongest predictive power improvement effect for the about 9 months ahead in the market prediction result of Chapter 4. Case4 used all inputs that showed the best performance compared to other cases. The results with the highest performance in Case4 indicates that the trend of individual stocks and the information of the network topology, represented by fractal measures, help each other. Similar results are observed when the score type is Sharpe ratio. The equally weighted portfolio method constructs the portfolio without going through a specific optimization process. The score type functions are all based on the return series; therefore similar results are obtained. In shorts, In every conclusion of Case2, Case3 and Case4 presented that proposed RankNet model help to classify better rebalancing points in the 3,6 months ahead. In other words, both the trend of individual stocks and the information of the network at the present time contribute to improve predictive power. In addition all additional inputs have little effect on improving predictive power if the input is longer than 1 year periods.

Table 5.3 summarizes the excess score which is difference in the relative score

type function of the portfolios that classified as optimal rebalancing model is better. This result is the average value (%) of excess score in Eq. (5.7). The accuracy result of Table 5.2 is only result of predicting performance, and there is a limitation to use as actual investment and its application. Therefore, it is necessary to use the performance measure that reflects the expected profit to examine the applicability of the actual trading strategy. The larger the expected return and sharpe ratio, the better the performance. Therefore, when the value of Table 5.3 is positive, it succeeded in selecting a better portfolio, and the larger the value, the better the portfolio is classified. For comparison between score periods, we used the annualize scaled results. Since it is a binary classification problem comparing portfolios of two points, the result is similar to the accuracy of Table 5.2. Regardless of the score type, the optimal rebalancing model with input nodes of Case2 and Case3 with a score period of 3 months shows significantly improved results compared to the results of Case1 or baseline. In addtion, if the score period was 6 months, the results of Case3 and Case4 showed good performance. In the results of models with a score type of Sharpe Ratio and a score period of 3 or 6 months, Case3 showed very high performance. According to previous studies, the Sharpe ratio is related to risk because it considers the variance of the portfolio, and we observed the relationship between the stability of the stock market network structure and the proposed fractal measures in Chapter 3. Therefore, it makes sense that fractal measures that represent the stability of the network work effectively in our model to find a better rebalancing point in the future.

**Analyses on maximized Sharpe ratio portfolio :** Table 5.4 summarizes the results for an maximized Sharpe ratio portfolio in which the portfolio selection

Table 5.2: **Accuracy rate of equally weighted portfolio with different parameters**

| | **Score Type : Expected Return(%)** | | | | **Score Type : Sharpe Ratio(%)** | | | |
|---|---|---|---|---|---|---|---|---|
| | Case1 | Case2 | Case3 | Case4 | Case1 | Case2 | Case3 | Case4 |
| 3M | 51.07 | 55.55 | 54.57 | 57.90 | 51.53 | 53.26 | 55.62 | 55.48 |
| 6M | 50.63 | 52.20 | 55.00 | 55.45 | 52.12 | 51.59 | 54.68 | 54.24 |
| 1Y | 49.04 | 50.32 | 48.88 | 51.27 | 47.94 | 50.81 | 47.01 | 49.66 |

Table 5.3: **Excess score of equally weighted portfolio with different parameters**

| | **Score Type : Expected Return(%)** | | | | **Score Type : Sharpe Ratio(%)** | | | |
|---|---|---|---|---|---|---|---|---|
| | Case1 | Case2 | Case3 | Case4 | Case1 | Case2 | Case3 | Case4 |
| 3M | 0.0339 | 0.0561 | 0.0444 | 0.0694 | 0.2506 | 0.2592 | 0.8672 | 0.5237 |
| 6M | -0.0161 | 0.0016 | 0.0115 | 0.0282 | 0.3252 | -0.0001 | 0.4042 | 0.2719 |
| 1Y | 0.0067 | 0.0181 | 0.0124 | 0.0192 | -0.2220 | 0.1715 | 0.0601 | 0.1679 |

method is calculated as Eq. (5.1). Similarly, the average accuracy of the optimal rebalancing model trained 20 times, and each row and column are the same as Table 5.2. In the result where the score type is expected return, results with a score period of 6 months and 1 years did not improve accuracy compared to Case1. On the other hand, when the score period is 3 month, Case2 shows slightly improved accuracy compared to Case1, and Case4 shows improved accuracy. Unlike the previous equally weighted portfolio result, the result of Case3 has no predictive power. In the result where score type is Sharpe ratio, the overall result is similar when the score type is expected return. If the score period is after 6 months, all results do not show good performance. In addition, the result of Case3 shows less than 50% accuracy. On the other hand, when the score period is 3 months, Case2 and Case4 models showed good performance compared to Case1 of model, and Case4 shows the highest performance.

Table 5.4: **Accuracy rate of maximized Sharpe ratio with different parameters**

| | **Score Type : Expected Return(%)** | | | | **Score Type : Sharpe Ratio(%)** | | | |
|---|---|---|---|---|---|---|---|---|
| | Case1 | Case2 | Case3 | Case4 | Case1 | Case2 | Case3 | Case4 |
| 3M | 50.38 | 52.89 | 47.52 | 54.23 | 49.05 | 54.45 | 47.21 | 55.49 |
| 6M | 53.73 | 50.83 | 49.94 | 51.72 | 53.75 | 50.34 | 48.76 | 51.97 |
| 1Y | 54.08 | 49.42 | 52.53 | 51.07 | 49.76 | 48.65 | 45.84 | 48.11 |

Table 5.5: **Excess score of maximized Sharpe ratio with different parameters**

| | **Score Type : Expected Return(%)** | | | | **Score Type : Sharpe Ratio(%)** | | | |
|---|---|---|---|---|---|---|---|---|
| | Case1 | Case2 | Case3 | Case4 | Case1 | Case2 | Case3 | Case4 |
| 3M | -0.0011 | 0.0338 | 0.0123 | 0.0387 | 0.0439 | 0.5271 | -0.1330 | 0.6398 |
| 6M | 0.0135 | 0.0177 | -0.0169 | 0.0099 | 0.1206 | 0.1550 | -0.1595 | 0.3477 |
| 1Y | 0.0071 | 0.0069 | 0.0021 | 0.0043 | -0.1054 | -0.0677 | -0.1615 | -0.1130 |

Table 5.5 summarizes the excess score of different portfolios that classified by the optimal rebalancing model. Similar to the result of the previously equally weighted portfolio, it shows a similar pattern to Table 5.4. Importantly, in Case3, all of them have poor performance, as is the accuracy result. This result is because fractal measures represent the structure of the entire network, while the maximized Sharpe ratio portfolio consists of only a few stocks. The difference from Table 5.4 is that Case2 and Case4 show performance improvement in both score types when the score period is 6 months. In other words, when using the proposed optimal rebalancing model for investment, investors must make decisions in consideration of both accuracy and excess score.

In summary, the optimal rebalancing model proposed in both portfolio selection methods improved accuracy when we used 3 months as score period. In addition, when we used the score period with 6 months, accuracy was improved only with the

equally weighted portfolio, and when we used the score period with 1 year, there are no significant improvement in accuracy. In other words, we thought that information from past data affects prediction no more than 3 months, and then gradually lose the information. In addition, despite the model is considered regularization, the Case4 results showed the highest accuracy in most of the two portfolio scenarios. These results indicate that the proposed fractal measures improve performance when it used with other meaningful prediction variables. We will compare the results of Case2 and Case3 in the two portfolio scenarios. In the equally weighted portfolio result, Case3 performed better than average. On the other hand, in the maximized Sharpe ratio portfolio result, Case2 showed good accuracy but Case3 showed lower accuracy than baseline performance. This is because the equally weighted portfolio evenly distributes all assets at every rebalancing point regardless of past trends. Conversely, the maximized Sharpe ratio protfolio consisted mostly with top 5 stocks added in Case2. It is concluded that the performance of the proposed optimal rebalancing model is improved when it construct with variables that are useful for prediction. In the maximized Sharpe ratio scenario, which is less relevant to the overall structure of the stock market, the result of Case4 has the highest performance. This implies that the proposed fractal measures have predictive power when it used together with other variables.

## 5.2 Simulation Analysis

### 5.2.1 Simulation Structure

Using the RankNet model trained in the previous Section 5.1, it is possible to classify points with a better score function, $f(X)$, for two different time points. If the latest rabalancing point is $t_R$ and the current point is $t$, $P(X_t, X_{t_R})$ in Eq. (5.4) can be calculated using the trained RankNet model. The probability that the score is better among the latest rebalancing points and the current points at each time point will be calculated. The higher this probability value, the model judges that the current point $t$ is better for rebalancing than the latest rebalancing point $t_R$. At this time, the rebalancing level can be determined by setting the threshold probability $\delta$, which is a key parameter of the trading strategy. The minimum value of $\delta$ is 0.5, and the bigger the $\delta$, the more strict rebalancing point is obtained. If this probability is greater than the threshold probability $\delta$, the rebalancing process is performed using the prescribed portfolio selection method discussed in Section 5.1.1. The proposed trading strategy can be ambiguous in that it compares the score functions of different time periods. However, in general, when previous portfolio study evaluated performance, the expected return or Sharpe ratio is used in terms of unit period. Similarly, the proposed trading strategy compares portfolios for different time periods, but it is also possible to compare the performance of unit periods. Finally, the latest rebalancing point is updated to the current point and this process is repeated during the simulation period.

Figure.5.2 shows the above process. The proposed trading strategy is expected

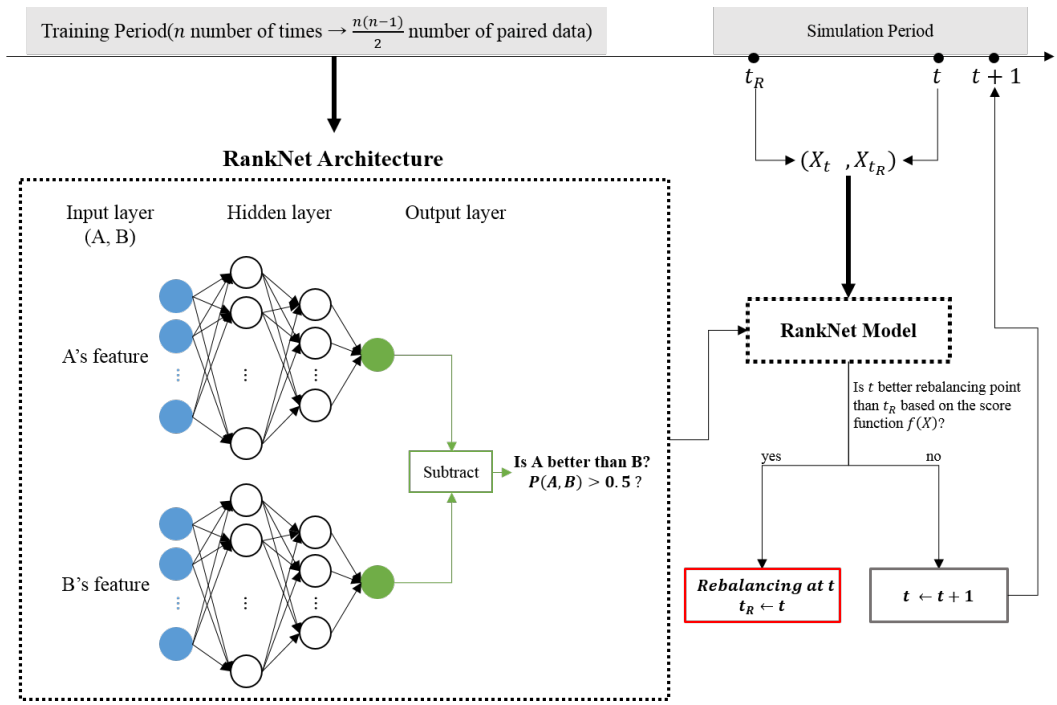


Figure 5.2: **Simulation structure of trading strategy**

to be easily applied to other studies.

The reason is that when the portfolio selection method and the target score function are determined, the proposed trading strategy works by predicting the rebalancing point with a better score in the future with the trained RankNet model. That is, it can be applied regardless of the methodology to the portfolio selection methodology in which a lot of research is being conducted. Also, since the output node of the RankNet model is composed of the desired target score function, the proposed trading strategy will be more suitable for the investor.

**Experiment set-up:** Trading simulation is conducted from 1999-01-01 to 2019-12-31. First, the data of the first 48 weeks in the period is used to construct a network for calculating Fractal measures; time points that excluded at this period is called $t = \{1, 2, ..., N\}$. Construct feature data $X$ required at all times within this period. In this chapter, the RankNet model used in trading simulation is trained for a period of 3 years, from 4 years to 1 year ago. As in the previous section, the score type functions $f(X)$ are the expected return and sharpe ratio, and the score periods $m$ are 3 months, 6 months, and 1 year. As a result, a total of 6 scenario trading simulations are conducted. The detailed trading simulation process is described in Algorithm1.

In the 14th line of Algorithm 1, the reason for rebalancing due to the expiration of the score period is that the score used to obtain a better rebalancing point is considered as the score type function in the future score period $m$. In other words, since RankNet model is to find better rebalancing point during the $m$ period of time, if there is no better rebalancing point found during the score period after rebalancing, it is reasonable to repeat the algorithm process after rebalancing by

**Algorithm 1:** Trading simulation process

**Data:** Input data $X_t$, $t = \{1, 2, .., N\}$, rebalancing probability $\delta$
**Result:** List of rebalancing points

Set portfolio selection algorithm
Set score function $f_{m,s}$ where $s$ = score type , $m$ = score period
$P_{AB} \leftarrow$ train RankNet in $(0, 3year]$
Recent Rebalance Point, $R \leftarrow 4year$(buffer period 1 year)
Initial portfolio composition at R
**for** $t = R + 1$ *to* $N$ **do**
    Calculate $P(X_t, X_R)$
    **if** $P(X_t, X_R) > \delta$ **then**
        Rebalancing at $t$
        $P_{AB} \leftarrow$ retrain RankNet in $(t - 4year, t - 1year]$
        $R \leftarrow t$
    **end**
    **if** $t - R = m$ **then**
        Rebalancing at t(due to expiration of score period)
        $P_{AB} \leftarrow$ retrain RankNet in $(t - 4year, t - 1year]$
        $R \leftarrow t$
    **end**
**end**
**Return** All rebalancing point

expiration. Also, in terms of investment strategy, since consistent results must always be guaranteed regardless of learning iteration, the model used in the trading strategy is the ensemble model of RankNets that learned 20 times at each time point.

Finally, we considered investor preference as a key parameter called $\delta$. $\delta$ is a parameter that determines rebalance when the calculated $P(X_t, X_R)$ exceeds a certain value. If $\delta$ is a large value, rebalancing will proceed only at the point where the model is classified as more certain, and if $\delta$ is a small value, the reliability of the model results will decrease slightly, but frequent rebalancing will proceed. If the number of rebalancing proceeding is increased, investors have to decide the appropriate level of $\delta$ because the it will increase the transaction cost. However, it is not possible to find the optimal $\delta$. This is because it is a parameter that varies by on RankNet model result and investor preferences. Therefore, in this section, the experiment was conducted with $\delta = \{0.5, 0.7, 0.9\}$.

### 5.2.2 Simulation Results

In this section, we examine the simulation results of the trading strategy using the proposed optimal rebalancing strategy. We verify the trading strategy in eight equally weighted portfolios and maximized Sharpe ratio portfolio scenarios, with a total of 8 parameter sets that performed well in Section 5.1.3. The detailed parameter set is shown in Table 5.6.

Before the simulation experiment, Table 5.7 represents the number of predictions that satisfies $P_{AB} > \delta$ when $\delta$ is changed in the optimal rebalancing model experiment in the previous section. In addition, the conditional excess score repre-

Table 5.6: **Description of selected parameters**

| **Portfolio Algorithm** | **Score Type** | **Input Variable** |
|---|---|---|
| Equally weighted | Expected Return | Case2 |
| | | Case4 |
| | Sharpe Ratio | Case3 |
| | | Case4 |
| Maximized Sharpe ratio | Expected Return | Case2 |
| | | Case4 |
| | Sharpe Ratio | Case2 |
| | | Case4 |

sents the average value of the excess score for the prediction that satisfies the above situation. As shown in the previous section, the reason for using the conditional excess score without using the average value of the excess score is that there is no guarantee that the optimal rebalancing model is symmetric when it is not $\delta = 0.5$. It cannot be guaranteed that $P(AB) = 1 - P(BA)$. Therefore, we used the average of conditional excess scores to measure the performance to determine better portfolio with different $\delta$.

As a result, the stricter standard $\delta$ reduces the number of rebalancing points. In addition, according to the conditional excess score, it can be seen that as $\delta$ increases, classify portfolios with good average performance. In terms of investment, rebalancing at a more certain point reduces the frequency of opportunities, but finds a rebalancing point that is expected to be better in the future. We verified that the key parameter $\delta$ works as intended by Section 4.2.1. Investors can choose different strategies by adjusting $\delta$ according to their preferences. In this experiment, we analyze with $\delta = 0.5, 0.7, 0.9$.

We aim to calculate a better rebalancing point for a given portfolio selection

Table 5.7: **Relationship between $\delta$ and optimal rebalancing model**

| **Portfolio algorithm** | **Score Type** | **Input Variable** | | $\delta$ | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 |
| Equally weighted | Expected Return | Case2 | Count(%) | 55.55 | 49.02 | 42.22 | 34.70 | 25.15 |
| | | | Conditional Excess Score | 0.0759 | 0.0776 | 0.0795 | 0.0815 | 0.0815 |
| | | Case4 | Count(%) | 57.90 | 50.11 | 42.31 | 34.28 | 23.22 |
| | | | Conditional Excess Score | 0.0770 | 0.0790 | 0.0814 | 0.0841 | 0.0880 |
| | Sharpe Ratio | Case3 | Count(%) | 55.62 | 39.42 | 25.08 | 12.57 | 3.64 |
| | | | Conditional Excess Score | 2.3031 | 2.3280 | 2.3814 | 2.4229 | 2.4970 |
| | | Case4 | Count(%) | 55.48 | 47.45 | 39.10 | 30.04 | 18.97 |
| | | | Conditional Excess Score | 2.2272 | 2.2513 | 2.2808 | 2.3204 | 2.3801 |
| Maximized Sharpe ratio | Expected Return | Case2 | Count(%) | 52.89 | 46.44 | 39.84 | 32.08 | 22.84 |
| | | | Conditional Excess Score | 0.0644 | 0.0651 | 0.0659 | 0.0677 | 0.0697 |
| | | Case4 | Count(%) | 54.23 | 46.88 | 39.11 | 29.94 | 19.38 |
| | | | Conditional Excess Score | 0.0648 | 0.0660 | 0.0676 | 0.0693 | 0.0736 |
| | Sharpe Ratio | Case2 | Count(%) | 54.45 | 47.12 | 39.47 | 31.13 | 21.46 |
| | | | Conditional Excess Score | 2.1923 | 2.2117 | 2.2375 | 2.2726 | 2.3006 |
| | | Case4 | Count(%) | 55.49 | 47.62 | 39.48 | 30.19 | 19.29 |
| | | | Conditional Excess Score | 2.2146 | 2.2631 | 2.3041 | 2.3712 | 2.4587 |

method. As a benchmark for performance comparison, the S&P 500 buy and hold strategy and the rebalanced strategy at regular intervals of 1 month, 3 months, and 6 months are used for a given portfolio selection method. In addition to the 3 month score period where the optimal rebalancing model performed best, the benchmark strategies for 1 and 6 months were also considered because the performance of the simulation may be related to the number of rebalancing. Return (%), volatility (%), and Sharpe Ratio are used as performance measures of the simulation.

Table 5.8 is the result of an equally weighted portfolio. You can see that the performance of benchmarks rebalancing at regular intervals is better than buy and hold in S&P500. As a result, we could see that evenly distributing the assets among the available stocks would result in a diversified portfolio of idosyncratic risks. This is the same result as claimed in the modern portfolio theory. In addition, when we compare the performance of the benchmarks, the return result is the best in 1 month, which is rebalanced with the most frequent cycle in the experiment. The Sharpe ratio results show similar results regardless of the rebalancing period. Overall, we observed that our proposed model is similar or slightly better results than 1 month-benchmark model. Specifically, the proposed strategy (Expected Return, Case4) in the case of $\delta = 0.7$ showed the best performance among the models that proposed by return and Sharpe ratio, while the number of rebalancing was much smaller than that of the 1 month benchmark. The parameters of this proposed strategy are the most excellent parameters in Table 5.2. It is consistent with the results of the previous optimal rebalancing model. The performance of the benchmarks and proposed strategies does not differ much, because the given portfolio selection method is a very simple. Therefore, in the proposed strategies, the composition of the

portfolio perfectly matches the benchmark strategy, and there is only an additional advantage to the proposed model with rebalancing time points. In addition, since all the proposed strategies have less rebalancing points than the 1 month benchmark, if the transaction cost is considered, the actual performance of the proposed strategies will be better.

Table 5.9 is the result of maximize Sharpe ratio portfolio. In this table we can see that the performance of the benchmark strategies that periodically rebalancing assets is better than that of the buy and hold S&P500 strategy. When we compare the performance of benchmarks, the 1 month benchmark strategy shows the best results. This is because the maximize Sharpe ratio portfolio is constructed with stocks with good trends in the past. It can be inferred that the past trend reflects more effectively as the rebalancing frequency increases. Compared to the best-performing 1 month benchmark strategy, the rebalancing strategies with $\delta = 0.9$ performed well on average. The rebalancing strategy of $\delta = 0.7$ has similar performance to the 1 month benchmark strategy, but less than that of $\delta = 0.5$. Among the rebalancing strategies with $\delta = 0.9$, we observed that the proposed model (Sharpe Ratio, Case4) has the best Sharpe ratio. This result is very encouraging from two perspectives. First, the case where the score type is Sharpe Ratio and the input variable is Case4 is the parameter set with the best performance in Section 5.1.4. Second, the "Maximized Sharpe ratio portfolio" is a portfolio consisting of stocks that literally maximize the Sharpe ratio. Since the optimal rebalancing model we used is learned to have a good rank as the score type (in this case, the Sharpe ratio) is large, a high Sharpe ratio performance is a consistent result that passes through the flow of the entire experimental logic. In addition, when the maximized Sharpe ratio portfolio

Table 5.8: **Performance of equally weighted portfolio strategy**

| | $\delta$ | **Rebalancing freq.** | **Return(%)** | **Volatility(%)** | **Sharpe Ratio** |
|---|---|---|---|---|---|
| Buy and Hold S&P500 | - | 0 | 7.92 | 15.58 | 0.5084 |
| Benchmark_1M | - | 204 | 14.31 | 17.29 | 0.8279 |
| Benchmark_3M | - | 68 | 13.99 | 16.99 | 0.8233 |
| Benchmark_6M | - | 34 | 13.86 | 16.86 | 0.8220 |
| Proposed(Expected Return, Case2) | 0.5 | 176 | 14.38 | 17.31 | 0.8304 |
| | 0.7 | 129 | 14.41 | 17.33 | 0.8310 |
| | 0.9 | 104 | 14.17 | 17.09 | 0.8294 |
| Proposed(Expected Return, Case4) | 0.5 | 161 | 14.28 | 17.16 | 0.8321 |
| | 0.7 | 139 | 14.43 | 17.30 | 0.8342 |
| | 0.9 | 110 | 14.42 | 17.31 | 0.8331 |
| Proposed(Sharpe Ratio, Case3) | 0.5 | 187 | 14.36 | 17.34 | 0.8279 |
| | 0.7 | 97 | 14.18 | 17.15 | 0.8270 |
| | 0.9 | 74 | 14.00 | 17.02 | 0.8223 |
| Proposed(Sharpe Ratio, Case4) | 0.5 | 170 | 14.29 | 17.13 | 0.8338 |
| | 0.7 | 130 | 14.36 | 17.28 | 0.8309 |
| | 0.9 | 93 | 14.17 | 17.14 | 0.8269 |

strategy for $\delta = 0.9$ is significantly different from the benchmark strategy when compared to the results of the equally weighted portfolio, the composition of the stocks in the portfolio is also different in that case. In other words, it can be considered that the rebalancing effect is greater for this strategy because the portfolio is optimized for every different rebalancing time points. Likewise, the proposed strategies are actually better than the results shown in Table 5.9 when we consider transaction cost because the number of rebalancing is less than that of 1 month benchmark strategy.

Figure 5.3 shows that the rebalancing frequency results of equally weighted portfolio. The x-axis represents the portfolio holding period, and the y-axis is frequency. At $\delta = 0.5$, the rebalancing periods of 1 week has the most frequency. As the rebalancing period increased, the frequency is gradually decreased. However, when the rebalancing period is 12 weeks, the frequency increases again because when it cannot find better rebalancing time point until 12 weeks, it automatically rebalance it. Similar trends are observed for strategies at $\delta = 0.7$ and 0.9. The difference is that the frequency decreases as $\delta$ increases, except when the rebalancing period is 12 weeks. On the other hand, when the rebalancing period is 12 weeks, increasing in $\delta$, leads the rebalancing frequency. This result means that rebalancing due to the expiration occurs more frequently, so the meaning of the key parameter $\delta$ is well reflected in the model. Figure 5.4 shows that the rebalancing frequency results of maximized Sharpe ratio portfolio. Overall, the results are similar to the previous analysis. As a result, regardless of the portfolio selection method, the optimal rebalancing model proposed in the simulation works in a similar way.

Figure 5.5 shows that the simulation results of the strategies with the best per-

Table 5.9: **Performance of maximized Sharpe ratio portfolio strategy**

| | $\delta$ | **Rebalancing freq.** | **Return(%)** | **Volatility(%)** | **Sharpe Ratio** |
|---|---|---|---|---|---|
| Buy and Hold S&P500 | - | 0 | 7.92 | 15.58 | 0.5084 |
| Benchmark_1M | - | 204 | 11.51 | 15.40 | 0.7475 |
| Benchmark_3M | - | 68 | 10.89 | 15.76 | 0.6911 |
| Benchmark_6M | - | 34 | 10.65 | 16.09 | 0.6623 |
| Proposed(Expected Return, Case2) | 0.5 | 184 | 11.07 | 15.50 | 0.7142 |
| | 0.7 | 124 | 11.49 | 15.48 | 0.7424 |
| | 0.9 | 108 | 11.80 | 15.61 | 0.7559 |
| Proposed(Expected Return, Case4) | 0.5 | 177 | 11.64 | 15.69 | 0.7419 |
| | 0.7 | 133 | 11.67 | 15.66 | 0.7457 |
| | 0.9 | 102 | 11.95 | 15.64 | 0.7636 |
| Proposed(Sharpe Ratio, Case2) | 0.5 | 177 | 11.35 | 15.96 | 0.7109 |
| | 0.7 | 116 | 11.65 | 15.68 | 0.7430 |
| | 0.9 | 106 | 12.48 | 15.60 | 0.7998 |
| Proposed(Sharpe Ratio, Case4) | 0.5 | 176 | 11.18 | 15.63 | 0.7153 |
| | 0.7 | 130 | 11.92 | 15.69 | 0.7598 |
| | 0.9 | 94 | 12.59 | 15.37 | 0.8191 |

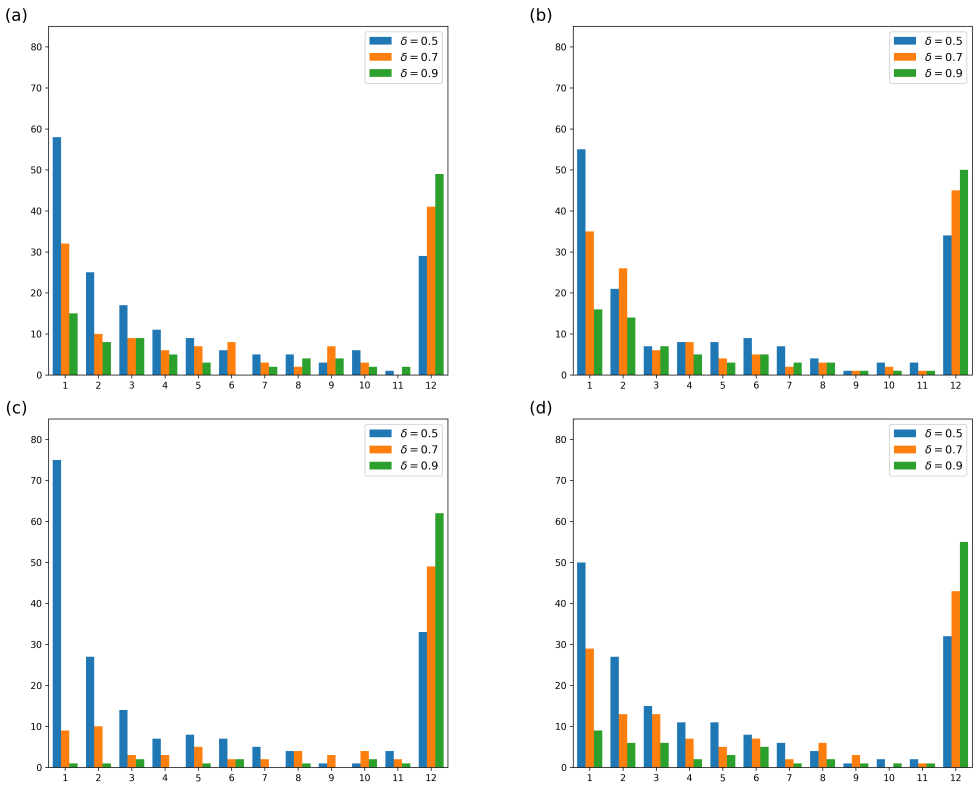


Figure 5.3: **Rebalancing frequency results of each trading strategies(Equally weighted portfolio)**

(a) Proposed strategy(Expected Return, Case2)
(b) Proposed strategy(Expected Return, Case4)
(c) Proposed strategy(Sharpe ratio, Case3)
(d) Proposed strategy(Sharpe ratio, Case4)

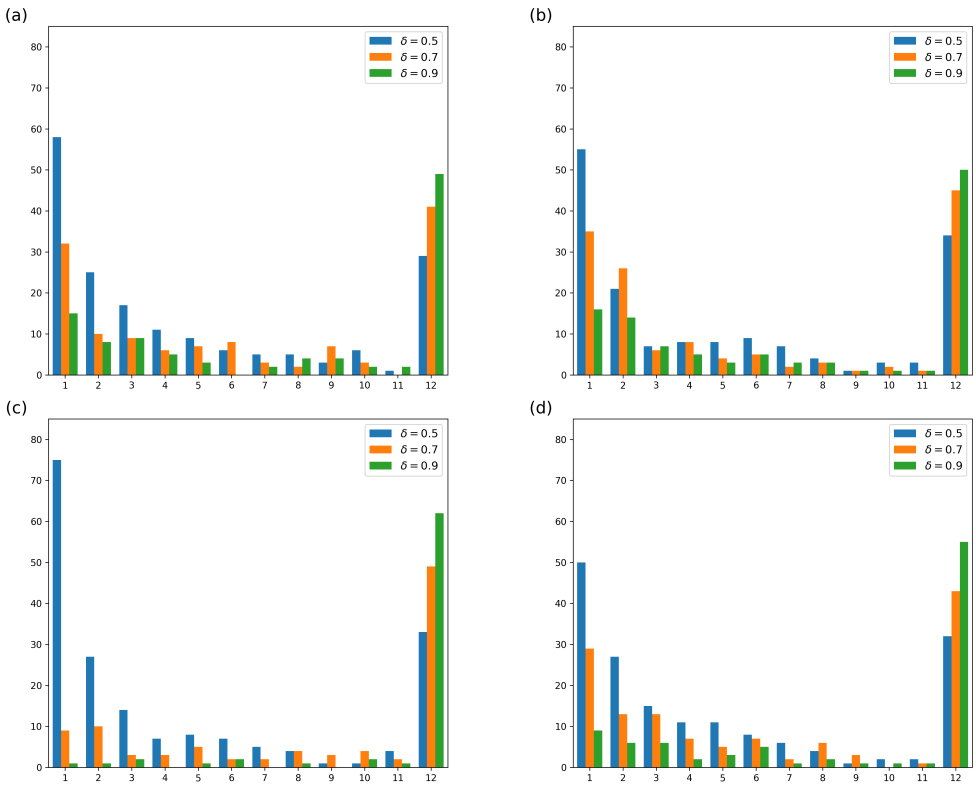


Figure 5.4: **Rebalancing frequency results of each trading strategies(Maximized Sharpe ratio portfolio)**

(a) Proposed strategy(Expected Return, Case2)
(b) Proposed strategy(Expected Return, Case4)
(c) Proposed strategy(Sharpe ratio, Case2)
(d) Proposed strategy(Sharpe ratio, Case4)

formance in each portfolio selection scenario and the comparison with the 1 month benchmark strategies. Fig. 5.5 (a) shows that the simulation result of $\delta = 0.7$ for the strategy with the highest performance (Expected return, Case4) in the equally weighted portfolio. The blue line indicates the rebalancing point of the 1 month benchmark strategy, and the red line indicates the rebalancing point of the proposed strategy. The number of rebalancing points of the proposed strategy is smaller than that of month benchmark strategy and has a slightly better balance than the benchmark in almost all periods. These strategy results are slightly improved on performance, and we used exactly the same portfolio selection method except using rebalancing points.

Fig. 5.5 (b) shows that the simulation result of $\delta = 0.9$ for the strategy with the highest performance (Sharpe ratio, Case4) in the maximized Sharpe ratio portfolio. Since $\delta$ is bigger than the above strategy, the rebalancing frequency of the proposed model (94 times) is less than half the frequency of the 1month benchmark strategy. However, it shows a larger balance difference compared to the 1 month benchmark strategy. In particular, the balance gap widened from the beginning of 2008 when the financial crisis began. This result is due to the effect of changes in portfolio composition stocks as discussed above.

In summary, both portfolio selection scenarios showed excellent performance in simulation experiments with a good performance parameter set in Section 5.1. Particularly, we observed that there is a better effects when the performance measure of the strategy is used with the right score type of the optimal rebalancing model. Since results of the training set and the consistency of results from the optimal

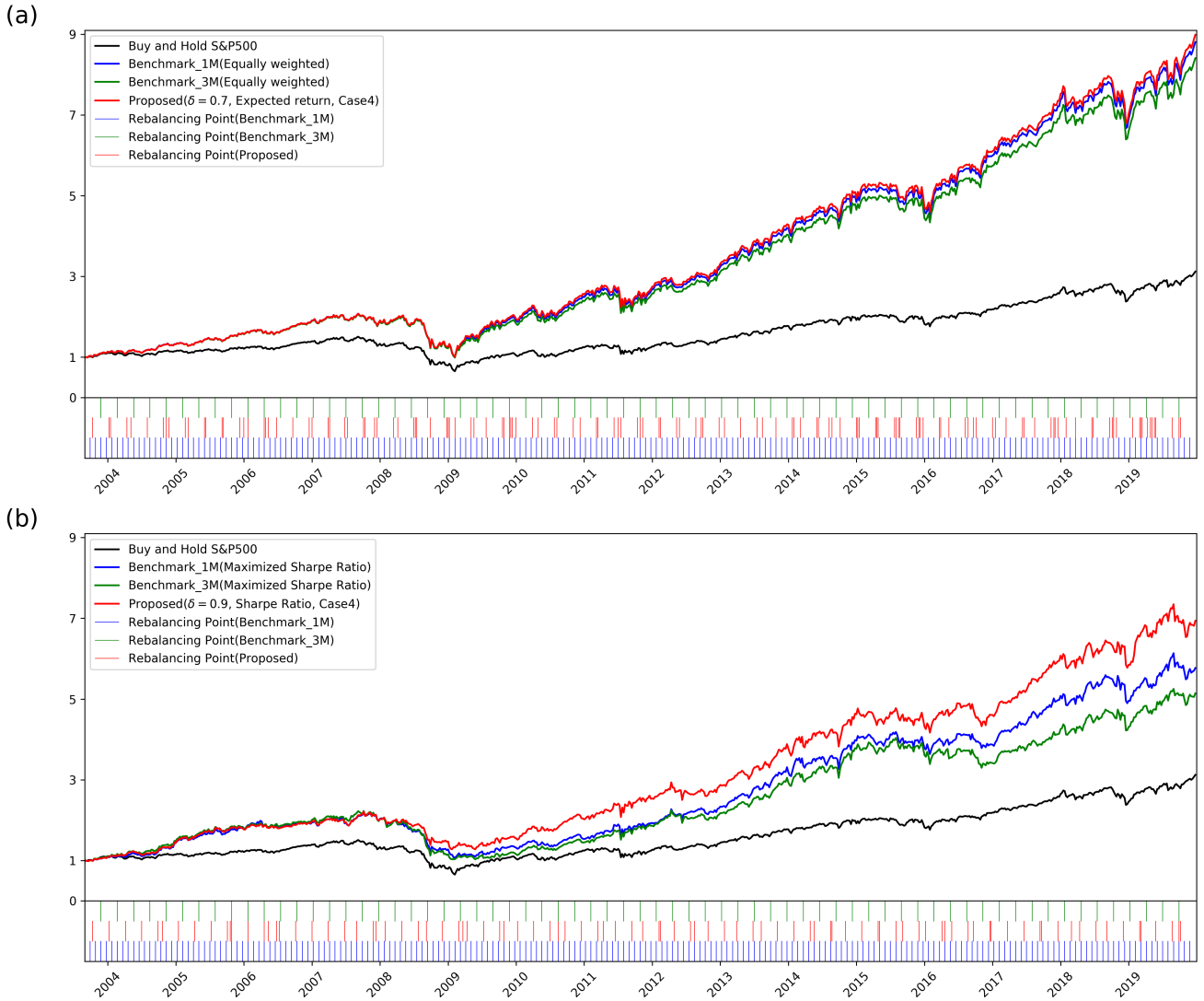


Figure 5.5: **Simulation results of each trading strategies and benchmark**

(a) Proposed strategy(Equally weighted portfolio, $\delta = 0.7$, Score type = Expected return, Input variable = Case4)

(b) Proposed strategy(Maximized Sharpe ratio portfolio, $\delta = 0.9$, Score type = Sharpe ratio, Input variable = Case4)

rebalancing model are introduced, investors can conduct trading simulations based on the performance evaluation results of models that trained with past data. Using rebalancing model performs better between the same portfolio selection method. The essence of our proposed strategy is that it can be applied to any portfolio selection method. Finally, if $\delta$ is 0.5, the proposed strategy will rebalance more frequently. Therefore, we recommend using $\delta$ higher than 0.5, depends investors' tendencies.

## 5.3 Summary and Discussion

In this chapter, we proposed a RankNet-based rebalancing strategy which identifies the optimal rebalancing time point using fractal measures. The key contribution is to develop a strategy to strengthen a given portfolio selection method by utilizing the proposed RankNet framework. To implement this strategy, input data related to portfolio selection, score type, score period, and future stock market forecasts was prepared.

Results are organized in three ways. First, an optimal rebalancing model was proposed to compare the prediction performance of portfolios of each models. Based on the result, we verified that it is possible to classify a better rebalancing point for a given portfolio selection methods. We also considered the economic significance of the model by analyzing the excess score. Second, a rebalancing strategy using an optimal rebalancing model was developed. Based on simulation analysis, the proposed rebalancing strategy outperformed than benchmark strategies with the same portfolio selection method. This results support our claim since the investment strategies can be developed using any portfolio selection strategies from previously studied. Finally, we showed that the fractal measures proposed in Chapter 3 are practical indicators that can be used in trading strategies. Based on results in this chapter, we can claim that models with fractal measures perform better than models without fractal measures.

We consider three limitations in this chapter. First, we assumed the consisting portfolio with fraction of stocks is possible in both scenarios of equally weighted portfolio and maximized Sharpe ratio portfolio. In other words, to consider practi-

cal position of stock purchase, the mixed-integer problem must be solved. However, among many previous studies, theoretical studies assumed an ideal situation similar to our portfolio selection method. Therefore, we also proceeded with an analysis that assumed purchasing fraction of stock is possible. Second, the study was conducted with stocks that had no defaults during the experiment period. The main purpose of the study is to evaluate the utility of proposed model. Therefore, we used same stock sets for every investment strategies. In addition, all experiments can be operated in a sliding window method composed of unit periods, so it is also possible to use them in practice. Lastly, various portfolio selection methods could be applied. However, in order to rigorously analyze each portfolio selection method, a standardized analysis method is needed to build with metadata and compare methods with different input variables. In order to solve these difficulties, previous studies related to portfolio selection also evaluate their proposed model based on the most representative strategies. Therefore, this chapter used the most widely used equally weighted portfolio and maximized shape ratio among portfolio selection methods. As future works, we will integrate several latest portfolio method by organizing several stock market related indicators and technical indicators into metadata.

# Chapter 6

# Conclusion

## 6.1 Conclusions

For the past several decades, financial markets have been extensively studied in combination with a variety of disciplines. Beginning with economics, countless studies have been conducted by combining theories of mathematics, statistics, and physics. Recently, due to the increase in availablity of data and computing power, machine learning techniques have become a trend in analyzing financial markets. Regardless of the field of study, research through the development of indicators that characterize the financial market is very concise and efficient. From technical indicators representing financial markets, indicators developed in mathematics, statistics, and physics, previous studies suggested new ways of research in financial markets, or acted to strengthen existing research. In addition, the indicators implying the characteristics of complex financial markets are efficient in terms of scalability of the research because of the advantage of being easily applicable to machine learning methodology. The goal of this study is to propose indicators to extract characteristics of financial markets that were not previously considered, and empirically analyze its effect on financial markets using them.

In this dissertation, we analyze the S&P500 network, one of the representatives global stock market, based on its network fractality. The research is conducted in the following steps. First, we proposed the concept of a correlation-based threshold network based on minimum spanning tree. Secondly, we investigated the fractal dimension of threshold networks and proposed suitable fractal measures($FD_t^{Mean}$, $FD_t^{Std}$, $FD_t^{Fit}$). Thirdly, we analyzed the S&P500 network based on the proposed measures and utilize them in the market prediction. Lastly, we proposed an optimal rebalancing model and trading strategy using the RankNet algorithm and fractal characteristics which describe the network topology of the S&P500.

Based on the results, we discover the self-similarity characteristic of the S&P500 network, and a strong effective repulsion phenomenon between hubs was detected by $FD_t^{Mean}$. Furthermore, we observe the different shortcut patterns of S&P500 network by $FD_t^{Std}$. In this study, we believe that the properties of these two network topologies could be useful for analyzing the stock market, especially for predicting stock market. Since we can see how effectively the positive and negative effects can propagate on the stock market by observing the repulsion phenomenon between hubs, we conjecture it would be closely related to the potential upward and downward trend in the stock market in the future. Also, if the shortcut pattern is different, the classification criteria of the stock market sector may be ambiguous, so it is essential to quantify this pattern in advance to analyze the structure of the stock market. Also, $FD_t^{Fit}$ explains the stability of the S&P500 network structure with its robustness. If there is a serious fluctuation in the value of $FD_t^{Fit}$, it is expected that the fractality of network may be broken, causing serious structural changes in the market.

There are two main experimental results for the market predictions. First, using logistic regression and random forest, we found that fractal measures improve the predictive power of the future S&P500 index. Second, we utilize the measures in the prediction of the cumulative log-return of S&P500 index through the simple artificial neural network and detect the improvement of prediction performance in the long-term development of the market. Our conclusion that $FD_t^{Mean}$ has the effect of improving predictive power of the S&P500 index, that can be inferred that structural stability of the network and changes is related to the future market. Furthermore, it can be inferred that the proposed fractal measures represent the two network topologies mentioned Chapter 3 are significant.

The results for the optimal rebalancing model and proposed trading strategy are as follows. First, we confirmed that the current network structure affects portfolio composition when optimal rebalancing model constructed by fractal measures. In particular, we showed that the performance of the optimal rebalancing model contains the fractal measures outperformed the benchmark; therefore, fractal measures should be to consider to be included in network structure. Second, a trading strategy was proposed and evaluated by simulation. As a result, we showed that the traditional two portfolio selection methodologies were strengthened by the proposed trading strategy. Lastly, we argued that our trading strategy can be easily applied to various general portfolio selection methodologies, and it can be a more enhanced strategy by using appropriate additional data.

## 6.2 Future Works

This dissertation can be develop in following ways. First, we could use our proposed model to different stock market so that we can compare the performance of the model from each different stock market. In this dissertation, the analysis was conducted with the S&P500 market, the most representative stock market most used in research, therefore, we assumed meaningful results can be obtained if the same analysis is performed on the stock market with different characteristics. Second, we intend to propose a machine learning model that obtains high predictive performance by constructing a dataset that includes technical indicators that contain various meanings in the financial field. Since the main purpose of this study was to insist that the proposed fractal measures have an effect of improving predictive power, we used simple models and essential datasets rather than using complex machine learning algorithms and various datasets. However, in order to practically use in stock market, it would be necessary to construct with various datasets and advanced models. Finally, we tried to apply an optimal rebalancing model to a wider range of trading strategies. In this dissertation, the proposed optimal rebalancing model was verified by using only stock prices and fractal measures for the two portfolio composition methodologies that most frequently used in related studies. However, the actual trading strategies used in the stock market will prefer hybrid trading strategies. Combined with the first proposed future work, if we develop a trading strategy that can flexibly applied to an existing trading strategy, market participants will be able to make better decision.

# Bibliography

Almahdi, S. and Yang, S. Y. (2017). An adaptive portfolio trading system: A risk-return portfolio optimization using recurrent reinforcement learning with expected maximum drawdown. *Expert Systems with Applications*, 87:267–279.

Ballings, M., Van den Poel, D., Hespeels, N., and Gryp, R. (2015). Evaluating multiple classifiers for stock price direction prediction. *Expert Systems with Applications*, 42(20):7046–7056.

Barabási, A.-L. and Bonabeau, E. (2003). Scale-free networks. *Scientific american*, 288(5):60–69.

Berkson, J. (1944). Application of the logistic function to bio-assay. *Journal of the American statistical association*, 39(227):357–365.

Blitz, D. C. and Van Vliet, P. (2007). The volatility effect. *The Journal of Portfolio Management*, 34(1):102–113.

Boginski, V., Butenko, S., and Pardalos, P. M. (2005). Statistical analysis of financial networks. *Computational statistics & data analysis*, 48(2):431–443.

Boginski, V., Butenko, S., and Pardalos, P. M. (2006). Mining market data: a network approach. *Computers & Operations Research*, 33(11):3171–3184.

Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.

Bunde, A. and Havlin, S. (2012). *Fractals and disordered systems*. Springer Science & Business Media.

Bunde, A. and Havlin, S. (2013). *Fractals in science*. Springer.

Burges, C., Shaked, T., Renshaw, E., Lazier, A., Deeds, M., Hamilton, N., and Hullender, G. (2005). Learning to rank using gradient descent. In *Proceedings of the 22nd international conference on Machine learning*, pages 89–96.

Burges, C. J., Ragno, R., and Le, Q. V. (2007). Learning to rank with nonsmooth cost functions. In *Advances in neural information processing systems*, pages 193–200.

Cao, Z., Qin, T., Liu, T.-Y., Tsai, M.-F., and Li, H. (2007). Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th international conference on Machine learning*, pages 129–136.

Caraiani, P. (2014). The predictive power of singular value decomposition entropy for stock market dynamics. *Physica A: Statistical Mechanics and its Applications*, 393:571–578.

Chai, S. H. and Lim, J. S. (2016). Forecasting business cycle with chaotic time series based on neural network with weighted fuzzy membership functions. *Chaos, Solitons & Fractals*, 90:118–126.

Chou, Y.-H., Kuo, S.-Y., and Lo, Y.-T. (2017). Portfolio optimization based on funds standardization and genetic algorithm. *IEEE Access*, 5:21885–21900.

Clarke, R. G., De Silva, H., and Thorley, S. (2006). Minimum-variance portfolios in the us equity market. *The Journal of Portfolio Management*, 33(1):10–24.

Coletti, P. (2016). Comparing minimum spanning trees of the italian stock market using returns and volumes. *Physica A: Statistical Mechanics and its Applications*, 463:246–261.

Costenbader, E. and Valente, T. W. (2003). The stability of centrality measures when networks are sampled. *Social networks*, 25(4):283–307.

Cox, D. R. (1958). The regression analysis of binary sequences. *Journal of the Royal Statistical Society: Series B (Methodological)*, 20(2):215–232.

Cramer, J. S. (2002). The origins of logistic regression.

Crammer, K. and Singer, Y. (2002). Pranking with ranking. In *Advances in neural information processing systems*, pages 641–647.

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, 2(4):303–314.

Darbellay, G. A. and Wuertz, D. (2000). The entropy as a tool for analysing statistical dependences in financial time series. *Physica A: Statistical Mechanics and its Applications*, 287(3-4):429–439.

Dempster, M. A. and Leemans, V. (2006). An automated fx trading system using adaptive reinforcement learning. *Expert Systems with Applications*, 30(3):543–552.

Doganoglu, T., Hartz, C., and Mittnik, S. (2007). Portfolio optimization when risk factors are conditionally varying and heavy tailed. *Computational Economics*, 29(3-4):333–354.

Feder, J. (2013). *Fractals*. Springer Science & Business Media.

Fischer, T. and Krauss, C. (2018). Deep learning with long short-term memory networks for financial market predictions. *European Journal of Operational Research*, 270(2):654–669.

Garey, M. R. and Johnson, D. S. (1979). *Computers and intractability*, volume 174. freeman San Francisco.

Goh, K.-I., Salvi, G., Kahng, B., and Kim, D. (2006). Skeleton and fractal scaling in complex networks. *Physical review letters*, 96(1):018701.

Guida, M. and Maria, F. (2007). Topology of the italian airport network: A scale-free small-world network with a fractal structure? *Chaos, Solitons & Fractals*, 31(3):527–536.

Harris, R. D., Stoja, E., and Tan, L. (2017). The dynamic black–litterman approach to asset allocation. *European Journal of Operational Research*, 259(3):1085–1096.

Ho, T. K. (1995). Random decision forests. In *Proceedings of 3rd international conference on document analysis and recognition*, volume 1, pages 278–282. IEEE.

Huang, J. and Ling, C. X. (2005). Using auc and accuracy in evaluating learning algorithms. *IEEE Transactions on knowledge and Data Engineering*, 17(3):299–310.

Ivanovici, M. and Richard, N. (2010). Fractal dimension of color fractal images. *IEEE Transactions on Image Processing*, 20(1):227–235.

Järvelin, K. and Kekäläinen, J. (2002). Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems (TOIS)*, 20(4):422–446.

Kaastra, I. and Boyd, M. (1996). Designing a neural network for forecasting financial. *Neurocomputing*, 10:215–236.

Kim, J., Goh, K.-I., Salvi, G., Oh, E., Kahng, B., and Kim, D. (2007). Fractality in complex networks: critical and supercritical skeletons. *Physical Review E*, 75(1):016110.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kruskal, J. B. (1956). On the shortest spanning subtree of a graph and the traveling salesman problem. *Proceedings of the American Mathematical society*, 7(1):48–50.

Ku, S., Lee, C., Chang, W., and Song, J. W. (2020). Fractal structure in the s&p500: A correlation-based threshold network approach. *Chaos, Solitons & Fractals*, 137:109848.

Kwapień, J. and Drożdż, S. (2012). Physical approach to complex systems. *Physics Reports*, 515(3-4):115–226.

Lahmiri, S. and Bekiros, S. (2019). Cryptocurrency forecasting with deep learning chaotic neural networks. *Chaos, Solitons & Fractals*, 118:35–40.

Lee, T. K., Cho, J. H., Kwon, D. S., and Sohn, S. Y. (2019). Global stock market investment strategies based on financial network indicators using machine learning techniques. *Expert Systems with Applications*, 117:228–242.

Li, D., Wang, X., and Huang, P. (2017). A fractal growth model: Exploring the connection pattern of hubs in complex networks. *Physica A: Statistical Mechanics and its Applications*, 471:200–211.

Longin, F. and Solnik, B. (1995). Is the correlation in international equity returns constant: 1960–1990? *Journal of international money and finance*, 14(1):3–26.

Lu, C.-J., Lee, T.-S., and Chiu, C.-C. (2009). Financial time series forecasting using independent component analysis and support vector regression. *Decision support systems*, 47(2):115–125.

Maasoumi, E. and Racine, J. (2002). Entropy and predictability of stock market returns. *Journal of Econometrics*, 107(1-2):291–312.

Mao, S. and Xiao, F. (2019). Time series forecasting based on complex network analysis. *IEEE Access*, 7:40220–40229.

Markowitz, H. (1952). Portfolio selection*. *The Journal of Finance*, 7(1):77–91.

Marti, G., Nielsen, F., Bińkowski, M., and Donnat, P. (2017). A review of two decades of correlations, hierarchies, networks and clustering in financial markets. *arXiv preprint arXiv:1703.00485*.

McCulloch, W. S. and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133.

Nie, C.-X. (2019). Applying correlation dimension to the analysis of the evolution of network structure. *Chaos, Solitons & Fractals*, 123:294–303.

Nie, C.-X. and Song, F.-T. (2018). Constructing financial network based on pmfg and threshold method. *Physica A: Statistical Mechanics and its Applications*, 495:104–113.

Patro, D. K., Qi, M., and Sun, X. (2013). A simple indicator of systemic risk. *Journal of Financial Stability*, 9(1):105–116.

Peitgen, H.-O., Jürgens, H., and Saupe, D. (2006). *Chaos and fractals: new frontiers of science*. Springer Science & Business Media.

Perra, N. and Fortunato, S. (2008). Spectral centrality measures in complex networks. *Physical Review E*, 78(3):036107.

Qiu, M., Song, Y., and Akagi, F. (2016). Application of artificial neural network for the prediction of stock market returns: The case of the japanese stock market. *Chaos, Solitons & Fractals*, 85:1–7.

Santos, N. and Santos, D. (2018). A fractal dimension minimum in electrodeposited copper dendritic patterns. *Chaos, Solitons & Fractals*, 116:381–385.

Saramäki, J., Kivelä, M., Onnela, J.-P., Kaski, K., and Kertesz, J. (2007). Generalizations of the clustering coefficient to weighted complex networks. *Physical Review E*, 75(2):027105.

Sharpe, W. F. (1966). Mutual fund performance. *The Journal of business*, 39(1):119–138.

Shashua, A. and Levin, A. (2003). Ranking with large margin principle: Two approaches. In *Advances in neural information processing systems*, pages 961–968.

Song, C., Gallos, L. K., Havlin, S., and Makse, H. A. (2007). How to calculate the fractal dimension of a complex network: the box covering algorithm. *Journal of Statistical Mechanics: Theory and Experiment*, 2007(03):P03006.

Song, C., Havlin, S., and Makse, H. A. (2005). Self-similarity of complex networks. *Nature*, 433(7024):392.

Song, C., Havlin, S., and Makse, H. A. (2006). Origins of fractality in the growth of complex networks. *Nature Physics*, 2(4):275.

Song, Q., Liu, A., and Yang, S. Y. (2017). Stock portfolio selection using learning-to-rank algorithms with news sentiment. *Neurocomputing*, 264:20–28.

Strub, O. and Baumann, P. (2018). Optimal construction and rebalancing of index-tracking portfolios. *European journal of operational research*, 264(1):370–387.

Tumminello, M., Aste, T., Di Matteo, T., and Mantegna, R. N. (2005). A tool for filtering information in complex systems. *Proceedings of the National Academy of Sciences*, 102(30):10421–10426.

Tumminello, M., Lillo, F., and Mantegna, R. N. (2010). Correlation, hierarchies, and networks in financial markets. *Journal of Economic Behavior & Organization*, 75(1):40–58.

Watts, D. J. and Strogatz, S. H. (1998). Collective dynamics of 'small-world'networks. *nature*, 393(6684):440.

Wei, D.-J., Liu, Q., Zhang, H.-X., Hu, Y., Deng, Y., and Mahadevan, S. (2013). Box-covering algorithm for fractal dimension of weighted networks. *Scientific reports*, 3:3049.

Xu, Q., Bo, Z., Jiang, C., and Liu, Y. (2019). Does google search index really help predicting stock market volatility? evidence from a modified mixed data sampling model on volatility. *Knowledge-Based Systems*, 166:170–185.

Yook, S.-H., Radicchi, F., and Meyer-Ortmanns, H. (2005). Self-similar scale-free networks and disassortativity. *Physical Review E*, 72(4):045105.

Yu, J.-R., Chiou, W.-J. P., Lee, W.-Y., and Lin, S.-J. (2020). Portfolio models with return forecasting and transaction costs. *International Review of Economics & Finance*, 66:118–130.

Yu, J.-R., Lee, W.-Y., and Chiou, W.-J. P. (2014). Diversified portfolios with different entropy measures. *Applied Mathematics and Computation*, 241:47–63.

Zhang, X., Qu, S., Huang, J., Fang, B., and Yu, P. (2018). Stock market prediction via multi-source multiple instance learning. *IEEE Access*, 6:50720–50728.

# 국문초록

금융시장에 대한 연구는 전반적인 경제 활동과 밀접한 연관성이 있기 때문에, 다양한 학계의 지식들과 연계되어 광범위하게 연구되고 있다. 전통적인 경제학 이론을 바탕으로 여러가지 경제 지표들이 개발되었고, 계량경제학의 발전으로 이를 정량적으로 분석하는 연구가 진행되었다. 하지만, 서로 다른 특징을 갖는 시장참여자들의 행위로 이루어진 금융시장의 복잡한 특성 때문에, 기존의 경제학 기반의 방법론들만으로 금융시장의 변화를 정밀하게 예측하기에는 한계가 있었다. 만약 금융시장의 변화를 효율적으로 예측 할 수 있다면, 국가 정책이나 기업들 및 시장 참여자들은 합리적인 의사결정을 통해서 건전한 금융 활동을 할 수 있을 것이다. 반면에 이러한 금융시장의 변화를 효율적으로 예측하지 못해 비이상적인 금융 활동이 지속된다면, 최악의 경우에는 글로벌 금융 위기와 같은 대규모 시장 붕괴 현상이 발생할 수 있을 것이아. 따라서 본 학위논문에서는 경제물리학과 머신러닝을 융합하여 체계적으로 금융시장 분석을 진행하고자 한다.

본 학위논문에서는 금융시장의 다양한 섹터 중에서 주식시장 네트워크 구조를 분석하는데 초점을 맞춘다. 미래의 주식시장의 변화를 올바르게 예측하기 위해서는 주식시장 구성원들간의 관계 파악이 선행되어야 하는데, 이에 대표적인 분석 방법이 복잡계 네트워크 분석(Complex network analysis)이기 때문이다. 본 연구에서는 주식시장 네트워크 구조를 분석하는 여러 방법론들 중 프랙탈 이론(Fractal theory)의 도입을 제안한다. 실험 결과 주식시장 네트워크의 구조는 프랙탈 특성을 가짐을 밝혀냈다. 또한, 측정된 프랙탈 차원(Fractal dimension)과 네트워크의 토폴로지(Topology)와의 관계를 살펴본 결과 두 가지 주요한 주식시장 네트워크의 구조적인 특징을 발견할

수 있었다. 첫번째는, 프랙탈 차원과 소위 허브(Hub)라고 불리우는 네트워크 상에서 연결이 많이된 노드들간의 강한 반발(Strong effective repulsion) 현상과 연관성이 있다는 점이다. 두번째는, 프랙탈 차원으로 네트워크의 지름길(Shortcut) 구조를 관찰할 수 있었다. 또한 이 두 가지 네트워크의 구조적인 특성은 주식시장의 위험 관리(Risk management) 관점에서 유용하게 쓰일 수 있음을 분석했다. 그리고, 위 특성들을 다른 연구들에 쉽게 적용 가능하도록 네트워크 구조를 표현하는 3가지 프랙탈 지표(Fractal measures)들을 제안한다.

다음 단계로 주식시장에서 측정한 프랙탈 지표가 미래에 주가 지수의 예측력 향상에 도움이 되는지를 검증했다. 최근 다양한 분야에서 새롭게 발견한 지표들이 금융 시계열 데이터에 대하여 예측력 향상에 도움이 되는지를 검증하는 연구들이 진행되고 있다. 이러한 연구들은 발견한 지표들 만을 사용하여 정밀한 예측을 하는 목적이 아닌, 발견한 지표들이 예측력 향상에 도움이 된다는 점을 밝혀내는데 주 목적이 있다. 이렇게 예측력 향상이 있는것이 밝혀진 지표들은 다른 연구나 산업에 쉽게 적용할 수 있는 장점이 있다. 본 학위논문에서는 몇 가지 머신러닝 알고리즘을 활용하여 측정한 프랙탈 지표가 미래의 주가 지수의 예측력 향상에 도움이 되는지를 검증했다. 검증 실험은 가장 단순한 미래 주가 지수의 방향 분류(Classification) 부터, 주가 지수 수익률의 예측(Prediction) 까지 이루어진다. 그 결과 제안한 프랙탈 지표들은 약 3개월 이후의 장기 미래의 주가 지수에 대해 일관성있는 예측력 향상 효과가 있음을 밝혀냈다.

마지막으로 제안한 프랙탈 지표들과 Learning-to-rank 알고리즘을 활용하여 기존의 주식시장에 연구되었던 거래 전략(Trading strategy)의 성능을 개선할 수 있는 모델을 제안한다. 기존의 주식시장에서 연구된 거래 전략들 중 큰 비율을 차지하는 연구들은 현대 포트폴리오 이론(Modern portfolio theory)에 기반한 포트폴리오 구성 방법에 대한 연구들이다. 하지만, 실제 투자에 적용하기 위해서는 포트폴리오를 구성하는 방법론

뿐 만 아니라, 언제 포트폴리오를 재구성해야 하는지를 판단하는것 또한 중요한 의사결정 요소이다. 본 학위논문에서 제안하는 모델은 기존에 연구된 방법론들에 유연하게 접목하여 활용할 수 있는 최적 리밸런싱 시점 판단 모델(Optimal rebalancing model)이다. 실험은 두단계로 진행된다. 먼저, 제안한 모델로 학습 데이터 내의 서로 다른 두 시점 중 미래의 더 나은 성능을 보이는 리밸런싱 지점을 예측할수 있는지를 학습한다. 그후에, 학습된 모델들 중 좋은 성능을 갖는 파라미터를 선택하고, 시뮬레이션 분석을 통해서 실제 거래전략에 적용 가능성을 평가한다. 실험에서 사용된 기존의 포트폴리오 구성 방법론은, 관련 연구들에서 가장 대표적인 벤치마크로 활용되는 자산 균등 분배 포트폴리오 방식과 샤프 비율(Sharpe ratio) 최대화 포트폴리오 방식이다. 실험결과 두 방식 모두에서 본 연구에서 제안한 최적 리밸런싱 시점 판단 모델이 더 나은 포트폴리오 구성 시점을 구해냈다. 또한 시뮬레이션 결과 일정주기로 리밸런싱하는 동일한 포트폴리오 구성방식보다 더 나은 성능을 보였다. 특히 입력 변수로 프랙탈 지표들을 추가했을 때 가장 좋은 성능을 보임을 관찰했다. 본 모델은 연구된 기존의 모든 포트폴리오 구성 방법론들에 적용할 수 있는 확장성의 관점에서 중요한 기여가 있다. 그리고 프랙탈 지표를 통해서 관찰되는 네트워크의 구조적 특징들이 미래 시장을 판단하는데 도움이 됨을 보임으로써, 제안한 프랙탈 지표들이 실제 주식시장에 적용 가능한 실용적인 특성을 나타냄도 검증했다.

**주요어**: 경제물리학, 복잡계 네트워크 분석, 프랙탈 차원, 네트워크 토폴로지, 위험 관리, 시장 지수 예측, 최적의 리밸런싱 시점 판단 모델, 거래 전략

**학번**: 2015-21126