Monitoring Evolving Stock Networks

Rui Fernando Lima Dias

2015

Master Thesis in Data Analytics

Supervised by

Professor João Manuel Portela da Gama
Professor Carlos Gomes Ferreira

Dedicated to my wife Patrícia.

Biography

Rui Dias was born and raised in the town of Rebordosa, in Porto surroundings, in the North of Portugal. In 2001 he graduated in Computer Science at the Faculty of Sciences of the University of Porto.

Before graduating, he started to work at Arrábida Informática that later became Finantech. Currently, he leads a software developing team that provides solutions for the investment banking. In his work he has contact with great amounts of historical data, which motivated him to intensify his knowledge in that field, and to initiate his studies in the master in Data Analytics.

Acknowledgements

I would like to thank my supervisors, Professor João Gama and Professor Carlos Ferreira, for their guidance, support, incredible patience and excellent advices.

Thanks to the support of project MAESTRA (Grant number ICT-2013-612944) funded by European Commission.

To my family, thanks for understanding when I was absent and for encouraging me when I was present. A special thanks to my parents Mário and Fátima, my sister Ângela, and my grandmother Madalena. I would also like to thank my wife's parents and grandparents, my sister-in-law, brother-in-law and my niece Núria.

And finally, a very special thanks to my wife Patrícia. For giving me love and support, specially in the moments that writing this master thesis seemed impossible.

To all of you, my sincere gratitude.

Abstract

The existence of interactions between stocks to each other is a well known fact (Lee and Djauhari, 2012). This means that fluctuations of a stock price might be influenced by the behaviour of other stocks in the market. As pairs of stocks are correlated, these correlations can be used to construct a network.

In this work we describe a method (MESN - Monitoring Evolving Stock Networks) to study stock market dynamics. This method is inspired in Social Network Analysis (SNA). SNA provides techniques to quantify the influence and importance of a vertex and to detect communities. The metrics and techniques provided by SNA were used in a stock networks context.

The method MESN purposes a temporal analysis by constructing several networks that correspond to different periods of time. It provides a technique to study the evolution of the influence and importance of a stock through the networks, and a technique to study the evolution of communities.

We applied MESN to data collected for U.S. stocks for the year 2014. The results were analysed, and a discussion of how they can help an investor in his decision making process is presented.

Keywords: Social Network Analysis, Stock Markets, Network Dynamics

Resumo

A existência de interações entre ações é um fato bem conhecido (Lee and Djauhari, 2012). Isto significa que as flutuações de preço de uma ação podem ser influenciadas pelo comportamento de outras ações do mercado. Como há pares de ações correlacionados, essas correlações podem ser usadas para construir uma rede.

Neste trabalho descrevemos um método (MESN - Monitorização da Evolução de Redes de Ações) para estudar as dinâmicas de mercados de ações. O método é inspirado na Análise de Redes Sociais (SNA). SNA disponibiliza técnicas para quantificar a influência e a importância de um vértice na rede, assim como para detetar comunidades. As técnicas e métricas disponibilizadas pelo SNA foram usadas no contexto de redes de ações.

O método propõe uma análise temporal através da construção de várias redes que correspondem a períodos diferentes no tempo. Disponibiliza uma técnica para estudar a evolução da influência e importância de uma ação entre as redes, e uma técnica para estudar a evolução das comunidades.

Aplicámos o método MESN a dados recolhidos de ações dos E.U.A. para o ano 2014. Os resultados foram analisados, e discutimos como podem ajudar um investidor no seu processo de tomada de decisão.

Palavras-Chave: Análise de Redes Sociais, Mercados Financeiros, Redes Dinâmicas

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

In this chapter we describe in Section 1.1 the problem that is the subject of this work: a study of the interactions of a set of stocks, and its evolution over time. In Section 1.2, we present the motivation and the challenges inherent to the problem in study, as well as how this work can be useful and used by the community. The software tools used are presented in Section 1.3, and Section 1.4 describes the structure of the thesis.

1.1 Problem Definition

A stock is an ownership share in a corporation. Each of these shares denote a part ownership for a shareholder of that company. Stocks are traded in stock markets all over the world (Scott, 2003).

There are some events directly associated to a company that affect its stock price (e.g. annual revenues, a new CEO, personnel strikes, launch of new products). If the company is influential, its stock price fluctuation can affect the price of other companies stocks, with which it is somehow related. However, identifying the important and influential stocks in a stock market can be a very difficult task, that is, it can be very hard to identify which stocks have great capability to influence other

stock's prices.

Moreover, stock prices are also driven by economical and political conditions. If groups of stocks respond the same way to an event, their prices will change together and their stock prices time series should be highly correlated. In this work we studied the problem of detecting groups of stocks and how they evolve over time.

1.2 Motivation

The study of correlations among stocks is the subject of research of many economists, mathematicians and physicists. A recent approach to this matter is based on network theory. The analysis of a network is a good solution for problems that explore the pair-wise relationship between a large number of variables. Typically, in financial markets, the variables represent a stock attribute (e.g. price, returns, volume) collected for a period of time, and relationships are determined based on the correlation between them.

The studies found in the literature that apply network theory to the study of stock markets are focused on the identification of the hierarchical organization of the stocks. The contribution of this work is the development of a method (Monitoring Evolving Stock Networks - MESN) that studies how the relations between stocks evolve over time, and how it affects the importance and influence of a stock and groups of stocks. This method was inspired in Social Network Analysis (SNA). SNA provides techniques to identify the most influential or important vertices, and to detect groups of vertices in a network (in the SNA taxonomy such groups are named communities). To study the evolution of the relations between stocks for a period of time, a set of stock networks corresponding to different time windows were constructed, and the differences between them analysed.

The goal of this work was to apply MESN to real data, and make the results

available to investors, to support them in their decision making process. This method can help them decide when to sell, when to buy, or do nothing. For that purpose, data from 975 stocks of the U.S. markets for the year 2014 was collected, with the objective of studying the evolution of the important and influential stocks, and the evolution of groups of stocks for that period of time.

For example, if a stock is defined at some point as important, and that importance was high in the past, it is expectable that such stock remains important in the future. Therefore, some research on that stock is worth making.

The same thing with communities. If a community returns are higher than the market, or any particular index, it can be a smart move to invest in it. Moreover, the identification of communities helps the investor to diversify his investments.

Those are some examples of how the knowledge retrieved by the use of this method can help an investor. If his attention is focused on the important and influential stocks, and if he perceives how groups of stocks evolve, he will improve his returns and minimize his losses.

1.3 Software Tools

In the implementation of the experiments of Chapter 4 three software tools were used: Microsoft Excel, Gephi and R.

Excel was used to save the data collected, to perform some computations and to construct the charts. Gephi computed the centralities of every vertex of the networks and detected communities. R was used to construct the networks and to analyse the output of Gephi.

1.4 Structure of the Thesis

The overall thesis is structured as follows:

In Chapter 2 we present background information as well as state-of-art techniques of Social Network Analysis. We also present some of the assumptions and approaches applied in this work.

Chapter 3 describes a method (MESN) to study the dynamics of a stock market. The method uses a criteria to identify important stocks and to detect communities in a stock network. Moreover, it explains a way to study their dynamics, that is, how they evolve over time.

Chapter 4 presents the application of the method described in Chapter 3 to data collected from 975 U.S. stocks. The experimental setup and results are discussed through this chapter. Finally, a discussion of how the results can be used by third parties is presented on the end of this chapter.

Chapter 5 presents the main conclusion and also some limitations of the method. It also discusses possible future paths of this work.

Chapter 2

Social Network Analysis Overview

The analysis of a network is a good solution for problems that explore the pair-wise relationship between a large number of variables. In this chapter we present the most important concepts of Social Network Analysis, and its utility in the context of a study on stock networks. We begin by introducing the concept of network, and explaining how networks are used in the real world. Further on, different centralities measures are presented, that is, measures that indicate how important and influential a vertex is. We also describe the problem of detecting the organization of the vertices in communities and we present some solutions for the problem. Finally, the last sections are dedicated to the study of the evolution of networks, centralities and communities over time.

2.1 Networks

The first reference to a network is credited to Euler (1741), in his approach to the to the Konigsberg bridge's problem. The city was set on both sides of a river, and included two large islands which were connected to each other and the mainland by seven bridges. The problem was to devise a walk through the city that would cross each bridge once and only once. Leonard Euler solved the problem representing the different spots of Konisberg with dots and the bridges with lines as shown in Figure

2.1.

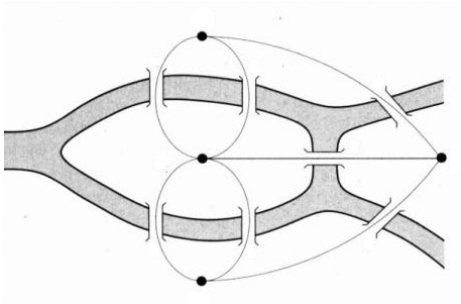


Figure 2.1: Euler's network of Konisberg.

The dots and lines structure introduced by Euler is called a graph. The term graph is credited to Sylvester (1878), although the first publication on graph theory is due to Denes Konig (Tutte, 2001).

Mathematically, networks may be represented as graphs. The Figure 2.2 is a network represented as a graph $G = (V, E)$, where $V = \{1, 2, 3, 4, 5, 6\}$ is the set of vertices, and $E = \{(1, 2), (1, 5), (2, 3), (2, 4), (3, 4), (3, 5), (3, 6)\}$ the set of edges. An edge is placed when any kind of relation between a pair of vertices exists.

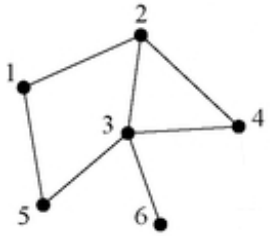


Figure 2.2: A network represented by a graph (Newman, 2010).

Formally, a graph $G = (V, E)$ consists of a non-empty set V of vertices or nodes

and a set E of edges.

The edges may either be directed or undirected. Associated to an edge may be a weight that represents the strength of the connection between two vertices. In this case the graphs are named weighted.

Given a graph $G = (V, E)$ the order of G is the total number of vertices n in V and is denoted mathematically as $|V(G)|= n$. The size of G is the total number of edges m in E and is denoted as $|E(G)|= m$. For undirected graphs the maximum number of edges is $m_{max} = \frac{n(n-1)}{2}$ and for directed graphs is $m_{max} = n(n-1)$ (Oliveira and Gama, 2012; Diestel, 2005).

Another representation of a network is the adjacency matrix. The adjacency matrix A of a graph is the matrix with elements A_{ij} such that

$$A_{ij} = \begin{cases} 1 & \text{if } (i,j) \in E \\ 0 & \text{otherwise} \end{cases} \tag{2.1}$$

For the graph of Figure 2.2 the adjacency matrix is

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

To undirected graphs the matrix is symmetric. For weighted graphs $A_{ij} = w$ where i and j are vertices and w represents the strength of their connection.

These structures are the underlying subject of study of Social Network Analysis. SNA is used widely in the social and behavioural sciences, as well as in economics,

marketing, and industrial engineering. The social network perspective focuses on relationships among social entities (Wasserman and Faust, 1994). Table 2.1 lists some of the many problems that can be represented by a network.

Network	Vertex	Edge
Internet	Computer or router	Cable or wireless data connection
World Wide Web	Web page	Hyperlink
Citation network	Article, patent, or legal case	Citation
Power grid	Generating station or substation	Transmission line
Friendship network	Person	Friendship
Metabolic network	Metabolite	Metabolic reaction
Neural network	Neuron	Synapse
Food Web	Species	Predation
Stock market	Stock	High correlation

Table 2.1: Networks of the real world (Newman, 2010).

The focus of this work is on the last entry of the Table 2.1, that is, the problem of studying the relations of stocks in a stock market. There are several works published that study the stock networks.

The first study of financial markets in the context of networks is credited to Mantegna (1999). The method started to compute the correlation coefficient between all the possible pairs of stocks present in a portfolio in a given time period. The second step of the method is to compute a distance d based in the correlations ($d(i,j) = \sqrt{2(1-\rho_{ij})}$). The distance between stocks highly correlated is lower than the distance of uncorrelated stocks. Then, a network is created and every edge is weighted with that distance. Finally, the method extracts the minimum spanning tree (MST) from the complete network. Figure 2.3 presents a minimum spanning tree, which is a connected sub-network with no cycles, that includes every vertex of the original network with the lowest cost (minimum sum of the edge weights) (Costa et al., 2011).

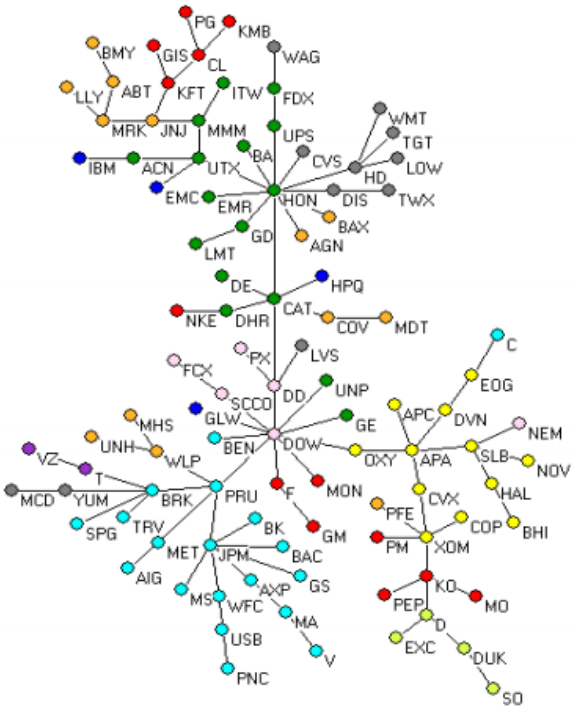


Figure 2.3: MST of the NYSE 100 (Lee and Djauhari, 2012).

The analysis of the MST showed that U.S. stocks were grouped accordingly with their industry sector (Mantegna, 1999).

Bonanno et al. (2003) applied the method introduced by Mantegna (1999) to compare the topological properties of the MST obtained from a portfolio of stocks traded at the New York Stock Exchange during a 12-year period with the one obtained by using simple market models. The author concluded that the topology of MST obtained from real financial markets showed large scale correlation properties characteristic of complex networks. The market models studied failed to reproduce such properties (Bonanno et al., 2003).

Eom et al. (2007) collected the daily returns for 400 stocks listed under the S&P 500 for a time period of 12 years and investigated the role of economic factors in the formation of stock networks. The author concluded that individual stocks with

a large number of links to other stocks in a network are more highly correlated with common economic factors than those with a small number of links.

The works presented above show that a structure of a set of stocks of the financial markets and their relations can be represented as a network. The stocks are the vertices and a function, that is typically correlation based, is used to weigh the edge that connects every pair of stocks.

An example of a function that measures the relation of two stocks was presented by Mantegna (1999) in his work to find the hierarchical structure of stock markets. The author defined a distance to relate a pair of stocks. This distance fulfils the three axioms of a Euclidean distance enumerated below:

1. $d(i,j) = 0 \Leftrightarrow i = j$
2. $d(i,j) = d(j,i)$
3. $d(i,j) \leq d(i,k) + d(k,j)$, for all practical purposes.

The first step was to quantify the degree of similarity between the synchronous time evolution of a pair of stock price by the correlation coefficient:

$$\rho_{ij} = \frac{\langle Y_i Y_j \rangle - \langle Y_i \rangle \langle Y_j \rangle}{\sqrt{(\langle Y_i^2 \rangle - \langle Y_i \rangle^2)(\langle Y_j^2 \rangle - \langle Y_j \rangle^2)}} \tag{2.2}$$

where i and j are the numerical labels of stocks, $Y_i = ln\ P_i(t) - ln\ P_i(t-1)$ and $P_i(t)$ is the close price of the stock i at the day t.

The function purposed is:

$$d(i,j) = \sqrt{2(1-\rho_{ij})} \tag{2.3}$$

The function $d(i,j)$ fulfils the three axioms of a Euclidean metric:

- The first axiom is valid because $d(i, j) = 0$ if and only if the correlation is total ($\rho = 1$, namely only if the two stocks perform the same stochastic process) (Mantegna, 1999).

- The second axiom is valid because the correlation coefficient matrix and hence the distance matrix D is symmetric by definition (Mantegna, 1999).

- The third axiom is valid because Equation 2.3 is equivalent to the Euclidean distance between two vectors $\tilde{Y}_i$ and $\tilde{Y}_j$ which are obtained from the time series Y_i and Y_j by considering each record of the time series a component of the vector (Mantegna, 1999).

2.2 Centralities

Given a network structure it is possible to calculate some measures that capture particular features of the network topology. These measures, known as centralities in SNA, quantify the importance and influence of a vertex in the network. This section presents four centrality measures: degree, betweenness, closeness and eigenvector.

2.2.1 Degree Centrality

Degree centrality is perhaps the simplest centrality measure in a network. It is calculated counting the number of edges that are connected to a vertex. Although degree centrality is a simple centrality measure, it seems reasonable to suppose that individuals that have connections to many others might have more influence than those who have fewer connections (Newman, 2010).

Formally, for a vertex i, the degree is denoted as k_i and is calculated as

$$k_i = \sum_{j=1}^{n} a_{ij} \tag{2.4}$$

where a_{ij} is the entry of the i-th row and j-th column of the adjacency matrix A.

The average degree is the mean of the degrees of all vertices in a network. This measure can be used to measure the global connectivity of a network (Costa et al., 2011).

$$\bar{k} = \frac{1}{n} \sum_{i=1}^{n} k_{ij} \tag{2.5}$$

For directed networks the degree centrality is divided in two measures: in-degree k_v^+ and out-degree k_v^-. In-degree counts the number of edges that begin on v (Equation 2.6), and the out-degree measures the number of edges that end on v (Equation 2.7).

$$k_i^+ = \sum_{j=1}^{n} a_{ji} \tag{2.6}$$

$$k_i^- = \sum_{j=1}^{n} a_{ij} \tag{2.7}$$

On weighted networks, strength k_v^w is the equivalent of degree, being computed as the sum of the weights of the edges adjacent to the vertex v (Oliveira and Gama, 2012).

$$k_i^w = \sum_{j=1}^{n} a_{ij}^w \tag{2.8}$$

The stocks with the higher scores directly influence the behaviour of more other stocks which are directly connected to it (Lee and Djauhari, 2012).

2.2.2 Betweenness Centrality

Betweenness centrality measures the extent to which a vertex lies on paths between other vertices (Gama et al., 2012). Vertices with high betweenness centrality may have considerable influence within a network by virtue of their control over information passing between others (Newman, 2010). These vertices are called gatekeepers

since they tend to control the flow of information between communities (Oliveira and Gama, 2012). Formally, the betweenness centrality of a vertex v is calculated as

$$b_v = \sum_{s,t \in V(G) \setminus v} \frac{\alpha_{st}(v)}{\alpha_{st}} \tag{2.9}$$

where α_{st} denotes the number of shortest paths between vertices s and t (usually $\alpha_{st} = 1$) and $\alpha_{st}(v)$ expresses the number of shortest paths passing through node v.

The betweenness can also be computed for an edge. The betweenness b_e denotes the number of shortest paths that run through a given edge e.

$$b_e = \sum_{s,t \in V(G)} \frac{\alpha_{uv}(e)}{\alpha_{uv}} \tag{2.10}$$

where $\alpha_{uv}(e)$ is the number of shortest paths passing trough edge e. This measure is very useful in SNA since it helps to discover bridges inside the network. Bridges are defined as edges that connect different communities of the network. This matter is presented in the Section 2.3.

The stocks with higher score are considered significant in terms of their role in coordinating the information among stocks (Lee and Djauhari, 2012).

2.2.3 Closeness Centrality

Closeness centrality measures the mean distance of a vertex to other vertices, giving an idea about how long it will take to reach other vertices from a given starting vertex (Oliveira and Gama, 2012).

Suppose d_{ij} is the length of the shortest path (also known as geodesic path) between vertex i and vertex j. Then, the mean geodesic distance from i to j,

averaged over all vertices j in the network is

$$l_i = \frac{1}{n-1} \sum_{j(\neq i)} d_{ij} \tag{2.11}$$

The mean distance l_i is not a centrality measure in the same sense as the others described in this work, since it gives low values for more central vertices and high values for less central ones (Newman, 2010). To avoid this issue, it should be considered the inverse of l_i. Therefore, closeness centrality of a vertex i shall be calculated as

$$C_i = \frac{1}{l_i} \tag{2.12}$$

In a financial market network, closeness centrality is a measure of how close a stock is to all other stocks. The higher the score of a particular stock the faster the stock spread the information from it to all others (Lee and Djauhari, 2012).

2.2.4 Eigenvector Centrality

Eigenvector centrality is a more elaborated version of the degree centrality. It assumes that not all neighbours of a vertex have the same importance. Therefore, what is taken into account is not the quantity of neighbours, but the quality of those neighbours.

This metric is based on the assignment of a relative score to each vertex and measures how well a given actor is connected to other well-connected actors. This score is given by the first eigenvector of the adjacency matrix. The basic idea behind eigenvector centrality is that the power and status of an actor is recursively defined by the power and status of his/her alters. Alter is a term frequently used in the egocentric approach of social networks analysis, and it refers to the actors that are directly connected to a specific actor, called ego. In other words, we can say that the centrality of a given vertex is proportional to the sum of the centralities of its

neighbours (Oliveira and Gama, 2012). Eigenvector centrality is calculated as

$$x_i = \frac{1}{\lambda} \sum_{j=1}^{n} a_{ij} x_j \tag{2.13}$$

where $x_i \backslash x_j$ denotes the centrality of vertex $i \backslash j$, a_{ij} represents an entry of the adjacency matrix A ($a_{ij} = 1$ if vertices i and j are connected by an edge and $a_{ij} = 0$ otherwise) and λ denotes the largest eigenvalue of A (Oliveira and Gama, 2012).

The stocks with higher scoring are connected to the high-scoring stocks and contribute more to the scores of their neighbours (Lee and Djauhari, 2012).

2.3 Communities

In a social network, a community is defined as a group of vertices densely connected to each other but less connected to the vertices outside (De Meo et al., 2013). Such clusters, or communities, can be considered as fairly independent compartments of a graph (Fortunato, 2010). Figure 2.4 is an example of a graph with three communities.

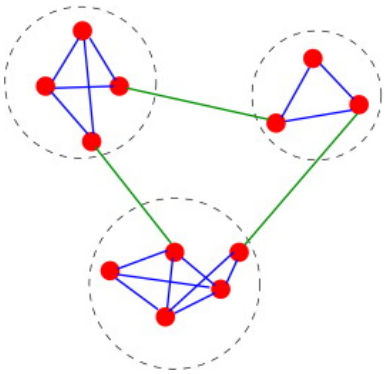


Figure 2.4: A graph with three communities (Fortunato, 2010).

Community detection in graphs is an interdisciplinary subject with a vast spec-

trum of applications. Practical applications can be found in many fields such as biology (metabolic networks, gene regulatory networks, other forms of interactions among proteins), computer science, sociology (families, working and friendship circles, villages are examples of social networks, and sociology studies how different people interact, how they decide to join or not a community), and marketing (identifying clusters of customers with similar interests enables to set up efficient recommendation systems, e.g., www.amazon.com) (De Meo et al., 2013).

There is not a consensual definition of the problem of community detection. However, the most popular definition may be enunciated as: Given a network represented by a graph $G = (V, E)$, the community structure is a partition $P = \{C_1, C_2, ..., C_r\}$ of the vertices of G such that, for each $C_i \in P$, the number of edges linking vertices in C_i is high in comparison to the number of edges linking vertices on two distinct sets. Each set C_i is a community of G (De Meo et al., 2013).

The aim of community detection is to identify the communities and, possibly, their hierarchical organization, by only using the information encoded in the graph topology (Fortunato, 2010).

Some of the most popular techniques used to detect communities are hierarchical clustering, Girvan-Newman algorithm, and modularity based methods.

2.3.1 Hierarchical Clustering

Hierarchical clustering first calculates a weight W_{ij} for every pair (i, j) of vertices in the network, which represents in some sense how closely connected the vertices are. Then one takes the n vertices in the network, with no edges between them, and adds edges between pairs one by one in order of their weights, starting with the pair with the strongest weight and progressing to the weakest. As edges are added, the resulting graph shows a nested set of increasingly large components (connected

subsets of vertices), which are taken to be the communities.

Because the components are properly nested, they all can be represented by using a tree of the type shown in Figure 2.5, in which the lowest level at which two vertices are connected represents the strength of the edge that resulted in their first becoming members of the same community.

A slice through this tree at any level gives the communities that existed just before an edge of the corresponding weight was added. Trees of this type are sometimes called dendrograms in the sociological literature (Girvan and Newman, 2002).

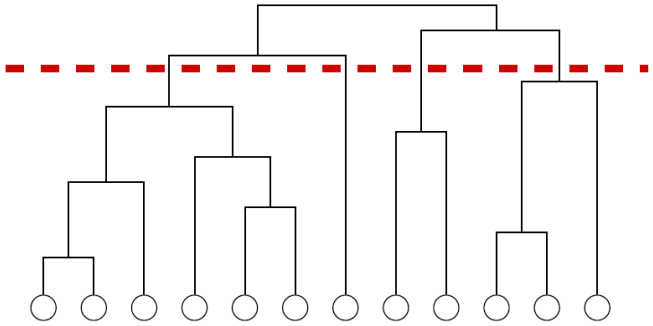


Figure 2.5: Hierarchical tree or dendrogram illustrating the type of output generated by the algorithm. As we move up the tree the vertices join together to form larger and larger communities (Newman and Girvan, 2004).

2.3.2 Girvan-Newman Algorithm

Instead of trying to construct a measure that tells which edges are the most central to communities, Girvan and Newman focused instead on those edges that are least central, the edges that are most between communities. Rather than constructing communities by adding the strongest edges to an initially empty vertex set, they are constructed by progressively removing edges from the original graph.

First proposed by Freeman, the betweenness centrality of a vertex i is defined as the number of shortest paths between pairs of other vertices that run through i. It

is a measure of the influence of a vertex over the flow of information between other vertices, especially in cases where information flow over a network primarily follows the shortest available path.

To find which edges in a network are most between other pairs of vertices, a generalization of Freeman's betweenness centrality to edges is defined. The betweenness of an edge is the number of shortest paths between pairs of vertices that run along it. Thus, the edges connecting communities will have high edge betweenness. By removing these edges, we separate groups from one another and so reveal the underlying community structure of the graph (Girvan and Newman, 2002).

Figure 2.6 presents a schematic representation of a network with community structure to demonstrate how edge betweenness may be used to detect communities.

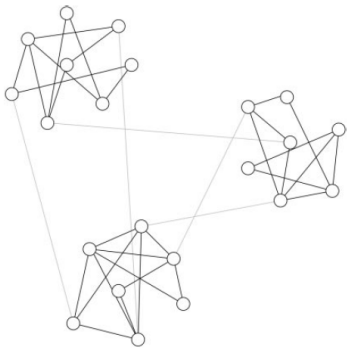


Figure 2.6: A schematic representation of a network with community structure. In this network there are three communities of densely connected vertices (circles with solid lines), with a much lower density of connections (grey lines) between them (Girvan and Newman, 2002).

Algorithm 1 is the one proposed for detecting communities.

Algorithm 1 Girvan-Newman Algorithm for community detection in a graph

Input: Graph $G = (V, E)$

1: **while** $E \neq \emptyset$ **do**
2: Calculate the betweenness for all edges in the E
3: Remove the edge with the highest betweenness from E
4: **end while**

The algorithm output is in the form of a dendrogram which represents an entire nested hierarchy of possible community divisions for the network as can be seen in Figure 2.5. There are a set of possible solutions, and a question remains: *Where should the dendrogram be cut to get a sensible division of the network?*

To answer this question Newman and Girvan (2004) introduced a benefit function called modularity. To explain modularity lets consider a particular division of a network into k communities and a $k \times k$ symmetric matrix E whose element e_{ij} is the fraction of all edges in the network that link vertices in community i to vertices in community j. The row (or column) sums $a_i = \sum_j e_{ij}$ represents the fraction of edges that connect to vertices in community i. Then, modularity Q is written as

$$Q = \sum_i (e_{ii} - a_i^2) \tag{2.14}$$

A cut in the dendrogram must be performed in a way that maximizes Q. A strategy may be to move the cut in the dendrogram, looking for local peaks in its value, which indicate particularly satisfactory splits (Newman and Girvan, 2004).

2.3.3 Modularity-based Methods

The modularity value is high for good community divisions and low for poor ones. Modularity is by far the most used and best known quality function. It quantifies the quality of a given division of the network into communities (Fortunato, 2010). There are three kinds of algorithms inspired on modularity: modularity optimization, greedy techniques and simulated annealing.

- **Modularity optimization**

  A way of finding a good community partition in a network is by looking for the divisions of a network that have positive, and preferably high, values of modularity. Another way is by automatically select the optimal number of communities p, by finding the value of p for which Q is maximized. An exhaustive optimization of Q is impossible, due to the huge number of ways in which it is possible to partition a graph. Besides, the true maximum is out of reach, as it has been recently proved that modularity optimization is an NP-complete problem. However, there are currently several algorithms able to find fairly good approximations of the modularity maximum in a reasonable time (Wang et al., 2008).

  Blondel et al. (2008) presented a simple heuristic method to extract the community structure of large networks that is based on modularity optimization - the Louvain Method for community detection.

  First, each vertex in the network is assigned to its own community. Then, for each vertex i, the change in modularity is calculated for removing i from its own community and moving it into the community of each neighbour j of i. Once this value is calculated for all communities that i is connected to, i is placed into the community that resulted in the greatest modularity increase. If no increase is possible, i remains in its original community. This process is applied repeatedly and sequentially to all vertices until no modularity increase can occur. Once this local maximum of modularity is hit, the first phase has ended.

  The second phase of the algorithm builds a new network where the vertices are the communities detected in the first phase. The edges that connected vertices in the same community are now represented by self loops. Edges that connected multiple vertices from a community to vertices of another community

are now represented by a weighted edge. After the new network is created, the second phase has ended and the first phase can be applied to the new network.

- **Greedy techniques**

  The first algorithm devised to maximize modularity was a greedy method of Newman. It is an agglomerative hierarchical clustering method, where groups of vertices are successively joined to form larger communities such that modularity increases after the merging. The algorithm starts by defining n clusters, each one containing a single vertex. Edges are not initially present, they are added one by one during the procedure. An edge is chosen such that this partition gives the maximum increase (minimum decrease) of modularity with respect to the previous configuration. If the insertion of an edge does not change the partition, that is, the edge is internal to one of the clusters previously formed, modularity stays the same (Fortunato, 2010).

- **Simulated annealing**

  Simulated annealing is a probabilistic procedure for global optimization used in different fields and problems. It consists of performing an exploration of the space of possible states, looking for the global optimum of a function F, say its maximum. Its standard implementation combines two types of moves: local moves, where a single vertex is shifted from one cluster to another, taken at random; global moves, consisting of mergers and splits of communities. Splits can be carried out in several distinct ways (Fortunato, 2010).

2.4 Evolving Networks

The use of a temporal analysis of a network gives an extra knowledge of the behaviour of the vertices. When a network of stocks is being studied, the analysis shows how the position and influence of a particular stock changes over time, and how the

network as a whole adapts to those changes.

Two approaches to collect the snapshots of a network can be sliding windows and cumulative windows. The sliding window approach applies an ageing variable, where old data is forgotten when a snapshot of the network is taken. In that way the most recent data is more valuable than old data. In the other hand, cumulative windows never "forgets" the data. In all snapshots the data collected so far is considered. Figure 2.7 illustrates the two concepts.

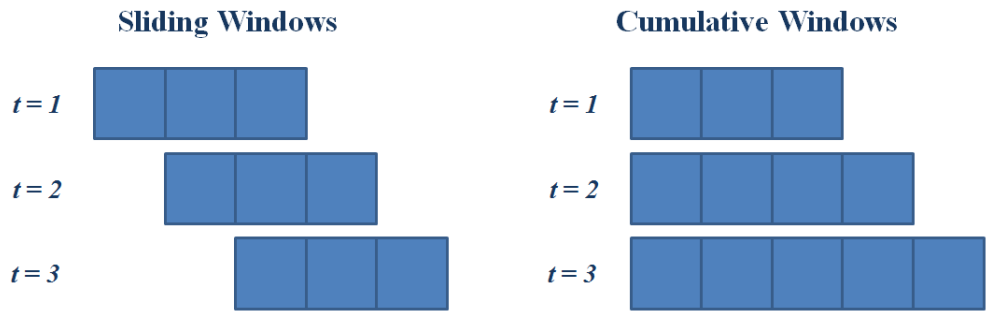


Figure 2.7: Left: Sliding windows. Right: Cumulative windows.

There are two types of sliding windows depending on whether the number of elements in the window is fixed (fixed-size sliding window) or variable (variable-size sliding window) (Arasu and Manku, 2004).

With the creation of snapshots a temporal analysis on the network dynamics is possible. Basically, it is the study of changes on consecutive snapshots of the network. These snapshots are used by the methods that study the evolution of the centralities and communities. These methods are discussed in more detail in Sections 2.5 and 2.6.

2.5 Evolution of the Centralities

Traditional data analysis techniques aim to extract knowledge from 2-dimensional data. Data is represented in a matrix form, with rows corresponding to the objects and columns to the variables. The collected data corresponds to a single snapshot in time. When the intention is to study the evolution of the variables values over time, a temporal dimension must be added. A data representation scheme able to model higher than 2-dimensional data is high-order tensors (also known as hypermatrices, multiway models, multiway arrays or multidimensional arrays) (Oliveira and Gama, 2013).

As Kiers (2000) defines, a tensor is an N-way data array, where N is the order of the tensor. A 3-order tensor ($N = 3$) encapsulates three modes: the row-entities mode (mode A), the column-entities mode (mode B) and the fiber-entities mode (mode C). The term mode is used to refer to a set of entities.

High-order tensors, denoted by calligraphic letter χ, are generalizations of scalars (order 0), vectors (order 1) and matrices (order 2 to 3 or higher orders). The element (i, j, k) of a 3-order tensor χ is denoted by x_{ijk}, where index i, j and k refers to the entities of mode A, B and C respectively.

The proposal of Oliveira and Gama (2013) is to represent dynamic networks as 3-order tensors where mode A gets the vertices, mode B gets SNA metrics (e.g. centralities) and mode C represents the time. Since the values of centralities are embedded in the matrix that represent each snapshot, it is possible to determine the importance and influence of each vertex of the network in a specific point in time.

A trajectory can be defined as a sequence of time-stamped points $Traj = p_0 \rightarrow p_1 \rightarrow ... \rightarrow p_k$, where $p_i = (x_i, y_i, t_i)(i = 0, 1, .., k)$ represents the position of a given object at time point t_i, and (x_i, y_i) are the coordinates of the object, in a 2D space. Typically, these trajectories are defined in low-dimensional representative spaces and

are graphically represented by a line that connects the coordinates of an object for different time points. It is common to resort to 2D, instead of 3D spaces, since they are simpler to analyse and, at the same time, allow an effective data analysis. Thus, the use of 2-dimensional projections that encode the third dimension as a trajectory over the plane enables to map a given actor's trajectory along time, by simply using 2-dimensional projections (Oliveira and Gama, 2013).

With this technique, the follow-up of the most important and most influential stocks trajectories is possible. Even better, is the fact that this trajectories are plotted in 2-dimension charts, which allows an easy and effective analysis of the centralities dynamics by an investor.

2.6 Evolution of the Communities

There are five events that describe how a cluster or community can evolve between two consecutive time intervals or snapshots. In related work it can be found different names to such events. Asur et al. (2009) named the events as: *Continue*, κ-Merge, κ-Split, *Form* and *Dissolve*. In the work Monitor of the Evolution of Clusters (MEC), Oliveira and Gama (2010) named those same events as *Survival*, *Merge*, *Split*, *Birth* and *Death*.

2.6.1 Method presented by Asur et al. (2009)

Let S_i and S_{i+1} be snapshots of the network S at two consecutive time intervals. C_i^k is the k-th cluster at snapshot S_i and V_i^k represents the set of vertices of cluster k in S_i. Figure 2.8 shows examples of those five events.

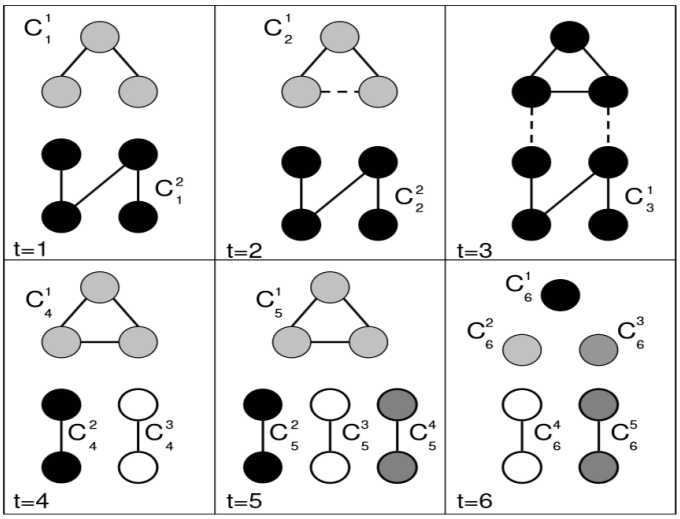


Figure 2.8: Temporal Snapshots at time $t = 1$ to 6 (Asur et al., 2009).

Asur et al. (2009) presented a method to detect the events that affect communities over time. A brief description of what each event is and how it can be detected is presented below.

- **Form**

  A new cluster C_{i+1}^k is said to have been formed if none of the vertices in the cluster were grouped together at the previous time interval, that is, no two vertices in V_{i+1}^k existed in the same cluster at time period i.

$$Form(C_{i+1}^k) = 1 \Leftrightarrow \nexists C_i^j : V_{i+1}^k \cap V_i^j > 1 \tag{2.15}$$

  Intuitively, a form indicates the creation of a new community or new collaboration. Figure 2.8 at time $t = 5$ shows a form event when two new nodes appear and a new cluster is formed.

- **Continue**

  A cluster C_{i+1}^j is marked as continuation of C_i^k if V_{i+1}^j is the same as V_i^k.

$$Continue(C_i^k, C_{i+1}^j) = 1 \Leftrightarrow V_i^k = V_{i+1}^j \tag{2.16}$$

  The main motivation behind this is that if certain vertices are always part of the same cluster, any information supplied to one vertex will eventually reach the others. Therefore, as long as the vertex set remains the same, the information flow is not hindered. The addition and deletion of edges merely indicates the strength between the nodes.

  An example of a continue event is shown at $t = 2$ in Figure 2.8. Note that an extra interaction appears between the nodes in Cluster C_2^1 but the clusters do not change.

- **κ-Merge**

  Two different clusters C_i^k and C_i^l are marked as merged if in the next timestamp exists a cluster that contains at least $\kappa\%$ of the nodes belonging to these two clusters. The essential condition for a merge is:

$$Merge(C_i^k, C_i^l, \kappa) = 1 \Leftrightarrow \tag{2.17}$$

$$\exists C_{i+1}^j : \frac{|(V_i^k \cup V_i^l) \cap V_i^j + 1|}{Max(|V_i^k\% \cup \ V_i^l|, |V_i^j + 1|)} > \kappa\% \ \wedge$$

$$|V_i^k\% \cap V_{i+1}^j| > \frac{|C_i^k|}{2} \ \wedge \ |V_i^l\% \cap V_{i+1}^j| > \frac{|C_i^l|}{2}$$

  This condition will only hold if edges between V_i^k and V_i^l exist in timestamp $i + 1$. Intuitively, it implies that new interactions have been created between nodes which previously were part of different clusters. This caused $\kappa\%$ of

nodes in the two original clusters to join the new cluster. Note that, in an ideal or complete merge, with $\kappa = 100$, all nodes in the two original clusters are found in the same cluster in the next timestamp. The two original clusters are completely lost in this scenario. Figure 2.8 shows an example of a complete merge event at $t = 3$. The dotted lines represent the newly created edges. All the nodes now belong to a single cluster (C_3^1).

- **κ-Split**

  A single cluster C_i^j is marked as split if $\kappa\%$ of nodes from this cluster are present in two different clusters in the next timestamp. The essential condition is that:

$$Split(C_i^j, \kappa) = 1 \Leftrightarrow \tag{2.18}$$

$$\exists C_{i+1}^k, C_{i+1}^l : \frac{|(V_{i+1}^k \cup V_{i+1}^l) \cap V_i^j|}{Max(|V_{i+1}^k \cup V_{i+1}^l|, |V_i^j|)} > \kappa\% \wedge$$

$$|V_{i+1}^k \cap V_i^j| > \frac{|C_{i+1}^k|}{2} \wedge |V_{i+1}^l \cap V_i^j| > \frac{|C_{i+1}^l|}{2}$$

  A split signifies that the interactions between certain nodes are broken and not carried over to the current timestamp, causing the nodes to part ways and join different clusters. Also note that a broken edge, by itself, does not necessarily indicate a split event, as there may be other interactions existing between vertices in the cluster (similar to the notion of k-connectivity). Time t=4 in Figure 2.8 shows a split event when a cluster gets completely split into three smaller clusters.

- **Dissolve**

  A single cluster C_i^k is said to have dissolved if none of the vertices in the cluster are in the same cluster in the next timestamp, that is, no two entities in the original cluster have an interaction between them in the current time interval.

$$Dissolve(C_i^k) = 1 \Leftrightarrow \nexists C_{i+1}^j : V_i^k \cap V_{i+1}^j > 1 \tag{2.19}$$

Intuitively, a dissolve indicates the lack of contact or interactions between a group of nodes in a particular time period. This might signify the breakup of a community or a workgroup. Figure 2.8 at time $t = 6$ shows a dissolve event when there are no longer interactions between the three nodes in Cluster C_5^1 resulting in a breakup of the cluster into 3 clusters - C_6^1, C_6^2 and C_6^3.

2.6.2 Monitor of the Evolution of Clusters

The method Monitor of the Evolution of Clusters (MEC) traces evolution by detecting and categorizing transitions on clusters in different snapshots of the network. In MEC method, the events discovery explores the concept of conditional probability and is restricted by a previously defined threshold - survival threshold τ - which assumes the minimum of $\tau = 0.5$ (intuitively, this means that a match must contain at least half of the objects of the previous cluster) (Oliveira and Gama, 2010).

The conditional probabilities are computed for every pair of possible connections between clusters obtained at different snapshots and they represent the edge's weights in a bipartite graph.

Given the clusterings $\xi_i, \xi_{i+\Delta t}$, obtained at $t_i, t_{i+\Delta t}$, a graph $G = (U, V, E)$ can be constructed, where U represents the first subset of vertices (clusters of t_i), V represents the second subset of vertices (clusters of $t_{i+\Delta t}$), and E denotes a set of weighted edges between any pair of clusters belonging to ξ_i and $\xi_{i+\Delta t}$. Formally, the weight assigned to the edge connecting clusters $C_m(t_i)$ and $C_u(t_{i+\Delta t})(m = (1, ..., k_{ti})$ and $u = (1, ..., k_{ti+\Delta t})$, where k_{ti} and $k_{ti+\Delta t}$ are the number of clusters returned by a given clustering algorithm in time points t_i and $t_{i+\Delta t}$, respectively) are estimated in accordance with the conditional probability:

$$weight(C_m(t_i), C_u(t_{i+\Delta t})) = P(X \in C_u(ti + \Delta t) | X \in C_m(t_i)) = \frac{\sum P(x \in C_m(t_i) \bigcap C_u(t_{i+\Delta t}))}{\sum P(x \in C_m(t_i))} \tag{2.20}$$

To detect changes, MEC defines the transitions that a cluster $C \in \xi_i$ can experience, with respect to $\xi_{i+\Delta t}$. A new threshold was introduced to help the definition of these transitions: the split threshold λ. The transitions are summarized in Table 2.2.

Event	Notation	Definition
Birth	$\phi \rightarrow C_u(t_{i+\Delta t})$	$0 < weight(C_m(t_i), C_u(t_{i+\Delta t})) < \tau \forall_m$
Death	$C_m(t_i) \rightarrow \phi$	$weight(C_m(t_i), C_u(t_{i+\Delta t})) < \lambda \forall_u$
Split	$C_m(t_i) \overset{\subset}{\rightarrow} \{C_1(t_{i+\Delta t}), \ldots, C_r(t_{i+\Delta t})\}$	$(\exists_u \exists_v : weight(C_m(t_i), C_u(t_{i+\Delta t})) \geq \lambda \wedge$ $weight(C_m(t_i), C_v(t_{i+\Delta t})) \geq \lambda) \wedge$ $\sum_{u=1}^{r} weight(C_m(t_i), C_u(t_{i+\Delta t})) \geq \tau$
Merge	$\{C_1(t_i), \ldots, C_p(t_i)\} \overset{\subset}{\rightarrow} C_u(t_{i+\Delta t})$	$weight(C_m(t_i), C_u(t_{i+\Delta t})) \geq \tau) \wedge$ $\exists C_p \in \xi_i \setminus \{C_m\} : weight(C_p(t_i), C_u(t_{i+\Delta t})) \geq \tau$
Survival	$C_m(t_i) \rightarrow C_u(t_{i+\Delta t})$	$weight(C_m(t_i), C_u(t_{i+\Delta t})) \geq \tau) \wedge$ $\nexists C_p \in \xi_i \setminus \{C_m\} : weight(C_p(t_i), C_u(t_{i+\Delta t})) \geq \tau$

Table 2.2: Communities transitions as described in MEC (Oliveira and Gama, 2010).

Chapter 3

Methodology to Study Evolving Stock Networks

In this chapter we present the methodology that we developed to study the dynamics of stocks. An overview of the methodology is presented, and further on, a detailed description of each step.

3.1 Overview of the Methodology

The analysis of a network is a good solution for problems that explore the pair-wise relationship between a large number of variables. In stock markets, such variables represent a stock attribute (e.g. price, returns, volume) collected for a period of time, and relationships are determined based on the correlation between them.

The method Monitoring Evolving Stock Networks (MESN) described in this chapter was designed to be applied on data collected about stocks for a given period of time. The time series collected for each stock is the close price. The data is used to construct several networks with the same stocks that correspond to different temporal windows. The goal of MESN is to provide a way to study the evolution of

the networks over time.

The method MESN has 6 steps. Step 1 and Step 2 describe how a single network is constructed. As previously referred, to study the evolution of the relations between stocks over time, several networks must be constructed. The parameters that need to be set to create the networks are described in Step 3. Step 4 indicates what measures should be computed for every network and how to detect the communities. Step 5 introduces the technique to study the evolution of the important and influential stocks, and finally, Step 6 presents the technique to study the evolution of communities.

Figure 3.1 presents an overall schema of the method MESN.

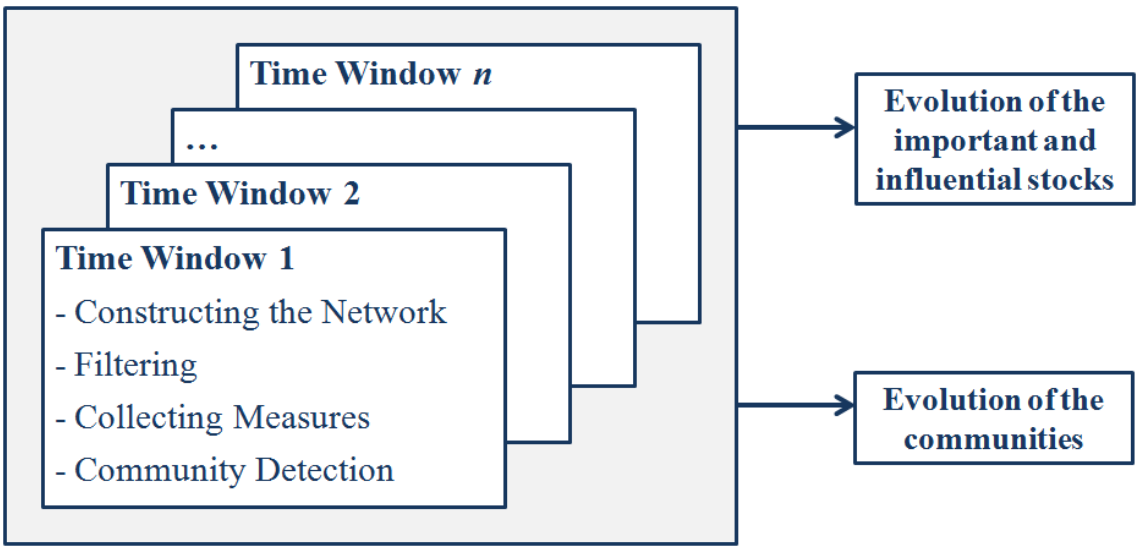


Figure 3.1: Schematic representation of the method MESN.

3.2 Step 1 - Constructing the Network

The first step of MESN is the definition of the conditions for a network construction.

A network in this context is an undirected graph $G = (V, E, f)$ where $V =$

$\{s_1, s_2, \ldots, s_n\}$ is the non-empty set of the n stocks of the data, E is a set of edges weighed upon the output of a function f. Typically, the function f is correlation based, and its output will be associated to the edges that connect every pair of stocks.

There are many options for f. The most basic is correlation on close price time series. An alternative is to correlate stocks based on the daily variations or returns, that can be computed from the close prices. The daily variation $var(i)$ at day i, can be computed as

$$var(i) = \begin{cases} 0 & \text{if } i = 0 \\ \frac{cp(i) - cp(i-1)}{cp(i-1)} & \text{otherwise} \end{cases} \tag{3.1}$$

where $cp(i)$ and $cp(i-1)$ is the close price at day i and day $i-1$ respectively. The variation time series of a set of stocks are at the same scale, and that has impact when computing the correlation.

Another option to f is to consider the function introduced by Mantegna (1999) and described in Section 2.1. It is widely used, mainly in studies where an MST is extracted from the complete network.

3.3 Step 2 - Filtering the Network

The second step of the method extracts from the network previously constructed an unweighted network by filtering some edges. This filter consists of a threshold α that is compared with the output of the function f chosen.

If f is the correlation function then an edge is removed from the network if it is not greater or equal to α. The edges that remain in the network follow the Condition 3.2.

$$(s_i, s_j) \in E \Leftrightarrow \rho_{ij} \geq \alpha \wedge i \neq j \tag{3.2}$$

The inequality $i \neq j$ is set to prevent a vertex from being connected to itself.

Correlation varies from -1 to 1, where 1 means that the series are positively correlated and -1 are negatively correlated. Therefore, if the goal is to study the impact of negatively correlated stocks in network's topology, another filter may be considered. The edges that remain in the network follow the Condition 3.3.

$$(s_i, s_j) \in E \Leftrightarrow |\rho_{ij}| \geq \alpha \wedge i \neq j \tag{3.3}$$

The result is a network where positively correlated stocks and negatively correlated stocks are connected.

The last filter that may be considered is used when the function f that is chosen is the distance of Mantegna. This function is a distance $d(i,j) = \sqrt{2(1-\rho_{ij})}$ (Section 2.1), therefore the stocks that remain connected are the closer ones. Thus, the edges that remain in the network follow the Condition 3.4.

$$(s_i, s_j) \in E \Leftrightarrow d(i,j) \leq \alpha \wedge i \neq j \tag{3.4}$$

3.4 Step 3 - Preparing the Study of the Evolution of Stocks Relations

After selecting a function to relate a pair of stocks (Step 1 of method MESN) and the choice of a threshold α (Step 2), an unweighted and undirected network is produced. However, in order to study the evolution, several networks of this type must be constructed. The period of time of the data is divided in temporal windows, and for each time window a network is built. A set N of networks is produced with

the same vertices.

With the objective of giving more importance to the most recent data, this method uses the sliding windows technique. Sliding windows requires a parameter β for the size of the window, and a parameter θ that represents the slide.

Example: For a time period of one year, a window size $\beta = 2$ months, and a slide $\theta = 1$ month, a set of networks N with 11 elements $\{Jan-Feb, Feb-Mar, \ldots, Oct-Nov, Nov-Dec\}$ is produced.

3.5 Step 4 - Collecting Measures of the Networks

In this step some measures are computed for each network of the set N constructed in Step 3.

The first measure is the average degree (Avg. Degree) of the network. First, the degree centrality is computed for each vertex of the network. Formally, for a vertex i, the degree centrality is denoted as k_i and is calculated as

$$k_i = \sum_{j=1}^{n} a_{ij} \tag{3.5}$$

where a_{ij} is the entry of the i-th row and j-th column of the adjacency matrix A that represents the network. If there is n stocks in the network, the average degree is given by:

$$Avg.Degree = \bar{k} = \frac{\sum_{i=1}^{n} k_i}{n} \tag{3.6}$$

The average degree is an indicator of the density of the network.

The next centrality to be computed is the eigenvector centrality that is calculated as

$$x_i = \frac{1}{\lambda} \sum_{j=1}^{n} a_{ij} x_j \tag{3.7}$$

where $x_i \backslash x_j$ denotes the centrality of vertex $i \backslash j$; a_{ij} represents an entry of the adjacency matrix A ($a_{ij} = 1$ if vertices i and j are connected by an edge and $a_{ij} = 0$ otherwise); and λ denotes the largest eigenvalue of the adjency matrix that represents the network (Oliveira and Gama, 2012).

The eigenvector centrality is what best defines the importance of a stock in the network, because it is computed based in the importance of the stocks to which it is connected.

Finally, communities are detected for each network of N. This method adopts the Louvain's Algorithm, presented by Blondel et al. (2008). It is described in Section 2.3.3.

3.6 Step 5 - Evolution of the Important and Influential Stocks

In the previous step, the eigenvector centrality x_i for every stock s_i was computed for each network of set N.

In this step, a threshold ϵ is set to detect important and influential stocks. A stock s_i is important if in at least one of the networks of set N its eigenvector centrality is greater of equal to ϵ ($x_i \geq \epsilon$).

It is of interest to analyse the trajectory of important and influential stocks, because they have the capability to influence the price of several stocks in the network.

3.7 Step 6 - Evolution of the Communities

Sometimes really small communities are detected, and it is of no interest to include them in the analysis. Therefore, a filter can be applied to exclude the communities with number of vertices below a threshold ω.

After the communities eligible for analysis are determined, a technique inspired in MEC (Section 2.6) is used to study their evolution through the networks. The focus is on survival transition described in MEC. To detect them, all conditional probabilities $p(C_{i+1}|C_i)$ must be computed, where C_i and C_{i+1} are the set of communities of two consecutive networks of set N. Then, a filter must be applied to consider only the probabilities above a survival threshold τ.

An example of what a transition among communities might be, is shown in Figure 3.2.

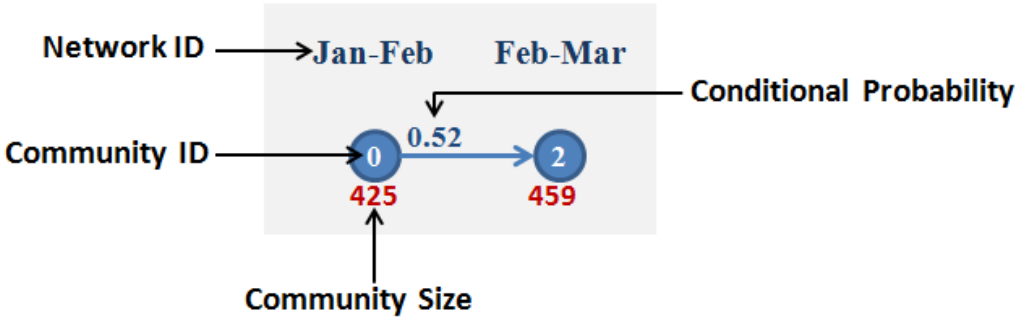


Figure 3.2: Example of a survival transition with $\tau = 0.5$.

In Figure 3.2 each circle represents a community. The number inside each circle is the community ID as defined by Gephi. The number under the circles defines the size of the community. The line that connects the circle indicates that the community with ID equal to 0 at time window Jan-Feb, survived in time window Feb-Mar. The value on the line is the conditional probability of a stock to be in the community with ID 2, knowing that it belonged to community with ID 0.

Chapter 4

Study of Evolving Stock Networks

This chapter describes our case study. We present the results of the experiments that were carried out applying the method MESN (Chapter 3) to data collected for U.S. stocks for the year 2014.

First, we present the results of using the correlation on close prices to construct the networks, as well as the impact of considering negatively correlated stocks in the topology of the network. Additionally, we describe the results obtained for several sets of networks N built with different time window sizes. Moreover, we present the results of using the correlation on daily variation prices and also the results of using Mantegna's distance to construct the networks.

Finally, we discuss the experiments results and how they can be used by an investor.

4.1 Data

The *Russell 1000 Index* provides a comprehensive and unbiased barometer for the large-cap segment and is completely redefined annually to ensure new and growing

equities are reflected (Investments, 2014). It represents 92% of the U.S. stock market.

This work considered the stocks of *Russel 1000* that belonged to NYSE and NASDAQ at December 12^{th} of 2014, which makes a total of 996 stocks. As required by MESN, the close prices of every stock were collected for the year 2014. A total of 21 of the collected stocks did not have a close price for all days of trading. Those stocks were deleted from the database, leaving 975 stocks for our experiments.

A list of the stocks used for this study is available at Appendix A in Tables A.1 and A.2.

4.2 Experiment 1 - Correlation on Close Prices

In the first experiment we used the data as it was collected, that is, we used the close prices to relate the stocks. The goal was to analyse the data and make a sensibility analysis of some parameters.

For this experiment we defined the following setup:

- The function f, used to weight the edge that connects a pair of stocks, is the correlation on close prices ρ.
- To filter the edges a set of threshold values $\alpha \in \{0.6, 0.7, 0.8\}$ were considered, and only positively correlated pairs of stocks remained connected.
- To construct the set of networks N, the window size was set as $\beta = 2$ months and the slide $\theta = 1$ month.
- To study the evolution of important and influential stocks the value of the threshold was defined as $\epsilon = 1$.
- The threshold ω, used to exclude small communities from the communities evolution analysis, was set to $\omega = 4\%$. This way, communities with less than 39 stocks were excluded.

- The survival threshold to consider a transition in MEC's graph was set to $\tau = 0.5$.

4.2.1 Results

In Table 4.1 we present, for each possible condition to filter the edges ($\rho \geq \alpha$), the average degree, and the number of communities obtained for each network of set N.

Network ID	$\rho \geqslant 0.6$		$\rho \geqslant 0.7$		$\rho \geqslant 0.8$	
	Avg.Degree	No.Communities	Avg.Degree	No.Communities	Avg.Degree	No.Communities
Jan-Feb	298.1	4	207.5	13	109.3	52
Feb-Mar	384.7	4	271.1	24	144.1	80
Mar-Apr	201.5	5	131.5	13	65.8	105
Apr-May	200.2	5	98.7	32	52.4	106
May-Jun	241.2	14	171.2	33	99.2	112
Jun-Jul	155.7	6	89.1	31	35.5	130
Jul-Aug	262.4	6	172.2	17	83.2	52
Aug-Sep	331.4	9	232.9	24	122.2	62
Sep-Oct	398.4	5	300.1	15	182.3	35
Oct-Nov	504.0	6	440.1	13	316.2	55
Nov-Dec	259.0	7	172.9	16	86.7	60

Table 4.1: Average degree and number of communities obtained for $\rho \geq \alpha$.

Analysing Table 4.1 we concluded that:

- The stocks are strongly correlated. In fact, even for the higher value of α (0.8) the values of the average degree are very high.

- Stocks are highly connected in the networks of the period October-November and less connected in the networks of April-May.

- There are a high number of communities, especially for high values of α. When the value of α goes up, the probability of a network's vertex to not be connected to another vertex is higher. In other words, the number of vertices with values of degree centrality equal to 0 ($k = 0$) increases. Those vertices are assigned to a community with size 1. To better show this effect, the number of stocks with centrality degree equal to 0 ($k = 0$) is presented in Table 4.2. We also present the total number of communities, and the number of communities with size

greater than 1, obtained for each network of N, and for each possible condition to filter the edges ($\rho \geq \alpha$).

Network ID	$\rho \geqslant 0.6$			$\rho \geqslant 0.7$			$\rho \geqslant 0.8$		
	No. Stocks with $k = 0$	No. Communities	No. Communities with size >1	No. Stocks with $k = 0$	No. Communities	No. Communities with size >1	No. Stocks with $k = 0$	No. Communities	No. Communities with size >1
Jan-Feb	1	4	3	7	13	6	47	52	5
Feb-Mar	1	4	3	15	24	9	71	80	9
Mar-Apr	1	5	4	9	13	4	97	105	8
Apr-May	1	5	4	26	32	6	95	106	11
May-Jun	10	14	4	24	33	9	99	112	13
Jun-Jul	3	6	3	24	31	7	121	130	9
Jul-Aug	1	6	5	13	17	4	43	52	9
Aug-Sep	5	9	4	19	24	5	56	62	6
Sep-Oct	2	5	3	12	15	3	31	35	4
Oct-Nov	3	6	3	9	13	4	48	55	7
Nov-Dec	3	7	4	12	16	4	55	60	5

Table 4.2: Comparing the number of stocks with degree centrality $k = 0$, with the total number of communities and the number of communities with size greater than 1, for different threshold values α.

Analysing Table 4.2 it can be concluded that:

- A high value for the threshold α results in a high number of stocks with degree centrality $k = 0$.
- The number of communities is strongly affected by the number of vertices disconnected of the network.

The threshold α that was used to filter the edges was set as $\alpha = 0.6$ from this moment on. The decision was backed up by the fact that it was the value of α that produced the lower number of vertices disconnected from the network. We were interested in studying the communities of stocks, and to consider this value for α is a way to force the stocks to group.

Table 4.2 is a strong argument in favour of the definition of threshold ω that excludes small communities from the analysis of the evolution of communities. If that threshold was not set, the number of communities available for that analysis

would be a large one, and that would make the analysis harder.

The chart in Figure 4.1 shows the evolution of the stocks that were identified as important and influential.

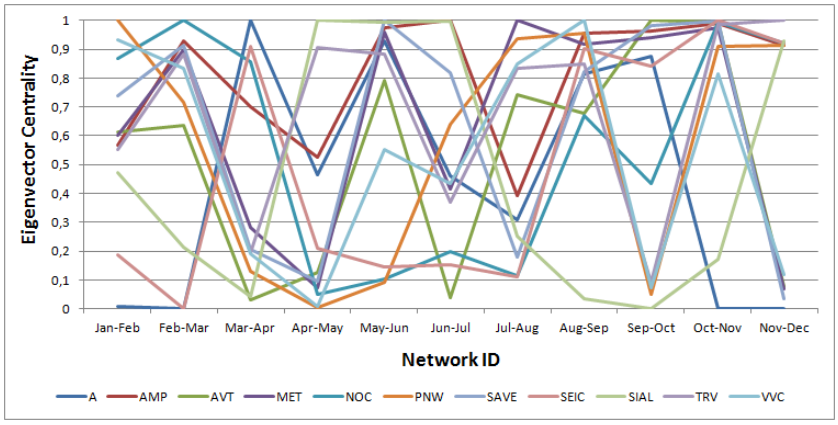


Figure 4.1: Evolution of the important and influential stocks using correlation on close prices.

The evolution of the eigenvector centrality for the stocks defined as important is highly irregular. In this work, we were looking for stocks that maintained their high importance and influence over time, that is, stocks that maintained the capability to influence other stocks. With that knowledge, an investor knows that fluctuations on that stock's prices, affects a large number of stocks.

The graph in Figure 4.2 shows how communities evolved over time.

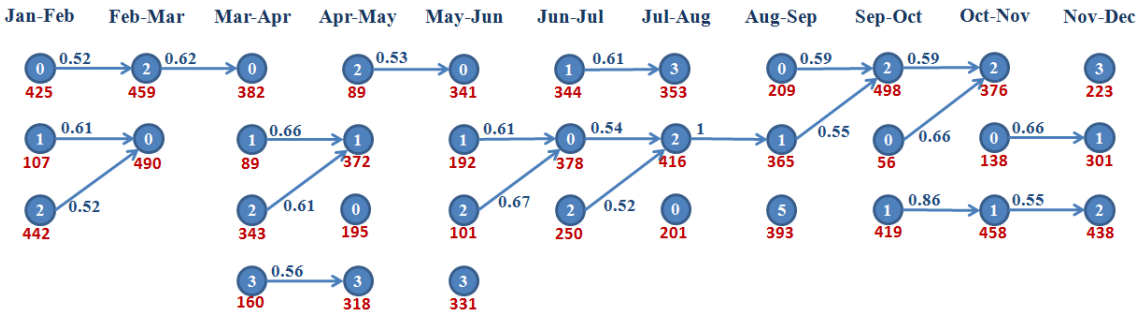


Figure 4.2: Communities evolution with $\rho \geq 0.6$.

Analysing Figure 4.2 we concluded that the evolution of the communities is highly irregular. There is no continuity or stability of the communities over time. Stocks are grouped in consecutive networks of set N in a way that communities die and born in a randomly fashion. We were looking for communities that survive over time. If a community survives through time, the performance of that community can be compared against other communities, the market, or stock indexes. Moreover, the past performance of that community can be used to predict its performance in the future.

4.3 Experiment 2 - Negative Correlations

The goal of this experiment was to study how negative correlated stocks affect the network's topology. It is possible that certain economic or political events cause the increase of price of some stocks, and at the same time the decrease of other stock prices, as an event can be positive for some companies, and at the same time negative to others. For example, the increase of oil price, is good for oil companies, but bad for distribution companies.

In this experiment we defined the following setup:

- The function f, used to weight the edge that connects a pair of stocks, is the correlation on close prices ρ.
- To filter the edges the threshold $\alpha = 0.6$ was considered, and positively and negatively correlated pairs of stocks remained connected.
- To construct the set of networks N the window size was set as $\beta = 2$ months and the slide $\theta = 1$ month.
- To study the evolution of the important and influential stocks the value of the threshold was defined as $\epsilon = 1$.

- The threshold ω, used to exclude small communities, was set as $\omega = 4\%$.
- To study the evolution of communities, the survival threshold was set to $\tau = 0.5$.

4.3.1 Results

Table 4.3 presents the average degree and the number of communities obtained for the networks of set N, and compares them with the values obtained in the previous experiment for the same value of α.

Network ID	$\rho \geqslant 0.6$		$\lvert\rho\rvert \geqslant 0.6$	
	Avg.Degree	No.Communities	Avg.Degree	No.Communities
Jan-Feb	298.1	3	346.3	4
Feb-Mar	384.7	3	409.3	3
Mar-Apr	201.5	4	290.6	4
Apr-May	200.2	4	213.3	4
May-Jun	241.2	4	296.4	3
Jun-Jul	155.7	3	225.6	4
Jul-Aug	262.4	5	315.5	3
Aug-Sep	331.4	4	370.6	3
Sep-Oct	398.4	3	420.0	3
Oct-Nov	504.0	3	576.3	3
Nov-Dec	259.0	4	323.0	3

Table 4.3: Average degree and number of communities for $\rho \geq 0.6$ and $|\rho| \geq 0.6$.

The objective of this experiment was to verify if the negatively correlated stocks affected significantly the topology of the network. Comparing the left columns ($\rho \geq 0.6$) with the right columns ($|\rho| \geq 0.6$) of Table 4.3 it can be concluded that this new connections are very significant inside the network.

The chart in Figure 4.3 shows the evolution of the stocks that were identified as important and influential.

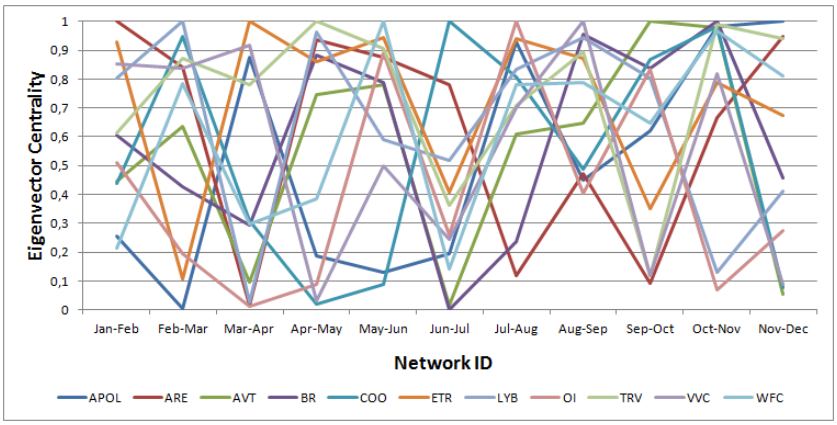


Figure 4.3: Evolution of the important and influential stocks considering negatively correlated stocks.

As in the previous experiment, the evolution of the eigenvector centrality for the important stocks is highly irregular. Again, it is not possible at a designated point in time to predict if the stock importance in a network $n_i \in N$ remains high in the network n_{i+1}. Considering the negatively correlated stocks was a step in the wrong direction in the context of the study of the evolution of important and influential stocks.

The graph in Figure 4.4 shows how communities evolved over time.

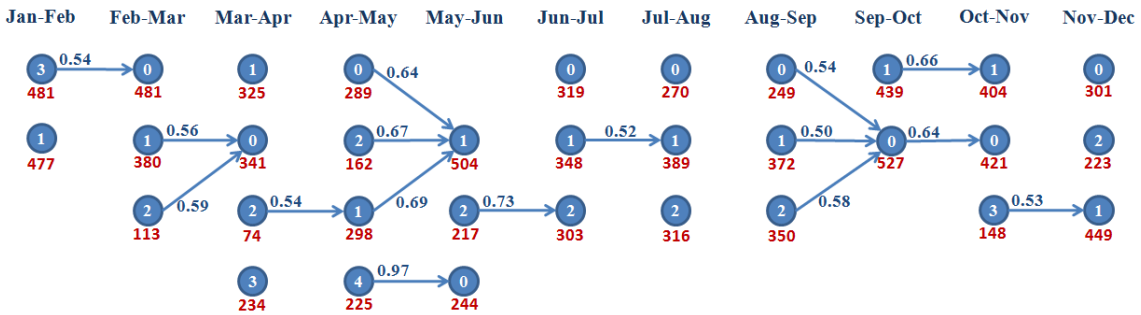


Figure 4.4: Communities evolution with $|\rho| \geq 0.6$.

Analyzing Figure 4.4, it can be concluded that the evolution of the communities

is highly irregular. As in the previous experiment, there is no continuity or stability of the communities over time. Stocks are grouped in consecutive networks of set N in a way that communities die and born in a randomly fashion. It is important that communities survive through the networks of set N, as the evolution of communities that born and die constantly are not interesting to study. To create the networks with this setup adds no gain from the point of view of community dynamics.

Connecting the negatively correlated pair of stocks seemed promising at first while constructing the networks. It showed an impact in the density of the networks, and that could lead to better results in the analysis of the important stocks and communities' evolution. However, those analyses revealed poor results, which led to the decision of abandoning the inclusion of negatively correlated stocks in further experiments.

4.4 Experiment 3 - Incrementing the Window Size

The value of the correlation depends on the length of the time series that is being used. In smaller time series, one element of the series has a higher importance than in larger series. The goal of this experiment was to verify if considering larger window sizes, which means larger time series to be correlated, produces more static networks, namely when studying the evolution of important and influential stocks and communities.

For this experiment we defined the following setup:

- The function f, used to weight the edge that connects a pair of stocks, is the correlation on close prices ρ.
- To filter the edges the threshold $\alpha = 0.6$ was considered, and only positively correlated pair of stocks remained connected.

- The window size β took values from 3 months until 8 months. For each value of β a set N of networks was constructed.
- To study the evolution of the important and influential stocks the value of the threshold ϵ was set as $\epsilon = 1$.
- The threshold ω, used to exclude small communities, was set as $\omega = 4\%$.
- To study the evolution of communities, the survival threshold was set to $\tau = 0.5$.

4.4.1 Results

Table 4.4 presents the average degree and the number of communities detected for each set of networks N for the different window sizes β.

$\beta = 3$ months			$\beta = 4$ months			$\beta = 5$ months		
Network ID	Avg. Degree	No.Communities	Network ID	Avg. Degree	No.Communities	Network ID	Avg. Degree	No.Communities
Jan-Mar	272.3	3	Jan-Apr	207.0	4	Jan-May	155.8	7
Feb-Apr	258.5	3	Feb-May	175.3	4	Feb-Jun	209.4	4
Mar-May	146.2	4	Mar-Jun	182.1	5	Mar-Jul	177.2	5
Apr-Jun	207.3	4	Apr-Jul	192.0	5	Apr-Aug	172.5	4
May-Jul	193.6	5	May-Aug	169.6	6	May-Sep	157.1	5
Jun-Aug	188.4	4	Jun-Sep	181.6	4	Jun-Oct	210.2	5
Jul-Sep	211.6	3	Jul-Oct	240.8	4	Jul-Nov	254.9	3
Aug-Oct	321.1	4	Aug-Nov	326.8	3	Aug-Dec	344.0	4
Sep-Nov	388.5	3	Sep-Dec	373.2	3			
Oct-Dec	457.2	4						

$\beta = 6$ months			$\beta = 7$ months			$\beta = 8$ months		
Network ID	Avg. Degree	No.Communities	Network ID	Avg. Degree	No.Communities	Network ID	Avg. Degree	No.Communities
Jan-Jun	188.3	4	Jan-Jul	194.6	4	Jan-Aug	184.1	4
Feb-Jul	205.8	5	Feb-Aug	189.0	5	Feb-Sep	179.8	4
Mar-Aug	163.4	5	Mar-Sep	155.8	5	Mar-Oct	146.5	5
Apr-Sep	160.5	5	Apr-Oct	155.3	5	Apr-Nov	170.2	4
May-Oct	161.2	5	May-Nov	176.7	4	May-Dec	213.5	4
Jun-Nov	225.4	4	Jun-Dec	267.8	3			
Jul-Dec	290.8	4						

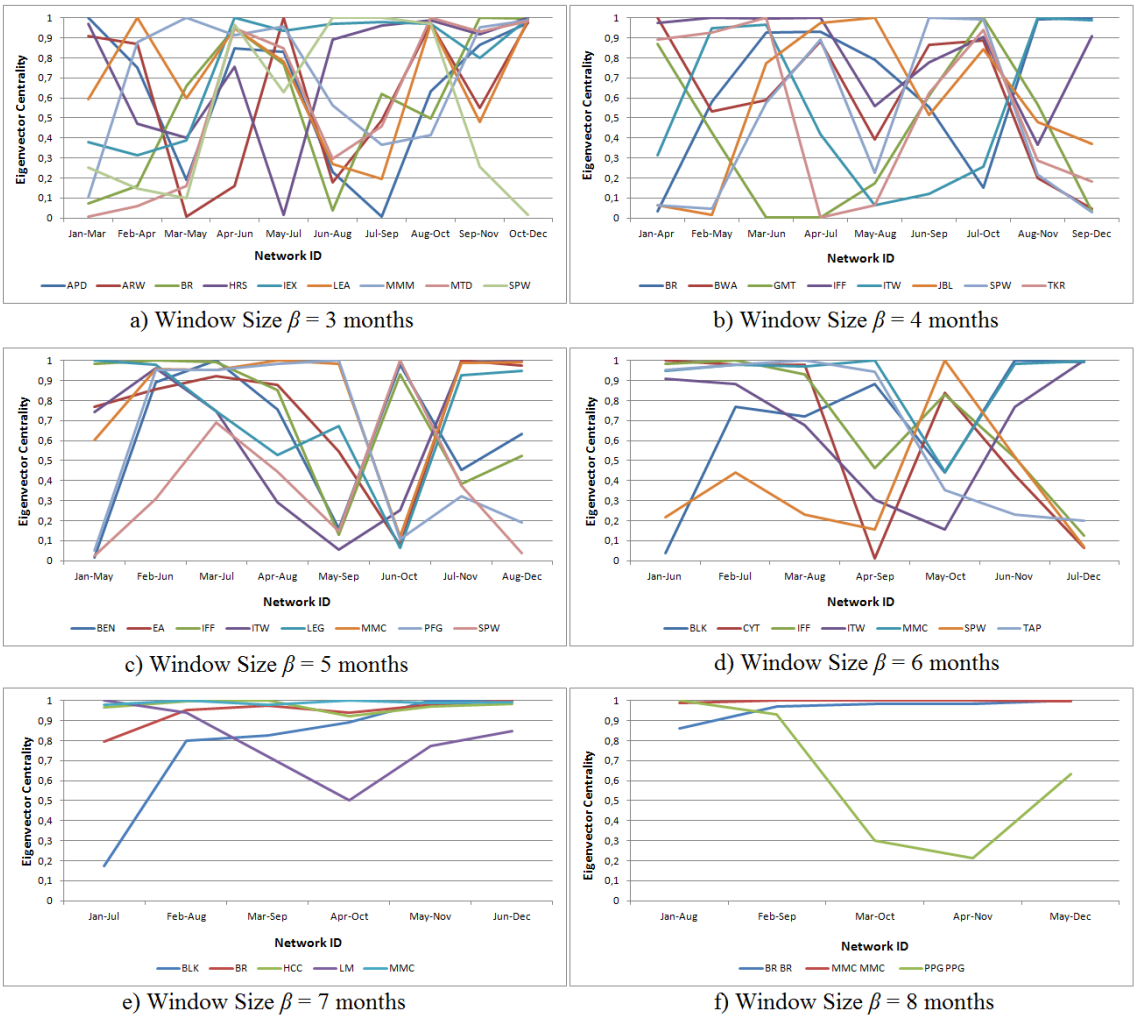
Table 4.4: Average degree and number of communities for networks with different window sizes $\beta \in \{3 \text{ months}, 4 \text{ months}, \ldots, 8 \text{ months}\}$.

Analysing Table 4.4, it can be concluded that:

- The stocks were again highly correlated with each other. However, the average degree decreases when the window size increases.

- There are no significant changes in terms of the number of communities detected.

The charts of Figure 4.5 show the important and influential stocks identified for the different sets of networks built.



a) Window Size $\beta = 3$ months

b) Window Size $\beta = 4$ months

c) Window Size $\beta = 5$ months

d) Window Size $\beta = 6$ months

e) Window Size $\beta = 7$ months

f) Window Size $\beta = 8$ months

Figure 4.5: Evolution of the important and influential stocks for different window sizes $\beta \in \{3 \text{ months}, 4 \text{ months}, \ldots, 8 \text{ months}\}$.

The evolution of the eigenvector centrality for the important stocks is very interesting. For the networks constructed for larger window sizes, the importance of a stock remained high during the time analysed. The stability of the evolution of this

centrality increases as the size of the windows is assigned with higher values. The window size that gives better results, from the point of view of the evolution of the stock's importance analysis, was $\beta = 7$ months.

The graphs in the Figure 4.6 show how communities evolved over time.

When the window size increases, the continuity of the communities became more evident. For example, in the first graph, the one corresponding to a 3 months window size, all communities of the network with ID Jan-Mar die. Moreover, at the graph that corresponds to the window size 6 months, there was only one community that survived through all time windows (community with ID = 0 detected in the network with ID Jan-Jun).

In the graph of window size 7 months, 3 of the 4 communities detected in the first network of set N, survived during the entire period. The same thing happened to window size 8 months.

Therefore, the choice was between window sizes 7 and 8 months. The 7 months window size led to satisfactory results in terms of community survival, with smaller time series needed to compute the correlations. Because of this, it was the window size chosen as reference for the next experiments. As said before, it is important that communities survive through the networks of set N, or the study of the evolution of that community is not useful.

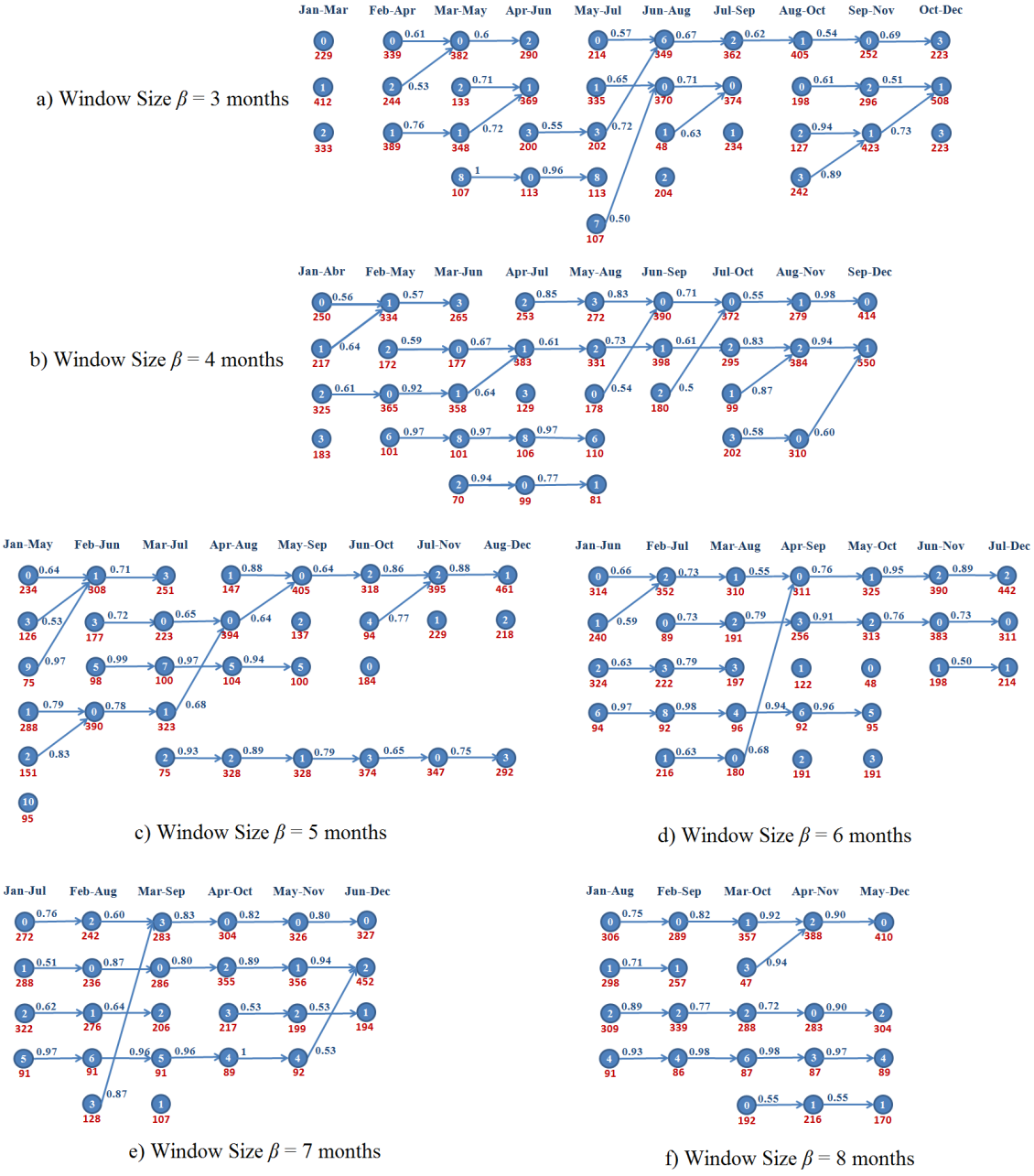


Figure 4.6: Evolution of the important and influential stocks for different window sizes $\beta \in \{3 \text{ months}, 4 \text{ months}, \ldots, 8 \text{ months}\}$.

4.5 Experiment 4 - Correlation on Variations

The goal of this experiment was to study how the network behaves considering the daily price variation for computing the correlations, instead of the close price as in the previous experiments.

Some stock prices are hundreds of dollars and there are stocks that have prices of cents of a dollar. The daily variation of the close prices makes the time series used on correlation to be on the same scale.

In this experiment we defined the following setup:

- The function f, used to weight the edge that connects a pair of stocks, is the correlation on daily variations ρ.
- To filter the edges the threshold $\alpha = 0.6$ was considered, and only positively correlated pair of stocks remained connected.
- The window size β has values from 2 months until 7 months. For each value of β a set N of networks was constructed.
- To study the evolution of the important and influential stocks the value of the threshold ϵ was set as $\epsilon = 1$.
- The threshold ω, used to exclude small communities, was set as $\omega = 4\%$.
- To study the evolution of communities the survival threshold was set to $\tau = 0.5$.

4.5.1 Results

Table 4.5 presents the average degree and the number of communities detected for each set of networks N for the different window size β.

$\beta = 2$ months			$\beta = 3$ months			$\beta = 4$ months		
Network ID	Avg. Degree	No.Communities	Network ID	Avg. Degree	No.Communities	Network ID	Avg. Degree	No.Communities
Jan-Feb	60.5	14	Jan-Mar	33.3	25	Jan-Apr	25.5	35
Feb-Mar	46.6	17	Feb-Apr	33.8	34	Feb-May	20.3	36
Mar-Apr	55.9	18	Mar-May	25.8	30	Mar-Jun	13.8	44
Apr-May	42.9	19	Apr-Jun	17.5	38	Apr-Jul	11.2	46
May-Jun	14.3	35	May-Jul	9.8	51	May-Aug	7.6	48
Jun-Jul	16.8	35	Jun-Aug	9.8	46	Jun-Sep	9.5	49
Jul-Aug	23.4	21	Jul-Sep	15.7	37	Jul-Oct	33.6	30
Aug-Sep	26.2	22	Aug-Oct	61.7	28	Aug-Nov	28.4	39
Sep-Oct	117.7	12	Sep-Nov	42.1	31	Sep-Dec	42.6	29
Oct-Nov	65.7	19	Oct-Dec	58.7	22			
Nov-Dec	50.4	10						

$\beta = 5$ months			$\beta = 6$ months			$\beta = 7$ months		
Network ID	Avg. Degree	No.Communities	Network ID	Avg. Degree	No.Communities	Network ID	Avg. Degree	No.Communities
Jan-May	17.7	44	Jan-Jun	12.6	49	Jan-Jul	9.5	50
Feb-Jun	13.0	51	Feb-Jul	9.5	50	Feb-Aug	8.0	46
Mar-Jul	9.4	55	Mar-Aug	7.6	50	Mar-Sep	7.5	56
Apr-Aug	8.3	52	Apr-Sep	7.9	53	Apr-Oct	13.7	43
May-Sep	7.7	57	May-Oct	15.3	50	May-Nov	11.6	48
Jun-Oct	20.8	40	Jun-Nov	13.8	46	Jun-Dec	16.3	35
Jul-Nov	19.0	42	Jul-Dec	21.7	33			
Aug-Dec	31.4	35						

Table 4.5: Average degree and number of communities for networks with different window sizes $\beta \in \{2 \text{ months}, 3 \text{ months}, \ldots, 7 \text{ months}\}$.

Analysing Table 4.5, it can be concluded that:

- Although the stocks are highly correlated, they are less correlated when compared with the experiments using correlations on close prices.
- To use the daily variation time series to compute the correlation increases the number of communities. In fact, when compared with the previous experiments, the number of communities is significantly larger.

The charts in Figure 4.7 show the important stocks.

a) Window Size $\beta = 2$ months

b) Window Size $\beta = 3$ months

c) Window Size $\beta = 4$ months

d) Window Size $\beta = 5$ months

e) Window Size $\beta = 6$ months

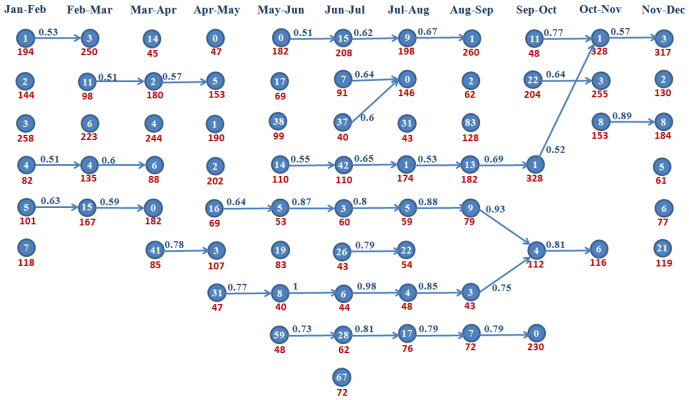
f) Window Size $\beta = 7$ months

Figure 4.7: Evolution of the important and influential stocks for different window sizes $\beta \in \{2 \text{ months}, 3 \text{ months}, \ldots, 7 \text{ months}\}$.

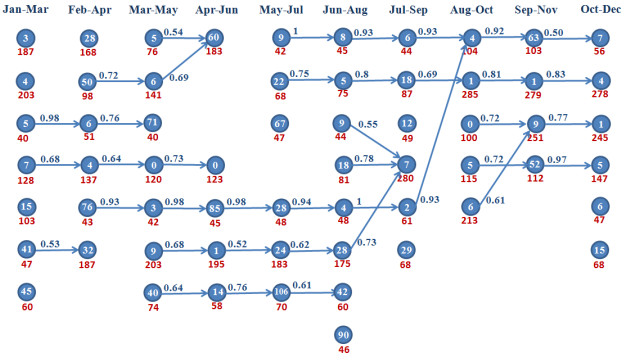
From the observation of the charts it can be concluded that incrementing the window size produces a more stable evolution of the eigenvector centrality through networks. Moreover, the number of stocks that reaches the value 1 decreases when the window size increases. As in the experiment 3 (Section 4.4), the window size that produced a better result was 7 months.

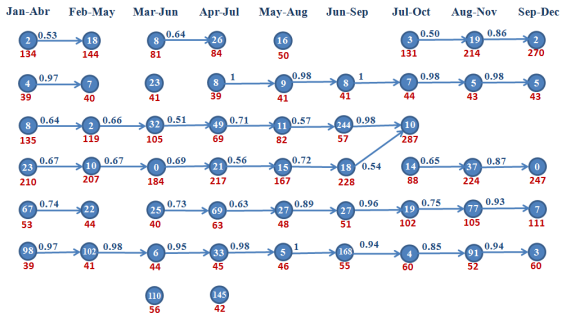
The graphs in Figure 4.8 show how communities evolved over time.

a) Window Size β = 2 months

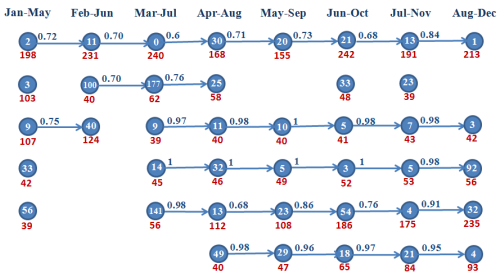


b) Window Size β = 3 months

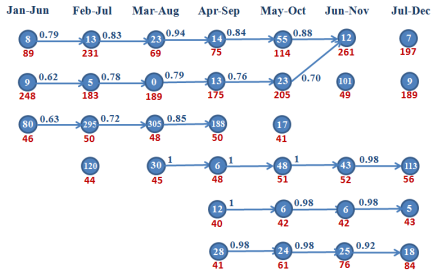




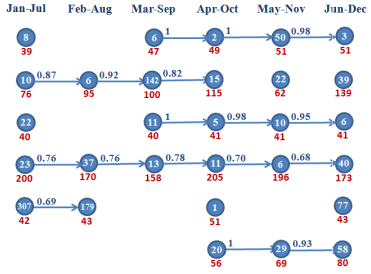
c) Window Size β = 4 months



d) Window Size β = 5 months



e) Window Size β = 6 months



f) Window Size β = 7 months

Figure 4.8: Evolution of the communities for different window sizes.

In the previous experiment (Section 4.4), the increment of the window size generated communities that were more likely to survive. In this experiment, that does not happen. Communities for networks with window size of 4 or 5 months are more likely to survive than the communities detected for networks produced with a window size of 7 months.

4.6 Experiment 5 - Mantegna's Distance

The goal of this experiment was to study how the networks behave considering the distance d introduced by Mantegna (1999).

In this experiment we define the following setup:

- The function f, used to weight the edge that connects a pair of stocks is the distance of Mantegna d.

- The function d is defined as $d(i,j) = \sqrt{2(1-\rho_{ij})}$. For a $\rho_{ij} = 0.6$ the value of d is 0.89. Therefore, to filter the edges, a threshold $\alpha = 0.89$ was considered.

- To construct the networks of set N the window size was assigned to $\beta =$ 7 months and the slide is $\theta = 1$ month.

- To study the evolution of the important and influential stocks the value of the threshold was defined as $\epsilon = 1$.

- The threshold ω, used to exclude small communities, was set as $\omega = 4\%$.

- To study the evolution of the communities, the survival threshold was set to $\tau = 0.5$.

4.6.1 Results

Table 4.6 presents side by side the average degree and the number of communities of the networks produced in this experiment and in the experiments described in Sections 4.4 and 4.5 for time window sizes $\beta = 7$ months.

Network ID	Experiment 3 (Section 4.4)		Experiment 4 (Section 4.5)		Experiment 5	
	Avg. Degree	No.Communities	Avg. Degree	No.Communities	Avg. Degree	No.Communities
Jan-Jul	194.6	4	9.5	50	39.5	39
Feb-Aug	189.0	5	8.0	46	38.0	35
Mar-Sep	155.8	5	7.5	56	37.8	36
Apr-Oct	155.3	5	13.7	43	42.6	31
May-Nov	176.7	4	11.6	48	40.9	30
Jun-Dec	267.8	3	16.3	35	16.4	41

Table 4.6: Average degree and number of communities for experiments 3, 4, and 5 respectively.

From the analysis of Table 4.6 it can be concluded that:

- The networks build using the correlation based on the close prices are denser than the networks that were built using the correlation based on the daily variations and the ones that were built using the distance of Mantegna.
- The density of a network appears to affect the number of communities detected. The denser the network is, less communities it has.

The chart in Figure 4.9 shows the evolution of the stocks that were identified as important and influential using Mantegna's distance.

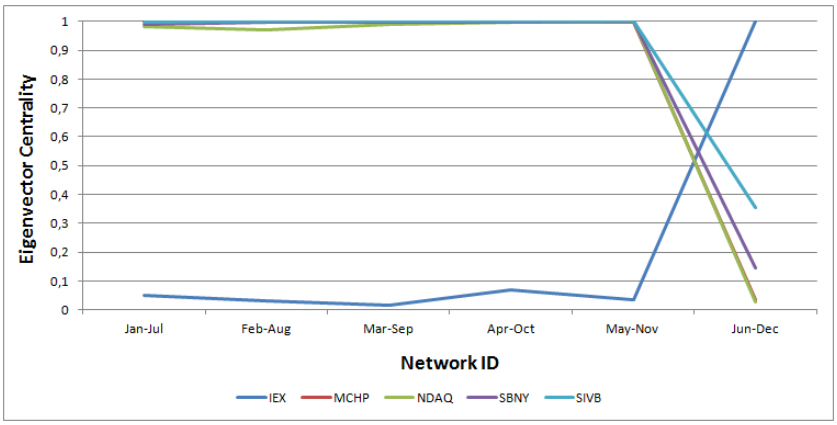


Figure 4.9: Evolution of the important and influential stocks using Mantegna's distance.

The stocks have a very stable value of the eigenvector centrality through the networks, but suddenly, at the network with ID Jun-Dec the value drops for the majority of the stocks and rises for one particular stock (IEX).

The graph in Figure 4.10 shows how communities evolve over time.

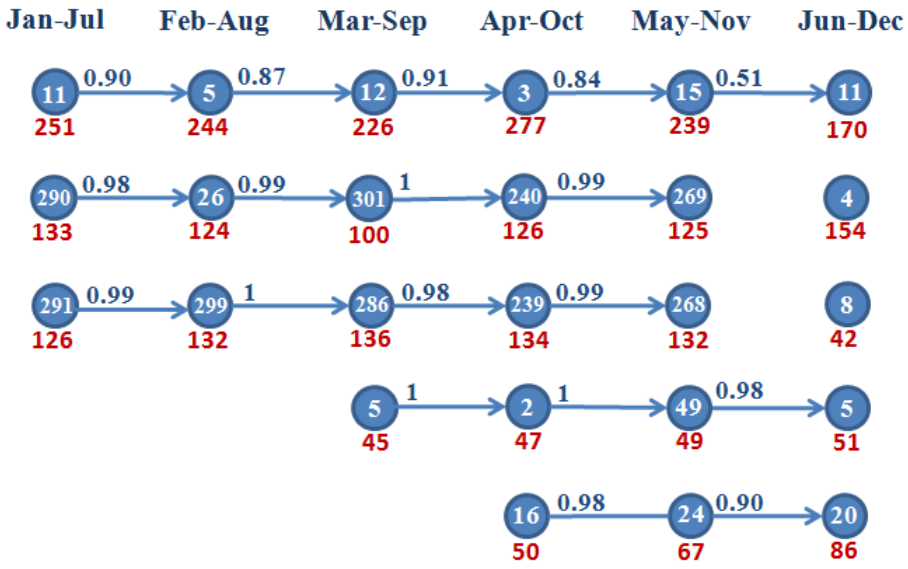


Figure 4.10: Communities evolution with $d(s_i, s_j) \leq 0.89$.

We verified that the communities have a long survival period. Moreover, the conditional probability associated to the transitions is very high. That means that the community detected at a determined network of set N, inherits a large number of stocks of the community detected in the previous network.

4.7 Discussion of the Results

The main goal of this work was to develop a method that could deliver a list of important stocks and groups of stocks to an investor. These could be the focus of his analysis and be part of his trading decision process. The method MESN presented in Chapter 3, that was applied to real data in this chapter, allowed the identification of the important stocks and the study of communities evolution. We had the best results for experiments performed for time window sizes of 7 months. The next paragraphs discuss how these results can be used by an investor.

The first task that was carried out was the normalisation the data collected, namely the close prices, to reduce the stock prices to the same scale. With that data the chart in Figure 4.11 was produced. It shows the evolution of the market for the year 2014.

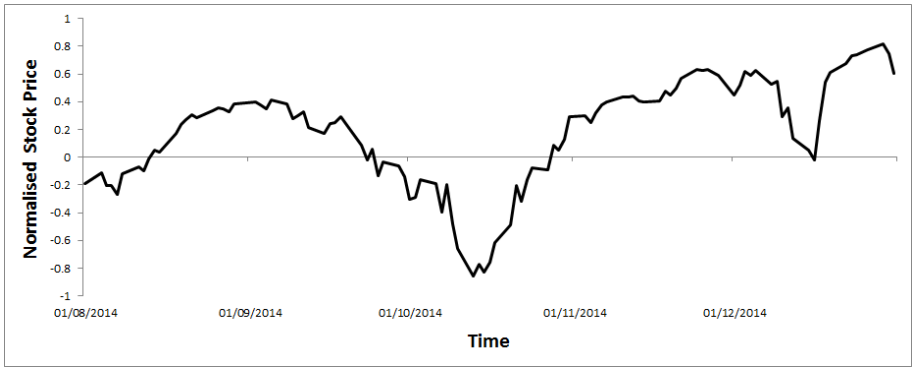


Figure 4.11: Market evolution for the time period August to December.

The reason why the first 7 months are not presented in Figure 4.11 is that the

window size is 7. August is the month when eigenvector centralities were computed and communities were detected for the first time. After that, those actions were carried out once a month, which corresponds to the value of the window slide used in the experiments.

The discovery of an important or influential stock can be defined as the moment that its eigenvector centrality reaches the value 1. Table 4.7 presented the important stocks of the experiments made for window sizes $\beta = 7$ months and their discovery date.

Experiment 3		Experiment 4		Experiment 5	
Stock	Discovery Date	Stock	Discovery Date	Stock	Discovery Date
BLK	01/12/2014	AFG	01/08/2014	IEX	01/01/2015
BR	01/01/2015	AMP	01/10/2014	MCHP	01/08/2014
HCC	01/10/2014	HON	01/09/2014	NDAQ	01/12/2014
LM	01/08/2014	IEX	01/01/2015	SBNY	01/12/2014
MMC	01/09/2014	PFG	01/12/2014	SIVB	01/09/2014

Table 4.7: Important and influential stocks and their discovery date.

The charts in Figures 4.12, 4.13, and 4.14 compare the market evolution with the stocks detected.

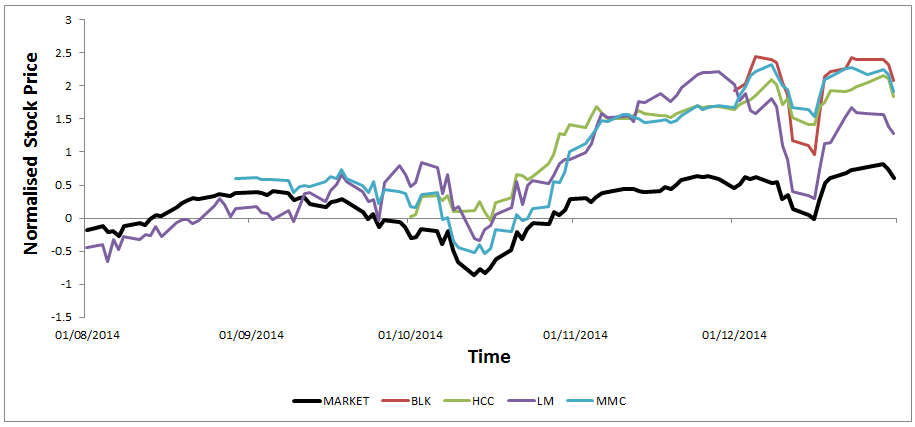


Figure 4.12: Comparing the market evolution with the important and influential stocks detected on the experiment performed using the correlation on close prices.

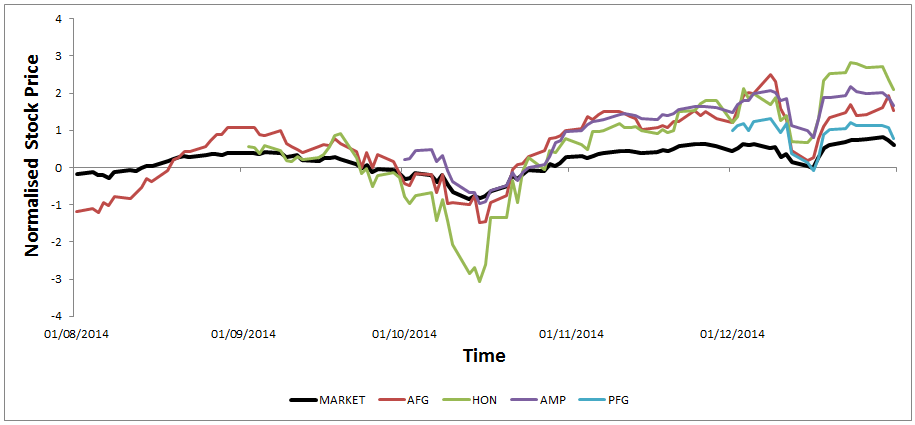


Figure 4.13: Comparing the market evolution with the important and influential stocks detected on the experiment performed using the correlation on daily variations of close prices.

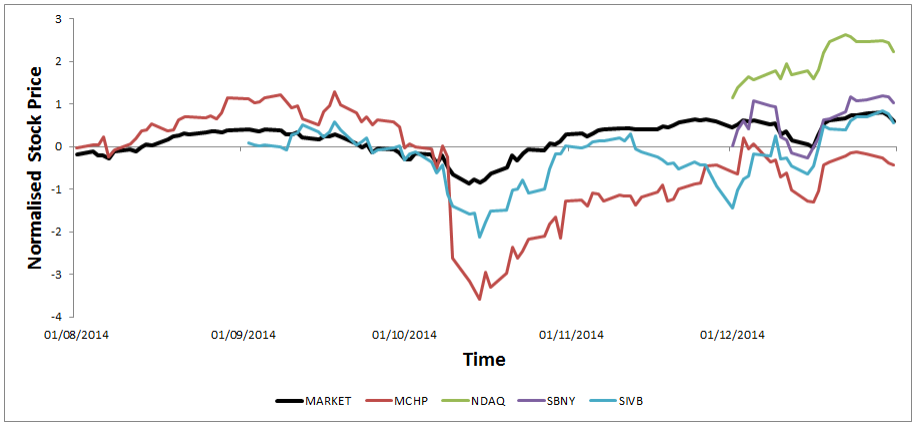


Figure 4.14: Comparing the market evolution with the important and influential stocks detected on the experiment performed using the Mantegna's distance.

A similar analysis can be carried out for communities that were detected in those experiments.

From the analysis of Figure 4.6 (Section 4.4), namely for the graph produced for window sizes of 7 months, it can be concluded that it is of interest to analyse the trajectory of the communities with ID C_0, C_1 and C_5 at the first window. The

chart that compares the performance of the market against these communities is presented in Figure 4.15.

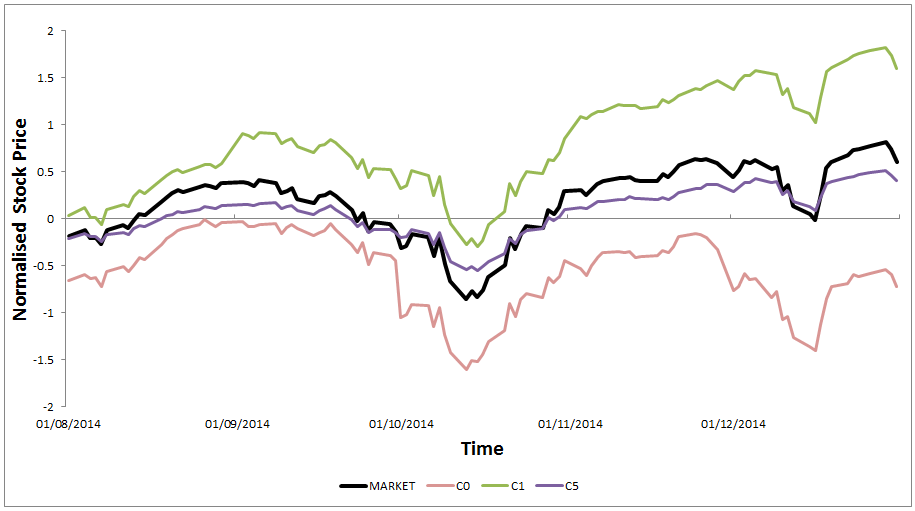


Figure 4.15: Comparing the market evolution with communities detected on the experiment performed using the correlation on close prices.

From the analysis of Figure 4.8 (Section 4.5), for the set of networks constructed from the correlation on daily variations with a window size of 7 months, it is of interest to analyse the trajectory of the community with ID C_{23}, and the communities with ID C_6 and C_{11} detected for the time window Mar-Sep. The chart that compares the performance of the market against these communities is presented in Figure 4.16.

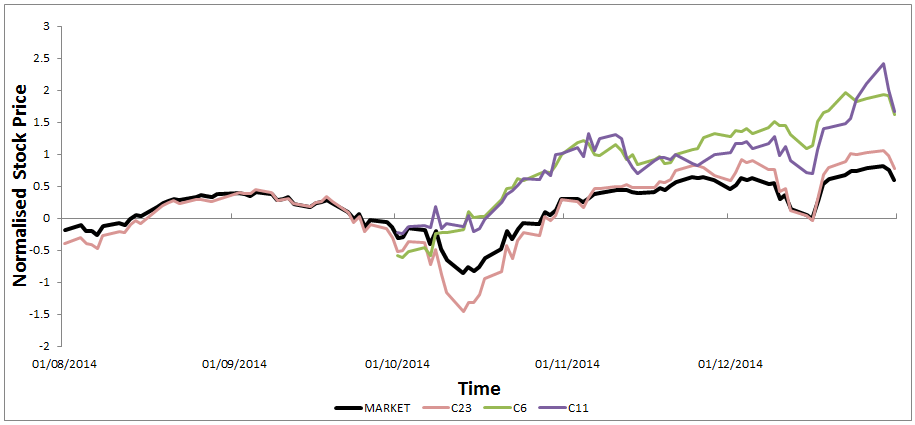


Figure 4.16: Comparing the market evolution with communities detected on the experiment performed using the correlation on daily variations of close prices.

From the analysis of Figure 4.10 (Section 4.6), for the set of networks made using the distance of Mantegna with a window size of 7 months, it is of interest to analyse the trajectory of the communities with ID C_{11}, C_{290} and C_{291} detected in the time window Jan-Jul, and the community with ID C_5 detected for the time window Mar-Sep. The chart that compares the performance of the market against these communities is presented in Figure 4.17.

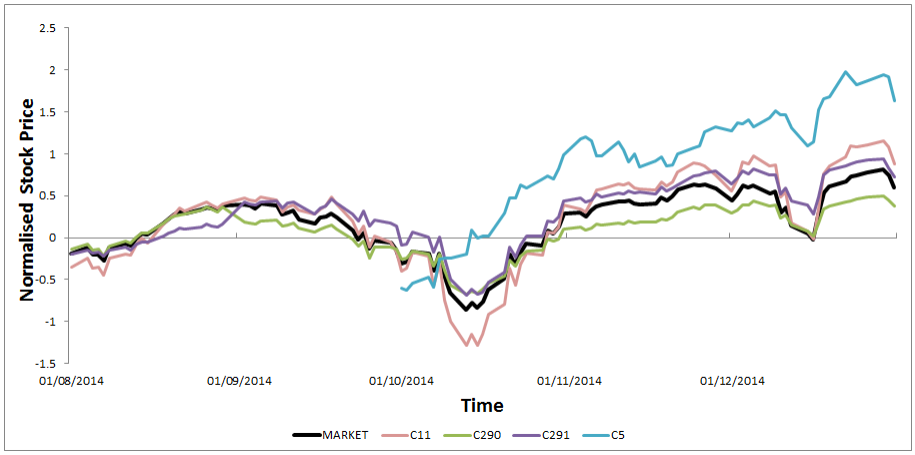


Figure 4.17: Comparing the market evolution with communities detected on the experiment performed using the Mantegna's distance.

An interesting aspect that is observed in the charts above, is the fact that the communities follow the evolution of the market. It seems that the communities are correlated with the market and with each other.

A different analysis that can be made is from the returns point of view. Table 4.8 presents the monthly returns or variations for the market as a whole, and the influential and important stocks detected in each experiment in analysis and the communities.

		August	September	October	November	December
	Market	4.44	-3.96	4.11	1.47	0.93
Experiment 3 Window Size of 7 months	BLK					0.71
	HCC			8.91	1.94	1.15
	LM	4.38	3.46	2.93	8.43	-4.76
	MMC		-1.52	5.10	3.38	1.34
	C_0	4.46	-3.62	3.98	-0.44	-0.40
	C_1	4.19	-2.87	4.33	2.94	1.71
	C_5	5.20	-3.81	3.09	3.38	1.32
Experiment 4 Window Size of 7 months	AFG	6.71	-3.48	4.11	0.77	0.86
	AMP			5.45	4.22	1.20
	HON		-2.49	4.87	3.56	2.59
	PFG					-1.37
	C_{23}	4.47	-4.69	5.28	2.37	1.25
	C_6			10.28	0.88	3.62
	C_{11}			8.18	-0.54	3.64
Experiment 5	MCHP	4.44	-3.17	-6.42	4.49	0.88
	NDAQ					8.32
	SBNY					5.16
	SIVB		-0.48	2.14	-6.09	14.8
	C_{11}	4.58	-3.99	4.78	2.28	1.78
	C_{290}	5.15	-3.53	3.43	2.89	0.78
	C_{291}	4.70	-2.27	5.01	2.13	0.73
	C_5			10.3	0.95	3.76

Table 4.8: Monthly returns for important stocks, communities and market in percentage.

The analysis of Table 4.8 gives an interesting insight of the stock market dynamics to an investor. Rapidly he can identify the most important stocks, the internal

groups of stocks, and how did they perform in the recent past. Hopefully, he will use this knowledge to maximize his profits and to avoid or at least reduce his losses.

Chapter 5

Conclusions

The challenge of this work was to develop a method to study the evolution of important and influential stocks and the evolution of communities in stock networks.

We proposed a method (MESN), that was described in Chapter 3, and was inspired in Social Network Analysis. First, a strategy to construct a weighed and undirected stock network was described. The edges were weighed using a function that was correlation based. After that, a threshold α was defined to filter edges, and finally an undirected and unweighted stock network was produced. The method MESN required the construction of several stock networks for different time windows. For each of these networks some vertex level metrics were computed, like the degree and eigenvector centralities, and communities were detected. Finally, a technique to study the evolution of important and influential stocks and a technique to study the evolution of stock's communities were presented.

Five experiments were carried out applying the method to data collected for U.S. stocks for the year 2014.

The first experiment considered correlation on stock close prices for sliding windows of 2 months. The important and influential stocks detected could not maintain their importance over time, and the communities detected did not survive consis-

tently in transitions through networks. These results were considered unsatisfactory. We were looking for stocks that maintain their importance and influence high and for communities that survive over time. It is on the best interest of an investor to analyse the stocks that have the capability to influence other stocks. Moreover, if a community survives through time, the performance of that community can be compared against the market, and some prediction can be made for the future.

The following experiment studied the effect of negatively correlated stocks in the topology of a network. The results did not improve when compared with the first experiment.

The third experiment revealed that the increment of the size of the time windows identifies stocks that maintain their importance high, and the communities detected survive more consistently. The results showed that the use of time windows of size 7 months could be a good choice for the data collected.

Additionally, two more experiments were carried out. One using correlation on daily variations of close prices, and the other the distance presented by Mantegna (1999). The results confirmed that the use of time windows of size 7 months was a good choice for the data collected.

Finally, we discussed what would be the most efficient way to provide the knowledge retrieved from the experiments to an investor. The investor needs some numbers to help him in his decision process of buying and/or selling stocks. Providing the monthly returns for the whole market, the influential and important stocks, and the communities, gives the investor a rapid way to compare performances. Based on these performances he can decide to buy or sell stocks (or communities of stocks) or simply do nothing.

5.1 Limitations and Future Work

The biggest limitation of the method purposed is its inadequacy to be applied to streaming data. The computation of the correlations between every pair of stocks, the metrics associated to each vertex and the detection of communities, are very demanding for today computers. A change on the method to be adequate to streaming data, maintaining however his SNA inspiration, can be made in the future.

Moreover, different metrics and algorithms can be considered to identify the important stocks and to detect communities in the stock networks.

Another study that can be carried out in the future is the characterization of the communities considering the different business sectors. This could allow us to verify if same business sectors stocks are grouped in the same communities.

A possible evolution of the method, is to use the monthly returns of the important stocks and communities that were shown in Table 4.8 to be used as training data of a classifier. The goal would be to predict if a determined stock should be bought, sold, or no action should be taken. This way, the investor has an automatic suggestion for an investment decision that can maximize his profits and avoid or at least reduce his losses.

References

Arasu, A. and Manku, G. S. (2004). Approximate counts and quantiles over sliding windows. In *Proceedings of the twenty-third ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 286–296. ACM.

Asur, S., Parthasarathy, S., and Ucar, D. (2009). An event-based framework for characterizing the evolutionary behavior of interaction graphs. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 3(4):16.

Blondel, V. D., Guillaume, J.-L., Lambiotte, R., and Lefebvre, E. (2008). Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10):P10008.

Bonanno, G., Caldarelli, G., Lillo, F., and Mantegna, R. N. (2003). Topology of correlation-based minimal spanning trees in real and model markets. *Physical Review E*, 68(4):046130.

Costa, L. d. F., Oliveira Jr, O. N., Travieso, G., Rodrigues, F. A., Villas Boas, P. R., Antiqueira, L., Viana, M. P., and Correa Rocha, L. E. (2011). Analyzing and modeling real-world phenomena with complex networks: a survey of applications. *Advances in Physics*, 60(3):329–412.

De Meo, P., Ferrara, E., Fiumara, G., and Provetti, A. (2013). Enhancing community detection using a network weighting strategy. *Information Sciences*, 222:648–668.

Diestel, R. (2005). Graph theory. *Grad. Texts in Math.*

Eom, C., Oh, G., and Kim, S. (2007). Deterministic factors of stock networks based on cross-correlation in financial market. *Physica A: Statistical Mechanics and its Applications*, 383(1):139–146.

Euler, L. (1741). Solutio problematis ad geometriam situs pertinentis. *Commentarii academiae scientiarum Petropolitanae*, 8:128–140.

Fortunato, S. (2010). Community detection in graphs. *Physics Reports*, 486(3):75–174.

Gama, C., Carvalho, A., Oliveira, M., Faceli, K., and Lorena, A. (2012). Extração de conhecimento de dados, data mining. *JC Gama, Extração de Conhecimento de Dados, Data Mining*, page 101.

Girvan, M. and Newman, M. E. (2002). Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826.

Investments, R. (2014). Russel indexes. https://www.russell.com/indexes/americas/indexes/fact-sheet.page?ic=US1000. Last accessed on Dec 12, 2014.

Kiers, H. A. (2000). Towards a standardized notation and terminology in multiway analysis. *Journal of chemometrics*, 14(3):105–122.

Lee, G. S. and Djauhari, M. A. (2012). An overall centrality measure: The case of us stock market. *International Journal of Electrical & Computer Sciences*, 12(6).

Mantegna, R. N. (1999). Hierarchical structure in financial markets. *The European Physical Journal B-Condensed Matter and Complex Systems*, 11(1):193–197.

Newman, M. (2010). *Networks: an introduction*. Oxford University Press.

Newman, M. E. and Girvan, M. (2004). Finding and evaluating community structure in networks. *Physical review E*, 69(2):026113.

Oliveira, M. and Gama, J. (2010). Bipartite graphs for monitoring clusters transitions. In *Advances in Intelligent Data Analysis IX*, pages 114–124. Springer.

Oliveira, M. and Gama, J. (2012). An overview of social network analysis. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2(2):99–115.

Oliveira, M. and Gama, J. (2013). Visualization of evolving social networks using actor-level and community-level trajectories. *Expert Systems*, 30(4):306–319.

Scott, D. L. (2003). *Wall Street words: an A to Z guide to investment terms for today's investor*. Houghton Mifflin Harcourt.

Sylvester, J. J. (1878). On an application of the new atomic theory to the graphical representation of the invariants and covariants of binary quantics, with three appendices. *American Journal of Mathematics*, 1(1):64–104.

Tutte, W. (2001). Graph theory (reprint of the 1984 original). *Encyclopedia of Mathematics and its Applications*, 21.

Wang, G., Shen, Y., and Ouyang, M. (2008). A vector partitioning approach to detecting community structure in complex networks. *Computers & Mathematics with Applications*, 55(12):2746–2752.

Wasserman, S. and Faust, K. (1994). *Social network analysis: Methods and applications*, volume 8. Cambridge university press.

Appendix A

List of Studied Stocks

Table A.1 presents the stocks considered in this case study that are listed in the New York Stock Exchange (NYSE).

A	AA	AAN	AAP	ABBV	ABC	ABT	ACC	ACE	ACN	ACT	ADM	ADS	ADT	AEE
AEP	AES	AET	AFG	AFL	AGCO	AGN	AGO	AHL	AIG	AIV	AIZ	AJG	AL	ALB
ALK	ALL	ALLE	ALR	ALSN	AMD	AME	AMG	AMH	AMP	AMT	AMTD	AN	ANF	AOL
AON	AOS	APA	APAM	APC	APD	APH	AR	ARE	ARG	ARMK	ARW	ASH	ATI	ATK
ATO	ATR	ATW	AVB	AVP	AVT	AVX	AVY	AWH	AWI	AWK	AXP	AXS	AYI	AZO
BA	BAC	BAH	BAX	BBT	BBY	BCR	BDN	BDX	BEN	BG	BHI	BIG	BIO	BK
BKD	BKU	BLK	BLL	BMR	BMS	BMY	BOH	BR	BRO	BRX	BSX	BTU	BWA	BWC
BXP	C	CAB	CAG	CAH	CAM	CAT	CB	CBG	CBI	CBL	CBS	CBT	CCE	CCI
CCK	CCL	CCO	CE	CF	CFN	CFR	CFX	CHD	CHH	CHK	CHS	CI	CIE	CIM
CIT	CL	CLF	CLGX	CLH	CLR	CLX	CMA	CMG	CMI	CMP	CMS	CNA	CNC	CNK
CNP	CNX	COF	COG	COH	COL	COO	COP	COTY	COV	CPA	CPB	CPN	CPT	CR
CRI	CRL	CRM	CRS	CSC	CSL	CST	CSX	CTL	CVA	CVC	CVD	CVI	CVS	CVX
CXO	CXP	CXW	CYH	CYN	CYT	D	DAL	DATA	DBD	DCI	DD	DDD	DDR	DDS
DE	DEI	DFS	DG	DGX	DHI	DHR	DIS	DKS	DLB	DLR	DNB	DNR	DO	DOV
DOW	DPS	DPZ	DRC	DRE	DRI	DRQ	DST	DSW	DTE	DUK	DV	DVA	DVN	EAT
ECL	ED	EFX	EGN	EIX	EL	ELS	EMC	EMN	EMR	ENH	ENR	EOG	EQR	EQT
ESS	ETN	ETR	EV	EVHC	EW	EXC	EXP	EXR	F	FBHS	FCX	FDO	FDS	FDX
FE	FHN	FI	FII	FIS	FL	FLO	FLR	FLS	FLT	FMC	FNF	FRC	FRT	FSL
FTI	G	GAS	GCI	GD	GE	GEF	GGG	GGP	GIS	GLW	GM	GME	GMT	GNC
GNW	GPC	GPN	GPS	GRA	GS	GWR	GWW	GXP	H	HAL	HAR	HBI	HCA	HCC
HCN	HCP	HD	HE	HES	HFC	HHC	HIG	HII	HLF	HLT	HME	HNT	HOG	HON
HOT	HP	HPQ	HPT	HRB	HRC	HRL	HRS	HSP	HST	HSY	HTA	HTZ	HUM	HUN
HXL	IBM	ICE	IEX	IFF	IGT	IHS	IM	INGR	INT	IP	IPG	IR	IRM	IT
ITC	ITT	ITW	IVZ	JAH	JBL	JCI	JCP	JEC	JLL	JNJ	JNPR	JOY	JPM	JWN
K	KAR	KBR	KEX	KEY	KIM	KMB	KMT	KMX	KO	KORS	KOS	KR	KRC	KSS
KSU	L	LAZ	LB	LDOS	LEA	LEG	LEN	LGF	LH	LII	LLL	LLY	LM	LMT
LNC	LNKD	LNT	LO	LOW	LPI	LUK	LUV	LVLT	LVS	LXK	LYB	LYV	M	MA
MAA	MAC	MAN	MAS	MBI	MCD	MCK	MCO	MCY	MD	MDT	MDU	MET	MFA	MGM
MHFI	MHK	MJN	MKC	MLM	MMC	MMM	MNK	MO	MON	MOS	MPC	MRC	MRK	MRO
MS	MSCI	MSI	MSM	MTB	MTD	MTW	MUR	MUSA	MWV	N	NAV	NBL	NBR	NCR
NEE	NEM	NEU	NFG	NFX	NI	NKE	NLSN	NLY	NNN	NOC	NOV	NOW	NRF	NRG
NSC	NSM	NU	NUE	NUS	NWL	NYCB	O	OAS	OC	OCN	OCR	OFC	OGE	OHI
OI	OII	OIS	OKE	OMC	ORCL	ORI	OSK	OXY	P	PAG	PANW	PAY	PBF	PBI

PCG	PCL	PCP	PEG	PEP	PF	PFE	PFG	PG	PGR	PH	PHM	PII	PKG	PKI
PL	PLD	PLL	PM	PNC	PNR	PNW	POM	PPG	PPL	PPS	PRA	PRE	PRGO	PRU
PSA	PSX	PVH	PWR	PX	PXD	Q	QEP	R	RAD	RAI	RAX	RBC	RCL	RDC
ROK	ROL	ROP	RPAI	RPM	RRC	RS	RSG	RTN	RYN	S	SBH	SCCO	SCG	SCHW
SCI	SD	SDRL	SE	SEAS	SEE	SFG	SHW	SIG	SIX	SJM	SKT	SLB	SLG	SLH
SM	SMG	SNA	SNH	SNI	SNV	SO	SON	SPB	SPG	SPN	SPR	SPW	SRC	SRE
STI	STJ	STR	STT	STWD	STZ	SUNE	SWI	SWK	SWN	SWY	SYK	SYY	T	TAHO
TAP	TCB	TCO	TDC	TDG	TDS	TDW	TE	TEG	TER	TEX	TFX	TGI	TGT	THC
THG	THO	TIF	TJX	TK	TKR	TMHC	TMK	TMO	TMUS	TOL	TPX	TRGP	TRI	TRN
TRV	TRW	TSN	TSO	TSS	TTC	TUP	TW	TWC	TWO	TWTR	TWX	TXT	TYC	UA
UAL	UDR	UFS	UGI	UHS	UNH	UNM	UNP	UNT	UPL	UPS	URI	USB	USG	USM
UTX	V	VAL	VAR	VC	VEEV	VFC	VLO	VMC	VMI	VMW	VNO	VNTV	VOYA	VR
VSH	VTR	VVC	VZ	WAB	WAT	WBC	WCC	WCN	WDAY	WDR	WEC	WFC	WHR	WLK
WLL	WM	WMB	WMT	WPC	WPX	WR	WRB	WRI	WSM	WTR	WU	WWAV	WY	WYN
X	XEC	XEL	XL	XLS	XOM	XRX	XYL	Y	YELP	YUM	ZMH	ZTS		

Table A.1: Stocks of NYSE considered in the case study.

Table A.2 presents the stocks considered in this case study that are listed in the NASDAQ Stock Exchange.

AAL	AAPL	ACGL	ADBE	ADI	ADP	ADSK	AGNC	AKAM	ALGN	ALKS	ALNY	ALTR	ALXN	AMAT
AMCX	AMGN	AMZN	ANSS	APOL	ARCP	ARRS	ASNA	ATHN	ATML	ATVI	AVGO	AWAY	BBBY	BEAV
BIIB	BMRN	BOKF	BPOP	BRCD	BRCM	BRKR	CA	CAR	CBOE	CBSH	CBST	CDNS	CDW	CELG
CERN	CHRW	CHTR	CINF	CMCSA	CME	COMM	COST	CPRT	CREE	CSCO	CSGP	CTAS	CTSH	CTXS
DISCA	DISH	DLTR	DNKN	DOX	DTV	DWA	EA	EBAY	ENDP	EQIX	ERIE	ESRX	ETFC	EWBC
EXPD	EXPE	FAST	FB	FEYE	FFIV	FISV	FITB	FLIR	FNFG	FOSL	FOXA	FSLR	FTNT	FTR
FULT	GILD	GLNG	GLPI	GMCR	GNTX	GOOG	GPOR	GRMN	GRPN	GT	HAIN	HAS	HBAN	HCBK
HDS	HOLX	HSIC	IACI	IBKR	ICPT	IDXX	ILMN	INCY	INFA	INTC	INTU	IPGP	ISRG	JAZZ
JBHT	JDSU	JKHY	KLAC	KRFT	LAMR	LECO	LKQ	LLTC	LMCA	LPLA	LPNT	LRCX	LSTR	LVNTA
MAR	MAT	MCHP	MDLZ	MDRX	MDVN	MNST	MORN	MRVL	MSFT	MSG	MU	MXIM	MYGN	MYL
NATI	NCLH	NDAQ	NDSN	NFLX	NTAP	NTRS	NUAN	NVDA	NWSA	ODFL	ONNN	ORLY	PACW	PAYX
PBCT	PCAR	PCLN	PCYC	PDCO	PETM	PINC	PNRA	PPC	PTC	PTEN	QCOM	QGEN	REGN	RGLD
ROST	ROVI	RRD	RVBD	SATS	SAVE	SBAC	SBNY	SBUX	SCTY	SEIC	SFM	SGEN	SHLD	SIAL
SIRI	SIRO	SIVB	SLGN	SLM	SLXP	SNDK	SNPS	SPLK	SPLS	SPWR	SRCL	SSYS	STLD	STRZA
SWKS	SYMC	TECD	TECH	TFSL	TRIP	TRMB	TROW	TSCO	TSLA	TXN	ULTA	URBN	UTHR	VIAB
VRSK	VRSN	VRTX	WDC	WEN	WFM	WIN	WOOF	WYNN	XLNX	XRAY	YHOO	Z	ZBRA	ZION
ZNGA	ZU													

Table A.2: Stocks of NASDAQ considered in the case study.