Online Network Analysis of Stock Markets

by

Pedro Miguel Pereira Mota da Costa

Master Thesis in Modeling, Data Analysis and Decision Support Systems

Supervised by

Professor João Manuel Portela da Gama

Faculdade de Economia

Universidade do Porto

2018

Dedicated to Alexandre, César and Catarina

Biographical Sketch

Pedro Costa was born in April 1974, in Porto, Portugal. He earned his graduate degree in Computer Science from the Faculdade de Ciências da Universidade do Porto in 2003 and his masters degree in Artificial Intelligence in 2007 from the same school.

While in college, he started his professional career as software developer in 1998. He did his internship in Faculdade de Medicina da Universidade do Porto and worked there for some years as a junior researcher. After that, he moved to the private sector and worked in companies such as Critical Software, Rocket Internet and lately at PaddyPower-Betfair, where he plays the role of delivery manager. Throughout his career, he has worked on multiple businesses, from electronic medical records to semi-conductors, telecommunications, online portfolio management, e-commerce and online gambling.

In 2015, he has enroled in the Masters of Data Analysis at Faculdade de Economia da Universidade do Porto.

Acknowledgements

I would like to thank my supervisor, Professor João Gama, for his guidance, support, patience, advice and sense of humour. I am honoured to have studied under him and extremely grateful for all that he has taught me over time.

I would like to thank to all my family, for their encouragement and support. Special thanks to my parents, Ana and Alberto, who taught me to always aim for excellence and to become a better man each passing day. To my brother and syster, José and Ana, for their everlasting friendship and love. To my grand-mother, Aurora, whom I dearly miss so much, for encouraging me to pursue my studies and never give up no matter what.

A very special thanks to my wife Catarina, for her support and patience during the full length of this course. To both my suns, Alexandre and César, for their unconditional love and all the joy they bring to my life every day.

To all of you, my sincere gratitude.

Resumo

Os mercados de acções são considerados sistemas imprevisíveis (Fama, 1965) e complexos (Bak et al., 1996; Mantegna and Stanley, 2000), estando as variações nos preços das acções correlacionadas entre si (Bak et al., 1996).

Existem várias metodologias para estudar as correlações entre pares de séries temporais representando atributos de acções. Uma abordagem possível é mapear estas correlações em árvores (Mantegna, 1999) e redes (Tse et al., 2010) que eventualmente poderão evoluir no tempo (Onnela et al., 2003). Em tais redes, os vértices representam acções e as arestas representam correlações entre as acções.

Este trabalho descreve uma metodologia para a interpretação em tempo real das dinâmicas dos mercados de acções. A análise de redes sociais (SNA) serve de fonte de inspiração, sendo utilizadas as suas técnicas e métricas para medir a evolução dos mercados, detectar comunidades e quantificar e qualificar a influência das acções.

O método propõe a análise temporal de dados financeiros transmitidos de forma contínua, sendo construidas várias redes correspondentes a diferentes períodos temporais. O fluxo contínuo de dados acarreta desafios importantes, como calcular estatísticas sobre dados infinitos, escolher amostras não enviesadas e lidar com mudanças de contexto.

O método é aplicado a dados referentes a acções transacionadas em mercados norte-americanos durante o período de 1997 a 2017, sendo os resultados analisados e interpretados. É feita uma breve discussão sobre a aplicabilidade do método na constituição de portfólios de investimento aceitáveis como motivação para trabalho futuro.

Palavras-Chave: mercados de acções, redes complexas, análise de redes sociais, fluxos de dados

Abstract

Stock markets are regarded as unpredictable (Fama, 1965) and complex systems (Bak et al., 1996; Mantegna and Stanley, 2000), where changes in stock price are correlated to each other (Bak et al., 1996).

Several methodologies exist to study pair-wise correlations between time series of stock attributes. A possible approach is to map correlation to trees (Mantegna, 1999) and networks (Tse et al., 2010) that may evolve in time (Onnela et al., 2003). In such networks, nodes represent stocks and edges represent correlations between stocks.

This work describes a methodology for the online interpretation of stock market dynamics. Inspiration is drawn from social network analysis (SNA). Its techniques and metrics are used to gauge the evolution of markets, detect communities and to quantify and qualify the influence of stocks.

The method proposes temporal analysis of streaming financial data, constructing several networks that correspond to different periods of time. Data streams raise important challenges, such as computing statistics over infinite data, picking unbiased samples and dealing with concept change.

The method is applied to data regarding stocks traded from 1997 through 2017 in United State's stock markets. Results are analysed and interpreted. A discussion is held on the method's applicability in the construction of acceptable investment portfolios as motivation for future work.

Keywords: stock market, complex networks, social network analysis, data streams

Contents

List of Tables

List of Figures

Chapter 1

Introduction

In this chapter we introduce the problem to be studied, the motivation and contributions of this work.

1.1 Problem definition

Stock markets are regarded as unpredictable and complex systems (Mantegna and Stanley, 2000). Stock prices follow non-stationary time series (Fama, 1965) that are correlated with changes in volume (Podobnik et al., 2009) and to each other (Bak et al., 1996; Gopikrishnan et al., 2001). Stock prices also have a social component, as many times volatility results from crown effect, with investors mimicking each others' behaviours in response to market trends (Bak et al., 1996).

Investors perceive the value of a stock as a function of its expected future dividends (Markowitz, 1952). As prices fluctuate through time, investors seek out tools that can help them in decision-making regarding investment spending and durable consumption (Raunig and Scharler, 2010), so that future returns can be maximized while minimizing risk (Lima, 2015). Markowitz (1952), Rosenow et al. (2002) and Roll (2013) show that the diversification of investment into non-correlated assets reduces risk, even in times of crisis (Preis et al., 2012).

The problem we face is then how to determine which communities develop in a stock market, how do those communities evolve over time, which stocks are most influential and how is that influence exerted over the remaining stocks according to investors' reactions to economical, financial and political stimuli.

1.2 Motivation

Stock markets have been studied by economists and mathematicians for a long time. In recent years, a growing number of physicists is engaging in a ground-breaking multidisciplinary field of of research know as *econophysics* (Mantegna and Stanley,

2000), where the analysis of economic systems is approached with techniques used to solve physics' problems.

One important line of work under econophysics umbrella is that of asset price dynamics in stock markets. Existing literature proposes several methodologies to study pair-wise correlations between time series of stock attributes (e.g. price, returns, volume). A possible approach is to map correlation to trees (Mantegna, 1999) and networks (Tse et al., 2010), possibly evolving in time (Onnela et al., 2003, 2004). Typically, in such networks, nodes represent stocks and edges represent strong correlation between stocks.

Interesting contributions can be brough in from other disciplines. Otte and Rousseau (2002) suggest the use of social network analysis (SNA) in information sciences. SNA studies how the relations between individuals evolve through time, how does this affect the importance and influence of those individuals and of the communities they are involved in.

In recent works, Roy and Sarkar (2011), Dimitrios and Vasileios (2015) and Lima (2015), among others, use SNA techniques to study the evolution of stock market networks . One major drawback in such studies is that the used data sets of financial time series are of a considerable size. Whenever processing such data batches, one must consider practical aspects. On the one hand, batch processing of data is hard to scale and evolve. Time and space complexity of processing algorithms are major concerns, especially when dealing with evolving systems where the arrival of new items implies the reprocessing of the whole batch. On the other hand, the optimal size of the data set is hard to determine. Statistics go in disarray when dealing with big data sets, naturally prone to concept drift. Conversely, small data sets may fail to represent the phenomenon under study.

Arguably, a streaming approach is preferable. Nowadays, vast amounts of information are streaming out of financial markets, day-in, day-out, down to the second. Moreover, the sources of information are diverse: published annual reports, business periodicals, specialized sites and investment advisory services. Despite its appeal, streaming data is also not easy to process. In fact, streams raise important challenges, such as computing statistics over infinite data, picking unbiased samples and dealing with concept change.

1.3 Contributions

The goal of this work is to establish a methodology that can help investors to analyse streaming stock market data, exposing the most important correlations between stocks returns in the form of networks. Communities, influential stocks, common stock-sets and network metrics are made available for the investor to assist in decision making while conducting portfolio management.

1.4 Organization

This thesis is structured as follows:

Chapter 2 presents background information and state-of-art techniques in Stock Networks, Social Network Analysis and Streaming.

Chapter 3 presents the methodology used to study the dynamics of a stock market, including the criteria used to detect communities, identify important stocks and frequent stock-sets. It includes a presentation of the experimental system.

Chapter 4 presents the application of the methodology to data regarding the components of a stock market industrial index. The experimental results are presented and discussed throughout this chapter. The applicability of the methodology to portfolio management is also approached.

Chapter 5 presents the conclusions, limitations and future paths of this work.

Chapter 2

State of the Art

In this chapter, we go through the state of the art and look for the different approaches to study the evolution of stock markets: how to collect, process and represent data, what metrics to collect and how to interpret the results.

The chapter is organized as follows: in the first section, we start by a brief presentation of graph theory essentials and then move on to relevant papers about financial networks, how to build and interpret them and the differences between several types of networks. In the second section, we present bibliography on the application of social networks analysis methodologies over financial networks. In the third and final section, we go through papers on streaming, seeking to summarize the best approaches to how to deal with financial data streams.

Throughout the entire chapter, we summarize the results, findings and suggestions of futures lines of work of the reviewed literature.

2.1 Networks

Networks have been used to study complex problems ever since Euler's solution for the *Königsberg bridge problem* (Euler, 1736). Networks are often represented by *graphs*, mathematical structures consisting of sets of objects that are connected together.

Formally, a graph $G = (V, E)$ is an ordered pair consisting of a non-empty set V of nodes and a set E of edges, which are pairs of nodes $(u, v) \in V$. A graph G is said to be directed if E is comprised of sets of ordered pairs, otherwise it is undirected. A graph G is said to be weighted if the edges in E are assigned with numerical weights representing the strength of the connections.

The adjacency matrix A of a graph G is a $n \times n$ matrix defined as

$$A_{uv} = \begin{cases} 1 \text{ if } (u, v) \in E \\ 0 \text{ otherwise} \end{cases} \tag{2.1}$$

For undirected graphs, matrix A is symmetric. For weighted graphs, the adjacency matrix is defined as

$$W_{uv} = \begin{cases} w \text{ if } (u,v) \in E \\ 0 \text{ otherwise} \end{cases} \tag{2.2}$$

where w represents the weight or strength of the edges.

The order of a graph G, denoted as $|V(G)| = n$, is the number of elements of the set V of nodes. The size of a graph G, denoted as $|E(G)| = m$ is the number elements of the set E of edges. For directed graphs, the maximum number of edges is $m_{\max} = n(n-1)$, for undirected graphs it is $m_{\max} = n(n-1)/2$.

An alternating sequence of nodes and edges $v_0, e_0, v_1, \ldots, v_{k-1}, e_{k-1}, v_k$ that begins and ends with nodes is called a walk (of length k). A trail is a walk in which all edges are distinct. A path is a trail in which all nodes are distinct. Two nodes u and v are connected if there is a path between them in G. A graph G is connected when there is a path between every pair of nodes; in particular, a graph with a single node is connected. A graph that is not connected is disconnected.

A connected undirected graph G with no cycles is called a tree. A tree T that spans the entire graph G and is a subgraph of G is called a spanning tree. If G is a weighted graph and T bears the minimum possible total edge weight, then T is called a minimal spanning tree (MST).

A compreehesive exposition on graph theory is presented in Bondy et al. (1976). Graph theory provides answers in many fields of research, from Mathematics to Social Sciences (Boccaletti et al., 2006).

2.1.1 Hierarchical structure in financial markets

In recent years, the involvement of physicists in the study of economical problems brought graph theory into Economy and Finance (Mantegna and Stanley, 2000).

Mantegna (1999) publishes a seminal paper proposing a topological approach to the characterization of stock markets. The author represents markets as graphs where stocks are nodes and edges are established based on similarities in stocks' return performance. Given a pair of stocks (i, j), similarity is determined by the correlation coefficient

$$\rho_{ij} = \frac{< X_i X_j > - < X_i >< X_j >}{\sqrt{(< X_i^2 > - < X_i >^2)(< X_j^2 > - < X_j >^2)}} \tag{2.3}$$

where $X_i = lnP_i(t) - lnP_i(t-1)$ is a series of daily logarithmic return values, $< X_i >$ is the average over all trading days in a studied time period T and $P_i(t)$ is the closing quote (price) of stock i at days $t \in T$.

The author introduces a distance function

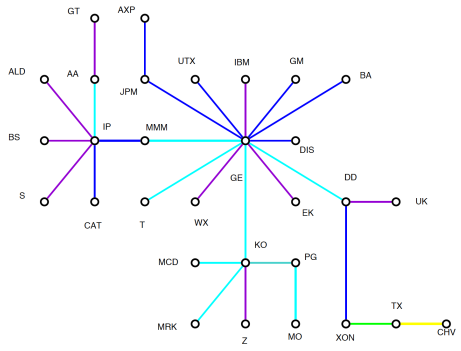


Figure 2.1: MST for Dow Jones Industrial.

$$d(i,j) = 1 - \rho_{ij}^2 \tag{2.4}$$

that, at once, is a function of the of stocks' i and j similarity and fulfils the four axioms of Euclidean metric:

1. $d(i,j) \geq 0$
2. $d(i,j) = 0 \iff i = j$
3. $d(i,j) = d(j,i)$
4. $d(i,j) \leq d(i,k) + d(k,j)$

The distance matrix $D = \{d_{ij} \mid \forall i, i\}$ is used to determine a minimal spanning tree (MST), a particular kind of weighted graph first described in Borůvka (1926)[1]. Figure 2.1 illustrates an MST of the Dow Jones Industrial index components from mid 1989 through late 1995.

Mantegna observes that MSTs exhibit interesting properties that ensure the preservation of the most relevant connections between stocks in a portfolio. Each n node of the MST is connected by $n-1$ edges without any loops. Moreover, the sum of the weights of edges is minimal. Later, Vandewalle et al. (2000) prove that the probability of nodes in stock-correlation MSTs to exhibit a degree of k is given by

$$P(k) \sim k^{-\alpha} \tag{2.5}$$

Networks where the distribution of node degrees follow such a power-law are knows as *scale-free networks*. Vandewalle et al. results are corroborated by Boginski et al. (2005) when authors conduct a statistical study of stock-correlation graphs to

[1]An excellent translation of this paper is presented in Nešetřil et al. (2001)

prove that the same power law also applies to nodes in graphs defined in according to the methodology of Onnela et al. (2004).

Sector-based clustering, volatility and the expressiveness of correlation

Bonanno et al. (2004) finds evidence of clustering based on economical sectors in MSTs obtained from the components of S&P 100 index from 1995 through 1998. Seeking to confirm the *Epps effect*, a decrease in stock return correlation caused by the increase in data sampling (Epps, 1979), Bonanno et al. change Mantegna's definition of return series to $X_i = lnP_i(t) - lnP_i(t - \Delta t)$, where Δt is the frequency of data sampling. Authors build several MSTs by picking different values for Δt, from one day down to nineteen minutes. In this process, they witness the transformation of the MSTs from multi-cluster hierarchical structures to simpler star-shaped graphs where intra-sector correlations decrease fast.

Bonanno et al. analyse the volatility of stock, a key financial indicator obtained by the ratio between daily maximum and minimum quotes. The authors notice that the probability distribution functions of return and volatility are different in nature; the former is symmetrical, the later is skewed. For this reason, Bonanno et al. suggest more robust non-parametric correlation coefficients, such as Spearman Rank, to study volatility. Results show that MSTs obtained from Spearman's correlation are more stable and present better characterized clusters than those obtained from Pearson's, but no major topological differences emerge between the two when the same sampling rate is used.

Finally, Bonanno et al. compare the topological properties of correlation with an uncorrelated Gaussian time series (Dudley, 1965) and the robust *one-factor* model (Sharpe, 1964). Authors notice that none of the MSTs resulting from the later models is able to capture the topological properties of those spawned by real-data's correlation. This is particularly interesting since the one-factor model is known to explain more than 80% of correlation coefficients observed in real-data.

Trees versus Graphs

Tumminello et al. (2005) argue that stock correlations disclosed by MSTs are too restrictive and that valuable information is lost. Tumminello et al. propose a richer correlation-based graph called *Planar Maximally Filtered Graph* (PMFG), which is iteratively built by adding highly correlated nodes as long as the resulting graph is planar and can be embedded in a surface with a fixed genus $G = 0$ (i.e., a plane or a sphere). The genus of a surface is the largest number of non-intersecting closed curves that can be drawn on the surface without separating it.

Tuminello et al. point out several advantages in PMFGs. First, every MST is always included in a PMFG. Second, for a set of n nodes, MST has $n - 1$ edges

whereas the PMFG has $3(n-2)$. Last, while the MST is a tree, the PMFG is a network with loops and cliques of 3 or 4 elements. The PMFG is therefore more appealing in terms of information, since 4-element cliques can reveal intra and inter-sector clusters.

2.1.2 Time-evolving trees and graphs

Onnela et al. (2003) introduce the concept of *dynamic asset trees*: a time-evolving MST. The authors collect daily closing quotes of stocks in the S&P500 index from 1982 through 2000 and section these time series into several consecutive windows of size w, displaced by a small Δw. One asset tree t is created out of each window. Them, trees are compared according to their *normalized length* $L(t)$ and *single-step surviving ratio* σ_t, respectively the average of the sum of distances between nodes and the average of the intersection of sets of edges in consecutive windows. Results show that asset trees undergo a drastic topological reconfiguration during financial crisis, such as 1987's *Black Monday*, when huge drops in $L(t)$ and σ_t lead to shorter, centralized and fast-changing trees with several high-degree nodes.

In a later work, Onnela et al. (2004) introduces *asset graphs*, an evolution of asset trees obtained by relaxing the non-looping constraint of Kurskal's algorithm (Kruskal, 1956). Onnela et al. claim that asset trees can be unrepresentative of the real stock's correlations because their spanning nature spawns edges that are less relevant than they seem. Conversely, asset graphs are more robust since the weak edges introduced in trees are prone to break. Moreover, graphs allow disconnected components, thus being able to capture clique phenomena naturally occurring in stock markets.

Complex networks

Tse et al. (2010) pursue an interesting variation of asset graphs (Onnela et al., 2004). The authors notice that both MSTs and PMFGs suffer from information loss, as topological conditions over-fitting the reduction criteria might remove edges representing high correlation coefficients while keeping others of lower interest. Tse et al. propose full complex networks where edges are added if and only if they represent correlation coefficients ρ_{ij} larger than a threshold value θ; this method is dubbed the *winner-take-all* approach.

Tse et al. study the prices, returns and trading volumes of stocks traded in the S&P 500, NASDAQ and Dow Jones indices from mid 2005 through mid 2007 and from mid 2007 to mid 2009, to prove that all obtained networks exhibit a scale-free degree distribution. Authors conclude that quote variation is driven by a small number of stocks, especially those in the financial sector.

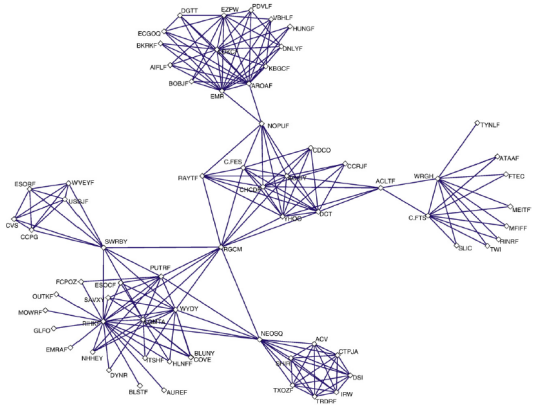


Figure 2.2: Partial network for US stock prices from July 1, 2005 through August 30, 2007.

Network stability and market cycles

Heiberger (2014) claims that topological properties of complex networks differ when stock markets go through calm or troubled times, extending the findings of Onnela et al. (2003). The author confirms evidences of the *May-Wigner* theorem (Hastings, 1982; Haldane and May, 2011) in complex stock networks. The theorem states that a system's stability is inversely proportional to the complexity of its network.

Heiberger mixes the methodologies of Onnela et al. (2003) and Tse et al. (2010) to build dynamic filtered networks out of stock in the S&P index from 2000 through 2012, thus covering the *dot-com bubble* and the *sub-prime crisis*. During calm times, the networks exhibit a clear and well-partitioned community structure, where clusters of companies from the same sector of activity often occur. Converselty, in troubled times networks become cohesive and centralized, with multiple communities quickly merging into massive clusters. Results prove that the May-Wigner theorem, devised for ecological networks, can be used as a predictor of turmoil in stock market networks.

2.2 Social Network Analysis

Social Networks Analysis (SNA) is an inter-disciplinary methodology started in the 1920s (Boccaletti et al., 2006). Social networks can be defined as the relationships, or *ties*, between social entities, or *actors*. Common tasks in SNA involve the identification of central actors, authorities, links, hubs and communities. Oliveira and Gama (2012b) and Adedoyin-Olowe et al. (2013) provide a good general overview on the essentials of SNA.

Social networks are represented as mathematical graphs. Therefore, concepts

originating from graph theory, such as centrality, density, distance, clustering, components, cliques, etc, are also present in SNA. This proximity leads to a growing application of SNA concepts in the field of econophysics.

2.2.1 Centralities

The concept of centrality in SNA is systematized in Freeman (1978). The author recognises centrality as an important structural attribute of social networks since it relates intimately to the way in which social groups are organized. Freeman covers three types of centrality: *degree*, *betweenness* and *closeness*.

The first and simplest centrality concept is *degree*. The degree of a node is the number of edges incident on it. Mathematically, the degree of node i it can be expressed as

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

where n is the number of nodes in the graph and a_{ij} is the entry of the adjacency matrix A corresponding to the ith row and jth column. In weighted networks, the *strength* of a node is the sum of the weights of edges incident on it. The degree of an actor is an indicator of its communication activity. Actors with high degree are focal points in information flow, given their direct contact with many other actors. Conversely, actors with low degree are regarded as peripheral.

The second centrality concept is *betweenness*, the frequency at which a node falls in the shortest path between other nodes. Betweenness is given by

$$b_k = \sum_{i,j \in V \setminus \{k\}} \frac{\sigma_{ij}(k)}{\sigma_{ij}} \tag{2.7}$$

where $\sigma_{ij}(k)$ is the number of shortest paths between nodes i and j passing through node k and $\sigma_{ij}(k)$ is the total number of shortest paths between nodes i and j. The betweenness of an actor is an indicator of its control over communication. Actors with high betweenness, also know as gatekeepers, act as interfaces between tightly-connected groups and therefore can exert control over them by withholding or distorting information. Gatekeepers are often regarded as coordinators of group processes. Betweenness can also be defined for edges.

The third and last centrality concept is *closeness*, the mean length of all shortest paths from one node to all other nodes in the network. Formally, closeness is given by

$$C_i = \frac{n-1}{\sum_{j \in V \setminus \{i\}} d_{ij}} \tag{2.8}$$

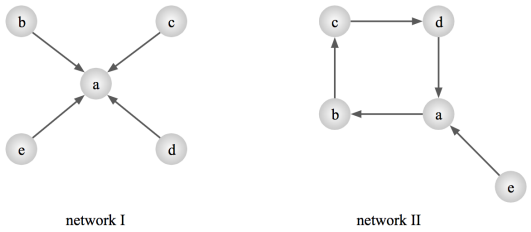


Figure 2.3: Hipothetical networks.

where d_{ij} is the length of the shortest path between node i and all nodes $j \neq i$. Closeness is a measure of efficiency, an indicator of how fast a given actor can reach every other actor in the network. It is also an indicator of independence, as actors with high closeness can reach other groups without the need of going through gatekeepers.

The quantity and quality of influence and status

Bonacich (1972, 1987) proposes *eigenvalue* centrality as a measure of influence and status for actors. The central idea is that the status of an actor is proportional to the weighted sum of the statuses of those connected to him. Let A be an adjacency matrix where a_{ij} is the contribute of actor i to actor j's status, let

$$\lambda x_i = a_{1i}x_1 + a_{2i}x_2 + \ldots + a_{ni}x_n \tag{2.9}$$

be a eigenvector of A and λ its largest associated eigenvalue. If A is a $n \times n$ matrix, the system has n solutions corresponding to the n values of λ. The eigenvalue centrality of actor i given by Eq. 2.9, commonly represented as

$$x_i = \frac{1}{\lambda} \sum_{j=1}^{n} a_{ij}x_j \tag{2.10}$$

Eigenvalue centrality is meaningless in some situations. If the status of an actor is a function of its neighbours statuses, then an actor with no neighbours has no status and thus contributes nothing to any other statuses as well. Figure 2.3 illustrates two cases from Bonacich and Lloyd (2001). All positions in network I have zero status, whereas in network II positions a, b, c, and d have the same status since e has no contribution to a's status.

Centrality measures for weighted networks

Opsahl et al. (2010) contributes with a generalization of centrality measures for weighted networks. Authors extend the notions of degree, betweenness and

closeness to weighted networks, trying to overcome the limitations of previous works based on edge weight alone.

Opsahl et al. revisit degree centrality as defined by Freeman (1978). The authors observe that degree can be generalized for weighted networks by replacing the binary adjacency matrix A by the weight matrix W. The strength of a node i can thus be defined as

$$s_i = \sum_{j=1}^{n} w_{ij} \tag{2.11}$$

where w_{ij} is the weight or strength of the edge between nodes i and j. The problem with strength alone is that it can be misleading in what regards to the node's involvement in the network; a node with a few strong connections can have a better index than another with more but weaker connections. Opsahl et al. propose a tuning parameter $\alpha > 0$ that determines the relative importance of connections over weights. The new definition of degree is

$$C_D^{w\alpha}(i) = k_i \times \left(\frac{s_i}{k_i}\right)^{\alpha} = k_i^{(1-\alpha)} \times s_i^{\alpha} \tag{2.12}$$

where k_i and s_i are respectively the degree (Eq. 2.6) and the strength (Eq. 2.11) of node i.

Opsahl et al. then turn on betweenness and closeness. Both measures rely on the notion of shortest paths, usually identified by Dijkstra (1959) algorithm. In unweighted networks, the binary shortest distance between nodes i and j is $d_{ij} = min(a_{ik} + \ldots + a_{kj})$ where k are intermediate nodes. In weighted networks, weights represent the strength of edges, so before applying Dijkstra's, the weights must first be inverted so that they are regarded as costs. That way, strong edges represent cheap connections and weak edges represent costly ones; in particular, an edge bearing a weight of zero will have an infinite cost. The shortest path is therefore defined as $d_{ij}^{w} = min(w_{ik}^{-1} + \ldots + w_{kj}^{-1})$ where k are intermediate nodes.

Once again, Opsahl et al. notice that this shortest path algorithm disregards the involvement of the node in the network. They again propose the adjustment of weights using a tuning parameter $\alpha > 0$ before applying Dijkstra's algorithm. By doing so, the shortest path between nodes i and j becomes

$$d_{ij}^{w\alpha} = min(\frac{1}{w_{ik}^{\alpha}}, \ldots, \frac{1}{w_{kj}^{\alpha}}) \tag{2.13}$$

The weighted version of shortest distance can be applied to the formula of closeness centrality, that becomes

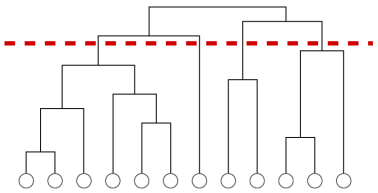


Figure 2.4: Hierarchical tree (dendrogram) with cut point determining the number of communities.

$$C_i^w = \frac{n-1}{\sum_{j \in V \setminus \{i\}} d_{ij}^{w\alpha}} \tag{2.14}$$

Likewise, the weighted shortest distance can be used to compute $\sigma_{ij}^{w\alpha}(k)$, the number of shortest paths between nodes i and j passing through node k, and $\sigma_{ij}^{w\alpha}$, the total number of of shortest paths between nodes i and j. Betweenness centrality is thus rewritten as

$$b_k^w = \sum_{i,j \in V \setminus \{k\}} \frac{\sigma_{ij}^{w\alpha}(k)}{\sigma_{ij}^{w\alpha}} \tag{2.15}$$

2.2.2 Communities

Social networks tend to show community structure. Communities, modules or clusters, are groups of similar nodes. A better definition is obtained resorting to density: communities are densely connected groups of nodes with sparse connections between them.

Hierarchical Clustering and *Graph Partitioning* are two main lines of research in community discovery, both seeking to determine groups of related nodes based on the information provided by the networks's topology (Oliveira and Gama, 2012b). Hierarchical clustering methods use measures of similarity between nodes (e.g. cosine similarity, Jaccard Index, Euclidean distance, etc.) and clusters (e.g. single linkage, complete linkage, Ward's method, etc.) to uncover the nested structure of networks. Graph Partitioning progressively divides graphs into sets of disjoint subgraphs, identifying and removing high betweenness edges (bridges) to isolate communities. Both classes of methods produce hierarchical structures, called dendograms, from which communities are produced by horizontal cut according to some criteria.

Finding communities in networks

Newman and Girvan (2004) use divisive algorithms to discover community struc-

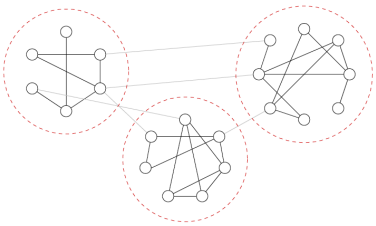


Figure 2.5: Graph with three communities, exhibiting dense internal links and sparse external links.

tures in networks. The authors notice higher betweenness in edges lying between communities and lower in intra-community edges. The proposed algorithms computes edge betweenness for all edges in the network, removes the edge with higher betweenness and then repeats the process for all remaining edges. Without the recalculation step, edge betweenness would be outdated after the first removal, thus rendering an inaccurate final result.

Newman and Girvan use *modularity* to assess the quality of the partitions (Newman, 2003). Given a particular partition of a network into k communities, authors define a matrix $E_{k\times k}$ where each entry e_{ij} is the fraction of edges that connect nodes in community i to nodes in community j. Modularity can then be defined as

$$Q = \sum_i \left(e_{ii} - a_i^2\right) = TrE - ||E^2|| \tag{2.16}$$

where $TrE = \sum_i e_{ii}$ is the trace of matrix E, $a_i = \sum_j a_{ij}$ is the row sum of the fractions of edges incident to nodes in community i and $||E^2||$ is the sum of all elements in matrix E^2.

Modularity Q is the difference between the number of edges connecting intra-community nodes and the expected number of edges connecting intra-community nodes in a network where intra-community edges are built at random. Q takes its values in $[-1, 1]$. A value of $Q = 0$ show that the found communities are no better than random connections. A positive value reveals community structure; the higher the value, the better the partition. Newman and Girvan state that empiric evidence shows that any value between 0.3 and 0.7 is a good indicator of meaningful communities; this observation is later confirmed by Clauset et al. (2004).

Community detection based in heuristics

Blondel et al. (2008) introduce the *Louvain method for community detection*, a two-phase heuristic approach based on modularity optimization. Blondel et al. define modularity as

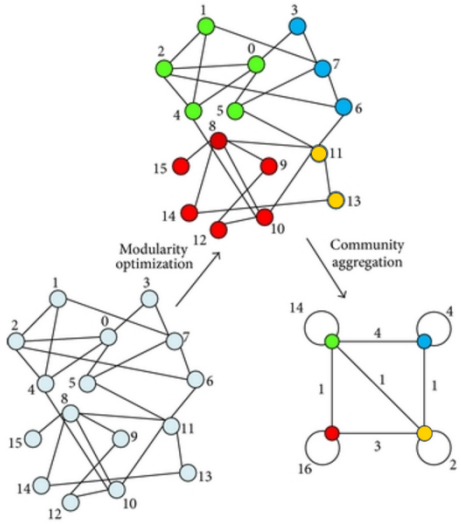


Figure 2.6: Louvain algorithm in process.

$$Q = \frac{1}{2m} \sum_{ij} \left[A_{ij} - \frac{k_i k_j}{2m} \right] \delta(c_i, c_j) \tag{2.17}$$

where m is the number of edges, k_i and k_j are respectively the degree of nodes i and j, A_{ij} is the cell of the adjacency matrix holding the number of edges between i and j, $\frac{k_i k_j}{2m}$ is the expected number of edges falling between i and j, c_i and c_j denote the groups to which nodes i and j belong and $\delta(c_i, c_j)$ is the Kronecker delta.

The algorithm's two phases, illustrated in Figure 2.6, are iteratively repeated. The first phase starts by creating communities of one node. Every node i is then removed from its community and added to the community of each neighbour j. Every time i changes community, modularity is recalculated. Once every community is visited, i is placed into the community that presents the largest modularity gain if one is found, otherwise i returns to its original community. The process is repeated for all nodes until no more gains in modularity are detected; by then, a local maximum is determined and phase one ends.

In the second phase, all nodes in a community are aggregated into a single node; the resulting network is therefore comprised of nodes representing the communities found in phase one. Edges connecting new nodes are weighted according to the sum of the weights of edges connecting the old communities (pre-aggregation); edges within the same old community are represented as self loops. Phase two ends once the new network is completely determined. The algorithm resumes phase one and the process goes on iteratively until no more changes in modularity are found.

Blondel et al. (2008) point out the advantages of the Louvain algorithm: it is

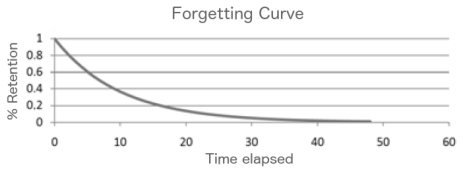


Figure 2.7: Ebbinghaus' forgetting curve.

easy to implement and very efficient, its outcome is unsupervised, and intermediate results are meaningful, even at small scale. The authors apply the algorithm to networks of assorted sizes and compare results with those obtained by other community detection algorithms by Clauset et al. (2004), Pons and Latapy (2005) and Wakita and Tsurumi (2007). Louvain outperforms all other methods, even in networks of unprecedented sizes.

2.2.3 Retention and Stability

Kudelka et al. (2010) propose the reduction of networks based on node and edge stability. The authors define two coefficients, *retention* and *stability*, that describe the relevance of nodes and edges in the social network based on their long-term behaviour. The network is reduced to smaller components by removal of irrelevant nodes and edges.

Kudelka et al. use the *forgetting curve* (Ebbinghaus, 1913), which defines the probability of one remembering information after t time has passed since a previous recall. The forgetting curve is given by

$$R = e^{-\frac{t}{S}} \tag{2.18}$$

where e is the Euler number, t is the time elapsed since last recall and S is an estimate of the time required to store information in memory after last recall. S changes over time. The time required to store new information is a constant $S_{ini} > 0$. For every recall after a time $t > 0$, the value of S_{new} can be computed by

$$S_{new} = ch(t, S, F, S_{ini}) \times S \tag{2.19}$$

where $F > 1$ is a multiplicative factor of S and ch is a function that determines the speed of the forgetting process.

Kudelka et al. suggest that edges connecting pairs of nodes can be regarded as experiences stored in memory. Stable networks result from connections that are frequently recalled. The authors identify three characteristics of nodes and edges that change over time depending on the frequency of interactions:

Node Retention (NR): probability of recalling an edge between this node and a previously incident node at time t;

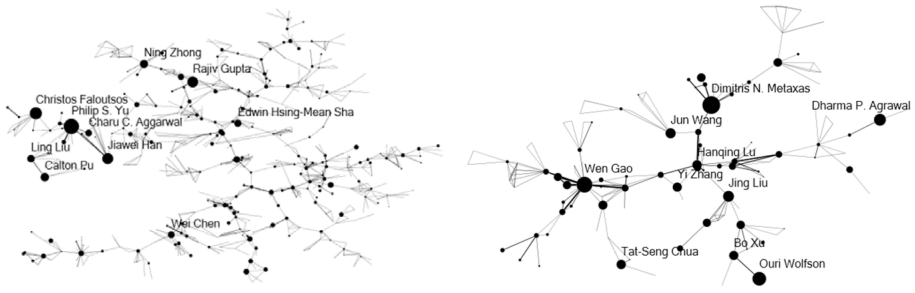


Figure 2.8: Contributions network before and after reduction.

Node Stability (NS): estimated time for which the node remains active;

Active Node: node for which $NS > 0$ in time t;

Edge Retention (ER): probability of recalling an edge between two previously incident nodes at time t;

Edge Stability (ES): estimated time for which the edge remains active;

Active Edge: edge for which $ES > 0$ in time t;

Kudelka et al. build networks for a dataset comprising 2 million co-authorship relationships between 443 thousand authors over a period of 45 years. Data is divided in non-overlapping monthly windows. Within each window span, an edge is added for every article published by an author and a co-author. The forgetting curve is used to calculate retention and stability monthly; edge stability is used as weight. Results show that stability changes every month and that retention quickly decreases due to the forgetting curve. Not surprisingly, authors with many publications in common are represented by nodes and edges with steady high retention and increasing stability over time. At the end, only 28% of the nodes and 12% of the edges are active. Figure 2.8 shows the networks before and after reduction. The final networks is comprised of small backbones of authors with high stability. Small cycles and cliques are also detected.

2.2.4 Tracking the evolution of networks

Complex networks modelling real-worlds phenomena are organized in community structures that evolve through time. While early research represented networks as static mathematical objects that seldom captured the evolving nature of the underlying concept, recent works acknowledge the importance of time in the shaping of network topologies (Rossetti and Cazabet, 2018).

Figure 2.9 illustrates four concepts that gradually introduce the temporal dimension into network models. A static network captures a single snapshot of reality. Edge weights can be used to measure strength over well-defined time-windows.

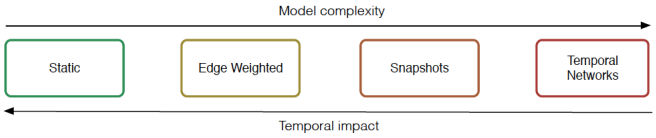


Figure 2.9: From static graph to temporal networks: complexity increasing with temporal information.

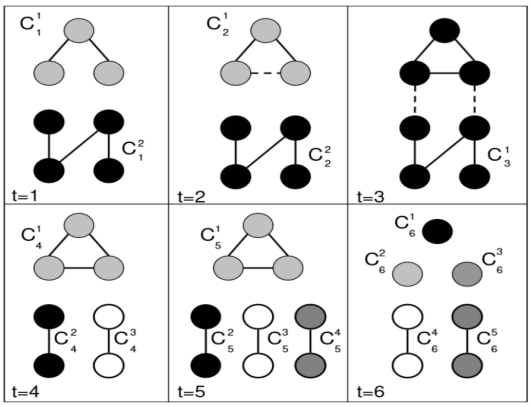


Figure 2.10: Temporal snapshots at time $t = 1$ to 6.

Snapshots are temporal ordered series of networks taken in (regular) intervals that can be used to keep track of changes in networks topologies. Temporal networks allow a fine-grain description of network dynamics, adding explicit birth and death timestamps to every node and edge. Complexity increases as one moves from static to temporal networks. One must then consider the trade-off between richness of information versus the performance in the storing/processing of data (Rossetti and Cazabet, 2018).

Characterizing the evolutionary behavior of interaction graphs

Asur et al. (2009) propose a snapshot-based framework to study the evolution of networks. The authors monitor transformations in communities through critical events detected in transitions between consecutive snapshots. These events characterize the behavioural patterns of both individuals and communities over time.

Let S_t and S_{t+1} be snapshots of network S in two consecutive times intervals t and $t+1$, C_t^n be the n-th cluster in snapshot S_t and V_t^n the set of nodes in cluster C_t^n. Asur et al. propose the following events:

Form A cluster C_{t+1}^n is *formed* if no two nodes in V_{t+1}^n existed in a same cluster C_t^n at time t. In Figure 2.10, snapshot $t = 5$ shows a new cluster C_5^3 forming.

$$Form(C_{t+1}^n) = 1 \iff \nexists C_t^n : V_t^n \cap V_{t+1}^n > 1 \tag{2.20}$$

Continue A cluster C_{t+1}^n is the *continuation* of C_t^n if V_{t+1}^n equals V_t^n. Figure 2.10 shows a continue event in snapshot $t = 2$. Despite the new interaction between nodes in cluster C_1^2, the clusters do not change.

$$Continue(C_t^n, C_{t+1}^n) = 1 \iff V_t^n = V_{t+1}^k \tag{2.21}$$

k-Merge Two distinct clusters C_t^i and C_t^j *merge* if there is a cluster C_{t+1}^n at time $t+1$ containing at least $k\%$ of the nodes belonging to C_t^i and C_t^j. Figure 2.10 shows a merge event in snapshot $t = 3$. The dashed lines are newly created edges, joining all nodes under a single cluster C_3^1.

$$\begin{aligned} Merge(C_t^i, C_t^j, k) = 1 \iff \nexists C_{t+1}^n : \frac{|(V_t^i \cup V_t^j) \cap V_{t+1}^k|}{Max(|V_t^i \cup V_t^j|, |V_{t+1}^n|)} > k\% \wedge \\ |V_t^i \cup V_{t+1}^n| > 0.5|C_t^i| \ \wedge \ |V_t^j \cup V_{t+1}^n| > 0.5|C_t^j| \end{aligned} \tag{2.22}$$

The condition holds if there are edges between V_t^i and V_t^j at time $t+1$.

k-Split A cluster C_t^n *splits* if $k\%$ of its nodes are in two different clusters at time $t+1$. Figure 2.10 shows a split event in snapshot $t = 4$, when cluster C_3^2 splits into two smaller clusters C_4^2 and C_4^3.

$$\begin{aligned} Split(C_t^n, k) = 1 \iff \exists C_{t+1}^i, C_{t+1}^j : \frac{|(V_{t+1}^i \cup V_{t+1}^j) \cap V_t^n|}{Max(|V_{t+1}^i \cup V_{t+1}^j|, |V_t^n|)} > k\% \wedge \\ V_{t+1}^i \cup V_t^n| > 0.5|C_{t+1}^i| \ \wedge \ |V_{t+1}^j \cup V_t^n| > 0.5|C_{t+1}^j| \end{aligned} \tag{2.23}$$

Dissolve A cluster C_t^n *dissolves* if no two nodes in the cluster are in a same cluster at time $t+1$. In Figure 2.8, snapshot $t = 6$ shows a dissolve event when the edges in cluster C_5^1 disappear, spawning three new clusters C_6^1, C_6^2 and C_6^3.

$$Dissolve(C_t^n, C_{t+1}^n) = 1 \iff \nexists C_{t+1}^n : V_t^n \cap V_{t+1}^n > 1 \tag{2.24}$$

Experimental results over two large datasets show that the value of k has great influence on the number of merges and splits: the number of events drops as the value of k raises. Asur et al. claim that a high value for k helps to capture interesting merge and split events in highly overlapping clusters.

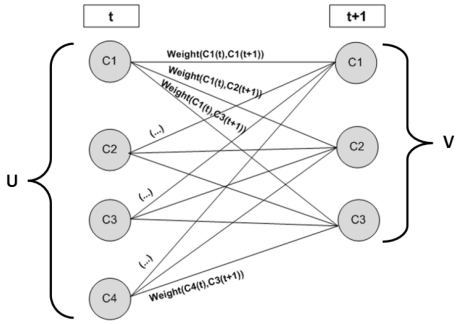


Figure 2.11: MEC bipartite graph: nodes as clusters and edges' weights as conditional probabilities.

Monitor of the Evolution of Clusters

Oliveira and Gama (2012a) develop a framework to *Monitor of the Evolution of Clusters* (MEC). The process consists on the detection of transitions undergone by mutually exclusive clusters observed in consecutive snapshots of a network taken in regular time intervals Δt.

For any given time interval $[t_i, t_{i+\Delta t}]$, Oliveira and Gama build a bipartite graph $G(U, V, E)$, where U is the set of clusters existing at time t_i, V is the set of clusters existing at time $t_{i+\Delta t}$ and E is the set of weighted edges connecting all possible pairs of clusters in U and V. The weight function is defined as

$$\begin{aligned} weight(C_u(t_i), C_v(t_{i+\Delta t})) &= P(X \in C_v(t_{i+\Delta t}) | X \in C_u(t_i)) \\ &= \frac{\sum P(x \in C_u(t_i) \cap C_v(t_{i+\Delta t}))}{\sum P(x \in C_u(t_i))} \end{aligned} \tag{2.25}$$

where X is the set of observations found in cluster $C_u(t_i)$ and $P(X \in C_v(t_{i+\Delta t}) | X \in C_u(t_i))$ is the conditional probability of X belonging to cluster $C_v(t_{i+\Delta t})$ at time $t+1$ given that X belongs to $C_u(t_i)$ at time t.

Oliveira and Gama use two thresholds to reduce the fully-connected graph. The *survival* (τ) threshold is used to remove edges whose weights indicate meaningless matching probabilities. The *split* (λ) threshold is used to detect changes in clusters. The authors consider five types of transitions in the reduced graph:

Birth A cluster C_v is born when all edges connecting it to any previous clusters C_u have weights below the survival threshold τ

$$0 < weight(C_u(t_i), C_v(t_{i+\Delta t})) < \tau, \ \forall u \tag{2.26}$$

Survival A cluster C_u survives if there is a single edge connecting it to a cluster C_v in the following snapshot and if that edge's weight is above the survival threshold τ

$$\begin{gathered} weight(C_u(t_i), C_v(t_{i+\Delta t})) \geq \tau \wedge \\ \nexists\ C_w \neq C_u : weight(C_w(t_i), C_v(t_{i+\Delta t})) \geq \tau \end{gathered} \tag{2.27}$$

Merge Two (or more) distinct clusters C_u and C_w merge if there are edges connecting them to a cluster C_v in the following snapshot and if those edges' weights are above the survival threshold τ

$$\begin{gathered} weight(C_u(t_i), C_v(t_{i+\Delta t})) \geq \tau \wedge \\ \exists\ C_w \neq C_u : weight(C_w(t_i), C_v(t_{i+\Delta t})) \geq \tau \end{gathered} \tag{2.28}$$

Split A cluster C_u is split if there are edges connecting it to clusters C_v and C_w in the following snapshot, if those edges' weights above the split threshold λ and if the sum of those weights is above the survival threshold τ

$$\begin{gathered} \exists\ C_v \neq C_w : weight(C_u(t_i), C_v(t_{i+\Delta t})) \geq \lambda \wedge \\ weight(C_u(t_i), C_w(t_{i+\Delta t})) \geq \lambda \wedge \\ \sum_{k=1}^{n} weight(C_u(t_i), C_k(t_{i+\Delta t})) \geq \tau \end{gathered} \tag{2.29}$$

Death A cluster C_u dies when all edges connecting it to clusters C_v in the following snapshot have weights below the split threshold λ

$$weight(C_u(t_i), C_v(t_{i+\Delta t})) < \lambda,\ \forall v \tag{2.30}$$

Oliveira and Gama conduct two independent sets of experiments over three large datasets, varying one threshold and keeping the other constant in each turn. The first set of experiences shows that high values of τ lead to few survivals and many splits, whereas births ans deaths remain constant. The second set shows that high values of λ lead to few splits and many births and deaths, and that λ as no effect on the number of survivals and merges. The authors conclude that the tuning of the split threshold λ is not critical since its impact is low for a significant number of transitions. Conversely, the tuning of the survival threshold τ is very important since transitions are more sensitive to this parameter and even small variations of this value may lead to very different results.

2.2.5 Centralities and communities in stock markets

Roy and Sarkar (2011) study the evolution of global stock markets before and after the 2008 financial crisis. The authors focus on the 2006-2010 period, seeking to understand the effect of Lehman Brothers' collapse in stock markets.

Roy and Sarkar follow the methodology of Onnela et al. (2004) to track the evolution of a network comprised of 93 global-wide indices. Data is split in overlapping windows displaced every 4 week. Each window originates one MST and one correlation network, for which *degree*, *betweenness*, *closeness* and *eigenvector* centralities are determined. Additionally, authors determine the tree length of every MST. The metrics are used to build normalized ranks for each of the indices; the ordering assigns the smallest rank to the most influential index. A final normalized rank is build as an average of the normalized ranks.

Empiric results show that the variance in ranks increases by 62% during the peak of 2008 crisis, pointing to turbulence in markets. North America indices in particular drop a few positions, while pan-European indices begin to rise. Changes are more prominent in indices with higher ranks (less influence). Moreover, there is a topological reconfiguration of MSTs and networks. The collapse of Lehman Brothers is captured by a clear decrease in the length of MSTs, thus showing a significant increase in correlation between indices around the globe. Pre-crisis communities, heavily influenced by regional and trading interactions, give place to new ones where smaller indices (South America, South-east Asia) move away from US and European indices into more economically stable communities featuring the UK and Japan.

In a different work, Dimitrios and Vasileios (2015) use SNA techniques to study the most influential stock and stock communities in the Greek stock exchange during the years 2007 and 2012.

The authors collect daily closing quotes to derive daily returns, used to obtain the cross-correlation matrices from which they build unweighted threshold-filtered networks according to methodology of Huang et al. (2009). Several networks are built using different threshold values, both positive and negative. Dimitrios and Vasileios conduct the analysis of *degree*,*closeness*, *betweenness* and *eigenvector* centralities and *local clustering coefficients* for every produced network[2].

Dimitrios and Vasileios find that small positive threshold values produce highly dense networks, thus yielding results of little use. As the threshold increases, many communities begin to emerge, reflecting a healthy state of the market. Conversely, cluster tend to merge as the network becomes highly connected during the peak of the *sovereign debt* crisis of 2012; in particular, the banking sector forms a tightly connected community, despite the small fluctuations in return values. This herd behaviour is considered a sign of financial turmoil (Hastings, 1982; Onnela et al., 2003; Heiberger, 2014). Authors also find a very small number of negative correlations; the percentage of negative correlated stock is 0.3% of the positive ones.

[2]Gephi (Bastian et al., 2009) is used for network representation and analysis.

2.2.6 The role of SNA in portfolio management

Portfolio management is an important topic since Markowitz (1952) proposed diversification as a tool to mitigate risk and optimize return. Several approached to the topic have been put forth, including Fama (1965), Ross (1976), Rosenow et al. (2002), Tola et al. (2008), Preis et al. (2012), Roll (2013) among others. In what follows, we focus on approaches based on SNA.

Networks of experts

Koochakzadeh et al. (2012) propose the use of social networks of investors based on the similarity of their publicly available portfolios. The authors find that investors may be fully characterized by the *expected return* and *risk level* of their portfolios.

Koochakzadeh et al. collect the investment portfolios of 125 experts and 57 amateurs, and the daily quotes from 2010 through 2011 for the stocks in those portfolios. The authors compute the semestral expected return and risk for each stock, using those values to compute the semestral expected return and risk of each portfolio. To measure portfolio performance, authors use *Sharpe ratio*, a measure of the amount of return added per unit of risk. Each expert is placed in one of five risk categories, according to the performance of his/her portfolio. The experts portfolio data set is used to train a classifier, which in turn is used to predict the risk level of the amateurs investors.

Koochakzadeh et al. use the *K-means* algorithm to determine the similarity between every pair of portfolios, as a function of the number of common clusters. The authors build a social network of experts where similarity values are used as weights for edges. Amateur investors are then inserted into communities of experts exhibiting similar propensity for risk. The expert with highest *degree* centrality in each community is chosen as its representative and his/her portfolio is recommended to amateurs in that community. Results show that in all cases, the mean Sharpe ratio of amateur investors is lower than that of experts belonging to the same class, meaning that amateurs get higher return for the same risk.

Evolution of communities in stock networks

Lima (2015) addresses portfolio management base on the evolution of communities of stocks traded in the *Russel 1000* index during 2014.

Lima (2015) starts by computing the time series of daily prices variations as

$$Var_i(t) = \frac{P_i(t) - P_i(t-1)}{P_i(t-1)} \tag{2.31}$$

where $P_i(d)$ and $P_i(d-1)$ are the closing prices of stock i at days d and $d-1$ respectively. Lima divides each time series in sliding windows $w_k,\ k = 1, \ldots, n$ of size t displaced by a small Δt and creates one correlation matrix $\rho_{ij,t}$ for each

window w_k, from which the author creates one unweighted, acyclical, threshold-filtered graph according to the metodology of Huang et al. (2009). From the graph, the author extracts[3] the *eigenvector* centrality of each node, the *average degree* of the network, and the communities as detected by the *Louvain*'s algorithm (Blondel et al., 2008).

Lima deems a stock influential if its *eingenvector* centrality is above a threshold ϵ at least in one network. Likewise, he considers networks to be eligible for study if its size is above a threshold ω. The author uses the MEC framework (Oliveira and Gama, 2012a) to monitor the survival of communities, setting the survival threshold τ according to the recommendations of Oliveira and Gama (2012a).

Lima conducts several experiments over the dataset, setting different sizes for sliding windows and both positive and negative threshold values for correlation. Results show that positive and negative correlations do not differ significantly in terms of results. Small windows prompt short-lived communities and transient influence in terms of stock. Conversely, longer windows yield long lasting communities and a stable behaviour. For comparative purposes, the author repeats the study using the ultrametric distance proposed by Mantegna (1999). Lima observes a decrease in network density and an increase on the number of detected communities, which last longer; influential stock also maintain their status for longer.

2.3 Streaming

Data streams are stochastic processes in which events occur continuously and independently of each other (Gama, 2010). Unlike data sets, streams are unbounded in length, its items arrive in any order and are discarded or archived as soon as they are processed. Algorithms dealing with streams must therefore be prepared to deal with limitations in time and space. Common approaches involve the adoption of synopses, summaries and sliding windows over past data, and sampling as a way to reduce the flow rate of input streams (Babcock et al., 2002; Gama, 2010).

Babcock et al. (2002) peruses a number of issues regarding data management, query processing and algorithmic problems that arise from dealing with continuous data streams. Gama (2010) presents a comprehensive approach to several aspects and applications of knowledge discovery from data streams.

2.3.1 Statistics over streams

Often, one has to compute statistics over streamed data. Since data arrives continuously, the calculation of statistics must be incremental. Arguably, the best way to achieve results is through *recursive* formulas. Given a stream x_i, the sample mean recursively obtained as

[3]Using Gephi (Bastian et al., 2009)

$$\bar{x}_i = \frac{(i-1)\bar{x}_{i-1} + x_i}{i} \tag{2.32}$$

for which one needs to keep in memory the number i of observations and the sum of all observed values $\sum x_i$. A similar formula can be deduced for the standard deviation

$$\sigma_i = \sqrt{\frac{\sum x_i^2 - \frac{(\sum x_i)^2}{i}}{i-1}} \tag{2.33}$$

for which one keeps the number i of observations, the sum of all observed values $\sum x_i$ and the sum of all squared observed values $\sum x_i^2$. Multiple stream statistics can be obtained as well. Given two streams x_i and y_i, the recursive formula for correlation is

$$\rho(X,Y) = \frac{i\sum(x_i \cdot y_i) - \sum x_i \sum y_i}{\sqrt{i\sum x_i^2 - (\sum x_i)^2}\sqrt{i\sum y_i^2 - (\sum y_i)^2}} \tag{2.34}$$

for which one keeps the number i of observations, the sum of the values for each stream $\sum x_i$ and $\sum y_i$, the sum of the squared values $\sum x_i^2$ and $\sum y_i^2$ and the sum of cross product $\sum(x_i \cdot y_i)$.

2.3.2 Statistics over sliding windows

Datar et al. (2002) address the problem of maintaining aggregates and statistics over data streams, seeking a way to conduct the analysis in one scan, within memory bounds and without resource to precomputed summaries of data. The authors notice that, for most applications, decision making is continuous and based on statistics gathered over recent events. This motivates the concept of *sliding window*: a subset $x_{i-w}, \ldots, x_i$ of the observed values, where i is the current element and w is the *fixed* size of the window. A sliding windows is a *first-in, first-out* data structure from which element x_{i-w} is removed (and forgotten) as the next observed element $x_j, j = i+1$ is inserted.

Datar et al. propose *exponential histograms* as a means to hold statistics, histograms and hash tables in memory. Exponential histograms keep arriving data in buckets, each of them tagged with the timestamp of arrival of its most recent item. Two auxiliary variables $LAST$ and $TOTAL$ are kept to record respectively the size of the last bucket and the total size of the buckets. The authors prove that the size of these histograms grows exponentially to 2^w, where w is the size of the sliding window.

Datar et al. illustrate the concept by consuming a binary stream and counting the number of 1's observed in the last N elements. The algorithm starts by checking

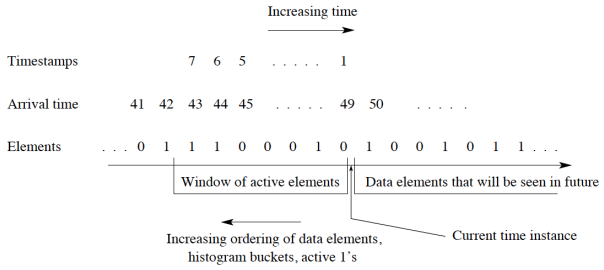


Figure 2.12: Data stream, sliding window and timestamps.

whether the last bucket's time stamp falls outside the range of the sliding window. If so, the bucket is discarded and $LAST$ and $TOTAL$ are updated accordingly. Then, the value of the arriving item is observed. Every 0 is immediately discarded. Conversely, every 1 is associated with a time stamp and inserted into a new bucket of size one, while $TOTAL$ is incremented. The list of buckets is then traversed in increasing order of sizes. Given a parameter ϵ, if there are $\lceil 1/\epsilon \rceil/2 + 2$ buckets of the same size, the oldest two of them are merged into a new bucket with the double of size, updating $LAST$ if the last bucket is the result of the merger. The process iterates over all buckets, cascading into a series of mergers. It is important to notice that the last bucket may contain 1's beyond the size of the sliding window. To avoid a miscount, Datar et al. establish that the last bucket holds an estimated number of $LAST/2$ active items. To compute the number of 1's in the sliding window, one computes the value of $TOTAL - LAST/2$.

The detail level in data becomes coarser as buckets grow old. As buckets are merged, time gets compressed by a factor equal to the growth rate and older information loses importance until it is discarded. Authors suggest ways to apply the same technique to other problems with a multiplicative overhead of $O(\frac{1}{\epsilon}log(w))$ and at the cost in accuracy with a $1 + \epsilon$ factor.

Statistics over variable-sized sliding windows

Bifet and Gavalda (2006, 2007) introduces ADWin (for ADaptive Window) as a variable-size sliding window algorithm capable of keeping updates statistics over streams of numeric data while acting as a change detector.

ADWin takes two inputs: a confidence level $\delta \in [0, 1]$ and an infinite stream $x_1, \ldots, x_t, \ldots$ of independent values generated according to an unknown distribution D_t with unknown expected value μ_t. One sliding window W of size n is used to keep the most recently observed values x_i. Let W_0 and W_1 be a partition of W, n_0 and n_1 their respective sizes, $\hat{\mu}_{w_0}$ and $\hat{\mu}_{w_1}$ their known averages and μ_{w_0} and μ_{w_1} their unknown expected values.

ADWin processes the incoming stream one item a time, adding it to the head

of the sliding window (W_1). The algorithm then looks for partitions $W = W_0.W_1$ where $|\hat{\mu}_{W_0} - \hat{\mu}_{W_1}| \geq \epsilon_{cut}$. If found, W_0 is dropped and the process resumes with a new window $W = W_1$. The cutting threshold is given by

$$\epsilon_{cut} = \sqrt{\frac{1}{2m}.ln\frac{4n}{\delta}} \tag{2.35}$$

where m is the harmonic mean of n_0 and n_1. Bifet and Gavalda claim that, in every step, the probability that ADWin shrinks W is at most δ when μ_t remains constant. Conversely, when μ_t diverges, the probability that W is cut down to $W1$ or shorter is $1 - \delta$.

Bifet and Gavalda (2007) introduce an improved version of ADWin that uses a variation of *exponential histograms* (Datar et al., 2002). Results prove that the difference in approximation power between the two versions is negligible. Tests over synthetic and real data streams prove that ADWin can be successfully used as an *estimator* for statistics since older parts of the window are dropped when their average diverges from that of the most recently ones. Additionally, it is also a good *change detector* because windows shrink if and only if there are significant changes in data.

Non-parametrical statistics over sliding windows

The Pearson correlation coefficient is not robust in the presence of outliers, missing points and heavy-tailed distributions, which makes it unfit for some streaming applications. Xiao (2017) proposes an online algorithm to compute Spearman's Rank (ρ_S) and Kendall's Tau (τ) non-parametric correlation coefficients.

Both ρ_S and τ require the sorting (ranking) of two series X and Y, a computationally heavy task. The recursive approach used for Pearson's cannot be used for the two non-parametric correlations; in ρ_S, past observations may have their ranks changed as new data arrives, and τ computes the correlation by comparison of new and past data. Xiao's algorithm deals with these limitations while keeping both time and memory cost complexities constant. The algorithm can work with both fixed-sized and variable-sized sliding windows.

Let (X, Y) be a pair of time series, $\{c_k^X \mid k = 0, \ldots, p\}$ and $\{c_l^Y \mid l = 0, \ldots, q\}$ sets of cut-points in increasing order of X and Y and $n_{gap} \geq 1$ a positive integer. Xiao uses the cut-point sets to build a matrix $M_{p \times q}$ where each cell $M[k, l]$ stores the frequency, up until time t, of sample values (x_t, y_t) falling into the range $[c_{k-1}^X, c_k^X[\times [c_{l-1}^Y, c_l^Y[$. When $(t \ mod \ n_{gap})$ is equal to zero, the non-parametric correlation coefficient of choice is computed and stored in the return array r. At any given time t, the the value so far of coefficients can be computed by observing the values in $(r[n_{gap}], r[2n_{gap}], \ldots)$.

Xiao tests his online algorithm on streams of data generated over an year by 4 sensors of an industrial plant. Several settings are defined for the number of

cut-points, gap and sliding windows sizes. Based on the experimental results, Xiao proposes different values of p, q for ρ_S and τ. For comparison purposes, a batched version is tested against the same streams and results show that the online version is always faster.

Other interesting approaches

We kindly refer the reader to two other interesting works in the field of statistics over sliding windows: *StatStream* (Zhu and Shasha, 2002) and *Local Correlation* (LoCo) score (Papadimitriou et al., 2006). The first is a parallel computing system based on *Discrete Fourier Transforms* capable of computing single and multiple stream statistics in one pass, constant time and bounded memory. The second is a sound alternative to Pearson correlation that tracks the evolution of local correlations among series using a joint model of the series without any assumption about stationarity.

2.3.3 Statistics with gradual forgetting

In most real world applications, such as biomedicine, industrial processes, stock markets and fault detection and diagnosis, data flows continuously according to non-stationary distributions for which the underlying concept changes over time (Gama, 2010). As the characteristics of data change, old data items become less significant than new ones for future behaviour prediction or resource allocation.

Linear gradual forgetting

Koychev (2000) proposes gradual forgetting, implemented by time-based weight functions $w = f(t),\ t \geq 0$, as a means to deal with concept drift in data streams. In a practical example, Koychev defines a linear forgetting function that takes item i's elapsed time since arrival, the total number of items n and a slope adjustment factor $k \in [0, 1]$ as parameters and returns a weight w_i. The function is subject to the following constraints: $w_i > 0$ and $(1/n) \sum_{i=1}^{n} w_i = 1$.

Koychev test his function in conjunction with ID3 (Quinlan, 1986) and Naïve Bayes Classifier (NBC) algorithms in two sets of experiments involving the dataset used to test STAGGER (Schlimmer and Granger, 1986). In the first set, the author uses a landmark window with the classifiers, in the second Koychev uses fixed-size non-overlapping sliding windows. Results show that gradual forgetting always increases the average prediction accuracy of both classifiers, especially when used together with sliding windows.

Koychev's findings prove that gradual forgetting can be used alone or together with other forgetting mechanisms for partial memory learning, improving the predictive accuracy of learning algorithms. The author suggest the use of logarithmic

and exponential functions for gradual forgetting.

Other gradual forgetting functions

Cohen and Strauss (2003) address the problem of maintaining time-decaying aggregates and statistics over data streams. The authors define storage-efficient algorithms that use non-increasing decay functions to transform data so that the relative contribution of each item to the aggregate is scaled down by a factor depending on elapsed time.

The authors start by presenting the *Decayed Count Problem* (DCP). Given a binary data stream $f(t) \geq 0$ and a non-increasing decay function $g(x) \geq 0$ defined for all $x \geq 0$, the decayed sum is

$$V_g(T) = \sum_{t<T} f(t)g(T-t) \tag{2.36}$$

where T is the current time and t is an integer. Given a acceptable error threshold $\epsilon > 0$, the goal is to produce an approximate estimate $V_g^{'}(T)$ such that

$$\frac{V_g^{'}(T) - V_g(T)}{V_g^{'}(T)} \leq \epsilon \tag{2.37}$$

Cohen and Strauss present several families of decay functions. One of such families is exponential decay, commonly used due to its simplicity. Given $\lambda > 0$, the function is defined as $g(x) = exp(-\lambda x)$. Given any current time T, the exponentially decayed count can be obtained by

$$V_{\text{exp}}(T) = f(T) + exp(-\lambda)V_{\text{exp}}(T-1) \tag{2.38}$$

The authors also study the sliding windows family. Given a window size W, the decay function is defined as $g(x) = 1$ for $x \leq W$ and $g(x) = 0$ otherwise. The authors focus on exponential histograms introduced by Datar et al. (2002) as a method to produce approximate estimates of sliding windows' decayed counts up to an acceptable error of $1 + \epsilon$. At any given current time T, the estimate for the decayed count can be obtained as

$$V_{\text{slidwin}}(T) = \sum_i C_i = \sum_i \sum_{w_{i-1}<t\leq w_i} f(t) \tag{2.39}$$

Cohen and Strauss prove that the decayed count problem using any decay function can be estimated using exponential histograms over window of size W. Using summation by parts, the decayed count problem is rewritten as

$$
\begin{aligned}
V_g(T) &= \sum_{T-W \leq t < T} f(t)g(T-t) \\
&= g(W) \sum_{T-W \leq t < T} f(t) \\
&+ \sum_{i=1}^{W-1} (g(W-i) - g(W+i-1)) \sum_{T-W+i \leq t < T} f(t) \\
&= g(W) V_{\text{slidwin}_{\text{W}}}(T) \\
&+ \sum_{i=1}^{W-1} (g(W-i) - g(W+i-1)) V_{\text{slidwin}_{\text{W-i}}}(T)
\end{aligned}
\tag{2.40}
$$

The approximate estimate for decayed count using exponential histograms can then be obtained by

$$
V_{\text{slidwin}}(T) = g(T-w_0)C_0 + \sum_{i \leq 1} (g(T-w_i) - g(T-w_{i-1}))C_i \tag{2.41}
$$

This result shows that sliding windows decay is the "hardest" decay function in terms of data storage, as other decay functions can be used to produce estimates equivalent to those produced by exponential histograms. Cohen and Strauss argue that exponential and sliding windows decays do not provide sufficient flexibility when it comes to fine tune the rate of decay and propose efficient algorithms for the *polynomial*, *polyexponential* and *polygonal* decay families.

2.4 Summary

In this chapter, we cover graph theory and its application to financial networks, different types of networks, their traits and capabilities in terms of expressiveness. We also peruse the application of social network analysis as a tool to extract information from financial networks. Finally, we introduce and summarize some approaches to streaming, seeking to learn how to best use the power of streamed data to deal with financial data streams. The goal of reviewing the state of the art is to guide us in the definition of a methodology, presented in the next chapter. The results, findings and suggested future lines of work found in literature motivate our choices.

Chapter 3

Methodology

In this chapter, we discuss the methodology of the study. We start by presenting the pre-processing method that turns a stream of stock prices into a stream of return values. We describe how to compute statistics over that stream using three window models, how to build the correlation networks and how to measure the evolution node centralities, communities and the networks themselves. We conclude with an overview of the experimental system implementing the methodology.

3.1 Overview

Stock market analysis is often approached as the study of relationships between variables that represent a particular stock attribute (e.g. price, return value, volume, etc.) over a period of time.

Network analysis has proven to be an excellent tool to address this problem: a market can be modelled as a network, an undirected weighted graph in which nodes represent stocks, edges represent correlation between stocks and the weight associated with each edge illustrates the strength of the relationship.

In what follows, we present a methodology that enables the use of network analysis on streams of stock data to study the evolution of correlation between return values of pairs of stocks over different periods of time. Markets are dynamic in the sense that new stocks may be added and to and/or removed from time to time (Rao et al., 2000; Harris and Gurel, 1986; Shoven and Sialm, 2000), therefore changes in the node set must be addressed. The goal is to provide a tool that can be used to drive decision-making for stock portfolio management.

3.2 Preprocessing data

Given a stock market M composed of n stocks s_i $(i = 1, \ldots, n)$, let

$$M(t) = (M_1, \ldots, M_T),\ 1 \leq t \leq T \tag{3.1}$$

be a matrix-based time-series (Tiao and Box, 1981; Antille, 2007) where each time point M_t is a matrix of size $n \times k$ bearing information about the n stocks traded in M at a given point in time t. Each row in M_t is a k-tuple

$$s_i = (t, i, m_i, p_i, h_i, l_i, a_i, t_i, v_i) \tag{3.2}$$

where t is a time stamp, i is the stock's ticker name, m_i is the name of the company, p_i is the quote (price) at time t, h_i and l_i are respectively the highest and lowest quotes up until t, a_i and t_i are respectively the absolute and percentage differences between the last quote at times t and $t-1$ and v_i is the traded volume until t. Due to changes in the market's components, it is possible that two consecutive matrices M_t and M_{t+1} exhibit a different number of rows, that is, it may happen that $dim(M_t) = n \times k$ and $dim(M_{t+1}) = m \times k$ where $n \neq m$.

The $M(t)$ time series meets the criteria defined by Datar et al. (2002) for a data stream: it is constantly changing, by insertion of new elements only, rendering repeated operations over its entire contents unnecessary. Let us then process $M(t)$ as a data stream, collecting one matrix M_t for every time period $\Delta t = t - (t-1)$ and extracting quotes p_i for each stock s_i in the market. Given two consecutive matrices M_{t-1} and M_t, we can then compute the return values (Mantegna, 1999; Vandewalle et al., 2000; Bonanno et al., 2001; Onnela et al., 2004; Tumminello et al., 2005; Tse et al., 2010) for each stock s_i as

$$r_{i,t} = \begin{cases} 0 & \text{if } \nexists\ p_{i,t-1} \\ log(p_{i,t}) - log(p_{i,t-1}) & \text{otherwise} \end{cases},\ t = 0, 1, \cdots, T \tag{3.3}$$

We can thus transform a stream of stock quotes into a stream of stock return values from which we can derive statistics.

3.3 Statistics and data streams

In a streaming context, data is unbound in size which makes the task of keeping all return values in memory impracticable. We solve the problem by adopting either summary or synopsis that allow us to compute statistics over an infinite sequence of return values without having to keep them all in memory and with an acceptable associated error ϵ (Gama, 2010). Since market components may change over time, we adapt by adding and/or removing summaries/synopsis following a *naïve* approach: summaries/synopsis are added and dropped as stocks enter and exit the market.

Gama (2010) presents a practical approach to maintain simple statistics over data streams. Cross-correlation, in particular, can be computed incrementally by

$$\rho_{ij} = \rho(r_{i,t}, r_{j,t}) = \frac{nS_{ij} - S_i S_j}{\sqrt{nS_i^2 - (S_i)^2}\sqrt{nS_j^2 - (S_j)^2}} \tag{3.4}$$

where i and j are stocks in market M, $t = 1, \ldots, n$ is a particular point in time, and n, S_i, S_j, S_i^2, S_j^2 and S_{ij} are summaries defined as

$n = \min(\#r_{i,t}, \#r_{j,t})$	minimum number of return values collected for stocks s_i and s_j until time t
$S_i = \sum_{t=1}^{n} r_{i,t}$	sum of return values for stock s_i until time t
$S_j = \sum_{t=1}^{n} r_{j,t}$	sum of return values for stock s_j until time t
$S_i^2 = \sum_{t=1}^{n} r_{i,t}^2$	sum of squared return values for stock s_i until time t
$S_j^2 = \sum_{t=1}^{n} r_{j,t}^2$	sum of squared return values for stock s_j until time t
$S_{ij} = \sum_{t=1}^{n} r_{i,t} \cdot r_{j,t}$	sum of cross-product of return values for stocks s_i and s_j until time t

The correlation table $\mathrm{P}_M = \{\rho_{ij} \mid \forall i, j \in M\}$ is built by computing cross-correlation of all pairs of stocks.

Computing statistics using landmark window

Landmark windows (Babcock et al., 2002; Sarmento et al., 2016) keep track of all data items from a given point in time onwards; no differentiation is made between data items as older ones are as important as the more recent ones. This model serves as a baseline of the study, helping helps us to determine how the return values of stocks correlate with each other in the entire time span of the study.

To compute the cross-correlation between the return values of any pair of stocks in the market, all it takes is to define a set of the previously enumerated summaries per stock and keep updating it. The use of summaries enables us to compute exact cross-correlations.

Computing statistics using a gradual forgetting

Stock markets are dynamic environments, where prices follow non-stationary distributions (Fama, 1965; French et al., 1987; Mantegna and Stanley, 2000) and return rates go through alternated periods of big and small fluctuations (Raunig and

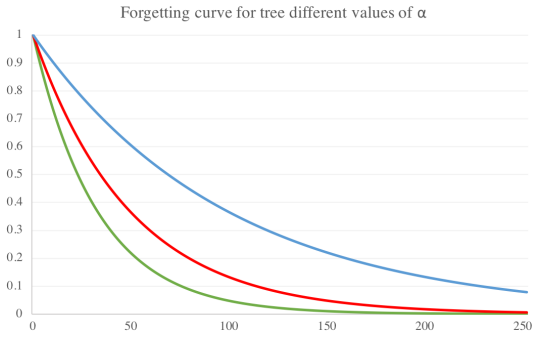


Figure 3.1: Forgetting curves of different intensities

Scharler, 2010). As prices and returns vary, we want to adapt and update relationships between stock.

Koychev (2000, 2002) and Klinkenberg (2004) present time-based forgetting functions based on weights as a means to deal with concept shift. Cohen and Strauss (2003) introduces time-decay forgetting functions and peruses its impact on statistics computed over data streams. In our work, we resort to an aggregation function that multiplies the previously enumerated summaries by a fading factor $\alpha \in]0, 1[$. Let us assume that, at a given point in time t, r_i is the observed return value for a particular stock s_i and that $A_i(t-1)$ is one of the five summaries storing data up until $t-1$. If we update the summary as

$$A_i(t) = r_{i,t} + \alpha A_i(t-1) \tag{3.5}$$

we manage to progressively adapt to the current context while keeping enough information to remember the recent performance of s_i in terms of return value. When we expand the right-hand side of the aggregation function

$$\begin{aligned} A_i(t) &= r_{i,t} + \alpha A_i(t-1) \\ &= r_{i,t} + \alpha(R_{i,t-1} + \alpha A_i(t-2)) \\ &= r_{i,t} + \alpha r_{i,t-1} + \alpha^2 r_{i,t-2} + \ldots + \alpha^n r_{i,t-n} \end{aligned} \tag{3.6}$$

we observe that it evolves exponentially (Cohen and Strauss, 2003) with the degree of coefficient α increasing as time goes by, thus rendering the older return values ever smaller. This fenomenon resembles the forgetting curve (Ebbinghaus, 1913), whose effect in social networks is studied by Kudelka et al. (2010). Figure 3.1 illustrates forgetting curves produced by three different values for α; one can clearly see the effect of the fading factor's value of the forgetting process' speed.

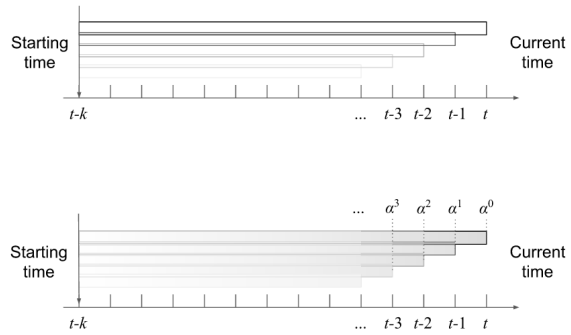


Figure 3.2: Landmark window, with (top) and without (bottom) forgetting

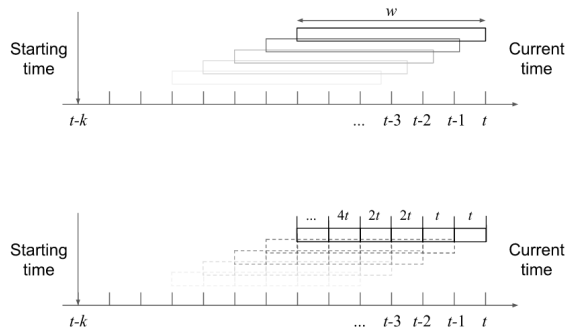


Figure 3.3: Sliding window, without (top) and with (bottom) exponential histogram

Computing statistics over sliding windows

Sliding windows (Babcock et al., 2002), whether fixed or adaptive, are another approach to force loss of relevance on older data. It is an abrupt forgetting mechanism where items falling out of the window span are deleted from memory never to be used again.

We define a fixed sliding window spanning over a period of $t = \omega$ trading days, displaced every Δt days. On top of that window, we define an *exponential histogram* synopsis (Datar et al., 2002) whose buckets hold return values computed in non-overlapping periods of $2^k t < T$, where $k = 0, 1, \ldots, n$ trading sessions. The admissible relative error in statistics is given by $0 < \epsilon < 1$.

Just like with summaries, we keep a set of five histograms for each stock in the stock market. To get the cross-correlations between a pair of stock, we compute the sum of the contents of all buckets in the respective histograms as described in Datar et al. (2002).

3.4 Building the network

The building blocks are now in place for us to produce market M's network as an undirected weighted graph

$$G_M = (V_M, E_M, w) \tag{3.7}$$

where $V_M = \{s_1, \ldots, s_i, s_j, \ldots, s_n\}$ is the set of nodes (the stocks), $E_M = \{(s_i, s_j) \mid \forall s_i, s_j \in V_M\}$ is the set of edges that connect every possible pair-wise combinations of stocks and w is a function that returns the weight of each edge as

$$\begin{array}{rccc} w: & E_M & \to & \mathrm{P}_M \\ & e_{ij} & \rightsquigarrow & \rho_{ij} \end{array} \tag{3.8}$$

By design G_M is undirected, since cross-correlation as defined in equation 3.4 is a symmetrical function in which $\rho_{ij} = \rho_{ji},\ \forall s_i \neq s_j \in V_M$. It is also fully connected, thus holding many edges that bring little information about the market dynamics. To filter out these edges, we follow the approach proposed in Tse et al. (2010); Namaki et al. (2011); Heiberger (2014) and define a threshold value $\theta > 0$ that is used to keep in the network any edge $e_{ij} \in E_M$ such that

$$\rho_{ij} \geq \theta \tag{3.9}$$

For the sake of completeness, we also wish to consider the impact of negative correlations in the structure of the networks. For that matter, we redefine the inclusion criterion to keep in the network any edge $e_{i,j} \in E_M$ such that

$$|\rho_{ij}| \geq \theta \tag{3.10}$$

3.5 Studying the evolution of the network

We wish to study the evolution of the network, its communities and actors over a period of time T. To do so, we split T in several small overlapping windows $\Delta t = t - (t-1)$ for which we build networks such as the one as described in section 3.4. Formally, we build time series of graphs or network snaphots (Park et al., 2013; Rossetti and Cazabet, 2018)

$$G(t) = (G_1, \ldots, G_t),\ 1 \leq t \leq T \tag{3.11}$$

where each time point is a graph $G_t = (V_t, E_t, w_t)$, V_t is the set of nodes, E_t is the set of edges and w_t is the function giving the weight of the relation at time t. The graph-based time series are as long as the matrix-based time series $M(t)$ from which we derive returns and correlations.

Once again, $G(t)$ meets the criteria defined by Datar et al. (2002) for a data stream, so we handle it as such. We build one graph-based stream for each possible combination of a window model (landmark, gradual forgetting and sliding windows) with a threshold filter (positive and absolute) and consume it one network at a time, collecting several metrics along the way.

3.5.1 Measuring the evolution of the network measures

The *density* of a network is an important measure that explains the general level of connectedness of a network. Being a ratio between the existing and the possible maximum number of of edges in the network, density is minimal when the network has no edges and maximal when the network is perfectly connected. In simple terms, high density values are associated with dense networks, whilst low density values are associated with sparse networks (Oliveira and Gama, 2012b). The value of the density of a network is given by

$$\rho(G) = \frac{m}{m_{max}},\ 0 < \rho < 1 \tag{3.12}$$

where m and $m_{\max}$ denote respectively the number of existing and all possible edges in G; $m_{\max}$ is $n(n-1)/2$ for undirected graphs and and $n(n-1)$ for directed ones.

The *average degree* is the mean of the degrees of all nodes in a network; it is a measure of the global connectivity of the network (Costa et al., 2011) and a powerful indicator of the network's density (Lima, 2015). The average degree is given by

$$\bar{k} = \frac{1}{n}\sum_{j=1}^{n} k_i \tag{3.13}$$

3.5.2 Measuring the evolution of communities

Modularity, as defined by Blondel et al. (2008), is a measure for the quality of the division of a network into communities. It is given by

$$Q = \frac{1}{2m}\sum_{ij}\left[A_{ij} - \frac{k_i k_j}{2m}\right]\delta(c_i, c_j) \tag{3.14}$$

where m is the number of edges, k_i and k_j are respectively the degree of nodes i and j, A_{ij} is the cell of the adjacency matrix A holding the number of edges between i and j, $\frac{k_i k_j}{2m}$ is the expected number of edges falling between i and j, c_i and c_j denote the groups to which nodes i and j belong and $\delta(c_i, c_j)$ is the Kronecker delta. The observed values for Q help us assess the presence of meaningful communities; in particular, we're looking for values within $[0.3, 0.7]$ (Newman and Girvan, 2004).

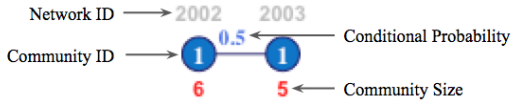


Figure 3.4: Example of survival transitions with $\tau = 0.5$.

Alongside with the modularity value, we capture comprehensive lists of communities' members, known as *enumerations* or *extensional definition of clusters*. According to Oliveira and Gama (2012a), a cluster can be represented as

$$C_k(t) = \{x_1, \ldots, x_n\} \tag{3.15}$$

where k is the index of the cluster, $x_i (1 \leq i \leq n)$ is an observation assigned to cluster k and $t = 1 \ldots T$ is the last analysed timestamp. In our case, x_i is a merely the stock i's ticker name.

We use the Louvain algorithm to compute modularity and stock membership in the different communities emerging over time.

We adopt the MEC framework (Oliveira and Gama, 2012a) to study the evolution of communities. We therefore need define the number of transitions to study. Depending on the size of the studied dataset, the analysis evolution of communities might be cumbersome. For large datasets, we turn to sampling to reduce the number of studied networks. We take n snapshots from the enumerations obtained by Louvain, corresponding to $n-1$ time intervals $[t_i, t_{i+\Delta t}]$, where $1 \leq i \leq n-1$ and $\Delta t = T/n$. Then, and for each snapshot, we build a bipartite graph $G = (U, V, E)$ where U is the set of clusters at time t_i, V is the set of clusters at time $t_{i+\Delta t}$ and E is the set of weighted edges connecting pairs of clusters $(C_u(t_i), C_v(t_{i+\Delta t}))$. The weight function is given by

$$\begin{aligned} weight(C_u(t_i), C_v(t_{i+\Delta t})) &= P(S \in C_v(t_{i+\Delta t}) | S \in C_u(t_i)) \\ &= \frac{\sum P(s \in C_u(t_i) \cap C_v(t_{i+\Delta t}))}{\sum P(s \in C_u(t_i))} \end{aligned} \tag{3.16}$$

where S are sets of stocks found in cluster $C_u(t_i)$ and $P(S \in C_v(t_{i+\Delta t}) | S \in C_u(t_i))$ is the conditional probability of S belonging to cluster $C_v(t_{i+\Delta t})$ given that S belongs to $C_u(t_i)$. Figure 3.4 illustrates a survival transition between communities.

We study *Birth*, *Merge*, *Split*, *Survival* and *Death* events and set appropriate values for the survival (τ) and split (λ) thresholds. The events are defined as follows:

Birth

$$0 < weight(C_u(t_i), C_v(t_{i+\Delta t})) < \tau, \ \forall u \tag{3.17}$$

Merge

$$\begin{gathered} weight(C_u(t_i), C_v(t_{i+\Delta t})) \geq \tau \wedge \\ \exists\, C_w \neq C_u : weight(C_w(t_i), C_v(t_{i+\Delta t})) \geq \tau \end{gathered} \tag{3.18}$$

Split

$$\begin{gathered} \exists\, C_v \neq C_w : weight(C_u(t_i), C_v(t_{i+\Delta t})) \geq \lambda \wedge \\ weight(C_u(t_i), C_w(t_{i+\Delta t})) \geq \lambda \wedge \\ \sum_{k=1}^{n} weight(C_u(t_i), C_k(t_{i+\Delta t})) \geq \tau \end{gathered} \tag{3.19}$$

Survival

$$\begin{gathered} weight(C_u(t_i), C_v(t_{i+\Delta t})) \geq \tau \wedge \\ \nexists\, C_w \neq C_u : weight(C_w(t_i), C_v(t_{i+\Delta t})) \geq \tau \end{gathered} \tag{3.20}$$

Death

$$weight(C_u(t_i), C_v(t_{i+\Delta t})) < \lambda,\ \forall v \tag{3.21}$$

3.5.3 Measuring the evolution of node centralities

Centrality is a measure an actor's position within a social network Oliveira and Gama (2012b). Centrality metrics such as *degree*, *betweenness*, *closeness* (Freeman, 1978) and *eigenvector* centralities (Bonacich, 1987) are used to establish the social rank of each actor in terms of involvement, control and reachability (Freeman, 1978). High centrality values are usually associated with powerful actors that have easy access other actors and/or control the flow of information between the other actors in the social network.

We study the evolution of *eigenvalue* centrality as a means to establish both the *quantity* and the *quality* of a stock's correlations. Eigenvalue centrality can be computed by

$$x_i = \frac{1}{\lambda} \sum_{j=1}^{n} a_{ij} x_j \tag{3.22}$$

where x_i and x_j are respectively the centrality of stocks i and j, a_{ij} is the a cell in the (binary) adjacency matrix A and λ is the first (or largest) eigenvector of A.

We also study node *betweenness* to assert which stocks lie in the paths between other stocks more often, thus controlling the propagation of trends between communities. Node betweenness is given by

$$b_k = \sum_{i,j \in V \setminus \{k\}} \frac{\sigma_{ij}(k)}{\sigma_{ij}} \tag{3.23}$$

where $\sigma_{ij}(k)$ is the number of shortest paths between stocks i and j passing through stock k and $\sigma_{ij}(k)$ is the total number of shortest paths between stocks i and j.

Finally, we study *closeness* to determine how fast a given stock can reach every other stock in the network. Formally, the closeness of stock i to all other stocks $j \neq i$ is given by

$$C_i = \frac{n-1}{\sum_{j \in V \setminus \{i\}} d_{ij}} \tag{3.24}$$

where d_{ij} is the length of the shortest path between stocks i and j.

3.5.4 Determining frequent stock-sets

To determine the most common associations of stocks, we turn to the discovery of frequent items-sets and associations rules as described by Agrawal et al. (1993); we freely adapt the work of the authors to meet our purposes.

Let us consider the sets S_M of stocks traded in a stock market M and C_M of communities discover in S_M over time. One can regard C_M as a set of transactions in M, since every community in $c_k \in C_M$ is comprised of stocks $s_i \in S_M$. Given a threshold $\sigma_{min} \in]0, 1]$, the frequent stock-sets are given by

$$\{c_k \subseteq S_M | P(c_k \subseteq C_M) \geq \sigma_{min}\} \tag{3.25}$$

where $P(c_k \subseteq C_M)$ is the probability of c_k occurring in C_M, also known as the *support* of c_k. The support can be used to measure of frequency of specific stocks-sets in the communities disclosed over time, complementing the information produced by the MEC framework.

Given the stock-sets, one can derive association rules from them. An association rule is an implication

$$c_k \Rightarrow \{s_i\} \tag{3.26}$$

where c_k and s_i are subsets of S_M and; c_k is called the antecedent and $\{s_i\}$ is called the consequent of the rule. It means that, whenever c_k is part of a community, s_i is also found in that community.

The quality of a rule can be assessed by measures. To select the most interesting rules, one can set thresholds over those measures. The *support* of the rule is the probability of finding communities that contains both the c_k and $\{s_i\}$

$$support(c_k \Rightarrow \{s_i\}) = P(c_k \cap \{s_i\})$$

The *confidence* of the rule is the probability of finding communities that containing c_k also contain $\{s_i\}$

$$confidence(c_k \Rightarrow \{s_i\}) = P(\{s_i\}|c_k) = \frac{P(c_k \cap \{s_i\})}{P(c_k)}$$

Finally, the *lift* (Brin et al., 1997) of the rule measures the deviation of the rule from the statistical independence of c_k and $\{s_i\}$

$$lift(c_k \Rightarrow \{s_i\}) = \frac{confidence(c_k \Rightarrow \{s_i\})}{support(\{s_i\})} = \frac{P(c_k \cap \{s_i\})}{P(c_k) \cdot P(\{s_i\})}$$

Lift takes values in $[0, +\infty[$. If $lift = 1$, c_k and $\{s_i\}$ are independent. If $lift < 1$, c_k and $\{s_i\}$ are negatively correlated. If $lift > 1$, c_k and $\{s_i\}$ are positively correlated.

3.6 Experimental system

The experimental system is implemented as a five-tier architecture, illustrated in Figure 3.5:

Input data collectors (web-crawlers, web-scrapers and parsers);

Transformation stream producers, transformers and consumers;.handles all business logic, such as computation of correlation between stocks, threshold-based filtering, the conversion to network format and network analysis tools;

Transport media for data streams; handles message transport between all components in the transformation tier;

Persistence persist data passing through the data streams in collections; introduces the ability to replay data streams;

Output data publishers.

Data flow

In what follows, we describe the flow of data from end to end. For simplicity, we describe a simplified version of the system where only one component of each type exists and a single stock market is studied.

For any given stock market, financial data enters the system through a collector, a component that reads stock quotes by some means (screen-scraping, API, web-socket, etc). The collector gathers data tuples as described in equation 3.2, one k-tuple for each stock s_i $(1 \leq i \leq n)$ in the market's portfolio. The tuples are passed

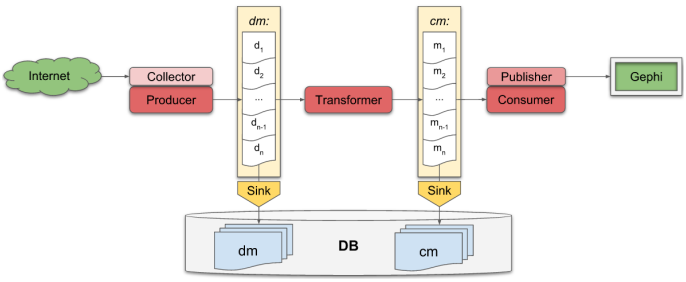


Figure 3.5: Conceptual architecture

on to the stream producer, whose function is to aggregate all tuples relative at a given time point t in a $k \times n$ table $M_i(t)$ and publish a message $m = \{t, M_i\}$ to the *data matrix* (dm) topic.

The stream transformer picks up and parses message m. For each stock s_i, it computes the return values $r_{i,t}$ and immediately adds them to the sufficient statistics of the stock. It then computes a $n \times n$ correlation matrix P, filters it according to the predefined threshold θ and publishes a message $m' = \{t, \mathrm{P}_\theta\}$ to the market's *correlation matrix* (cm) topic.

The stream consumer picks up and parses message m'. It first builds the undirected weighed graph $G = (V, E, w)$ out of P_θ. The graph is the passed on to the publisher, that decorates each node with labels indicating the stock ticker symbol and activity sector before sending it to the visualizer through a specialized web API.

At start-up time, each topic is attached to a data sink that traps each transported message and persists it within a specific database collection; the collection bear the same name as the topic.

3.7 Summary

In this chapter, we present the pre-processing method used to produce streams of stock return values. We describe how to compute statistics over that stream using three window models, how to build the correlation networks and how to measure the evolution node centralities, communities and the networks themselves. We conclude with an overview of the experimental system implementing the methodology. To validate our methodology, we apply it over a data set of financial data regarding the Dow Jones Industrial Index. The outcome is presented in the next chapter.

Chapter 4

Study of the Evolution of a Stock Market

In this chapter we present and discuss the experimental evaluation of our methodology over a data set of financial data regarding a major Unites States index. We motivate and describe different experimental settings, interpret and compare results, discuss the adherence of our findings to the reality of the studied index and the applicability of the methodology in portfolio management.

4.1 Data

We collect our data from *www.investing.com*, a global financial portal that provides news and analysis, streaming quotes and charts, historical data and other useful information about global financial markets.

Among other major world indices, *investing.com* offers information about the Dow Jones Industrial (DJI) average, an index created by Charles Dow and Edward Jones in May 26, 1896 (Shoven and Sialm, 2000). The index is comprised of 30 *blue-chip* stocks, of large publicly owned American companies, traded in the New York Stock Exchange (NYSE) and the National Association of Securities Dealers Automated Quotations (NASDAQ) stock exchange. Table A.1 contains the list of DJI components by December 29^{th} 2017, their ticker names, industries and sectors of activity. Figure A.1 shows the distribution of these stocks by sector of activity.

We study DJI's historical data regarding the period from January 2, 1997 to December 29, 2017, a total of 5285 trading days and 158550 closing quotes illustrated in Figure A.3. We convert this data set into a data stream, producing one complete trading day (30 quotes) every $\Delta t = 300$ seconds. We keep our focus on the daily closing quotes despite the availability of several items of financial data, such minimum and maximum quotes or traded volume. Future work can study networks obtained from correlation between other, and perhaps several, variables.

Two notes are in order regarding the collected historical data:

- `Goldman Sach` and `Visa` time series are missing data. Goldman's initial public offering (IPO) took place in May 1999, Visa went through a merger and became a publicly traded company in October 2006. As such, we are only able to collect data for 4696 trading days for GS and 2464 trading days for Visa;
- `Apple`, `Cisco Systems`, `Intel` and `Microsoft` time series have closing quotes for February 27, 2016. This is due to the fact that these stocks are traded in NASDAQ whereas all others are traded in NYSE, which operate in different trading days. The additional values are discarded.

Historical *versus* online data

We choose to study the time period from 1997 through 2017 due to the richness of financial, economical and historical events that have take place then. Given that this historical data is no longer available as a data stream, we simulate one resorting to a mock web-server that publishes tables in the exact same format of *investing.com*, illustrated in Figure A.2. This way, we can change our data source seamlessly.

Let us assume that we use the actual online source and configure the collector with a sampling rate of $\Delta t = 300$ seconds. DJI trades from Monday through Friday, from 9:30 to 16:00 Eastern Standard Time (4:30 to 11:00 Coordinated Universal Time), 5 days a week. With the defined sampling rate, we are able to scrape 78 matrices $M_{30\times 8}$ a day, for a total of 2340 quotes. If we keep the system running for 13 weeks (one quarter), we are able to consume a 5070 matrix-value data points, for a total of 152100 quotes. This volume of data is roughly equivalent to that of the mocked stream of historical data.

4.2 Experimental Evaluation

We conduct a series of experiments to assess the differences between two window models when applied to the DJI data stream: gradual forgetting over landmark windows and exponential histograms over sliding windows, henceforth referred to as *gradual forgetting* and *sliding windows* respectively. We also use a plain landmark window as baseline. Our goal is to monitor the way in which stocks cluster within the social network over time, to determine the most influential stock in communities and possibly in the entire network.

Throughout the chapter, we often refer to the cross-correlation coefficient ρ, a value with continuous distribution in $[-1, 1]$. For commodity, we split it into three discrete categories. Table 4.1 illustrates the distribution of correlation coefficients over the defined categories. We also refer often to the size of communities. For

strength	ρ	
	$-$	$+$
weak	$]-0.3, 0.0[$	$]0.0, 0.3[$
moderate	$]-0.5, -0.3]$	$[0.3, 0.5[$
strong	$[-1.0, -0.5]$	$[0.5, 1.0]$

Table 4.1: Cross-correlation categories

Type	Size
very small	$1-4$
small	$5-10$
medium	$11-20$
big	$21-26$
very big	$27-30$

Table 4.2: Community size categories

commodity, we split communities by size into five categories. Table 4.2 illustrates the distribution of community sizes over the defined categories.

4.2.1 First experience: Defining a correlation level

We start by familiarizing ourselves with data, studying the distribution of correlations between return values for the three window types and exploring towards a good correlation threshold to use while constructing the networks. For this experience, the following parameters are defined:

- The baseline landmark window spans the entire period of 20 years, with a data point acquired every $t = 1$ trading days;
- To study the impact of gradual forgetting, a fading factor of $\alpha = 0.996$ is imposed over the data items in a landmark window matching the baseline;
- Exponential histograms with a admissible relative error $\epsilon \leq 0.02$ are built over sliding windows of $\omega = 252$ trading days, displaced every $\Delta t = 1$ trading days.

Results

The distributions of correlations between return values are illustrated by the histograms in Figure 4.1. For landmark windows, most correlations are moderate ($[0.3, 0.4[$), whereas for other two models correlations are moderate-strong ($[0.3, 0.6[$). There are few correlations above 0.8 and, most interesting, values are mostly positive. For gradual forgetting and sliding windows, there are less that 1% of negative correlation values, for landmark windows, there are no negative correlations at all.

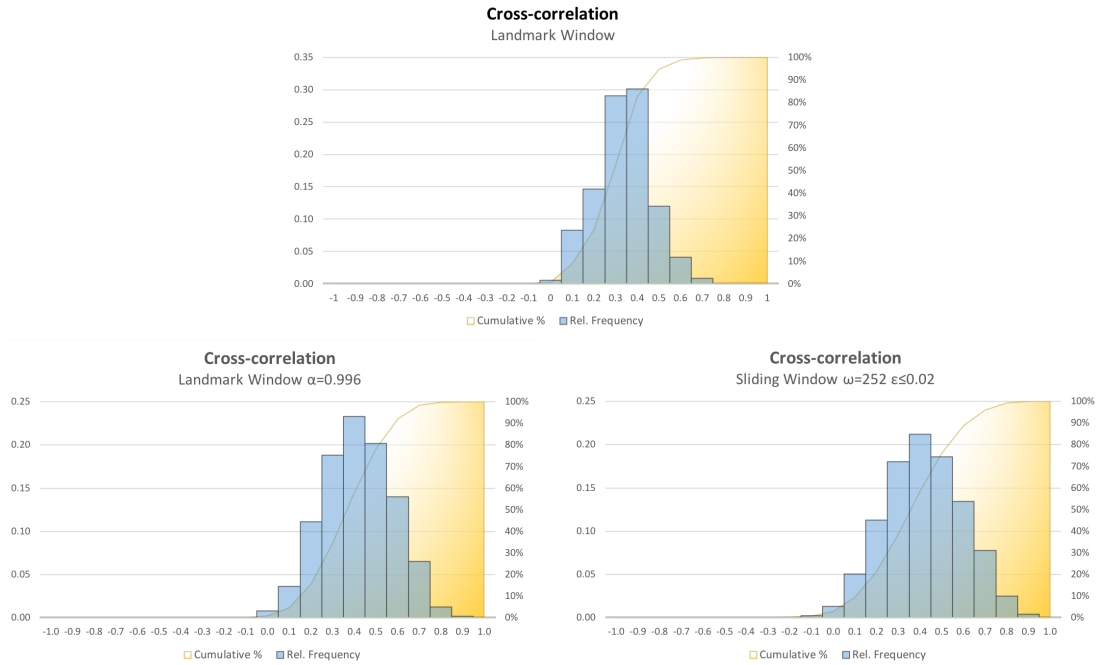


Figure 4.1: Cross-correlation distributions for landmark windows, gradual forgetting and sliding windows of one year

Let us take a look to the evolution of the network and community measures over the entire studied period. Table 4.3 shows the mean and standard deviation (within round brackets) for density, average degree, modularity and number of communities from January 1997 through December 2017. Table 4.4 shows the relative frequency, mean and standard deviation (within round brackets) of the size of communities larger than two members.

Interpretation of results

The following facts emerge from this experience:

- Density and average degree decrease as the correlation threshold θ increases. This is expected. Both measures are related to the number of edges and edges represent correlations that match or exceeds a given threshold. The higher the threshold value, the fewer edges.
- Modularity and the number of communities rise with the value of the threshold θ. This is also expected. As the threshold increases, the intra-community edges are more likely to be kept than the inter-community ones. The division of the network into modules is then clearer and thus modularity rises. The number of communities rises because many nodes become isolated.

$\rho \geq \theta$	**Landmark window**		**Landmark window** $\alpha = 0.996$		**Sliding window** $\omega = 252$ $\epsilon \leq 0.02$	
θ	**Density**	**Av.Deg.**	**Density**	**Av.Deg.**	**Density**	**Av.Deg.**
0.2	0.76 (0.11)	21.65 (3.32)	0.84 (0.19)	23.86 (6.73)	0.82 (0.20)	23.26 (5.90)
0.3	0.47 (0.18)	13.37 (5.18)	0.65 (0.26)	18.48 (7.77)	0.64 (0.27)	18.13 (7.90)
0.4	0.17 (0.07)	4.87 (1.99)	0.41 (0.29)	11.85 (8.59)	0.42 (0.29)	12.09 (8.57)
0.5	0.05 (0.03)	1.47 (0.76)	0.21 (0.24)	6.16 (7.08)	0.24 (0.26)	6.81 (7.51)
0.6	0.01 (0.01)	0.32 (0.37)	0.08 (0.12)	2.23 (3.57)	0.11 (0.18)	3.19 (5.35)
0.7	0.01 (0.03)	0.41 (0.80)	0.04 (0.10)	1.14 (2.93)	0.03 (0.06)	0.78 (1.66)
θ	**Modular.**	**#Comm.**	**Modular.**	**#Comm.**	**Modular.**	**#Comm.**
0.2	0.06 (0.04)	3.46 (0.59)	0.05 (0.07)	2.60 (0.91)	0.06 (0.07)	2.62 (0.94)
0.3	0.15 (0.09)	5.82 (1.17)	0.10 (0.12)	4.15 (2.58)	0.10 (0.12)	3.92 (2.31)
0.4	0.33 (0.14)	10.81 (1.99)	0.19 (0.17)	7.26 (4.34)	0.18 (0.18)	7.16 (4.30)
0.5	0.61 (0.15)	17.22 (3.10)	0.33 (0.23)	13.26 (6.55)	0.29 (0.22)	12.36 (6.79)
0.6	0.61 (0.07)	25.37 (1.55)	0.42 (0.24)	20.41 (6.76)	0.36 (0.22)	18.53 (7.38)
0.7	0.22 (0.23)	26.08 (3.79)	0.25 (0.25)	24.32 (5.89)	0.21 (0.23)	24.99 (5.40)

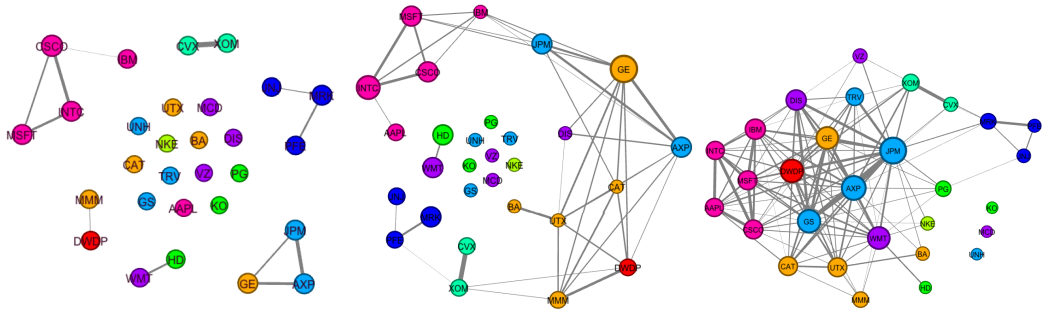
Table 4.3: Mean and standard deviation of measures observed in networks of different correlation levels

$\rho \geq \theta$	**Landmark window**		**Landmark window** $\alpha = 0.04$		**Sliding window** $\omega = 252$ $\epsilon \leq 0.02$	
θ	Rel.freq.	|Comm.|	Rel.freq.	|Comm.|	Rel.freq.	|Comm.|
0.2	0.67	12.18 (3.65)	0.92	12.17 (4.66)	0.94	11.89 (5.10)
0.3	0.56	8.24 (2.90)	0.68	9.87 (4.93)	0.75	9.67 (5.09)
0.4	0.39	5.42 (2.59)	0.48	7.31 (4.54)	0.49	7.31 (4.70)
0.5	0.30	3.34 (1.51)	0.29	5.20 (3.77)	0.31	5.61 (3.90)
0.6	0.12	2.30 (0.55)	0.16	3.70 (2.47)	0.31	5.49 (3.75)
0.7	0.03	2.04 (0.51)	0.06	2.94 (1.63)	0.08	3.32 (2.15)

Table 4.4: Relative frequency, mean (and standard deviation) of sizes for communities larger than two stocks in networks of different correlations levels

- Networks obtained from correlations $\rho \geq \{0.2, 0.3\}$ are too dense and show low values of modularity. As such, there is too much noise in the networks and the found communities are of little interest to the study.
- Modularity value drops in all window models for correlation values $\rho \geq 0.7$. Setting a large threshold θ removes many edges, both intra or inter-community. Therefore, modularity drops.
- Modularity in $[0.3, 0.7]$ (Newman and Girvan, 2004) appear in correlation values $\rho \geq \{0.5, 0.6\}$. However, the number of communities for those thresholds levels is high relatively to the number of stocks represented in DJI, meaning that most of those communities have a single stock and are not interesting. The number of relevant communities, as well as their sizes, is higher for $\rho \geq 0.5$.

All these facts, corroborated by similar findings over different window sizes and forgetting factors (Appendix B), lead us to peek a correlation threshold of 0.5 to built the networks.



(a) (b) (c)

Figure 4.2: Correlation networks for landmark window (a), gradual forgetting (b) and sliding windows (c) by the end of 2003

4.2.2 Second experience: Communities and centralities

The goal of this experience is to assess how communities and centralities evolve throughout the studied time period. Our goal is to study the birth, merge, split, survival and death of communities and determine their most relevant stocks. For this experience, the following parameters are defined:

- The baseline landmark window spans the entire period of 20 years, with a data point acquired every $t = 1$ trading days;
- A fading factor $\alpha = 0.996$ is imposed over a landmark window similar to the baseline;
- Exponential histograms with a admissible relative error $\epsilon \leq 0.02$ are built over sliding windows of $\omega = 252$ trading days, displaced every $\Delta t = 1$ trading days;
- A positive thresholds $\theta = 0.5$ is used to filter noise out of networks;
- The survival threshold τ is set to 0.5, the split threshold λ is set to 0.4;
- Communities with a single member are disregarded while building the MEC graphs;
- To study the dynamics of eigenvalue centrality, we impose a threshold of 1;
- To study the dynamics of betweenness centrality, we impose a threshold of 0.1;
- To study the dynamics of closeness centrality, we pick the top 10% values.

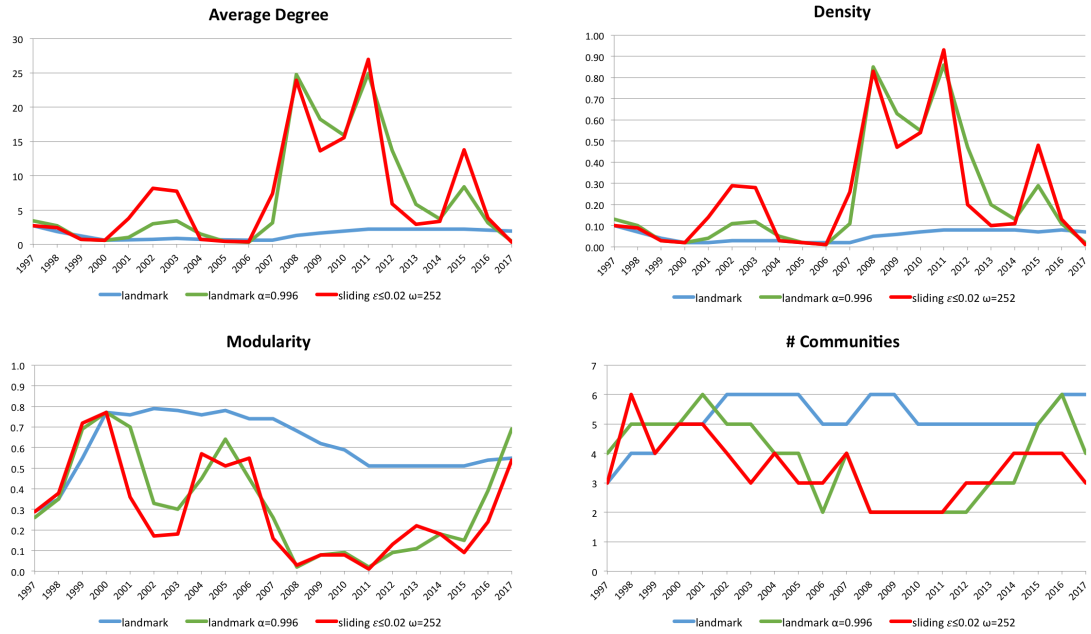


Figure 4.3: Average degree, density, modularity and number of communities for all window models, annual values from 1997 through 2017

Results

Figure 4.3 illustrates the evolution of the average degree, density, modularity and the number of communities by the end of each year. Table B.3 holds the data relative to the charts.

The behaviour of the landmark window is relatively stable. The average degree is low in the first decade, picking up after 2007. Density follow the exact same pattern. This indicates smooth transitions in network configurations, with few connections between nodes and few neighbours for each node. Modularity quickly rises until 2000, keeping above 0.7 until 2007, when it starts to slowly drop until 2011, where it stabilizes. All values are above 0.3, indicating well-established communities. The number of communities rises until 2002, oscillating between 5 and 6 communities thereafter.

The behaviour of the other two models is far from stable and similar to a certain extent, despite the quicker reaction of sliding windows to concept change. The average degree moves up and down moderately in the first decade, soars between 2007 and 2011, then plummets to a more moderate range with the exception of the peak in 2015. Once again, density follows the pattern of average degree. Modularity shows a similar behaviour of inverse direction; when density rises, modularity drops and vice versa. In the first decade, networks are generally stable and well defined, with modularity values over 0.3. Then, in 2008, modularity plummets, hinting high

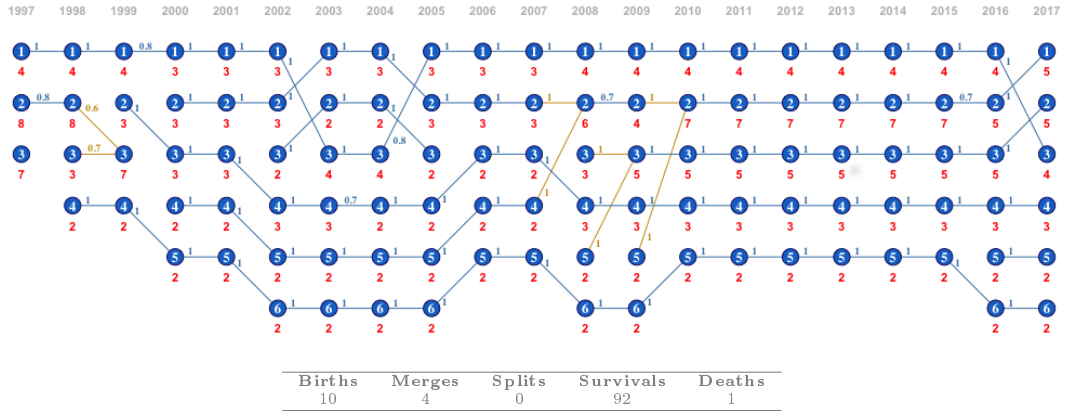


Figure 4.4: MEC graph for landmark window, annual snapshots from 1997 though 2017.

instability and fast changing networks; the turmoil lasts until 2015. The number of communities shows trends similar to modularity.

We now turn to the MEC framework to analyse communities evolution. For each window model, we take a snaphot of the network every 252 trading days, at the last day of trading in each year, for a total of 20 snapshots from 1997 through 2017. We use the snapshots to plot the MEC graph, gauging the evolution of communities in consecutive years. We also take samples of the centrality measures. Sampling intervals are chosen to match the length of the sliding windows; this rationale is applied in this and in the following experiences.

Communities, centralities and landmark window Figure 4.4 illustrates the evolution of detected communities; members are listed in Table C.1. This is a picture of high stability with a high survival rate. The edges's weights are in general very high. Communities form in early years and survive for a long time. A few merges take place and only one death occurs. The vast majority of communities is very small, with 2 to 4 elements; this trend spans the entire studied period.

Figure 4.5 illustrates the centrality measures. Four stocks feature in the eigenvalue chart: `Cisco`, `Microsoft`, `American Express` and `General Electric`. While the technological stocks play a major role in the early years of the XXI century, the services stocks assume a predominant place by 2006 that spans the following decade. Regarding betweenness, one can observe that `General Electric` is predominant until the turn of the century. A period of low values follows, hinting a highly disconnected network comprised of small communities. By 2008, three stocks assume alternately roles of gate-keeping, with particular relevance to `American Express`. `General Electric`, among others, sees its initial high closeness drop until 2000.

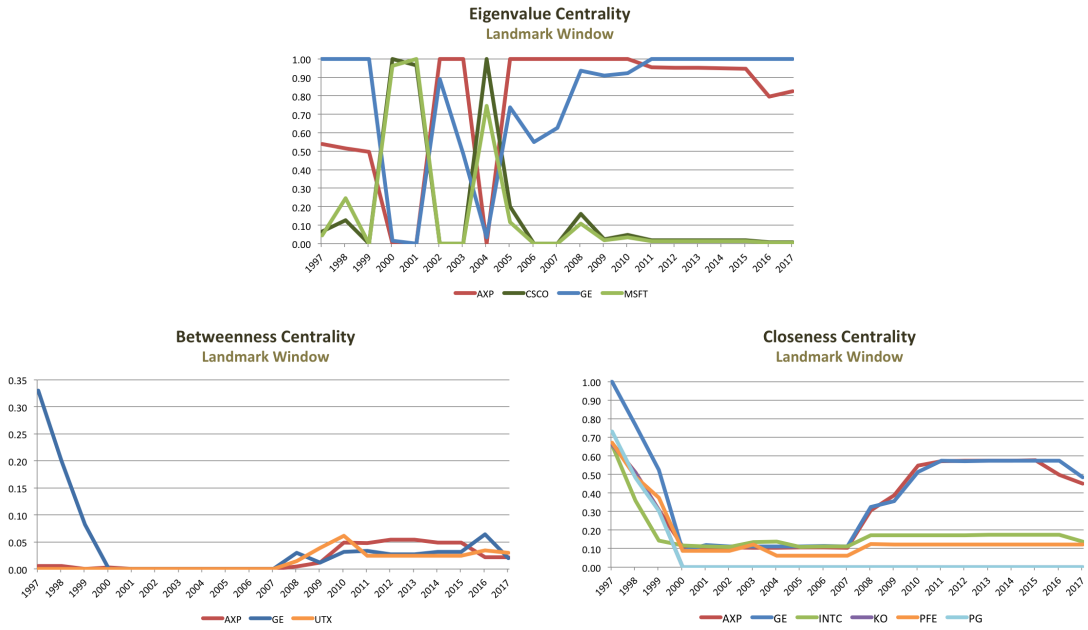
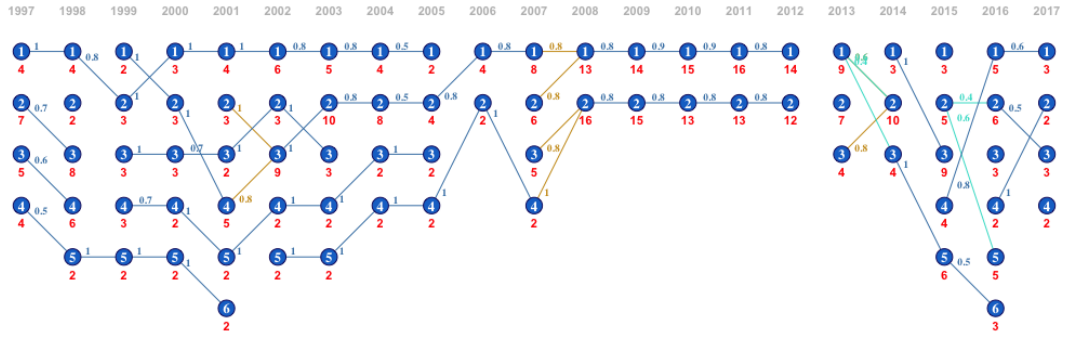


Figure 4.5: Centrality measures for most relevant stocks for networks obtained by the landmark window, annual snapshots from 1997 through 2017.



	Births	Merges	Splits	Survivals	Deaths
1997-2007	10	1	0	36	7
2008-2012	2	2	0	6	0
2013-2017	11	1	2	9	5
Total	**23**	**4**	**2**	**51**	**12**

Figure 4.6: MEC graph for gradual forgetting, annual snapshots from 1997 though 2017.

From that moment on, a period of low values extends for several years as nodes grow apart. By the end of 2007, the closeness of `American Express` and `General Electric` grows to medium values.

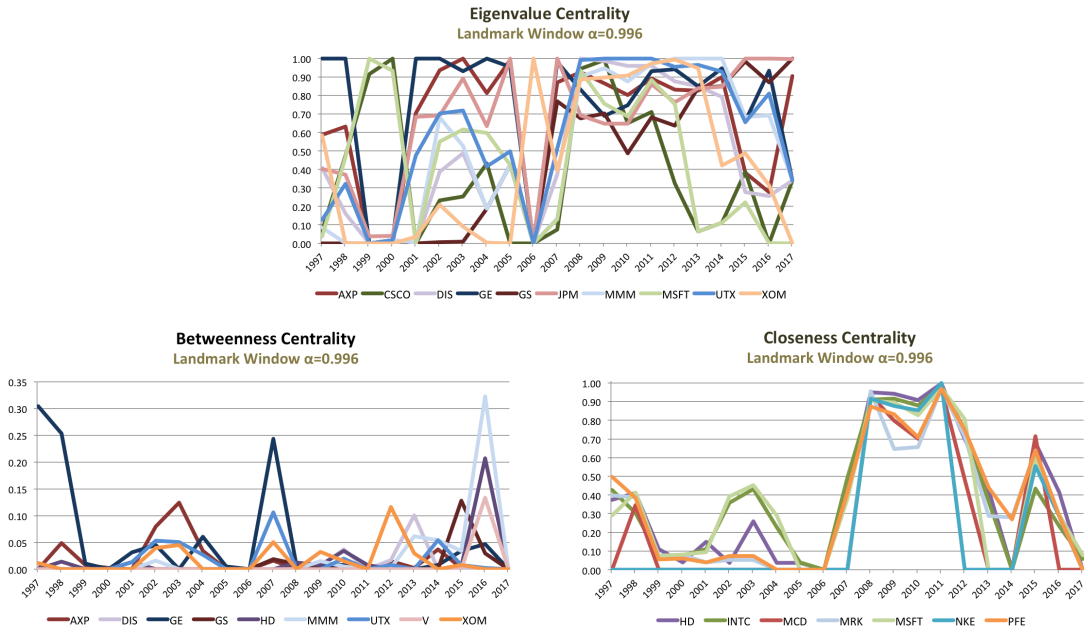
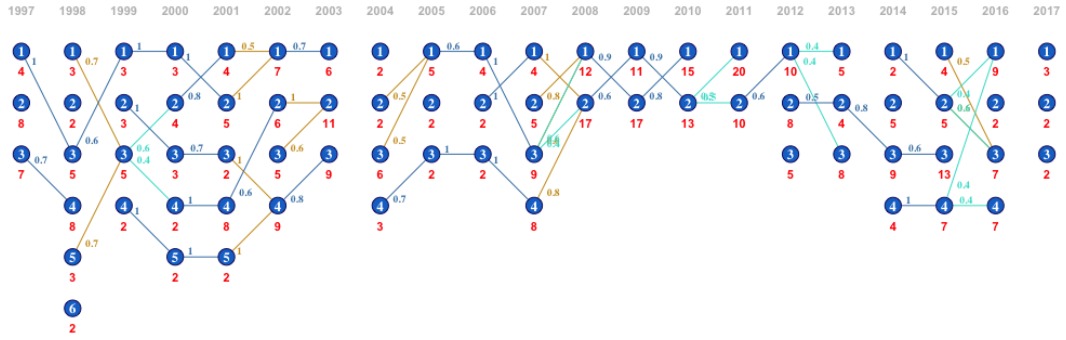


Figure 4.7: Centrality measures for most relevant stocks for networks obtained by gradual forgetting, annual snapshots from 1997 through 2017.

Communities, centralities and gradual forgetting Figure 4.6 illustrates the evolution of communities; members listed in Table C.2. The scenario is different from the baseline. Some merges and splits take place, the number of births and deaths increases considerably. There is still a sense of stability, but three distinct cycles are visible. The first, ranging from 1997 through 2007, exhibits communities with distinct life spans. Edges's weights prove high overlapping. The number of members is usually small (2-5), despite the existence of some larger ones. The second cycle, from 2008 through 2012, exhibits two communities that traverse the entire cycle and grow to a large number of members (13-15). The overlapping probabilities remain generally high. The third and last cycle, from 2013 onwards, is one of short lived, small communities (3-6 members). The weights of edges are usually near the survival threshold $\tau = 0.5$.

Centrality wise, the scenario in Figure 4.7 is also distinct from that of the baseline. The eigenvalue centrality is irregular, with several stocks reaching the highest value. `General Electric` keeps its initial influence and extends it into 2005, alternating with `Cisco` and `Microsoft` circa 2000 and `American Express` and `J.P. Morgan` from 2001 through 2005. The period between 2008 and 2014 is lead by `United Technologies`, `3M` and `Exxon`. The last years witness the comeback of financial companies, especially `J.P. Morgan`. Betweenness is simpler to interpret. Two stock clearly stand out: `General Electric` in the late nineties and circa 2007



	Births	Merges	Splits	Survivals	Deaths
1997-2007	15	5	1	20	9
2008-2012	2	2	2	4	0
2013-2017	13	1	3	6	7
Total	**30**	**8**	**6**	**30**	**16**

Figure 4.8: MEC graph for sliding windows, annual snapshots from 1997 though 2017.

and `3M` in the 2015-2017 period. In between, `American Express`, `Exxon`, `Goldman Sachs`, `Disney` and `Home Depot` play important but more discrete roles. Closeness centrality decreases until 1999. `Intel` and `Microsoft` share a leading role between 2001 and 2005. In 2006, closeness soars for all stocks, staying high until 2011 and plummeting afterwards. By 2014, only `Merck` and `Pfizer` have some degree of closeness to the remaining stocks. A new peak takes place in 2015, lead by `McDonalds` and `Home Depot`.

Communities, centralities and sliding windows Figure 4.8 illustrates the evolution of communities; members listed in Table C.3. It is the least stable scenario in terms of community survival. Three cycles are again visible. In the first cycle, from 1997 through 2007, communities are generally small (2-6 members) and short-lived. The overlap probability is usually near the survival threshold $\tau = 0.5$, but increases towards the end. The second cycle, from 2008 through 2012, shows communities merging into large structures (10-16 members) with high overlap probability. The third cycle, from 2013 through 2017, is highly unstable. With few exceptions, communities' birth and death occurs within the same year; exception are short-lived. Communities are general small (3-7) and get smaller towards the end.

The centrality measures are illustrated by Figure 4.9. Eigenvalue centrality remains unstable, with predominance of a few stocks: `General Electric` in the late nineties and early two-thousands, `J.P. Morgan` and `Goldman Sachs` between 2000 and 2007, and `United Technologies` and `3M` between 2008 and 2015. `General Electric` shows high betweenness until 1999, then fades away to make a moder-

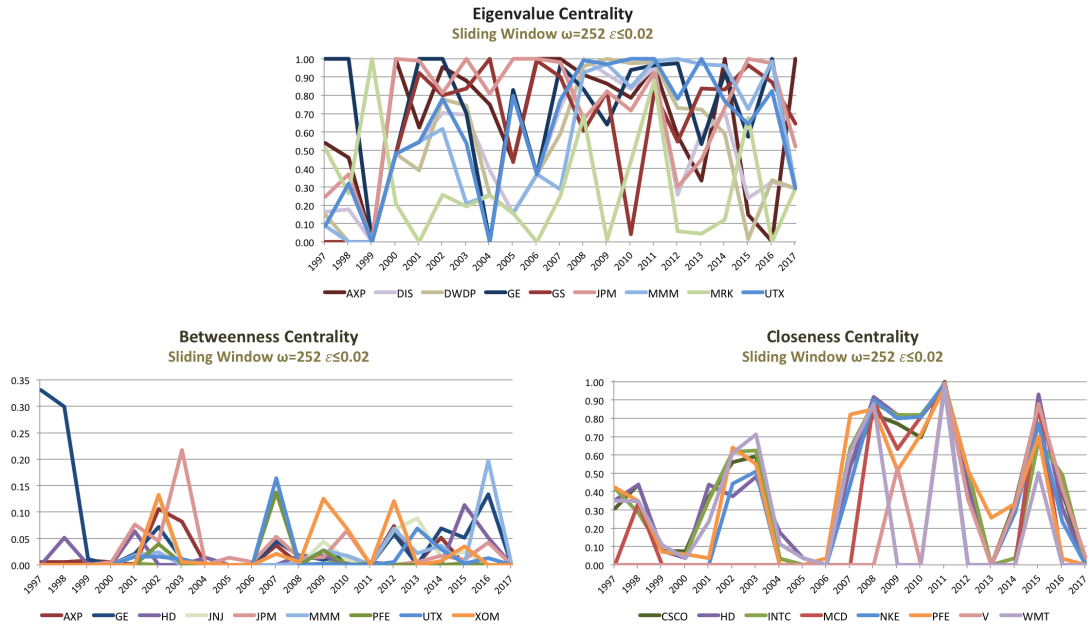


Figure 4.9: Centrality measures for most relevant stocks for networks obtained by sliding windows, annual snapshots from 1997 through 2017.

ate come back by 2015. Other important gatekeepers are `J.P. Morgan` by 2003, `Pfizer` and `United Technologies` by 2007, `Exxon` by 2009 and 2012 and `3M` by 2016. Four closeness peaks are visible. The first occurs during 2002-2003, spearheaded by `Pfizer`, `Intel`, `Cisco` and `Walmart`. Another peak takes place by 2008, with seven stocks reaching the top value; a plateaux of two years follows. A new peak occurs in 2011, followed by a general drop, interrupted by the last peak in 2015.

Interpretation of results

The results of the experience show that gradual forgetting and sliding windows come to similar results:

- The overall trends for networks measures are similar. Average degree and density show similar patterns, with very high values between 2007 and 2012. Modularity generally oscillates between 0.3 and 0.7 during the first decade but drops considerably from 2008 through 2012 to recover by 2015;
- Both MEC charts detect three distinct economical cycles that generally coincide in the start and end years. The first cycle, from 1997 through 2007, exhibits many communities, small and long-lived. The second, from 2008

through 2011, exhibits few communities, large and long-lived. The third, from 2012 through 2017, exhibits many communities, small and short-lived;

- The number of stock to which both models agree to assign relevant centrality values is significant. A comparison using the Jaccard index finds similarities of 0.62, 0.55 and 0.5 for eigenvalue, betweenness and closeness centrality sets.

4.2.3 Third experience: Remembering statistics for longer

The goal of this experience is to study the influence of a longer retention in memory of statistics over communities and centralities. To this purpose, we increase the fading factor and the size of the sliding windows. The sampling rate is adjusted to match the size of the sliding windows. Henceforth, we refer to these settings as *longer memory*. For this experience, the following parameters are defined:

- The baseline landmark window spans the entire period of 20 years, with a data point acquired every $t = 1$ trading days;
- A fading factor of $\alpha = 0.998$ is imposed over a landmark window similar to the baseline;
- Exponential histograms with admissible relative error $\epsilon \leq 0.02$ are built on top of sliding windows of $\omega = 504$ trading days, displaced every $\Delta t = 1$ trading days;
- A positive thresholds $\theta = 0.5$ such that $\rho \geq \theta$ is used to filter noise out of networks;
- The survival threshold τ is set to 0.5, the split threshold λ is set to 0.4;
- Communities with a single member are disregarded while building the MEC graphs;
- To study the dynamics of eigenvalue centrality, we impose a threshold of 1;
- To study the dynamics of betweenness centrality, we impose a threshold of 0.1;
- To study the dynamics of closeness centrality, we pick the top 10% higher values.

Results

Figure 4.10 illustrates the evolution of the average degree, density, modularity and the number of communities by the end of every second year. For reference, we show the measures obtained in section 4.2.2, exhibited as dimmed lines. Table

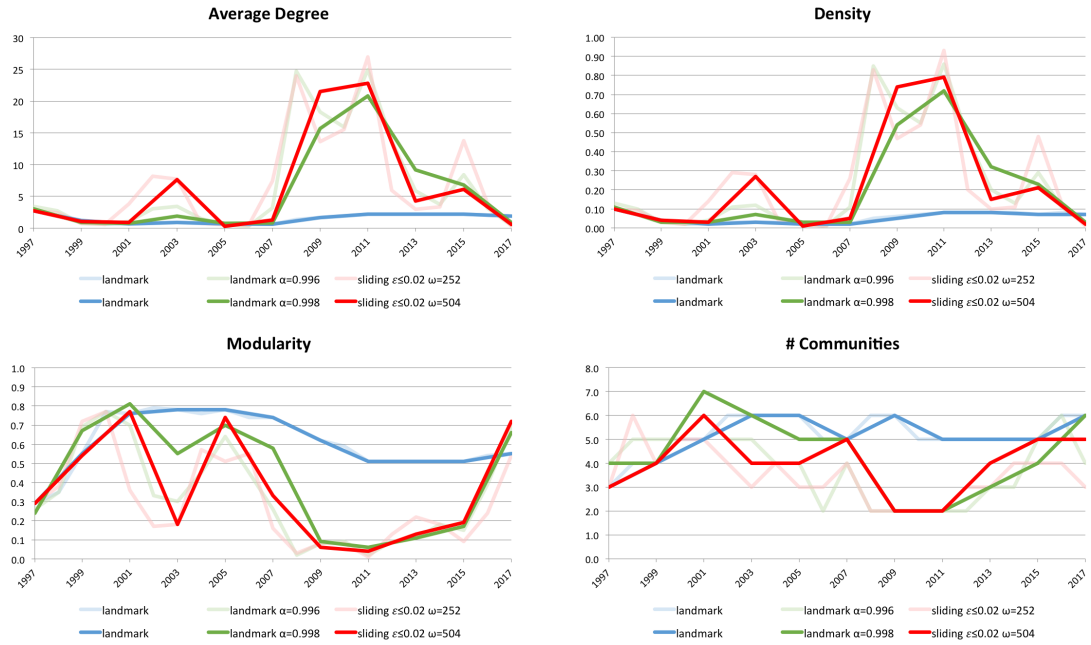


Figure 4.10: Average degree, density, modularity and number of communities for all window models, biennual values from 1997 through 2017

B.4 holds the data relative to the charts. Overall, the values obtained for all measures, in all window models, are similar those found in previous experience. Trends are in general smooth. The difference between gradual forgetting and sliding windows in terms of concept change detection is very pronounced, especially in the first decade, where the reaction of gradual forgetting is evidently slower.

For each window model, we take a snapshot every 504 days, at the end of every second year, for a total of 10 snapshots in the period between 1997 and 2017. The snapshots are used to build the MEC graphs. We also collect samples of the centrality measures at those instants.

Communities, centralities and landmark window Figure 4.11 illustrates the evolution of communities; members listed in Table C.4. One can observe stable and long-lived communities, with high overlapping probabilities, and very small number of members (2-4) throughout the entire period.

Figure 4.12 illustrates the centrality measures. The eigenvalue centrality charts shows three influential stocks: `General Electric` until 1999 and after 2011, `Microsoft` by 2001, and `American Express` from 2003 through 2009. Betweenness is shared by `General Electric` (GE) and `American Express`: the first leads

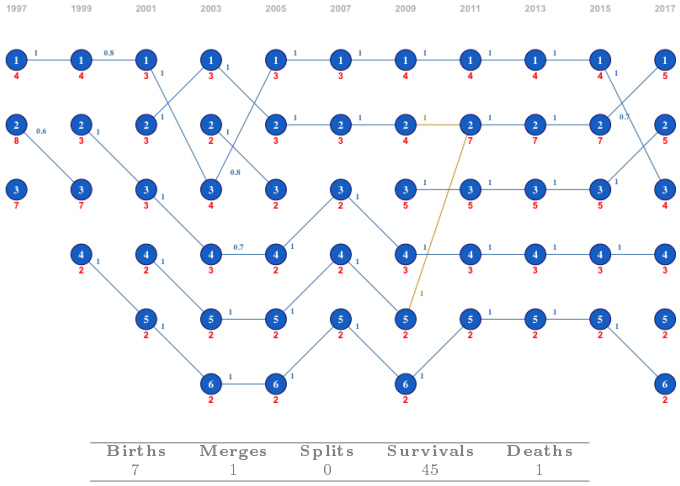


Figure 4.11: MEC graph for landmark window, biennual snapshots from 1997 though 2017.

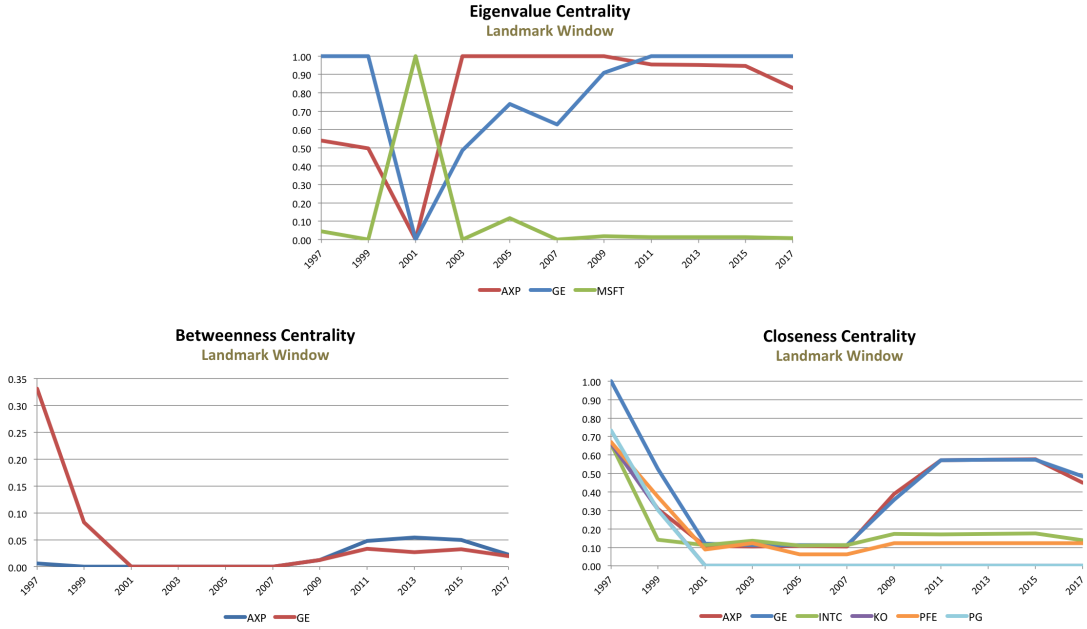
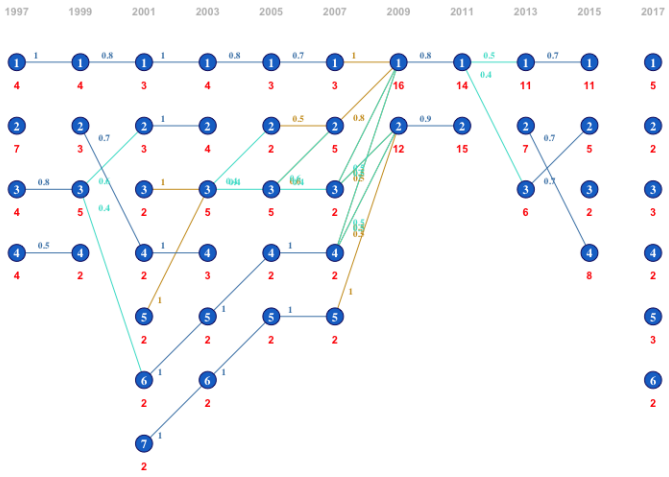


Figure 4.12: Centrality measures for most relevant stocks for networks obtained by landmark window, biennual snapshots from 1997 through 2017.

until 2001, the second predominates from 2009 onwards. Overall, closeness drops until 2001; `General Electric` having the highest values. A period of low closeness follows until 2007, when `Pfizer` and `General Electric` make a strong comeback. By 2015, these stocks begin to slowly drift away again.



	Births	Merges	Splits	Survivals	Deaths
1997-2007	5	2	3	16	2
2009-2011	3	2	2	2	0
2013-2017	6	0	1	3	3
Total	**14**	**4**	**6**	**21**	**5**

Figure 4.13: MEC graph for gradual forgetting, biennual snapshots from 1997 though 2017.

d

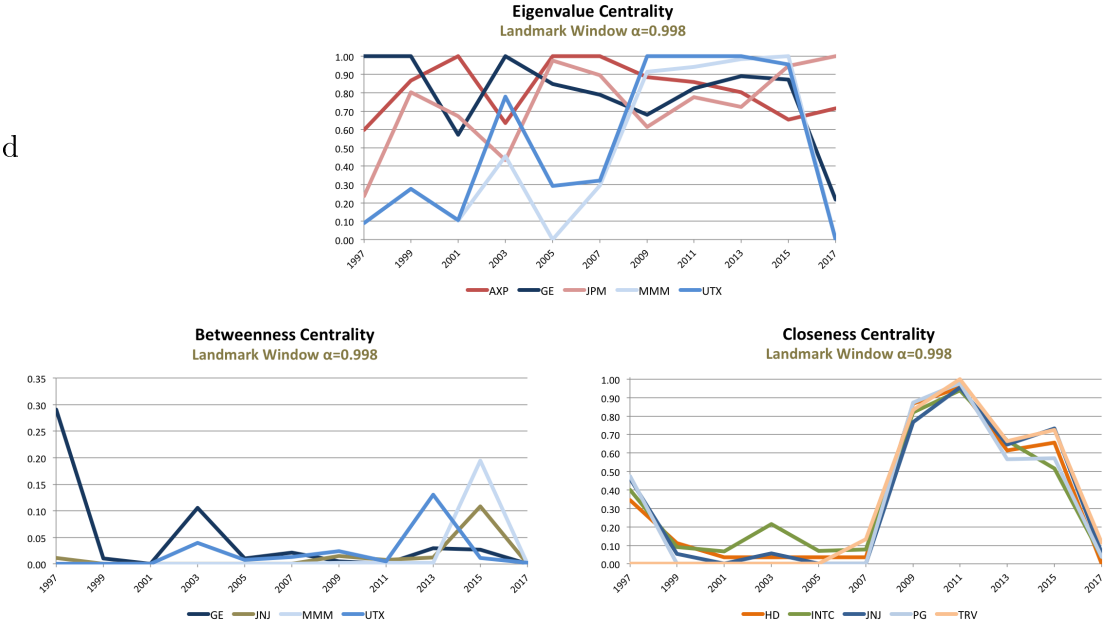
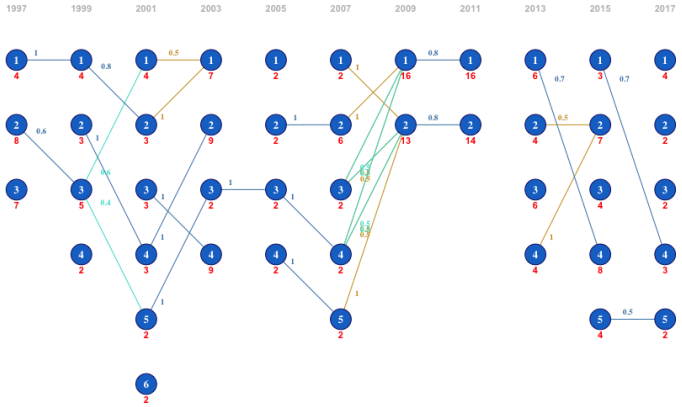


Figure 4.14: Centrality measures for most relevant stocks for networks obtained by gradual forgetting, biennual snapshots from 1997 through 2017.

Communities, centralities and gradual forgetting The MEC graph in Figure 4.13 illustrates the evolution of communities; members listed in Table C.5.



	Births	Merges	Splits	Survivals	Deaths
1997-2007	8	1	1	11	6
2009-2011	3	2	2	2	2
2013-2017	7	1	0	3	4
Total	**18**	**4**	**3**	**16**	**12**

Figure 4.15: MEC graph for sliding windows, biennual snapshots from 1997 though 2017.

There are three distinct cycles, with some splits and merges taking place. The first cycle, from 1997 through 2007, exhibits small communities (2-7 members) with distinct life cycles, the edges' weights showing high overlap probability. The second cycle, from 2009 through 2011, sees the merging of communities into two large structures (12-16 members) with high overlap probabilities. The third cycle, from 2013 through 2017, is comprised of short-lived, small communities (2-7 members), with overlap probabilities below the survival threshold most of the times.

Figure **??** shows the evolution of centrality metrics. During the first decade, eigenvalue centrality alternates between `General Electric`, `American Express` and `J.P. Morgan`. By 2009, `3M` and `United Technologies` assume command until 2015. The final years witness a comeback of financial companies, particularly of `J.P. Morgan`. Three stocks exhibit high betweenness: `General Electric` until 1999 and by 2003, `United Technologies` by 2013 and `3M` by 2015; `Johnson & Johnson` has a discrete contribution by 2015. The closeness of Proctor & Gamble, `Johnson & Johnson`, `Intel` and `Home Depot` drops until 2001. `Intel` manages to recover its position up until 2005. Closeness soars in the 2007-2009 period due to high connectivity and short paths in the network. The high values smoothly build to a peak in 2011 and then plummet to new lows until 2017, with a small inflexion by 2015.

Communities, centralities and sliding windows Figure 4.15 illustrates the evolution of communities; members listed in Table C.6. Once again, three cycles

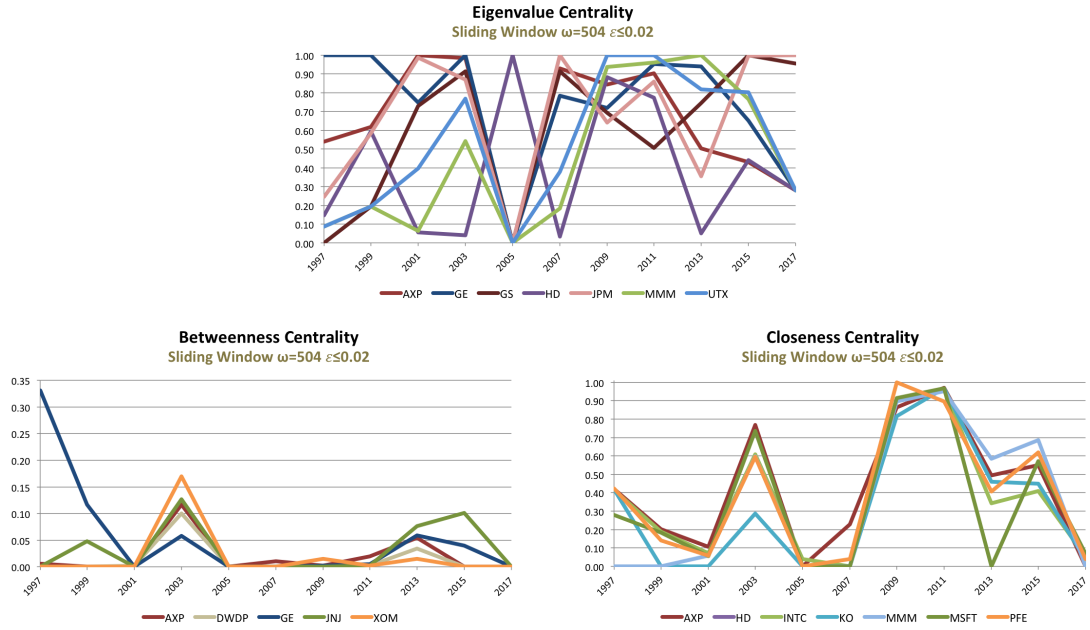


Figure 4.16: Centrality measures for most relevant stocks for networks obtained by sliding windows, biennual snapshots from 1997 through 2017.

occur. From 1997 through 2007, communities are small (2-5 members), long-lived and have moderate/high overlap probability. The second cycle, from 2009 up to 2011, is comprised of medium sized communities (14-15); the overlap probability is high throughout the entire cycle. The third cycle, from 2013 through 2017, shows small (2-7) short-lived communities, getting smaller towards the end of the cycle.

The centrality measures for this memory model are illustrated by Figure 4.16. Several stocks take turns in terms of influence. Circa 2001, `General Electric` gives place to `American Express` and briefly to `J.P. Morgan`. In 2003, `Home Depot` assumes leadership briefly. Financial stocks come bake strong between 2005 and 2007, but lose influence to `United Technologies` by 2009 and `General Electric` by mid 2011. By 2015, `J.P. Morgan` and `Goldman Sachs` assume leadership again. `General Electric` shows high betweenness until 2001. Circa 2003, several stocks lead by `Exxon` exhibit high betweenness. Minimum values are hit between 2005 and 2011, when `Johnson & Johnson`, `American Express` and `General Electric` make a comeback; the first peaking by 2015. The closeness chart shows three peaks of different intensity. The first occurs in 2002-2003, when `American Express` and `Microsoft`, closely followed by `Pfizer` have the highest closeness in the network. In 2008, another peak takes place with several stocks reaching high value. A smooth increase follows until 2011, when general closeness starts to drop until 2017, exception made for the inflexion lead by `3M` by 2015.

Interpretation of results

The results of the experience show that gradual forgetting and sliding windows come to similar results:

- The overall trends for networks measures are similar but the difference in concept change detection is very pronounced, especially in the first decade, where the reaction of gradual forgetting is evidently slower. Average degree and density show similar patterns, with very high values between 2007 and 2013. Modularity generally oscillates between 0.3 and 0.7 during the first decade but drops considerably from 2008 through 2012 to recover by 2015;
- Both MEC charts detect three distinct economical cycles that generally coincide in the start and end years. The first cycle, from 1997 through 2007, exhibits many communities, small and long-lived. The second, from 2009 through 2011, exhibits few communities, large and long-lived. The third, from 2013 through 2017, exhibits many communities, small and short-lived;
- The number of stock to which both models agree to assign relevant eigenvalue centrality values is significant. A comparison using the Jaccard index finds similarities of 0.57. The same does not happen for betweenness and closeness sets, whose Jaccard indices are 0.29 and 0.20 respectively.

4.2.4 Forth experience: Forgetting statistics faster

The goal of this experience is to study the influence of a shorter retention in memory of statistics over communities and centralities. To this purpose, we decrease the fading factor and the size of the sliding windows. The sampling rate is adjusted to match the size of the sliding windows. Henceforth, we refer to these settings as *shorter memory*. For this experience, the following parameters are defined:

- The landmark window spans the entire period of 20 years, with a data point acquired every $t = 1$ trading days;
- A fading factor of $\alpha = 0.992$ is imposed over a similar landmark window;
- Exponential histograms with admissible relative error $\epsilon \leq 0.02$ are built on top of sliding windows of $\omega = 126$ trading days, displaced every $\Delta t = 1$ trading days;
- A positive thresholds $\theta = 0.5$ such that $\rho \geq \theta$ is used to filter noise out of networks;
- The survival threshold τ is set to 0.5, the split threshold λ is set to 0.4;

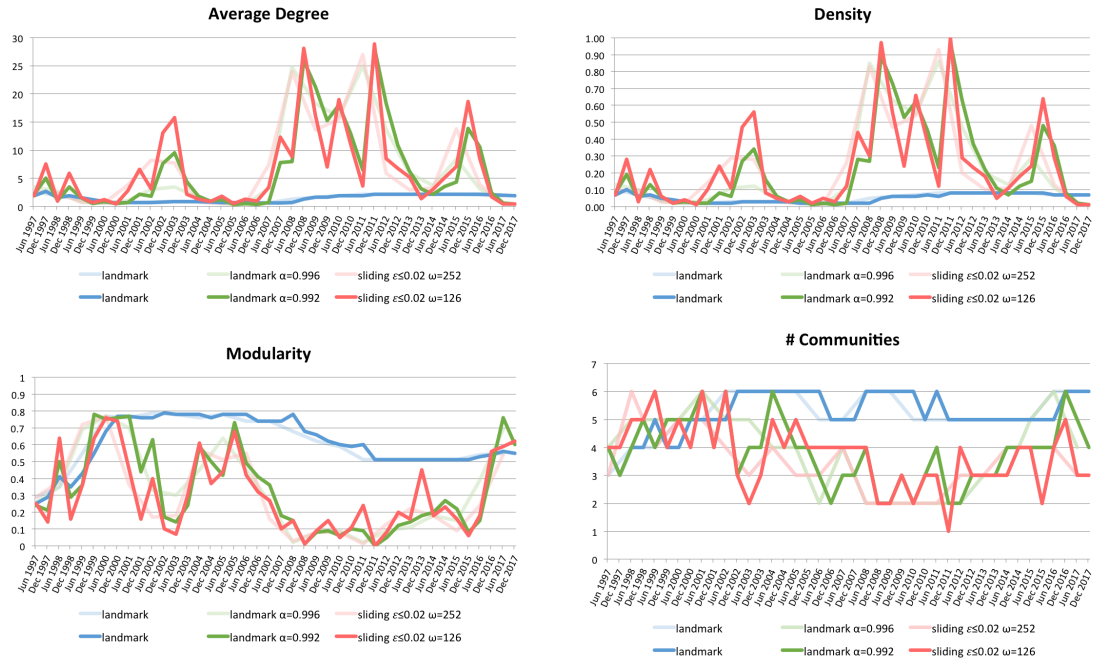


Figure 4.17: Average degree, density, modularity and number of communities for all window models, semestral values from 1997 through 2017

- Communities with a single member are disregarded while building the MEC graph;
- To study the dynamics of eigenvalue centrality, we impose a threshold of 1;
- To study the dynamics of betweenness centrality, we impose a threshold of 0.1;
- To study the dynamics of closeness centrality, we pick the top 10% values.

Results

Figure 4.17 illustrates the evolution of the average degree, density, modularity and the number of communities by the end of every semester. For reference, we keep the measures obtained in section 4.2.2, exhibited as dimmed lines. Table B.5 holds the data relative to the charts. Overall, the values obtained for all measures, in all window models, are similar to those found in previous experiences. The lines also appear more jagged, as the response to changes in market dynamics is faster. The difference between gradual forgetting and sliding windows in concept change detection is still evident but less intense than ever.

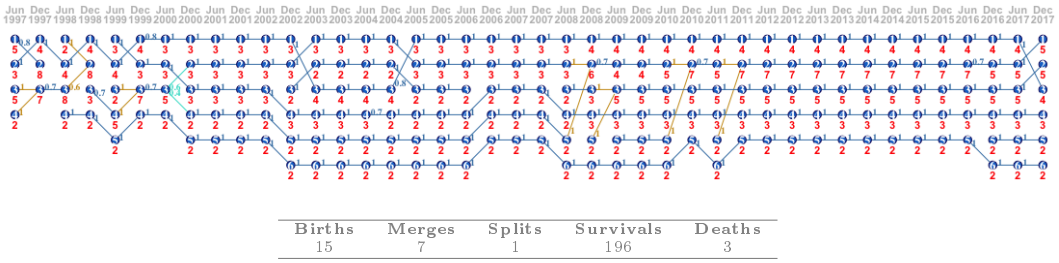


Figure 4.18: MEC graph for landmark window, semestral snapshots from 1997 though 2017.

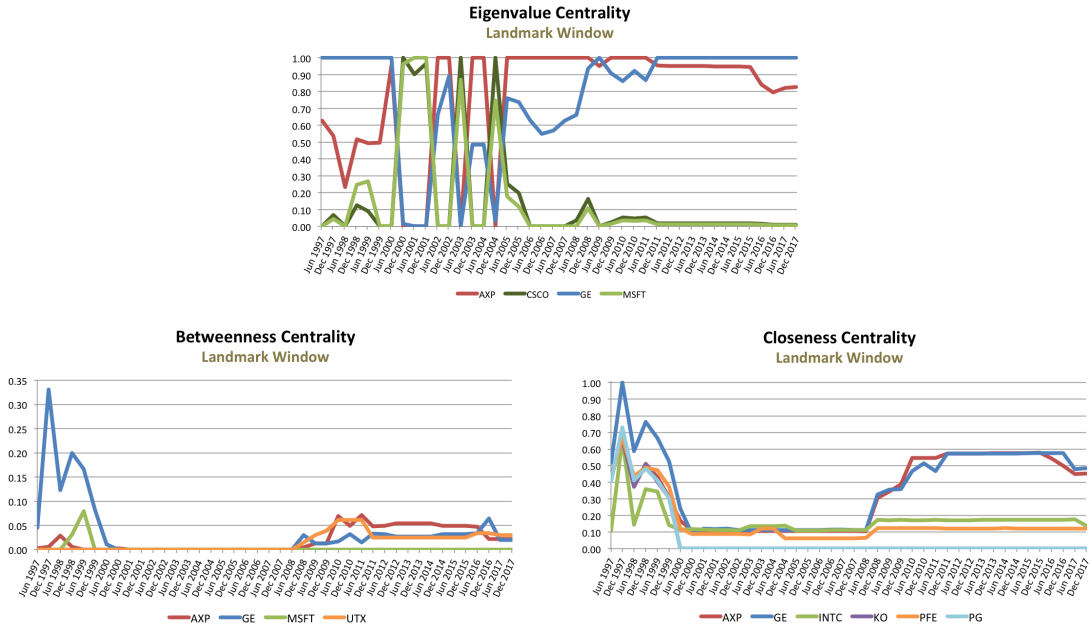
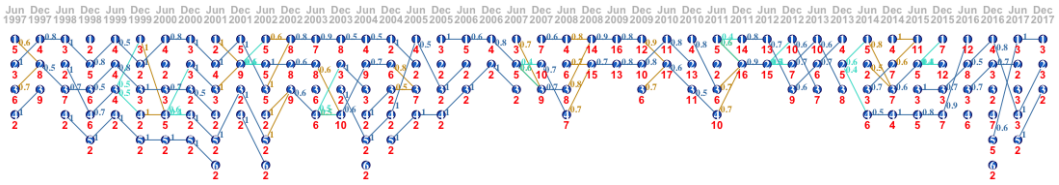


Figure 4.19: Centrality measures for most relevant stocks for networks obtained by landmark window, semestral snapshots from 1997 through 2017.

For each window model, we take a snapshot every 126 days, at the end of every semester, for a total of 40 snapshots in the period between 1997 and 2017. The snapshots are used to build the MEC graphs. We also collect samples of the centrality measures at those instants.

Communities, centralities and landmark window Figure 4.18 illustrates the evolution of communities; members listed in Table C.7. To no surprise, one finds stable and long-lived communities, with high overlap probability, and generally a very small number of members (2-4) throughout the entire period.



	Births	Merges	Splits	Survivals	Deaths
1997-2007	29	10	5	59	12
2008-2012	6	2	1	11	1
2013-2017	16	2	3	22	11
Total	**51**	**14**	**9**	**92**	**24**

Figure 4.20: MEC graph for gradual forgetting, semestral snapshots from 1997 though 2017.

Figure 4.19 illustrates the centrality measures. The eigenvalue centrality lead is shared by four stocks. `General Electric` is the most influential stock until mid 2000 and after 2011, `Cisco` and `Microsoft` take turns during 2001, and `American Express` leads from mid 2005 through 2011 with a small meddling by `General Electric`. Betweenness is lead by `General Electric` until mid 2000 and `American Express` from mid 2010 onwards, with a brief help of `United Technologies` from 2009 through 2011. `General Electric` exhibit the highest values of closeness until 2001. A period of low closeness follows until mid 2008, when `American Express` and `General Electric` start a strong recovery that stabilizes by June 2011 and shows a small decline again by 2016.

Communities, centralities and gradual forgetting Figure 4.20 illustrates the evolution of communities; members listed in C.8. Three distinct cycles are visible. The first cycle, from 1997 through mid 2008, exhibits small communities (2-6 members) with mixed life cycles and medium/high overlap probabilities. The second cycle, from late 2008 through late 2011 exhibits long-lasting, large communities (9-14 members); overlap probabilities stays high. The third cycle, from mid 2012 through late 2017, is comprised of short-lived, small communities (3-7 members), with medium overlap probabilities.

Figure 4.21 shows the evolution of centrality measures. Several stocks exhibit high eigenvalue centrality simultaneously throughout the entire period, exception made for `General Electric` until 1999 and during 2000-2001. Betweenness highest value is shared by `General Electric`, `3M`, `ExxonJ.P. Morgan` in several distinct occasions. The closeness chart exhibits 9 different stocks peaking simultaneously for 7 times, three of those time by the end of 1997, 1998 and mid 2003. Closeness soars in the period of 2007-2010, drops in early 2011 to soar again by the end of that year and plummet again from 2012 through 2013, exception made for Goldman Sachs that drops slower than other. A last peak is visible by late 2015.

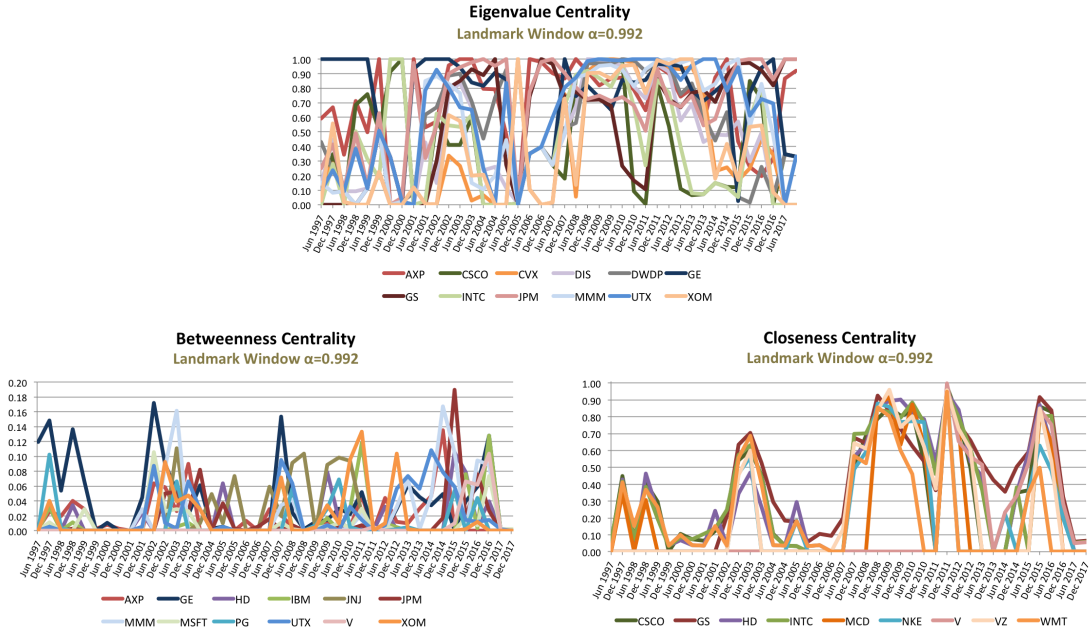
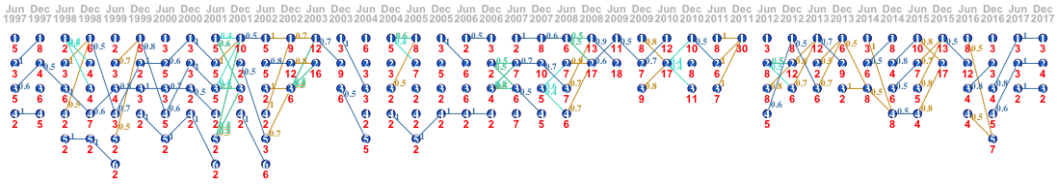


Figure 4.21: Centrality measures for most relevant stocks for networks obtained by gradual forgetting, semestral snapshots from 1997 through 2017.



	Births	Merges	Splits	Survivals	Deaths
1997-2007	44	13	10	41	24
2008-2012	9	2	1	5	3
2013-2017	17	8	1	14	6
Total	**70**	**23**	**12**	**60**	**33**

Figure 4.22: MEC graph for sliding windows, semestral snapshots from 1997 though 2017.

Communities, centralities and sliding windows Figure 4.22 illustrates the evolution of communities; members listed in Table C.9. The usual three cycles occur. The first cycle, from 1997 through mid 2008, is comprised of short-lived, small communities (2-6 members) with medium/high overlap probability. The second cycle, from 2009 through 2011, is comprised of medium sizes communities (8-14) with high overlap probabilities. The third cycle, from 2013 through 2017, shows unstable, small communities (3-8 members) with a very high frequency of births and deaths; the overlap probability is generally small and gets smaller towards the end.

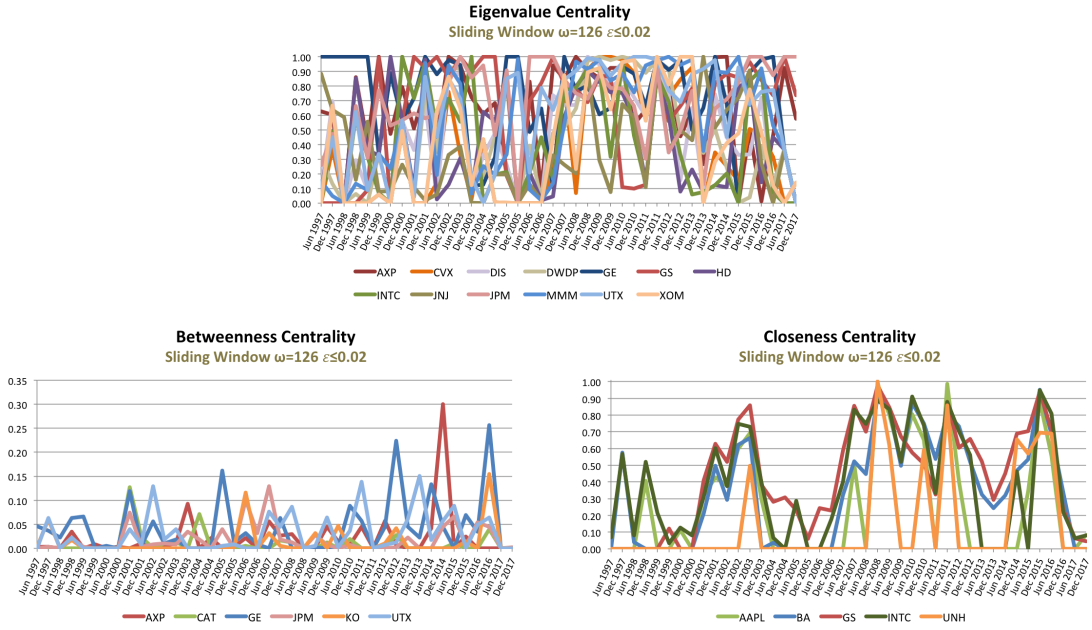


Figure 4.23: Centrality measures for most relevant stocks for networks obtained by sliding windows, semestral snapshots from 1997 through 2017.

The centrality measures are illustrated by Figure 4.9. The eigenvalue centrality chart depicts a scenario of high instability, exceptions made for `General Electric` until 1999 and `J.P.Morgan` from mid 2006 through 2007. Betweenness is also very unstable, with many stocks becoming important gatekeepers in the studied period. Two stocks stand out: `General Electric` and `American Express`. Closeness centrality is no different from the other two measures, with several peaks between 1997 and 2017, none lasting more than a quarter. Two stocks appear frequently in the top values: `Intel` and `Goldman Sachs`.

Interpretation of results

The results of the experience show that gradual forgetting and sliding windows come to similar results:

- The overall trends for networks measures are similar.The difference in concept change detection is less pronounced, despite the slower reaction of gradual forgetting. Average degree and density show similar patterns, with very high values by 2002, 2008, 2010, 2011 and 2015. Modularity generally stays above 0.3 during the first decade but lower values are often observed in that period, especially in 2002-2003. Modularity also drops considerably from 2008 through

2012 to recover by 2015. The reaction to market changes is fast and charts show jagged lines;

- Both MEC charts detect three distinct economical cycles that generally coincide in the start and end years. The first cycle, from 1997 up to 2007, exhibits many communities small and long-lived. The second, from 2008 up to 2011, exhibits few communities, large and long-lived communities. The third, from 2012 up to 2017, exhibits many communities, small and short-lived;

- The number of stock to which both models agree to assign relevant eigenvalue centrality values is significant. A comparison using the Jaccard index finds similarities of 0.79. The same does not happen for betweenness and closeness sets, whose Jaccard index is 0.30 and 0.09 respectively.

4.3 Discussion

We move to the discussion of the obtained results, comparison the different memory settings, making a parallel with major economical, financial and historical events taking place during the studied period. Additionally, we disclose the frequent communities, which influential stocks take place in those clusters and how does this relate to DJI's performance from 1997 through 2017.

4.3.1 Comparison of the different memory settings

We elaborate on the advantages and disadvantages of keeping statistics in memory for different time spans. A proper setting of this parameter, one that best fits the natural evolution of market dynamics while keeping the analysis of result easy to understand ans assimilate, is of paramount importance.

A comparison of the results collected from the different settings reveals a common trend that comes across in all tables and charts: the capacity to retain events in memory for longer periods induces a simplification of scenarios for all window models. The overall tendencies in the MEC graphs, node centrality, community and network measures are similar but the level of detail increases proportionally to the forgetting capacity. Longer memory prompts smoother charts and more stable communities, shorter memory prompts jagged charts and more instability in the MEC graphs. Other than this, one can draw the following conclusions:

- The landmark window is the least affected of all. Since it keeps all events in memory, correlations converge to the population value as the sample size increases (Cohen, 1977; Schönbrodt and Perugini, 2013). Therefore, the evolution of the network and the formed communities are similar enough to say that no major gain in information is attainable by probing the network more

frequently. There is, however, some gain in detail on the study of influential stocks, as the number of stock and richness in patterns of interaction increase in direct proportion to the sampling rate;

- The MEC graphs produced out of networks based on gradual forgetting exhibit similar enough shapes for all three fading factors. There are always evidences of three distinct cycles: 1997-2007, 2008-2012, 2013-2017. In general, the number of communities in the second cycle is smaller than in the other two; often there are only 2 communities during this time period. Merging and splitting phenomena are common. The overlap probability drops in the third cycle;

- Similar evidences are visible in the MEC graph of sliding windows. The produced MEG graphs exhibit similar enough shapes for all three window sizes. There are also evidences of three distinct cycles: 1997-2007, 2008-2012, 2013-2017. In general, the number of communities in the second cycle is smaller; often there are only 2 communities during this time period and, exceptionally, a single community is detected by the smallest window model by the end of 2011. Merging and splitting phenomena are common as well. The overlap probability also drops in the third cycle;

- Longer memory produces a shorter number of stocks with high eigenvalue centrality. No correlation is observed for other centralities;

- All models, in all memory settings, assign high eigenvalue centrality to `American Express` and `General Electric`. Gradual forgetting and sliding windows agree to highlight `J.P. Morgan` and `United Technologies`. Landmark window always highlights `Microsoft`, gradual forgetting always highlights `3M`, sliding windows always highlight `Goldman Sachs`;

- All models, in all memory settings, assign high betweenness centrality to `General Electric`. Landmark windows and sliding windows agree to highlight `American Express`. Gradual forgetting always highlights `3M` and `United Technologies`;

- All models, in all memory settings, assign high closeness centrality to `Intel`. Gradual forgetting always highlights `Home Depot`. Landmark window always highlights `Pfizer`, `American Express`, `General Electric`, `Coca-Cola` and `Proctor & Gamble`;

- Overall, the same essential information about market dynamics is kept. A shorter memory reveals more detail, but make the overall tendencies harder to determine and interpret. A longer memory promotes generalization and loss of information.

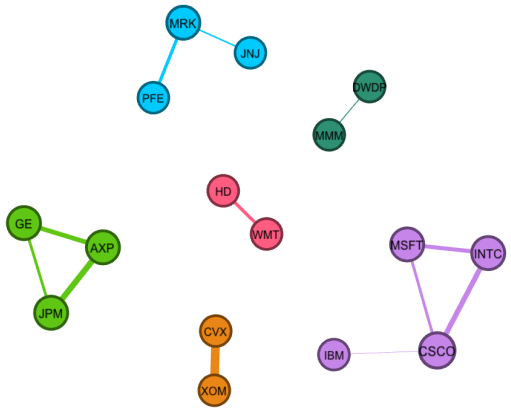


Figure 4.24: Communities detected in correlation networks built using the landmark window

4.3.2 Frequent communities, its members and sectors of activity

The MEC framework helps us to understand how communities are born, evolve and die, how members join, stay and leave different communities over time. However, it is not as helpful in determining the most common associations of stocks, either as communities or its sub-components. Moreover, gives little information on how many times a community is reborn; to obtain such information, one must observe every new-born community and try to match its members to a dying community in a previous (possibly remote) snapshot.

We seek to know how frequently do communities or its sub components occur and if there are common patterns emerging in the networks generated resorting to the different window models. To that purpose, we we apply a technique called frequent pattern mining (Agrawal et al., 1993) over the communities described in Tables C.1 to C.6. We address each community as a single transaction and determine frequent stock-sets. We then cross this information with the paths in MEC graphs to learn the history of the most common stock-sets. We focus on the communities obtained in the experiment described in Section 4.2.2. The full lists of frequent stock-sets presented in Appendix D.

Frequent communities and landmark windows

The landmark window helps to detect mostly small, stable, sector-based communities; examples are illustrated in Figure 4.24.

Technological stocks `Cisco`, `Intel` and `Microsoft` are in the same community for the entire period; `IBM` is also a member, but leaves during during short periods.

Two other stock-sets are also common: one is `Home Depot` and `Walmart` (ser-

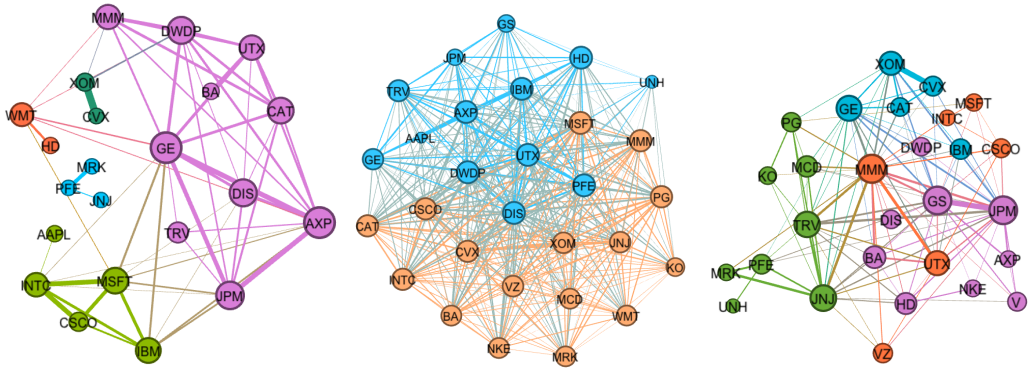


Figure 4.25: Communities detected in correlation networks built using gradual forgetting, during the first (a), second (b) and third (c) cycles

vices), the other is `General Electric` (cap. goods) and `American Express` and `J.P. Morgan` (financial). The two groups, initially together in the same community, split and evolve separately most of the time, merge again during the period of 2010 up to 2015. During this period, they are joined by `Disney` (services) and `Travelers` (financial).

`Chevron` and `Exxon`, both in energy, come together for 20 years. `Merck`, `Pfizer` and `Johnson & Johnson` (healthcare) join for 19 years; the last leaving for 4 years. Both energy and healthcare communities evolve alone, despite the fact that its members are initially together.

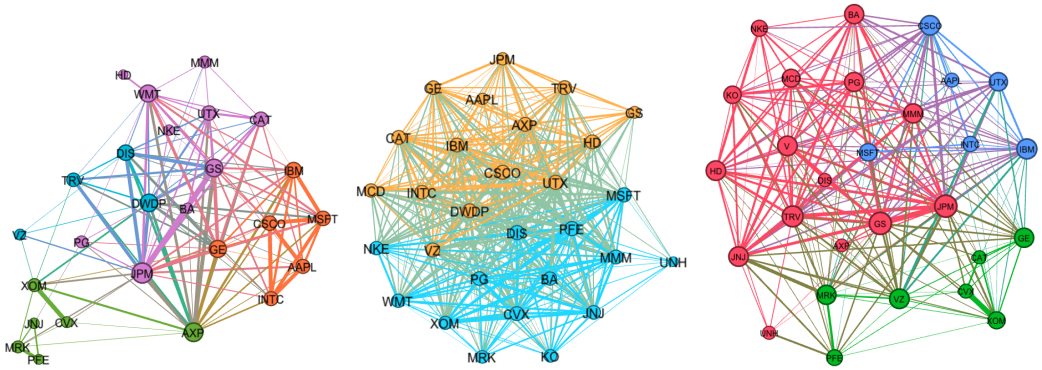
`Dupont` and `3M`, frequently together with `Caterpillar`, form a recurrent community, joined by `Boeing` and `United Technologies` join for half the studied period. All companies operate in the capital goods and basic materials sectors.

Frequent communities and gradual forgetting

Gradual Forgetting show mixed trends, framed within the three detected cycles as illustrated by Figure 4.25.

In the first cycle, Technological stocks `Cisco` and `Intel` are always together, occasionally joined by `IBM`, `Microsoft` and `Apple`; no other stocks join the cluster. `Chevron` and `Exxon` (energy) are also always together, with no other partners other than `Dupont` and `Proctor & Gamble` in the first year.

Two multi-sector communities form at first: one comprises `Johnson & Johnson`, `Merck` and `Pfizer` (healthcare), `Coca-Cola` (consumer/non-cyclical) and `Disney` (services), the other comprises `Home Depot` and `Walmart` (services - retail) and `Boeing`, `United Technologies` and `General Electric` (cap. goods). As the communities break up, the healthcare and retail companies form communities of their own; the financial cluster is often joined by `General Electric` and `United`



(a) (b) (c)

Figure 4.26: Communities detected in correlation networks built using sliding windows, during the first (a), second (b) and third (c) cycles

`Technologies`, among others.

Two large communities during the second cycle. The first is centred around `Cisco`, `IBM`, `Intel` and `Apple` (technological), `American Express`, `J.P. Morgan` and `Goldman Sachs` (Financial), `Boeing`, `United Technologies`, `3M` and `Caterpillar` (cap. goods) and `Dupont` (bas. materials).

The second's core is formed by `Pfizer`, `Proctor & Gamble`, `Johnson & Johnson` and `Merck` (healthcare), `Chevron` and `Exxon` (energy) and `Coca-Cola` (consumer/non-cyclical), `McDonalds` and `Verizon` (services).

The third cycle is marked by a mixed tendency between medium and small communities, mostly short-lived. `American Express`, `J.P. Morgan` and `Goldman Sachs` (Financial) keep together for some time, attracting several other companies along the way. `Chevron` and `Exxon` (energy), `General Electric` and `Caterpillar` (cap. goods) come together, briefly joined by the financial cluster.

`Boeing`, `United Technologies` and `3M` (cap. goods) briefly merge with the financial cluster, occasionally attracting `Intel` and `IBM` (technological). `Johnson & Johnson`, `Merck` and `Pfizer` (healthcare) regroup, occasionally joined by `Proctor & Gamble` and other companies in assorted sectors. By the end of the cycle, `Intel`, `Microsoft` and `Cisco` (technological) also regroup, joined by `Visa` (financial).

Frequent communities and gradual forgetting

Sliding windows exhibit the less stable patterns, framed in three cycles as illustrated in Figure 4.26.

In the first cycle, `Cisco`, `Intel` and `Microsoft` (technological) form a community

joined by Apple and IBM most of times; Goldman Sachs, J.P. Morgan, American Express (financial) and General Electric (cap. goods) briefly join in different occasions.

American Express and J.P. Morgan (financial) also come together in the first half of the cycle, joined at times by Goldman Sachs and General Electric, Home Depot and Walmart (cap. goods). American Express, J.P. Morgan and Goldman Sachs rejoin by the end of the cycle.

Home Depot and Walmart (cap. goods) form a community during the first and last thirds of the cycle; they are joined by American Express, Goldman Sachs and J.P. Morgan (financial) occasionally.

Boeing, Caterpillar, United Technologies and (cap. goods) come together episodically, joined by the pair Dupont (bas. materials) and Walmart (services), 3M and General Electric (cap. goods).

Johnson& Johnson, Merck and Pfizer (healthcare) are together for most of the cycle, occasionally joined by Proctor & Gamble (healthcare), Chevron and Exxon (energy) and Coca-Cola (Consumer/Non-cyclical).

Chevron and Exxon (energy) form a community that lasts the entire cycle, briefly merging with the healthcare cluster and General Electric (cap. goods) mid-cycle.

The second cycle is again defined by two large communities. The first is formed by Johnson & Johnson, Merck, Pfizer and Proctor & Gamble (healthcare), Coca-Cola (consumer/non-cyclical), McDonalds and Verizon (Services) and Chevron and Exxon (energy).

The second is formed by Apple, Cisco and Intel (technological), American Express, Goldman Sachs and J.P. Morgan (financial), Caterpillar and General Electric (cap. goods); it is joined by IBM and Microsoft occasionally, and Chevron, Exxon, Dupont and 3M by the end.

By the end of the cycle, the communities slit in three, as technological and financial stocks drift apart; capital goods companies stay together with financial ones.

In the third cycle, patterns seldom apply, with the exception of Chevron and Exxon, joined by Caterpillar at a given time. Boeing and United Technologies (services) are together at times in an aerospace cluster.

Goldman Sachs and J.P. Morgan (financial) are are also together sometimes, joined briefly by the aerospace cluster, American Express (financial), Disney (services), IBM and Intel (technological), Chevron and Exxon (energy) and Dupont and 3M (bas. materials and cap. goods).

Another community congregates Johnson & Johnson and Pfizer (healthcare) for in the first half of the cycle; they are commonly joined by Proctor & Gamble (healthcare) and briefly by Coca-Cola (consumer/non-cyclical), McDonalds, Home Depot and Walmart (services).

By the end of this cycle, Intel, Microsoft, Apple, Cisco and IBM (technologi-

cal) join together in two small communities that quickly converge, attracting `Nike` (consumer/cyclical), `Disney` (services) and `Visa` (financial) in the process.

Interpretation

Landmark windows often discloses stable communities of companies that have the same sector of activity, common interests or long-lasting commercial relationships. Conversely, gradual forgetting and sliding windows disclose a more dynamic scenario, with members shifting communities many times and frequent relationships between companies working in different sectors. Nevertheless, all models disclose a tendency for companies of the same sector to walk side-by-side, either as communities of their own or as as subcomponents of larger communities.

We illustrate these findings with associations rules derived from communities discovered in networks generated using landmark window (4.1), gradual forgetting (4.2) and sliding windows (4.3). The support, confidence and lift are presented within round brackets. The first rule shows the association of companies in the technological sector, the second shows the association of `3M` and `Boeing` (commercial relationship) and the third shows the association of `Coca-Cola` (consumer/non-cyclical), `Pfizer` and `Johnson & Johnson` (healthcare). All examples come from the experience described in Section 4.2.2. Additional rules and items-sets are presented in Appendix D.

$$\texttt{CSCO}, \texttt{IBM}, \texttt{MSFT} \Rightarrow \texttt{INTC} \quad (supp: 0.24,\ conf: 1.00,\ lift: 3.00) \tag{4.1}$$

$$\texttt{MMM} \Rightarrow \texttt{BA} \quad (supp: 0.18,\ conf: 0.77,\ lift: 3.37) \tag{4.2}$$

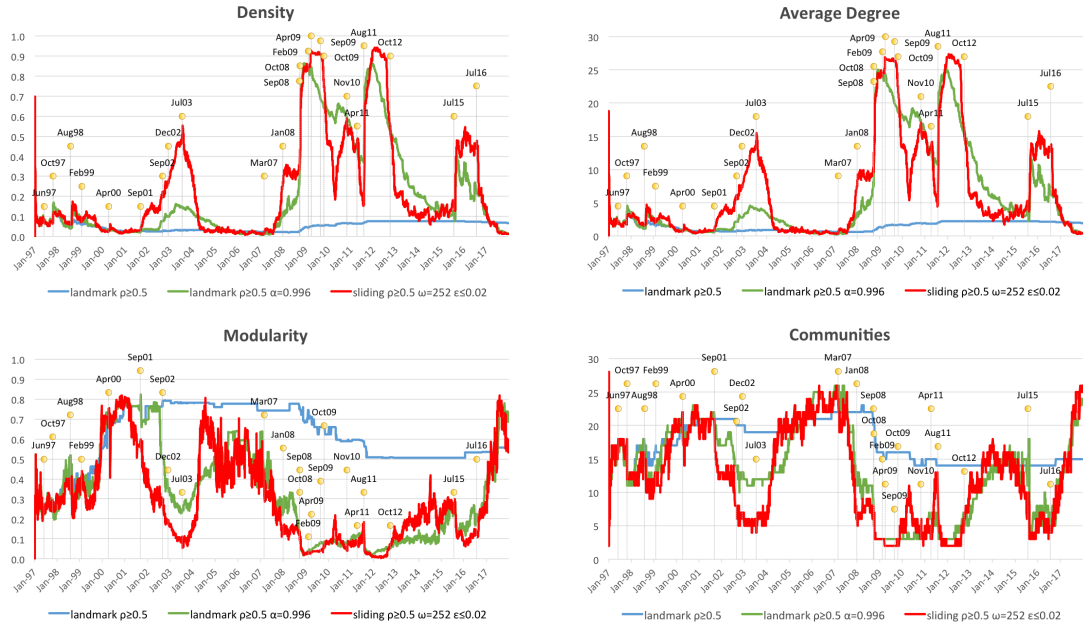
$$\texttt{KO}, \texttt{PFE} \Rightarrow \texttt{JNJ} \quad (supp: 0.15,\ conf: 1.00,\ lift: 4.54) \tag{4.3}$$

4.3.3 Adherence to financial market dynamics

We wish to validate whether our findings relate to the major economical, financial and political events that took place from 1997 through 2017. Figure 4.27 illustrates the evolution of density, average degree, modularity and number of communities for networks of correlations $\rho \geq 0.5$ enhanced with a time line of some important events.

Figure 4.27 shows that density and average degree raise in periods of great turbulence whereas modularity and the number of communities decrease in such times. Four clear cases of such behaviours are evident during the 2003 stock market panic, the 2008 stock market crash and U.S. banking crisis, the 2011 European sovereign debts crises and the 2015 Chinese stock market slowdown.

These results consistent with previous cases in the reviewed literature: correlation between financial turmoil and network's measures is reported in Onnela et al. (2003), Tse et al. (2010), Roy and Sarkar (2011), Heiberger (2014), Dimitrios and Vasileios (2015) and Vodenska et al. (2016). Authors refer to small,



Time	Event
Jun 1997	Asian financial crisis
Oct 1997	Hang Seng Index downturn
Aug 1998	Ecuador and Russia financial crisis
Feb 1999	Brasil samba effect
Apr 2000	DotCom bubble burst
Sep 2001	September 11 attacks
Sep 2002	02 stock market downturn
Dec 2002	Venezuelan oil strike
Jul 2003	03 stock market panic
Mar 2007	US subprime loan meltdown
Jan 2008	08 stock market downturn
Sep 2008	Stock Market crash
Oct 2008	US Banking Crisis
Feb 2009	Congress Economic Stimulus Plan
Apr 2009	Greece IMF/EU Intervention
Sep 2009	ObamaCare
Oct 2009	Bank of America loan program
Nov 2010	Ireland IMF/EU Intervention
Apr 2011	Portugal IMF/EU Intervention
Aug 2011	August 2011 stock markets fall
Oct 2012	Spannish banking bailout
Jul 2015	Chinese stock market turbulence
Jul 2016	Brexit referendum

Figure 4.27: Density, average degree, modularity and communities for networks of correlation levels $\rho \geq 0.5$ enhanced with the timeline of major economical, financial and political events.

stable communities in times of financial stability and economical growth. By contrast, the authors suggest contractions in stock networks, with increased correlation between stocks of different sectors, and fast changing communities taking shape.

The identification of influential stocks is of paramount importance (Roy and Sarkar, 2011; Heiberger, 2014), as their performance defines trends that propagate throughout the entire market. One can understand their contribution to market dy-

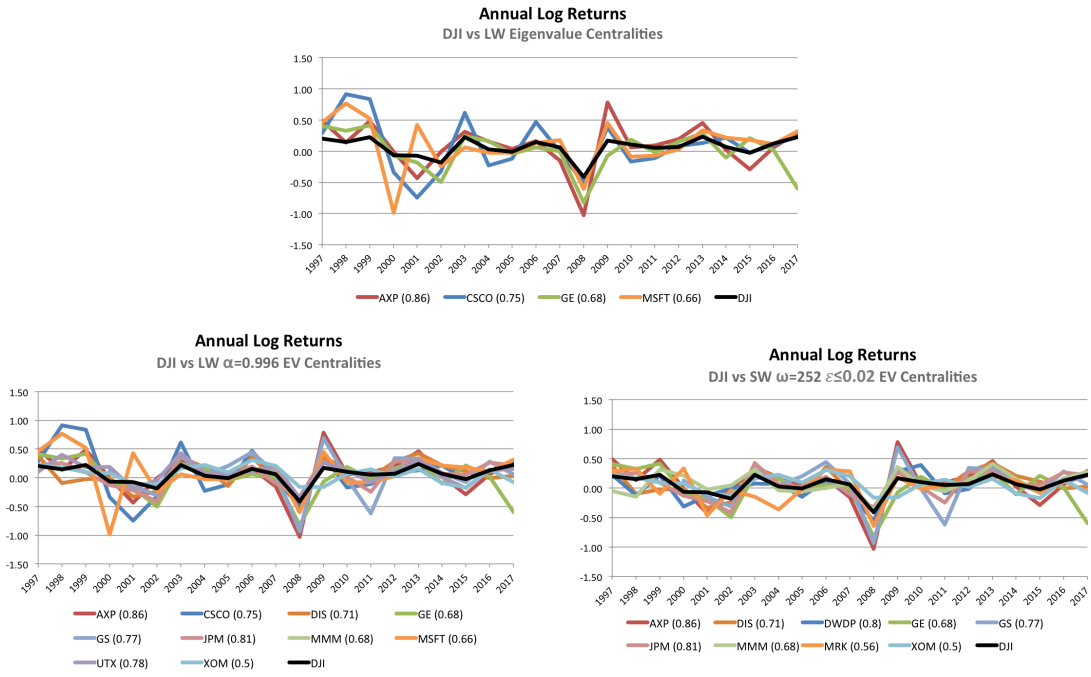


Figure 4.28: Comparing the logarithmic returns of DJI to the individual logarithmic returns of high eigenvalue centrality stocks

namics by comparing their performance to that of the market. Centrality measures are useful tools when it comes to the identification of influential stocks. Eigenvalue centrality in particular is very useful, since it measures both the quantity and quality of connections in the network.

The charts in Figure 4.28 compare the logarithmic returns of DJI and the stocks with higher eigenvalues centrality in the networks obtained from the three window model, using the settings used in subsection 4.2.2; the values within round brackets are the correlation coefficient with DJI return value. One can observe that the returns values of all stocks are correlated to those of the market, but none are capable of fully match it. However, if one computes the average return value of these stocks and compare it to the return value of the market, one finds that this average has a correlation level of over 90%; the charts in Figure 4.29 illustrate this result. Similar result are empirically confirmed in all window models, with all memory settings.

4.3.4 Applicability to portfolio management

Markowitz (1952) proposes diversification in portfolio to mitigate risk and optimize return. The author proposes investors to hold securities in considerable number, to peek stocks from companies operating in several industries, and (most important)

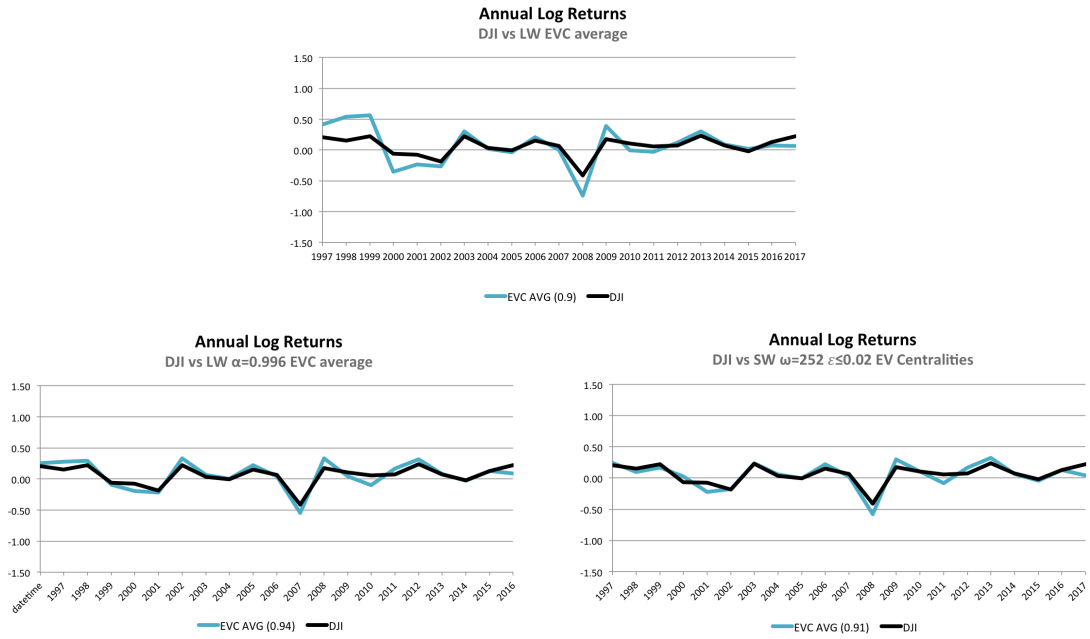


Figure 4.29: Comparing the logarithmic returns of DJI to the average logarithmic returns of high eigenvalue centrality stocks

to avoid highly correlated assets.

The goal of this work is to establish a methodology that investors can use to build themselves acceptable investment portfolios. The online tracking of evolving networks and active communities helps the investor many different ways.

The status of the correlation network, in particular, the values of density, average degree and modularity can help in the identification of the market cycle. Low density and average degree, together with high modularity are a sign of a stable, mature market. Conversely, high density/average degree and low modularity are frequently associated with a shallow, unstable market.

The study of communities and their members, of the most frequent stock-sets and association rules, complemented with the temporal perspective of longevity and churn rate given by MEC chats, helps in the identification of recurring patterns that must be avoided to guarantee the definition of heterogeneous portfolios both in terms of correlation and diversity of activity sector.

The quality and quantity of each stock neighbours (eigenvalue), links to other members of other communities (betweenness) and the distance to all other members (closeness) can help to determine both the ability and speed of influence in price and return value volatility.

A well-balanced use of the information disclosed by the three window models can help in the discrimination of small market disturbances from long-term trends.

Landmark windows disclose long-term, stable patterns that occur over a long period of time, being helpful for historical perspective. The introduction of gradual forgetting enables the detection of changes in concept, while retaining the capacity to recall old patterns of stability, thus providing milder responses in troubled times. Sliding windows are excellent in terms of change detection. They can be very helpful in times of turmoil, prompting quick warning signs at the smallest perturbation in the network. Used together, the three models produce richer results.

4.4 Summary

In this chapter we present and discuss the experimental evaluation of the devised methodology over a data set of financial data regarding the Dow Jones Industrial index. We motivate and describe experiences devised to study the application of different memory models, with different lengths, to the target data, comparing and drawing conclusions on results. We discuss the adherence of our findings to the performance of the studied index and the applicability of the methodology in portfolio management. The final conclusions regarding this work are presented on the next chapter.

Chapter 5

Conclusion

In this chapter, we present our closing remarks and discuss the limitations of our work, as well as some recommendations for future lines of investigation.

5.1 Closing remarks

In this work, we address the problem of determining which communities develop in a stock market, how do those communities evolve over time, which stocks are most influential and how is that influence exerted over the remaining stocks according to investors' reactions to economical, financial and political stimuli. The goal is to establish a methodology that investors can use in the analysis of streaming financial data relative to stock markets to improve decision making in portfolios management.

We present a methodology inspired in Social Network Analysis (SNA). The method produces series of return values out of streams of stock quotes, computes correlations between those returns and produces weighted undirected correlation networks from which metrics are taken that gauge the evolution of communities, node centralities and the networks themselves. Different window models are available to hold the statistics in memory: landmark window, gradual forgetting and sliding windows. Snapshots are used to track of changes in networks topologies.

The experimental evaluation of the methodology, conducted over financial data regarding the daily prices of stock in the portfolio of the Dow Jones Industrial index from 1997 through 2017, leads to interesting conclusions. Landmark windows define relatively sparse and small networks, comprised of medium-small communities whose members usually have long-term commercial relations, share common interests or are in the same sector of activity. Gradual forgetting and sliding windows detect three distinct network dynamics before, during and after the 2008-2012 period. Before the 2008 crisis, networks are sparse and communities are medium-small and mostly industry based, with a high survival rate and medium churn rate. The crisis period prompts highly dense networks where communities collapse to big, diverse, fast-

changing structures. In the aftermath of the 2012 sovereign debts crises, networks regain some sparsity, but the life span of communities of stocks remains low while the diversity stays high. The results are corroborated by the literature and show adherence to historical, financial and economic events.

We discuss the applicability of the methodology in portfolio management. It is our firm conviction that the devised methodology is capable of providing solid, constantly evolving evidence. The status of the correlation network can help in the identification of the market cycle. The study of communities and their members can help in diversification. Centrality measures can help to determine the both the ability and speed of influence in stock prices and return values volatilities. The use of different window models at once produces richer results, with each model showing helpful insights in different contexts. The combination of all this information, together with each stock's expected return values and the risk profile of the investor may produce interesting and appealing results in terms of portfolio management.

5.2 Limitations and Future Work

One limitation of this work regards the need to fine tune data sampling. Computing frequent stock-sets and plotting the MEC graphs comes with a heavy price in terms time and computational power. Without appropriate sampling, one risks either stalling the system or loosing vital information. Ideally, all the tasks should be fulfilled by streaming counterparts of the algorithms.

Another limitation is the naïve approach of dropping the sets of summaries when stock is missing in a given time point. This may lead to loss of important information in cases where the index components are traded in different stock exchange markets. A possible and more robust approach is to mark those sets as idle and start a countdown by the end of which the sets are dropped if no update is received in the meantime.

Future work should include the online study of other stock markets with larger sets of components.

The analysis of other stock attributes such as traded volume and stock volatility should be used to compute correlation alongside with return values. The introduction multi-dimensional variables might produce more interesting results.

Another interesting evolution would the use of the mean-variance model with the network-filtered correlation matrices (Tola et al., 2008). This could lead to a practical portfolio recommendation system.

Bibliography

Adedoyin-Olowe, M., Gaber, M. M., and Stahl, F. (2013). A survey of data mining techniques for social media analysis. *arXiv preprint arXiv:1312.4617*.

Agrawal, R., Imieliński, T., and Swami, A. (1993). Mining association rules between sets of items in large databases. In *Acm sigmod record*, volume 22, pages 207–216. ACM.

Antille, G. (2007). Descriptive analysis of matrix-valued time-series. *Applied Econometrics*, 8(4):45–54.

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.

Babcock, B., Babu, S., Datar, M., Motwani, R., and Widom, J. (2002). Models and issues in data stream systems. In *Proceedings of the twenty-first ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 1–16. ACM.

Bak, P., Paczuski, M., and Shubik, M. (1996). Price variations in a stock market with many agents. 246:430–453.

Bastian, M., Heymann, S., and Jacomy, M. (2009). Gephi: An open source software for exploring and manipulating networks.

Bifet, A. and Gavalda, R. (2006). Learning from time-changing data with adaptive windowing.

Bifet, A. and Gavalda, R. (2007). Learning from time-changing data with adaptive windowing. In *Proceedings of the 2007 SIAM International Conference on Data Mining*, pages 443–448. SIAM.

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.

Boccaletti, S., Latora, V., Moreno, Y., Chavez, M., and Hwang, D.-U. (2006). Complex networks: Structure and dynamics. *Physics reports*, 424(4):175–308.

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

Bonacich, P. (1972). Factoring and weighting approaches to status scores and clique identification. *Journal of mathematical sociology*, 2(1):113–120.

Bonacich, P. (1987). Power and centrality: A family of measures. *American journal of sociology*, 92(5):1170–1182.

Bonacich, P. and Lloyd, P. (2001). Eigenvector-like measures of centrality for asymmetric relations. *Social networks*, 23(3):191–201.

Bonanno, G., Caldarelli, G., Lillo, F., Miccichè, S., Vandewalle, N., and Mantegna, R. (2004). Networks of equities in financial markets. *The European Physical Journal B*, 38(2):363–371.

Bonanno, G., Lillo, F., and Mantegna, R. (2001). High-frequency cross-correlation in a set of stocks. *Quantitative Finance*, 1(1):96–104.

Bondy, J. A., Murty, U. S. R., et al. (1976). *Graph theory with applications*, volume 290. Citeseer.

Borůvka, O. (1926). O Jistém Problému Minimálním (About a Certain Minimal Problem) (in Czech, German summary). *Práce Mor. Prírodoved. Spol. v Brne III*, 3.

Brin, S., Motwani, R., and Silverstein, C. (1997). Beyond market baskets: Generalizing association rules to correlations. In *Acm Sigmod Record*, volume 26, pages 265–276. ACM.

Clauset, A., Newman, M. E., and Moore, C. (2004). Finding community structure in very large networks. *Physical review E*, 70(6):066111.

Cohen, E. and Strauss, M. (2003). Maintaining time-decaying stream aggregates. In *Proceedings of the twenty-second ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 223–233. ACM.

Cohen, J. (1977). Chapter 4 - differences between correlation coefficients. In Cohen, J., editor, *Statistical Power Analysis for the Behavioral Sciences*, pages 109 – 143. Academic Press.

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.

Datar, M., Gionis, A., Indyk, P., and Motwani, R. (2002). Maintaining stream statistics over sliding windows. *SIAM journal on computing*, 31(6):1794–1813.

Dijkstra, E. W. (1959). A note on two problems in connexion with graphs. *Numerische Mathematik*, (1):269 – 271.

Dimitrios, K. and Vasileios, O. (2015). A network analysis of the greek stock market. *Procedia Economics and Finance*, 33:340–349.

Dudley, R. M. (1965). Gaussian processes on several parameters. *The Annals of Mathematical Statistics*, 36(3):771–788.

Ebbinghaus, H. (1913). Memory: A contribution to experimental psychology.

Epps, T. (1979). Comovements in stock prices in the very short run. *Journal of the American Statistical Association*, 74(366a):291–298.

Euler, L. (1736). Solutio problematis ad geometriam situs pertinentis. *Commentarii Academiae Scientiarum Imperialis Petropolitanae*, 8:128–140.

Fama, E. (1965). The behaviour of stock market prices. *Journal of Business*, pages 34–105.

Freeman, L. C. (1978). Centrality in social networks conceptual clarification. *Social networks*, 1(3):215–239.

French, K. R., Schwert, G. W., and Stambaugh, R. F. (1987). Expected stock returns and volatility. *Journal of Financial Economics*, pages 3–30.

Gama, J. (2010). *Knowledge discovery from data streams*. CRC Press.

Gopikrishnan, P., Rosenow, B., Plerou, V., and Stanley, H. (2001). Quantifying and interpreting collective behavior in financial markets. *Physical Review E*, 64(3):035106.

Haldane, A. G. and May, R. M. (2011). Systemic risk in banking ecosystems. *Nature*, 469(7330):351–355.

Harris, L. and Gurel, E. (1986). Price and volume effects associated with changes in the s&p 500 list: New evidence for the existence of price pressures. *the Journal of Finance*, 41(4):815–829.

Hastings, H. (1982). The may-wigner stability theorem. *Journal of Theoretical Biology*, 97(2):155 – 166.

Heiberger, R. H. (2014). Stock network stability in times of crisis. *Physica A: Statistical Mechanics and its Applications*, 393(C):376–381.

Huang, W., Zhuang, X., and Yao, S. (2009). A network analysis of the chinese stock market. *Physica A: Statistical Mechanics and its Applications*, 388(14):2956–2964.

Klinkenberg, R. (2004). Learning drifting concepts: Example selection vs. example weighting. *Intelligent data analysis*, 8(3):281–300.

Koochakzadeh, N., Kianmehr, K., Sarraf, A., and Alhajj, R. (2012). Stock market investment advice: A social network approach. In *Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2012)*, pages 71–78. IEEE Computer Society.

Koychev, I. (2000). Gradual forgetting for adaptation to concept drift. Proceedings of ECAI 2000 Workshop on Current Issues in Spatio-Temporal Reasoning,.

Koychev, I. (2002). Tracking changing user interests through prior-learning of context. In *International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, pages 223–232. Springer.

Kruskal, J. (1956). On the Shortest Spanning Subtree of a Graph and the Traveling Salesman Problem. *Proceedings of the American Mathematical Society*, 7(1):48–50.

Kudelka, M., Horak, Z., Snasel, V., and Abraham, A. (2010). Social network reduction based on stability. In *Computational Aspects of Social Networks (CASoN), 2010 International Conference on*, pages 509–514. IEEE.

Lima, R. (2015). *Evolution of Centralities and Communities in Stock Market Networks*. Master thesis, Faculdade de Economia da Universidade do Porto.

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

Mantegna, R. and Stanley, H. (2000). *An Introduction to Econophysics: Correlations and Complexity in Finance*. Cambridge University Press, New York, NY, USA.

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

Namaki, A., Shirazi, A., Raei, R., and Jafari, G. (2011). Network analysis of a financial market based on genuine correlation and threshold method. *Physica A: Statistical Mechanics and its Applications*, 390(21-22):3835–3841.

Nešetřil, j., Milková, E., and Nešetřilová, H. (2001). Otakar borůvka on minimum spanning tree problem, translation of both the 1926 papers, comments, history. *Discrete Mathematics*, 233(1):3 – 36.

Newman, M. E. (2003). Mixing patterns in networks. *Physical Review E*, 67(2):026126.

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. (2012a). A framework to monitor clusters evolution applied to economy and finance problems. *Intelligent Data Analysis*, 16(1):93–111.

Oliveira, M. and Gama, J. a. (2012b). An overview of social network analysis. *Wiley Int. Rev. Data Min. and Knowl. Disc.*, 2(2):99–115.

Onnela, J., Kaski, K., and Kertész, J. (2004). Clustering and information in correlation based financial networks. *The European Physical Journal B*, 38:353–362.

Onnela, J.-P., Chakraborti, A., Kaski, K., and Kertész, J. (2003). Dynamic asset trees and black monday. *Physica A: Statistical Mechanics and its Applications*, 324(1):247 – 252.

Opsahl, T., Agneessens, F., and Skvoretz, J. (2010). Node centrality in weighted networks: Generalizing degree and shortest paths. *Social networks*, 32(3):245–251.

Otte, E. and Rousseau, R. (2002). Social network analysis: a powerful strategy, also for the information sciences. *Journal of Information Science*, 28(6):441–453.

Papadimitriou, S., Sun, J., and Philip, S. Y. (2006). Local correlation tracking in time series. In *Data Mining, 2006. ICDM'06. Sixth International Conference on*, pages 456–465. IEEE.

Park, Y., Priebe, C. E., and Youssef, A. (2013). Anomaly detection in time series of graphs using fusion of graph invariants. *IEEE journal of selected topics in signal processing*, 7(1):67–75.

Podobnik, B., Horvatic, D., Petersen, A. M., and Stanley, H. E. (2009). Cross-correlations between volume change and price change. *Proceedings of the National Academy of Sciences*, 106(52):22079–22084.

Pons, P. and Latapy, M. (2005). Computing communities in large networks using random walks. In *International symposium on computer and information sciences*, pages 284–293. Springer.

Preis, T., Kenett, D. Y., Stanley, H. E., Helbing, D., and Ben-Jacob, E. (2012). Quantifying the behavior of stock correlations under market stress. *Scientific reports*, 2:752.

Quinlan, J. R. (1986). Induction of decision trees. *Machine learning*, 1(1):81–106.

Rao, H., Davis, G. F., and Ward, A. (2000). Embeddedness, social identity and mobility: Why firms leave the nasdaq and join the new york stock exchange. *Administrative Science Quarterly*, 45(2):268–292.

Raunig, B. and Scharler, J. (2010). Stock market volatility and the business cycle. pages 54–63.

Roll, R. (2013). Volatility, correlation, and diversification in a multi-factor world.

Rosenow, B., Plerou, V., Gopikrishnan, P., and Stanley, H. (2002). Portfolio optimization and the random magnet problem. *EPL (Europhysics Letters)*, 59(4):500.

Ross, S. A. (1976). The arbitrage theory of capital asset pricing. *Journal of Economic Theory*, 13(3):341–60.

Rossetti, G. and Cazabet, R. (2018). Community discovery in dynamic networks: a survey. *ACM Computing Surveys (CSUR)*, 51(2):35.

Roy, R. B. and Sarkar, U. K. (2011). Identifying influential stock indices from global stock markets: A social network analysis approach. *Procedia Computer Science*, 5:442–449.

Sarmento, R., Oliveira, M., Cordeiro, M., Tabassum, S., and Gama, J. (2016). Social network analysis in streaming call graphs. In *Big Data Analysis: New Algorithms for a New Society*, pages 239–261. Springer.

Schlimmer, J. C. and Granger, R. H. (1986). Incremental learning from noisy data. *Machine learning*, 1(3):317–354.

Schönbrodt, F. D. and Perugini, M. (2013). At what sample size do correlations stabilize? *Journal of Research in Personality*, 47(5):609–612.

Sharpe, W. F. (1964). Capital asset prices: A theory of market equilibrium under conditions of risk. *The journal of finance*, 19(3):425–442.

Shoven, J. B. and Sialm, C. (2000). The dow jones industrial average: the impact of fixing its flaws. *The Journal of Wealth Management*, 3(3):9–18.

Tiao, G. C. and Box, G. E. P. (1981). Modeling multiple time series with applications. *Journal of the American Statistical Association*, 76(376):802–816.

Tola, V., Lillo, F., Gallegati, M., and Mantegna, R. N. (2008). Cluster analysis for portfolio optimization. *Journal of Economic Dynamics and Control*, 32(1):235–258.

Tse, C., Liu, J., and Lau, F. (2010). A network perspective of the stock market. *Journal of Empirical Finance*, 17(4):659 – 667.

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

Vandewalle, N., Brisbois, F., and Tordoir, X. (2000). Self-organized critical topology of stock markets. *eprint arXiv:cond-mat/0009245*.

Vodenska, I., Becker, A. P., Zhou, D., Kenett, D. Y., Stanley, H. E., and Havlin, S. (2016). Community analysis of global financial markets. *Risks*, 4(2):13.

Wakita, K. and Tsurumi, T. (2007). Finding community structure in mega-scale social networks. In *Proceedings of the 16th international conference on World Wide Web*, pages 1275–1276. ACM.

Xiao, W. (2017). An online algorithm for nonparametric correlations. *arXiv preprint arXiv:1712.01521*.

Zhu, Y. and Shasha, D. (2002). Statstream: Statistical monitoring of thousands of data streams in real time. In *Proceedings of the 28th international conference on Very Large Data Bases*, pages 358–369. VLDB Endowment.

Appendix A

Dow Jones Industrial components, prices and returns

Company	Symbol	Industry	Sector	Exchange
3M	MMM	Constr. - Supplies & Fixtures	Capital Goods	NYSE
American Express	AXP	Consumer Financial Services	Financial	NYSE
Apple	AAPL	Communications Equipment	Technology	NASDAQ
Boeing	BA	Aerospace & Defense	Capital Goods	NYSE
Caterpillar	CAT	Constr. & Agric. Machinery	Capital Goods	NYSE
Chevron	CVX	Oil & Gas - Integrated	Energy	NYSE
Cisco Systems	CSCO	Communications Equipment	Technology	NASDAQ
Coca-Cola	KO	Beverages (Nonalcoholic)	Consumer/Non-Cyclical	NYSE
Dupont	DWDP	Chemical Manufacturing	Basic Materials	NYSE
Exxon Mobil	XOM	Oil & Gas Operations	Energy	NYSE
General Electric	GE	Aerospace & Defense	Capital Goods	NYSE
Goldman Sachs	GS	Investment Services	Financial	NYSE
Home Depot	HD	Retail (Home Improvement)	Services	NYSE
IBM	IBM	Computer Services	Technology	NYSE
Intel	INTC	Semiconductors	Technology	NASDAQ
Johnson & Johnson	JNJ	Biotechnology & Drugs	Healthcare	NYSE
J.P. Morgan Chase	JPM	Investment Services	Financial	NYSE
McDonalds	MCD	Restaurants	Services	NYSE
Merck	MRK	Biotechnology & Drugs	Healthcare	NYSE
Microsoft	MSFT	Software & Programming	Technology	NASDAQ
Nike	NKE	Footwear	Consumer Cyclical	NYSE
Pfizer	PFE	Biotechnology & Drugs	Healthcare	NYSE
Proctor & Gamble	PG	Personal & Household Prods.	Consumer/Non-Cyclical	NYSE
Travelers	TRV	Insurance (Prop. & Casualty)	Financial	NYSE
United Technologies	UTX	Aerospace & Defense	Capital Goods	NYSE
United Health Group	UNH	Insurance (Accident & Health)	Financial	NYSE
Verizon	VZ	Communications Services	Services	NYSE
Visa	V	Business Services	Services	NYSE
Walmart	WMT	Retail (Grocery)	Services	NYSE
Walt Disney	DIS	Broadcasting & Cable TV	Services	NYSE

Table A.1: DJI components by December 29$^{\text{th}}$, 2017.

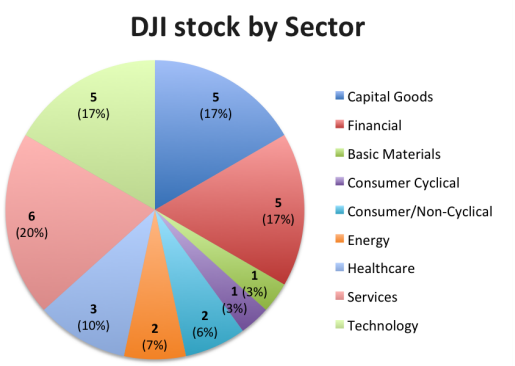
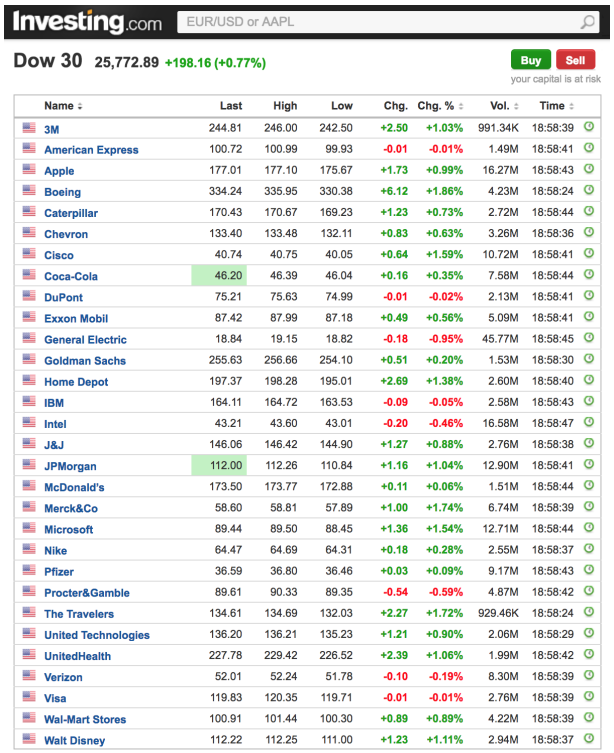


Figure A.1: Activity sectors for DJI components by December 29th, 2017.



Name	Last	High	Low	Chg.	Chg. %	Vol.	Time
3M	244.81	246.00	242.50	+2.50	+1.03%	991.34K	18:58:39
American Express	100.72	100.99	99.93	-0.01	-0.01%	1.49M	18:58:41
Apple	177.01	177.10	175.67	+1.73	+0.99%	16.27M	18:58:43
Boeing	334.24	335.95	330.38	+6.12	+1.86%	4.23M	18:58:24
Caterpillar	170.43	170.67	169.23	+1.23	+0.73%	2.72M	18:58:44
Chevron	133.40	133.48	132.11	+0.83	+0.63%	3.26M	18:58:36
Cisco	40.74	40.75	40.05	+0.64	+1.59%	10.72M	18:58:41
Coca-Cola	46.20	46.39	46.04	+0.16	+0.35%	7.58M	18:58:44
DuPont	75.21	75.63	74.99	-0.01	-0.02%	2.13M	18:58:41
Exxon Mobil	87.42	87.99	87.18	+0.49	+0.56%	5.09M	18:58:41
General Electric	18.84	19.15	18.82	-0.18	-0.95%	45.77M	18:58:45
Goldman Sachs	255.63	256.66	254.10	+0.51	+0.20%	1.53M	18:58:30
Home Depot	197.37	198.28	195.01	+2.69	+1.38%	2.60M	18:58:40
IBM	164.11	164.72	163.53	-0.09	-0.05%	2.58M	18:58:43
Intel	43.21	43.60	43.01	-0.20	-0.46%	16.58M	18:58:47
J&J	146.06	146.42	144.90	+1.27	+0.88%	2.76M	18:58:38
JPMorgan	112.00	112.26	110.84	+1.16	+1.04%	12.90M	18:58:41
McDonald's	173.50	173.77	172.88	+0.11	+0.06%	1.51M	18:58:44
Merck&Co	58.60	58.81	57.89	+1.00	+1.74%	6.74M	18:58:39
Microsoft	89.44	89.50	88.45	+1.36	+1.54%	12.71M	18:58:44
Nike	64.47	64.69	64.31	+0.18	+0.28%	2.55M	18:58:37
Pfizer	36.59	36.80	36.46	+0.03	+0.09%	9.17M	18:58:43
Procter&Gamble	89.61	90.33	89.35	-0.54	-0.59%	4.87M	18:58:42
The Travelers	134.61	134.69	132.03	+2.27	+1.72%	929.46K	18:58:24
United Technologies	136.20	136.21	135.23	+1.21	+0.90%	2.06M	18:58:29
UnitedHealth	227.78	229.42	226.52	+2.39	+1.06%	1.99M	18:58:42
Verizon	52.01	52.24	51.78	-0.10	-0.19%	8.30M	18:58:39
Visa	119.83	120.35	119.71	-0.01	-0.01%	2.76M	18:58:39
Wal-Mart Stores	100.91	101.44	100.30	+0.89	+0.89%	4.22M	18:58:39
Walt Disney	112.22	112.25	111.00	+1.23	+1.11%	2.94M	18:58:37

Figure A.2: DJI components page with ongoing trade

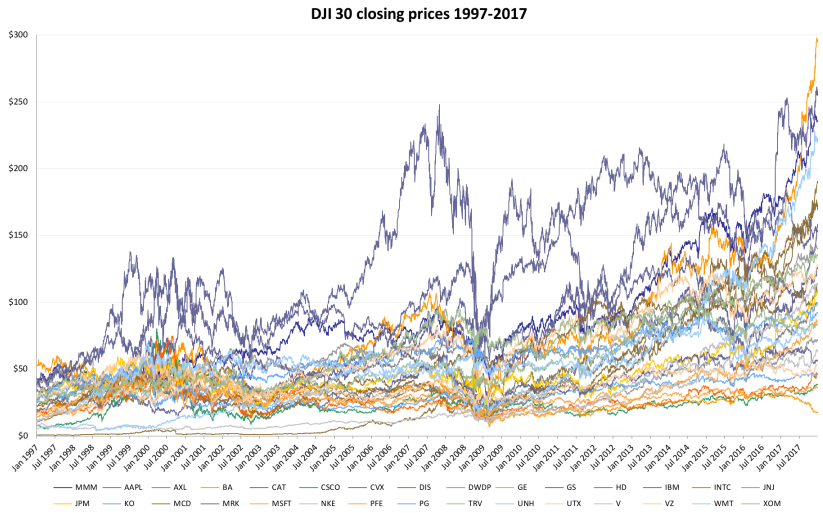


Figure A.3: Daily closing prices for DJI components from January 1st, 1997 through December 29th, 2017.

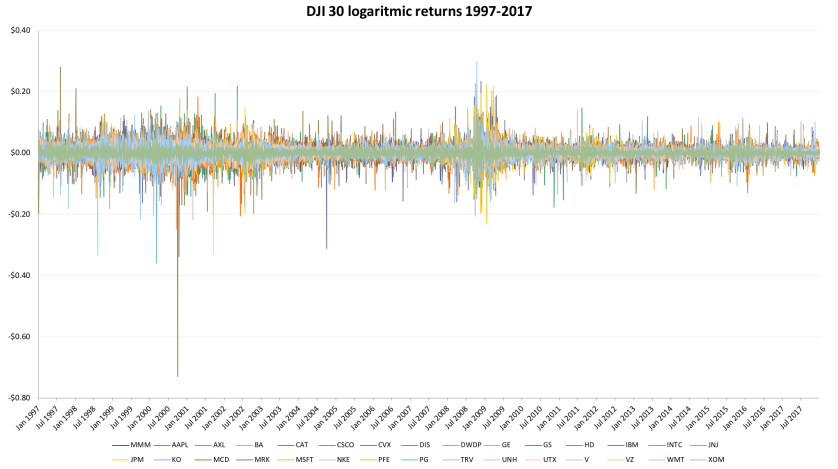


Figure A.4: Daily logarithmic returns for DJI components from January 1st, 1997 through December 29th, 2017.

Appendix B

Correlation distributions, network and community measures

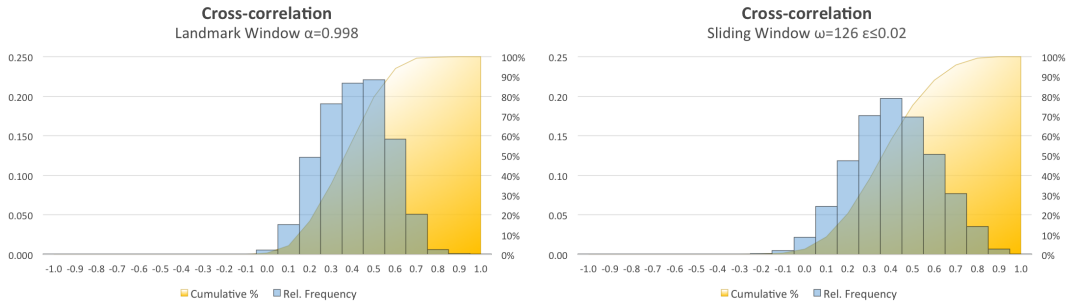


Figure B.1: Cross-correlation distributions for landmark window, gradual forgetting and sliding windows of six months

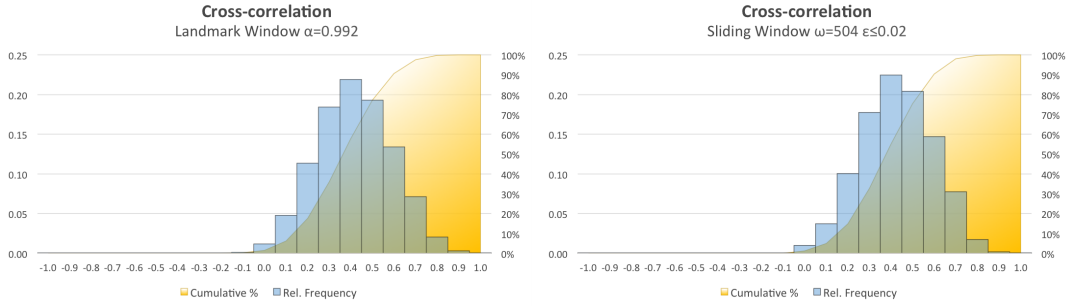


Figure B.2: Cross-correlation distributions for landmark window, gradual forgetting and sliding windows of two years

$\rho \geq \theta$	**Landmark window** $\alpha = 0.992$		**Sliding window** $\omega = 126$ $\epsilon \leq 0.02$	
θ	**Density**	**Av.Deg.**	**Density**	**Av.Deg.**
0.2	0.82 (0.20)	23.35 (5.98)	0.79 (0.20)	22.44 (5.96)
0.3	0.63 (0.27)	18.09 (7.90)	0.61 (0.26)	17.44 (7.64)
0.4	0.41 (0.29)	11.87 (8.47)	0.24 (0.25)	6.94 (7.31)
0.5	0.22 (0.24)	6.41 (7.11)	0.24 (0.25)	6.93 (7.33)
0.6	0.09 (0.16)	2.65 (4.53)	0.12 (0.19)	3.35 (5.59)
0.7	0.02 (0.05)	0.66 (1.59)	0.04 (0.10)	1.19 (2.95)
θ	**Modular.**	**#Comm.**	**Modular.**	**#Comm.**
0.2	0.06 (0.07)	2.64 (1.04)	0.07 (0.08)	2.73 (0.91)
0.3	0.10 (0.12)	3.97 (2.42)	0.11 (0.12)	3.79 (2.02)
0.4	0.19 (0.18)	7.34 (4.53)	0.28 (0.22)	11.78 (6.47)
0.5	0.31 (0.23)	12.84 (6.91)	0.28 (0.22)	11.83 (6.47)
0.6	0.38 (0.24)	19.78 (7.11)	0.36 (0.21)	18.24 (7.57)
0.7	0.22 (0.24)	25.53 (4.76)	0.27 (0.26)	24.03 (6.21)

Table B.1: Mean and standard deviation of measures observed in networks of different correlation levels obtained from six-month windows

$\rho \geq \theta$	**Landmark window** $\alpha = 0.998$		**Sliding window** $\omega = 126$ $\epsilon \leq 0.02$	
θ	**Density**	**Av.Deg.**	**Density**	**Av.Deg.**
0.2	0.83 (0.18)	23.55 (5.27)	0.85 (0.19)	24.08 (5.58)
0.3	0.63 (0.28)	18.08 (8.15)	0.67 (0.27)	18.98 (7.91)
0.4	0.41 (0.31)	11.91 (8.99)	0.44 (0.31)	12.56 (8.98)
0.5	0.20 (0.21)	5.72 (6.25)	0.24 (0.26)	6.84 (7.61)
0.6	0.06 (0.07)	1.64 (2.15)	0.09 (0.14)	2.73 (3.97)
0.7	0.01 (0.01)	0.22 (0.41)	0.02 (0.03)	0.57 (0.95)
θ	**Modular.**	**#Comm.**	**Modular.**	**#Comm.**
0.2	0.05 (0.06)	2.68 (0.88)	0.05 (0.07)	2.58 (0.96)
0.3	0.10 (0.11)	4.53 (2.32)	0.09 (0.11)	4.02 (2.52)
0.4	0.20 (0.18)	7.71 (4.16)	0.17 (0.17)	7.06 (4.21)
0.5	0.38 (0.25)	13.22 (6.39)	0.30 (0.23)	12.50 (7.11)
0.6	0.47 (0.20)	20.10 (5.89)	0.39 (0.25)	19.55 (7.39)
0.7	0.27 (0.28)	26.77 (2.36)	0.25 (0.23)	25.36 (4.39)

Table B.2: Mean and standard deviation of measures observed in networks of different correlation levels obtained from two-year windows

Year	Landmark window				Landmark window $\alpha = 0.996$				Sliding Window $\omega = 252$ $\epsilon \leq 0.02$			
	Density	Av.Deg.	Mod.	#Comm.	Density	Av.Deg.	Mod.	#Comm.	Density	Av.Deg.	Mod.	#Comm.
1997	0.10	2.71	0.29	3	0.13	3.43	0.26	4	0.10	2.71	0.29	3
1998	0.07	1.86	0.35	4	0.10	2.71	0.35	5	0.09	2.43	0.38	6
1999	0.04	1.24	0.55	4	0.03	0.76	0.69	5	0.03	0.76	0.72	4
2000	0.02	0.62	0.77	5	0.02	0.62	0.77	5	0.02	0.62	0.77	5
2001	0.02	0.69	0.76	5	0.04	1.03	0.70	6	0.14	3.79	0.36	5
2002	0.03	0.76	0.79	6	0.11	3.03	0.33	5	0.29	8.21	0.17	4
2003	0.03	0.90	0.78	6	0.12	3.45	0.30	5	0.28	7.72	0.18	3
2004	0.03	0.76	0.76	6	0.05	1.52	0.45	4	0.03	0.76	0.57	4
2005	0.02	0.69	0.78	6	0.02	0.48	0.64	4	0.02	0.48	0.51	3
2006	0.02	0.62	0.74	5	0.01	0.28	0.45	2	0.01	0.41	0.55	3
2007	0.02	0.62	0.74	5	0.11	3.17	0.26	4	0.26	7.38	0.16	4
2008	0.05	1.33	0.68	6	0.85	24.73	0.02	2	0.83	23.93	0.03	2
2009	0.06	1.67	0.62	6	0.63	18.27	0.08	2	0.47	13.60	0.08	2
2010	0.07	1.93	0.59	5	0.55	15.93	0.09	2	0.54	15.53	0.08	2
2011	0.08	2.20	0.51	5	0.86	24.93	0.02	2	0.93	26.93	0.01	2
2012	0.08	2.20	0.51	5	0.47	13.67	0.09	2	0.20	5.93	0.13	3
2013	0.08	2.20	0.51	5	0.20	5.87	0.11	3	0.10	2.93	0.22	3
2014	0.08	2.20	0.51	5	0.13	3.73	0.18	3	0.11	3.33	0.18	4
2015	0.08	2.20	0.51	5	0.29	8.40	0.15	5	0.48	13.80	0.09	4
2016	0.07	2.07	0.54	6	0.11	3.13	0.39	6	0.13	3.87	0.24	4
2017	0.07	1.93	0.55	6	0.02	0.47	0.69	4	0.01	0.33	0.54	3

Table B.3: Density, average degree, modularity and number of communities for all window models, annual values from 1997 through 2017

Year	Landmark window				Landmark window $\alpha = 0.998$				Sliding Window $\omega = 504$ $\epsilon \leq 0.02$			
	Density	Av.Deg.	Mod.	#Comm.	Density	Av.Deg.	Mod.	#Comm.	Density	Av.Deg.	Mod.	#Comm.
1997	0.10	2.71	0.29	3	0.11	3.07	0.24	4	0.10	2.71	0.29	3
1999	0.04	1.24	0.55	4	0.03	0.90	0.67	4	0.04	1.03	0.54	4
2001	0.02	0.69	0.76	5	0.03	0.76	0.81	7	0.03	0.90	0.77	6
2003	0.03	0.90	0.78	6	0.07	1.93	0.55	6	0.27	7.66	0.18	4
2005	0.02	0.69	0.78	6	0.03	0.76	0.70	5	0.01	0.28	0.74	4
2007	0.02	0.62	0.74	5	0.03	0.83	0.58	5	0.05	1.31	0.33	5
2009	0.06	1.67	0.62	6	0.54	15.67	0.09	2	0.74	21.47	0.06	2
2011	0.08	2.20	0.51	5	0.72	20.80	0.06	2	0.79	22.80	0.04	2
2013	0.08	2.20	0.51	5	0.32	9.20	0.11	3	0.15	4.27	0.13	4
2015	0.08	2.20	0.51	5	0.23	6.80	0.17	4	0.21	6.07	0.19	5
2017	0.07	1.93	0.55	6	0.03	0.93	0.66	6	0.02	0.60	0.72	5

Table B.4: Density, average degree, modularity and number of communities for all window models, biennual values from 1997 through 2017

Year	Landmark window				Landmark window $\alpha = 0.992$				Sliding Window $\omega = 126$ $\epsilon \leq 0.02$			
	Density	Av.Deg.	Mod.	#Comm.	Density	Av.Deg.	Mod.	#Comm.	Density	Av.Deg.	Mod.	#Comm.
Jun 1997	0.07	1.93	0.25	4	0.08	2.14	0.24	4	0.07	1.93	0.25	4
Dec 1997	0.10	2.71	0.29	3	0.19	5.00	0.21	3	0.28	7.50	0.14	4
Jun 1998	0.06	1.57	0.41	4	0.04	1.07	0.50	4	0.03	0.93	0.64	5
Dec 1998	0.07	1.86	0.35	4	0.13	3.43	0.29	5	0.22	5.93	0.16	5
Jun 1999	0.05	1.52	0.43	5	0.06	1.59	0.36	4	0.06	1.66	0.36	6
Dec 1999	0.04	1.24	0.55	4	0.02	0.48	0.78	5	0.02	0.69	0.64	4
Jun 2000	0.03	0.90	0.68	4	0.03	0.90	0.75	5	0.04	1.24	0.76	5
Dec 2000	0.02	0.62	0.77	5	0.02	0.55	0.76	5	0.01	0.41	0.74	4
Jun 2001	0.02	0.69	0.77	5	0.02	0.69	0.77	6	0.10	2.76	0.45	6
Dec 2001	0.02	0.69	0.76	5	0.08	2.14	0.44	4	0.24	6.62	0.16	4
Jun 2002	0.02	0.69	0.76	5	0.06	1.79	0.63	6	0.11	3.10	0.40	6
Dec 2002	0.03	0.76	0.79	6	0.27	7.66	0.17	3	0.47	13.10	0.10	3
Jun 2003	0.03	0.83	0.78	6	0.34	9.52	0.14	4	0.56	15.79	0.07	2
Dec 2003	0.03	0.90	0.78	6	0.16	4.41	0.24	4	0.08	2.14	0.30	3
Jun 2004	0.03	0.90	0.78	6	0.06	1.79	0.59	6	0.05	1.31	0.61	5
Dec 2004	0.03	0.76	0.76	6	0.03	0.97	0.50	5	0.03	0.90	0.37	4
Jun 2005	0.02	0.69	0.78	6	0.04	1.24	0.42	4	0.06	1.79	0.44	5
Dec 2005	0.02	0.69	0.78	6	0.01	0.34	0.73	4	0.02	0.55	0.68	4
Jun 2006	0.02	0.69	0.78	6	0.02	0.48	0.49	3	0.05	1.31	0.42	4
Dec 2006	0.02	0.62	0.74	5	0.01	0.34	0.41	2	0.03	0.97	0.32	4
Jun 2007	0.02	0.62	0.74	5	0.02	0.69	0.36	3	0.12	3.38	0.27	4
Dec 2007	0.02	0.62	0.74	5	0.28	7.79	0.18	3	0.44	12.34	0.10	4
Jun 2008	0.02	0.67	0.78	6	0.27	7.93	0.15	4	0.30	8.80	0.15	4
Dec 2008	0.05	1.33	0.68	6	0.89	25.93	0.01	2	0.97	28.07	0.01	2
Jun 2009	0.06	1.60	0.66	6	0.73	21.27	0.08	2	0.56	16.13	0.09	2
Dec 2009	0.06	1.67	0.62	6	0.53	15.27	0.09	3	0.24	7.00	0.15	3
Jun 2010	0.06	1.87	0.60	6	0.62	18.00	0.06	2	0.66	19.00	0.05	2
Dec 2010	0.07	1.93	0.59	5	0.45	13.00	0.10	3	0.38	11.13	0.11	3
Jun 2011	0.06	1.87	0.60	6	0.23	6.53	0.09	4	0.12	3.60	0.24	3
Dec 2011	0.08	2.20	0.51	5	0.97	28.20	0.00	2	1.00	28.87	0.00	1
Jun 2012	0.08	2.20	0.51	5	0.63	18.27	0.05	2	0.29	8.53	0.08	4
Dec 2012	0.08	2.20	0.51	5	0.37	10.80	0.12	3	0.23	6.73	0.20	3
Jun 2013	0.08	2.20	0.51	5	0.22	6.27	0.14	3	0.18	5.20	0.16	3
Dec 2013	0.08	2.20	0.51	5	0.11	3.13	0.18	3	0.05	1.40	0.45	3
Jun 2014	0.08	2.20	0.51	5	0.07	2.13	0.20	4	0.11	3.07	0.18	3
Dec 2014	0.08	2.20	0.51	5	0.12	3.53	0.27	4	0.18	5.13	0.23	4
Jun 2015	0.08	2.20	0.51	5	0.15	4.33	0.22	4	0.24	7.07	0.16	4
Dec 2015	0.08	2.20	0.51	5	0.48	13.87	0.08	4	0.64	18.67	0.06	2
Jun 2016	0.07	2.13	0.53	5	0.36	10.53	0.15	4	0.29	8.53	0.18	4
Dec 2016	0.07	2.07	0.54	6	0.08	2.40	0.46	6	0.06	1.73	0.56	5
Jun 2017	0.07	2.00	0.56	6	0.02	0.60	0.76	5	0.01	0.40	0.59	3
Dec 2017	0.07	1.93	0.55	6	0.01	0.40	0.60	4	0.01	0.40	0.62	3

Table B.5: Density, average degree, modularity and number of communities for all window models, semestral values from 1997 through 2017

Appendix C

Community members

Year	Communities					
	1	2	3	4	5	6
1997	CSCO, IBM, INTC, MSFT	AXP, BA, DIS, DWDP, GE, HD, JPM, WMT	CVX, JNJ, KO, MRK, PFE, PG, XOM			
1998	CSCO, IBM, INTC, MSFT	AXP, DIS, GE, HD, JPM, MRK, PFE, WMT	JNJ, KO, PG	CVX, XOM		
1999	CSCO, IBM, INTC, MSFT	JNJ, MRK, PFE	AXP, GE, HD, JPM, KO, PG, WMT	CVX, XOM		
2000	CSCO, INTC, MSFT	AXP, GE, JPM	JNJ, MRK, PFE	HD, WMT	CVX, XOM	
2001	CSCO, INTC, MSFT	AXP, GE, JPM	JNJ, MRK, PFE	HD, WMT	CVX, XOM	
2002	CSCO, INTC, MSFT	AXP, GE, JPM	DWDP, MMM	JNJ, MRK, PFE	HD, WMT	CVX, XOM
2003	AXP, GE, JPM	DWDP, MMM	CSCO, IBM, INTC, MSFT	JNJ, MRK, PFE	HD, WMT	CVX, XOM
2004	AXP, GE, JPM	DWDP, MMM	CSCO, IBM, INTC, MSFT	MRK, PFE	HD, WMT	CVX, XOM
2005	CSCO, INTC, MSFT	AXP, GE, JPM	DWDP, MMM	MRK, PFE	HD, WMT	CVX, XOM
2006	CSCO, INTC, MSFT	AXP, GE, JPM	MRK, PFE	HD, WMT	CVX, XOM	
2007	CSCO, INTC, MSFT	AXP, GE, JPM	MRK, PFE	HD, WMT	CVX, XOM	
2008	CSCO, IBM, INTC, MSFT	AXP, GE, HD, JPM, TRV, WMT	CAT, DWDP, MMM	JNJ, MRK, PFE	BA, UTX	CVX, XOM
2009	CSCO, IBM, INTC, MSFT	AXP, GE, JPM, TRV	BA, CAT, DWDP, MMM, UTX	JNJ, MRK, PFE	HD, WMT	CVX, XOM
2010	CSCO, IBM, INTC, MSFT	AXP, DIS, GE, HD, JPM, TRV, WMT	BA, CAT, DWDP, MMM, UTX	JNJ, MRK, PFE	CVX, XOM	
2011	CSCO, IBM, INTC, MSFT	AXP, DIS, GE, HD, JPM, TRV, WMT	BA, CAT, DWDP, MMM, UTX	JNJ, MRK, PFE	CVX, XOM	
2012	CSCO, IBM, INTC, MSFT	AXP, DIS, GE, HD, JPM, TRV, WMT	BA, CAT, DWDP, MMM, UTX	JNJ, MRK, PFE	CVX, XOM	
2013	CSCO, IBM, INTC, MSFT	AXP, DIS, GE, HD, JPM, TRV, WMT	BA, CAT, DWDP, MMM, UTX	JNJ, MRK, PFE	CVX, XOM	
2014	CSCO, IBM, INTC, MSFT	AXP, DIS, GE, HD, JPM, TRV, WMT	BA, CAT, DWDP, MMM, UTX	JNJ, MRK, PFE	CVX, XOM	
2015	CSCO, IBM, INTC, MSFT	AXP, DIS, GE, HD, JPM, TRV, WMT	BA, CAT, DWDP, MMM, UTX	JNJ, MRK, PFE	CVX, XOM	
2016	CSCO, IBM, INTC, MSFT	AXP, DIS, GE, JPM, TRV	BA, CAT, DWDP, MMM, UTX	JNJ, MRK, PFE	HD, WMT	CVX, XOM
2017	AXP, DIS, GE, JPM, TRV	BA, CAT, DWDP, MMM, UTX	CSCO, IBM, INTC, MSFT	JNJ, MRK, PFE	HD, WMT	CVX, XOM

Table C.1: Communities disclosed by landmark window, annual snapshots from 1997 through 2017.

Year	Communities					
	1	2	3	4	5	6
1997	CSCO, IBM, INTC, MSFT	AXP, DIS, JNJ, JPM, KO, MRK, PFE	BA, GE, HD, UTX, WMT	CVX, DWDP, PG, XOM		
1998	CSCO, IBM, INTC, MSFT	DWDP, MMM	DIS, GE, JNJ, KO, MCD, MRK, PFE, PG	AXP, HD, JPM, TRV, UTX, WMT	CVX, XOM	
1999	AXP, JPM	CSCO, INTC, MSFT	JNJ, MRK, PFE	GE, HD, WMT	CVX, XOM	
2000	CSCO, INTC, MSFT	AXP, GE, JPM	JNJ, MRK, PFE	HD, WMT	CVX, XOM	
2001	AAPL, CSCO, INTC, MSFT	CAT, DWDP, MMM	MRK, PFE	AXP, BA, GE, JPM, UTX	HD, WMT	CVX, XOM
2002	AAPL, CSCO, IBM, INTC, JPM, MSFT	JNJ, MRK, PFE	AXP, BA, CAT, DIS, DWDP, GE, MMM, TRV, UTX	HD, WMT	CVX, XOM	
2003	AAPL, CSCO, IBM, INTC, MSFT	AXP, BA, CAT, DIS, DWDP, GE, JPM, MMM, TRV, UTX	JNJ, MRK, PFE	HD, WMT	CVX, XOM	
2004	CSCO, IBM, INTC, MSFT	AXP, BA, CAT, DIS, DWDP, GE, JPM, UTX	HD, WMT	CVX, XOM		
2005	CSCO, INTC	AXP, GE, JPM, UTX	HD, WMT	CVX, XOM		
2006	AXP, GE, GS, JPM	CVX, XOM				
2007	AXP, DIS, DWDP, GS, HD, JPM, TRV, WMT	BA, CAT, CSCO, IBM, INTC, UTX	GE, MMM, MSFT, PFE, VZ	CVX, XOM		
2008	AAPL, AXP, CAT, CSCO, DWDP, GE, GS, HD, IBM, INTC, JPM, TRV, UTX	BA, CVX, DIS, JNJ, KO, MCD, MMM, MRK, MSFT, NKE, PFE, PG, UNH, VZ, WMT, XOM				
2009	AAPL, AXP, BA, CAT, CSCO, DWDP, GE, GS, HD, IBM, INTC, JPM, MMM, NKE	CVX, DIS, JNJ, KO, MCD, MRK, MSFT, PFE, PG, TRV, UNH, UTX, VZ, WMT, XOM				
2010	AAPL, AXP, BA, CAT, CSCO, GE, GS, HD, IBM, INTC, JPM, MMM, MSFT, NKE, TRV	CVX, DIS, DWDP, JNJ, KO, MCD, MRK, PFE, PG, UTX, VZ, WMT, XOM				
2011	AAPL, AXP, BA, CAT, CSCO, DIS, DWDP, GE, GS, IBM, INTC, JPM, MMM, MSFT, NKE, UTX	CVX, HD, JNJ, KO, MCD, MRK, PFE, PG, TRV, UNH, VZ, WMT, XOM				
2012	AXP, BA, CAT, CSCO, DIS, DWDP, GS, HD, IBM, INTC, JPM, MMM, MSFT, UTX	CVX, GE, JNJ, KO, MCD, MRK, PFE, PG, TRV, UNH, VZ, XOM				
2013	AXP, CAT, CVX, DWDP, GE, GS, JPM, TRV, XOM	DIS, HD, JNJ, KO, MRK, PFE, PG	BA, INTC, MMM, UTX			
2014	JNJ, MRK, PFE	AXP, BA, DIS, DWDP, GS, JPM, MMM, TRV, UTX, V	CAT, CVX, GE, XOM			
2015	CSCO, INTC, MSFT	BA, HD, MMM, NKE, UTX	JNJ, KO, MCD, MRK, PFE, PG, TRV, UNH, VZ	AXP, DIS, GS, V	CAT, CVX, GE, IBM, JPM, XOM	
2016	AXP, DIS, GS, JPM, TRV	CSCO, HD, INTC, MSFT, NKE, V	JNJ, MRK, PFE	KO, PG	BA, GE, IBM, MMM, UTX	CAT, CVX, XOM
2017	AXP, GS, JPM	KO, PG	INTC, MSFT, V	CVX, XOM		

Table C.2: Communities disclosed by gradual forgetting, annual snapshots from 1997 through 2017

Year	Communities					
	1	2	3	4	5	6
1997	CSCO, IBM, INTC, MSFT	AXP, BA, DIS, DWDP, GE, HD, JPM, WMT	CVX, JNJ, KO, MRK, PFE, PG, XOM			
1998	AXP, JPM, UTX	DWDP, MMM	AAPL, CSCO, IBM, INTC, MSFT	DIS, GE, JNJ, KO, MCD, MRK, PFE, PG	HD, TRV, WMT	CVX, XOM
1999	CSCO, INTC, MSFT	JNJ, MRK, PFE	AXP, GE, HD, JPM, WMT	CVX, XOM		
2000	CSCO, INTC, MSFT	AXP, GE, GS, JPM	JNJ, MRK, PFE	HD, WMT	CVX, XOM	
2001	AXP, DIS, GS, JPM	AAPL, CSCO, IBM, INTC, MSFT	MRK, PFE	BA, CAT, DWDP, GE, HD, MMM, UTX, WMT	CVX, XOM	
2002	AAPL, CSCO, GS, IBM, INTC, JPM, MSFT	BA, CAT, DIS, DWDP, MMM, UTX	AXP, HD, NKE, TRV, WMT	CVX, GE, JNJ, KO, MRK, PFE, PG, VZ, XOM		
2003	AAPL, AXP, CSCO, IBM, INTC, MSFT	BA, CAT, DIS, DWDP, GE, GS, MMM, NKE, TRV, UTX, WMT	CVX, HD, JNJ, JPM, MRK, PFE, PG, VZ, XOM			
2004	CSCO, INTC	BA, UTX	AXP, DIS, GS, HD, JPM, WMT	CVX, GE, XOM		
2005	AXP, GE, GS, JPM, UTX	HD, WMT	CVX, XOM			
2006	AXP, GS, JPM, TRV	MRK, PFE	CVX, XOM			
2007	JNJ, KO, MRK, PFE	CSCO, GE, IBM, INTC, MSFT	AXP, DIS, GS, HD, JPM, PG, TRV, VZ, WMT	BA, CAT, CVX, DWDP, MMM, NKE, UTX, XOM		
2008	AAPL, AXP, CAT, CSCO, DWDP, GE, GS, HD, IBM, INTC, JPM, TRV	BA, CVX, DIS, JNJ, KO, MCD, MMM, MRK, MSFT, NKE, PFE, PG, UNH, UTX, VZ, WMT, XOM				
2009	CVX, JNJ, KO, MCD, MRK, PFE, PG, TRV, UTX, VZ, XOM	AAPL, AXP, BA, CAT, CSCO, DIS, DWDP, GE, GS, HD, IBM, INTC, JPM, MMM, MSFT, NKE, V				
2010	AAPL, AXP, BA, CAT, CSCO, DIS, GE, GS, HD, INTC, JPM, MMM, MSFT, NKE, TRV	CVX, DWDP, IBM, JNJ, KO, MCD, MRK, PFE, PG, UNH, UTX, VZ, XOM				
2011	AAPL, AXP, BA, CAT, CSCO, CVX, DIS, DWDP, GE, GS, IBM, INTC, JPM, MMM, MSFT, NKE, UNH, UTX, V, XOM	HD, JNJ, KO, MCD, MRK, PFE, PG, TRV, VZ, WMT				
2012	CVX, DIS, GE, HD, JNJ, KO, MMM, MRK, PFE, TRV	AXP, BA, CAT, DWDP, GS, JPM, UTX, V	CSCO, IBM, INTC, MSFT, XOM			
2013	DIS, JNJ, KO, PFE, PG	AXP, BA, CAT, UTX	CVX, DWDP, GE, GS, JPM, MMM, TRV, XOM			
2014	INTC, MSFT	DIS, DWDP, GS, JPM, NKE	AXP, BA, CSCO, JNJ, MMM, PFE, TRV, UTX, V	CAT, CVX, GE, XOM		
2015	AXP, DIS, GS, V	AAPL, CSCO, IBM, INTC, MSFT	BA, HD, JNJ, KO, MCD, MMM, MRK, NKE, PFE, PG, TRV, UNH, WMT	CAT, CVX, GE, JPM, UTX, VZ, XOM		
2016	BA, DWDP, GE, GS, IBM, INTC, JPM, TRV, UTX	MRK, PFE	AAPL, CSCO, DIS, HD, MSFT, NKE, V	CAT, CVX, JNJ, KO, MMM, PG, XOM		
2017	AXP, GS, JPM	MSFT, V	CVX, XOM			

Table C.3: Communities disclosed by sliding windows, annual snapshots from 1997 through 2017.

Year	Communities					
	1	2	3	4	5	6
1997	CSCO, IBM, INTC, MSFT	AXP, BA, DIS, DWDP, GE, HD, JPM, WMT	CVX, JNJ, KO, MRK, PFE, PG, XOM			
1999	CSCO, IBM, INTC, MSFT	JNJ, MRK, PFE	AXP, GE, HD, JPM, KO, PG, WMT	CVX, XOM		
2001	CSCO, INTC, MSFT	AXP, GE, JPM	JNJ, MRK, PFE	HD, WMT	CVX, XOM	
2003	AXP, GE, JPM	DWDP, MMM	CSCO, IBM, INTC, MSFT	JNJ, MRK, PFE	HD, WMT	CVX, XOM
2005	CSCO, INTC, MSFT	AXP, GE, JPM	DWDP, MMM	MRK, PFE	HD, WMT	CVX, XOM
2007	CSCO, INTC, MSFT	AXP, GE, JPM	MRK, PFE	HD, WMT	CVX, XOM	
2009	CSCO, IBM, INTC, MSFT	AXP, GE, JPM, TRV	BA, CAT, DWDP, MMM, UTX	JNJ, MRK, PFE	HD, WMT	CVX, XOM
2011	CSCO, IBM, INTC, MSFT	AXP, DIS, GE, HD, JPM, TRV, WMT	BA, CAT, DWDP, MMM, UTX	JNJ, MRK, PFE	CVX, XOM	
2013	CSCO, IBM, INTC, MSFT	AXP, DIS, GE, HD, JPM, TRV, WMT	BA, CAT, DWDP, MMM, UTX	JNJ, MRK, PFE	CVX, XOM	
2015	CSCO, IBM, INTC, MSFT	AXP, DIS, GE, HD, JPM, TRV, WMT	BA, CAT, DWDP, MMM, UTX	JNJ, MRK, PFE	CVX, XOM	
2017	AXP, DIS, GE, JPM, TRV	BA, CAT, DWDP, MMM, UTX	CSCO, IBM, INTC, MSFT	JNJ, MRK, PFE	HD, WMT	CVX, XOM

Table C.4: Communities disclosed by landmark window, biennual snapshots from 1997 through 2017.

Year	Communities						
	1	2	3	4	5	6	7
1997	CSCO, IBM, INTC, MSFT	AXP, DIS, JNJ, JPM, KO, MRK, PFE	BA, GE, HD, WMT	CVX, DWDP, PG, XOM			
1999	CSCO, IBM, INTC, MSFT	JNJ, MRK, PFE	AXP, GE, HD, JPM, WMT	CVX, XOM			
2001	CSCO, INTC, MSFT	AXP, GE, JPM	DWDP, MMM	MRK, PFE	BA, UTX	HD, WMT	CVX, XOM
2003	CSCO, IBM, INTC, MSFT	AXP, DIS, GE, JPM	BA, CAT, DWDP, MMM, UTX	JNJ, MRK, PFE	HD, WMT	CVX, XOM	
2005	CSCO, INTC, MSFT	DWDP, MMM	AXP, BA, GE, JPM, UTX	HD, WMT	CVX, XOM		
2007	CSCO, IBM, INTC	AXP, DWDP, GE, JPM, TRV	BA, UTX	HD, WMT	CVX, XOM		
2009	AAPL, AXP, BA, CAT, CSCO, DWDP, GE, GS, HD, IBM, INTC, JPM, MMM, MSFT, NKE, VZ	CVX, DIS, JNJ, KO, MCD, MRK, PFE, PG, TRV, UTX, WMT, XOM					
2011	AAPL, AXP, BA, CAT, CSCO, DWDP, GE, GS, IBM, INTC, JPM, MSFT, NKE, UTX	CVX, DIS, HD, JNJ, KO, MCD, MMM, MRK, PFE, PG, TRV, UNH, VZ, WMT, XOM					
2013	AXP, BA, CAT, DIS, DWDP, GE, GS, HD, JPM, MMM, TRV	CVX, JNJ, KO, MRK, PFE, PG, XOM	CSCO, IBM, INTC, MSFT, NKE, UTX				
2015	AXP, BA, CAT, CVX, DIS, DWDP, GE, GS, JPM, UTX, XOM	CSCO, IBM, INTC, MMM, MSFT	HD, NKE	JNJ, KO, MCD, MRK, PFE, PG, TRV, VZ			
2017	AXP, CAT, GS, JPM, TRV	INTC, MSFT	JNJ, MRK, PFE	KO, PG	BA, MMM, UTX	CVX, XOM	

Table C.5: Communities disclosed by gradual forgetting, biennual snapshots from 1997 through 2017.

Year	Communities					
	1	2	3	4	5	6
1997	CSCO, IBM, INTC, MSFT	AXP, BA, DIS, DWDP, GE, HD, JPM, WMT	CVX, JNJ, KO, MRK, PFE, PG, XOM			
1999	CSCO, IBM, INTC, MSFT	JNJ, MRK, PFE	AXP, GE, HD, JPM, WMT	CVX, XOM		
2001	AXP, GE, GS, JPM	CSCO, INTC, MSFT	CAT, DWDP, MMM	JNJ, MRK, PFE	HD, WMT	CVX ,XOM
2003	AAPL, CSCO, GS, IBM, INTC, JPM, MSFT	AXP, CVX, JNJ, KO, MRK, NKE, PFE, PG, XOM	HD, WMT	BA, CAT, DIS, DWDP, GE, MMM, TRV, UTX, VZ		
2005	CSCO, INTC	GS, JPM	HD, WMT	CVX, XOM		
2007	MRK, PFE	AXP, DWDP, GE, GS, JPM, TRV	BA, UTX	HD, WMT	CVX, XOM	
2009	AAPL, AXP, BA, CAT, CSCO, DWDP, GE, GS, HD, IBM, INTC, JPM, MMM, MSFT, NKE, TRV	CVX, DIS, JNJ, KO, MCD, MRK, PFE, PG, UNH, UTX, VZ, WMT, XOM				
2011	AAPL, AXP, BA, CAT, CSCO, DIS, DWDP, GE, GS, INTC, JPM, MMM, MSFT, NKE, UTX, V	CVX, HD, IBM, JNJ, KO, MCD, MRK, PFE, PG, TRV, UNH, VZ, WMT, XOM				
2013	AXP, DWDP, GE, GS, JPM, V	DIS, KO, TRV, XOM	BA, CAT, CVX, INTC, MMM, UTX	JNJ, MRK, PFE, PG		
2015	CSCO, INTC, MSFT	JNJ, KO, MRK, PFE, PG, TRV, UNH	HD, MCD, MMM, NKE	AXP, BA, DIS, GS, IBM, JPM, UTX, V	CAT, CVX, GE, XOM	
2017	CAT, DWDP, GS, JPM	MRK, PFE	KO, PG	INTC, MSFT, V	CVX, XOM	

Table C.6: Communities disclosed by sliding windows, biennual snapshots from 1997 through 2017.

Year	Communities					
	1	2	3	4	5	6
Jun 1997	AXP, DWDP, GE, JPM, MMM	CSCO, INTC, MSFT	JNJ, KO, MRK, PFE, PG	CVX, XOM		
Dec 1997	CSCO, IBM, INTC, MSFT	AXP, BA, DIS, DWDP, GE, HD, JPM, WMT	CVX, JNJ, KO, MRK, PFE, PG, XOM			
Jun 1998	AXP, JPM	CSCO, IBM, INTC, MSFT	DIS, GE, JNJ, KO, MRK, PFE, PG, WMT	CVX, XOM		
Dec 1998	CSCO, IBM, INTC, MSFT	AXP, DIS, GE, HD, JPM, MRK, PFE, WMT	JNJ, KO, PG	CVX, XOM		
Jun 1999	JNJ, MRK, PFE	CSCO, IBM, INTC, MSFT	AXP, JPM	GE, HD, KO, PG, WMT	CVX, XOM	
Dec 1999	CSCO, IBM, INTC, MSFT	JNJ, MRK, PFE	AXP, GE, HD, JPM, KO, PG, WMT	CVX, XOM		
Jun 2000	CSCO, INTC, MSFT	JNJ, MRK, PFE	AXP, GE, HD, JPM, WMT	CVX, XOM		
Dec 2000	CSCO, INTC, MSFT	AXP, GE, JPM	JNJ, MRK, PFE	HD, WMT	CVX, XOM	
Jun 2001	CSCO, INTC, MSFT	AXP, GE, JPM	JNJ, MRK, PFE	HD, WMT	CVX, XOM	
Dec 2001	CSCO, INTC, MSFT	AXP, GE, JPM	JNJ, MRK, PFE	HD, WMT	CVX, XOM	
Jun 2002	CSCO, INTC, MSFT	AXP, GE, JPM	JNJ, MRK, PFE	HD, WMT	CVX, XOM	
Dec 2002	CSCO, INTC, MSFT	AXP, GE, JPM	DWDP, MMM	JNJ, MRK, PFE	HD, WMT	CVX, XOM
Jun 2003	AXP, GE, JPM	DWDP, MMM	CSCO, IBM, INTC, MSFT	JNJ, MRK, PFE	HD, WMT	CVX, XOM
Dec 2003	AXP, GE, JPM	DWDP, MMM	CSCO, IBM, INTC, MSFT	JNJ, MRK, PFE	HD, WMT	CVX, XOM
Jun 2004	AXP, GE, JPM	DWDP, MMM	CSCO, IBM, INTC, MSFT	JNJ, MRK, PFE	HD, WMT	CVX, XOM
Dec 2004	AXP, GE, JPM	DWDP, MMM	CSCO, IBM, INTC, MSFT	MRK, PFE	HD, WMT	CVX, XOM
Jun 2005	CSCO, INTC, MSFT	AXP, GE, JPM	DWDP, MMM	MRK, PFE	HD, WMT	CVX, XOM
Dec 2005	CSCO, INTC, MSFT	AXP, GE, JPM	DWDP, MMM	MRK, PFE	HD, WMT	CVX, XOM
Jun 2006	CSCO, INTC, MSFT	AXP, GE, JPM	DWDP, MMM	MRK, PFE	HD, WMT	CVX, XOM
Dec 2006	CSCO, INTC, MSFT	AXP, GE, JPM	MRK, PFE	HD, WMT	CVX, XOM	
Jun 2007	CSCO, INTC, MSFT	AXP, GE, JPM	MRK, PFE	HD, WMT	CVX, XOM	
Dec 2007	CSCO, INTC, MSFT	AXP, GE, JPM	MRK, PFE	HD, WMT	CVX, XOM	
Jun 2008	CSCO, INTC, MSFT	AXP, GE, JPM	DWDP, MMM	MRK, PFE	HD, WMT	CVX, XOM
Dec 2008	CSCO, IBM, INTC, MSFT	AXP, GE, HD, JPM, TRV, WMT	CAT, DWDP, MMM	JNJ, MRK, PFE	BA, UTX	CVX, XOM
Jun 2009	CSCO, IBM, INTC, MSFT	AXP, GE, JPM, TRV	BA, CAT, DWDP, MMM, UTX	JNJ, MRK, PFE	HD, WMT	CVX, XOM
Dec 2009	CSCO, IBM, INTC, MSFT	AXP, GE, JPM, TRV	BA, CAT, DWDP, MMM, UTX	JNJ, MRK, PFE	HD, WMT	CVX, XOM
Jun 2010	CSCO, IBM, INTC, MSFT	AXP, DIS, GE, JPM, TRV	BA, CAT, DWDP, MMM, UTX	JNJ, MRK, PFE	HD, WMT	CVX, XOM
Dec 2010	CSCO, IBM, INTC, MSFT	AXP, DIS, GE, HD, JPM, TRV, WMT	BA, CAT, DWDP, MMM, UTX	JNJ, MRK, PFE	CVX, XOM	
Jun 2011	CSCO, IBM, INTC, MSFT	AXP, DIS, GE, JPM, TRV	BA, CAT, DWDP, MMM, UTX	JNJ, MRK, PFE	HD, WMT	CVX, XOM
Dec 2011	CSCO, IBM, INTC, MSFT	AXP, DIS, GE, HD, JPM, TRV, WMT	BA, CAT, DWDP, MMM, UTX	JNJ, MRK, PFE	CVX, XOM	
Jun 2012	CSCO, IBM, INTC, MSFT	AXP, DIS, GE, HD, JPM, TRV, WMT	BA, CAT, DWDP, MMM, UTX	JNJ, MRK, PFE	CVX, XOM	
Dec 2012	CSCO, IBM, INTC, MSFT	AXP, DIS, GE, HD, JPM, TRV, WMT	BA, CAT, DWDP, MMM, UTX	JNJ, MRK, PFE	CVX, XOM	
Jun 2013	CSCO, IBM, INTC, MSFT	AXP, DIS, GE, HD, JPM, TRV, WMT	BA, CAT, DWDP, MMM, UTX	JNJ, MRK, PFE	CVX, XOM	
Dec 2013	CSCO, IBM, INTC, MSFT	AXP, DIS, GE, HD, JPM, TRV, WMT	BA, CAT, DWDP, MMM, UTX	JNJ, MRK, PFE	CVX, XOM	
Jun 2014	CSCO, IBM, INTC, MSFT	AXP, DIS, GE, HD, JPM, TRV, WMT	BA, CAT, DWDP, MMM, UTX	JNJ, MRK, PFE	CVX, XOM	
Dec 2014	CSCO, IBM, INTC, MSFT	AXP, DIS, GE, HD, JPM, TRV, WMT	BA, CAT, DWDP, MMM, UTX	JNJ, MRK, PFE	CVX, XOM	
Jun 2015	CSCO, IBM, INTC, MSFT	AXP, DIS, GE, HD, JPM, TRV, WMT	BA, CAT, DWDP, MMM, UTX	JNJ, MRK, PFE	CVX, XOM	
Dec 2015	CSCO, IBM, INTC, MSFT	AXP, DIS, GE, HD, JPM, TRV, WMT	BA, CAT, DWDP, MMM, UTX	JNJ, MRK, PFE	CVX, XOM	
Jun 2016	CSCO, IBM, INTC, MSFT	AXP, DIS, GE, HD, JPM, TRV, WMT	BA, CAT, DWDP, MMM, UTX	JNJ, MRK, PFE	CVX, XOM	
Dec 2016	CSCO, IBM, INTC, MSFT	AXP, DIS, GE, JPM, TRV	BA, CAT, DWDP, MMM, UTX	JNJ, MRK, PFE	HD, WMT	CVX, XOM
Jun 2017	CSCO, IBM, INTC, MSFT	AXP, DIS, GE, JPM, TRV	BA, CAT, DWDP, MMM, UTX	JNJ, MRK, PFE	HD, WMT	CVX, XOM
Dec 2017	AXP, DIS, GE, JPM, TRV	BA, CAT, DWDP, MMM, UTX	CSCO, IBM, INTC, MSFT	JNJ, MRK, PFE	HD, WMT	CVX, XOM

Table C.7: Communities disclosed by landmark window, semestral snapshots from 1997 through 2017.

Year	Communities					
	1	2	3	4	5	6
Jun 1997	AXP, DIS, GE, JPM, MMM	CSCO, INTC, MSFT	DWDP, JNJ, KO, MRK, PFE, PG	CVX, XOM		
Dec 1997	CSCO, IBM, INTC, MSFT	AXP, DIS, JNJ, JPM, KO, MRK, PFE, UTX	BA, CVX, DWDP, GE, HD, MCD, PG, WMT, XOM			
Jun 1998	CSCO, INTC, MSFT	AXP, JPM	GE, JNJ, KO, MRK, PFE, PG, WMT	CVX, XOM		
Dec 1998	DWDP, MMM	CSCO, IBM, INTC, MSFT, PFE	GE, JNJ, KO, MCD, MRK, PG	AXP, HD, JPM, TRV, UTX, WMT	CVX, XOM	
Jun 1999	CSCO, IBM, INTC, MSFT	GE, JNJ, MRK, PFE	AXP, HD, JPM, WMT	CVX, XOM		
Dec 1999	AXP, GE, JPM	INTC, MSFT	JNJ, MRK, PFE	HD, WMT	CVX, XOM	
Jun 2000	AAPL, CSCO, INTC, MSFT	DWDP, MMM	JNJ, MRK, PFE	AXP, GE, HD, JPM, WMT	CVX, XOM	
Dec 2000	CSCO, INTC, MSFT	AXP, GE, JPM	JNJ, MRK	HD, WMT	CVX, XOM	
Jun 2001	CAT, DWDP	AAPL, CSCO, INTC, MSFT	AXP, GE, JPM	MRK, PFE	HD, WMT	CVX, XOM
Dec 2001	AAPL, CSCO, IBM, INTC, MSFT	AXP, BA, CAT, DIS, DWDP, GE, JPM, MMM, UTX	HD, WMT	CVX, XOM		
Jun 2002	AXP, DIS, GE, GS, JPM	BA, CAT, DWDP, MMM, UTX	AAPL, CSCO, IBM, INTC, MSFT	MRK, PFE	HD, WMT	CVX, XOM
Dec 2002	AAPL, CSCO, DIS, GS, IBM, INTC, JPM, MSFT	AXP, BA, CAT, DWDP, MMM, NKE, TRV, UTX	CVX, GE, HD, JNJ, MRK, PFE, VZ, WMT, XOM			
Jun 2003	AAPL, CSCO, GE, GS, IBM, INTC, MSFT	BA, CAT, DIS, DWDP, JPM, TRV, UTX, VZ	HD, KO, MMM, NKE, PG, WMT	AXP, CVX, JNJ, MRK, PFE, XOM		
Dec 2003	AAPL, CSCO, GS, IBM, INTC, JPM, MSFT, TRV	JNJ, MRK, PFE	HD, WMT	AXP, BA, CAT, CVX, DIS, DWDP, GE, MMM, UTX, XOM		
Jun 2004	CSCO, IBM, INTC, MSFT	AXP, CAT, DIS, DWDP, GE, GS, JPM, MMM, TRV	JNJ, MRK, PFE	BA, UTX	HD, WMT	CVX, XOM
Dec 2004	CSCO, INTC	AXP, DIS, DWDP, GE, GS, JPM	BA, UTX	HD, WMT	CVX, XOM	
Jun 2005	HD, IBM, NKE, WMT	CSCO, INTC	AXP, DWDP, GE, GS, JPM, MMM, UTX	CVX, XOM		
Dec 2005	AXP, GS, JPM	CAT, UTX	HD, WMT	CVX, XOM		
Jun 2006	AXP, GE, GS, JPM, KO	HD, WMT	CVX, XOM			
Dec 2006	AXP, GS, JPM, TRV	CVX, XOM				
Jun 2007	AXP, DIS, VZ	DWDP, GS, JPM, TRV, UTX	CVX, XOM			
Dec 2007	AAPL, CSCO, GE, IBM, INTC, MSFT, PG	BA, CAT, CVX, DIS, DWDP, KO, MMM, NKE, UTX, XOM	AXP, GS, HD, JPM, MRK, PFE, TRV, VZ, WMT			
Jun 2008	CSCO, IBM, INTC, MSFT	GE, JNJ, KO, PFE, PG, VZ	AXP, DIS, GS, HD, JPM, NKE, TRV, WMT	BA, CAT, CVX, DWDP, MMM, UTX, XOM		
Dec 2008	AAPL, AXP, CAT, CSCO, DWDP, GE, GS, HD, IBM, INTC, JPM, NKE, TRV, VZ	BA, CVX, DIS, JNJ, KO, MCD, MMM, MRK, MSFT, PFE, PG, UNH, UTX, WMT, XOM				
Jun 2009	AAPL, AXP, BA, CAT, CSCO, DWDP, GE, GS, HD, IBM, INTC, JPM, MMM, MSFT, NKE, UTX	CVX, DIS, JNJ, KO, MCD, MRK, PFE, PG, TRV, UNH, VZ, WMT, XOM				
Dec 2009	AAPL, AXP, CAT, CSCO, DWDP, GE, GS, IBM, INTC, JPM, MMM, MSFT	CVX, DIS, JNJ, KO, MRK, PFE, PG, TRV, VZ, XOM	BA, HD, MCD, NKE, UTX, WMT			
Jun 2010	CVX, DIS, DWDP, JNJ, KO, MCD, MRK, PG, VZ, WMT, XOM	AAPL, AXP, BA, CAT, CSCO, GE, GS, HD, IBM, INTC, JPM, MMM, MSFT, NKE, PFE, TRV, UTX				
Dec 2010	GE, GS, JPM, TRV	CSCO, CVX, DWDP, JNJ, KO, MCD, MRK, PFE, PG, UNH, UTX, VZ, XOM	AAPL, AXP, BA, CAT, DIS, HD, IBM, INTC, MMM, MSFT, NKE			
Jun 2011	INTC, JNJ, MMM, TRV, UTX	MRK, PFE	AXP, GE, GS, HD, JPM, VZ	AAPL, BA, CAT, CVX, DIS, DWDP, IBM, KO, MSFT, XOM		
Dec 2011	CVX, HD, JNJ, KO, MCD, MRK, NKE, PFE, PG, TRV, UNH, VZ, WMT, XOM	AAPL, AXP, BA, CAT, CSCO, DIS, DWDP, GE, GS, IBM, INTC, JPM, MMM, MSFT, UTX, V				

Continued on next page

Table C.8: Communities disclosed by gradual forgetting, semestral snapshots from 1997 through 2017.

Continued from previous page

Year	Communities					
	1	2	3	4	5	6
Jun 2012	CVX, GE, JNJ, KO, MCD, MMM, MRK, PFE, PG, TRV, UNH, VZ, XOM	AAPL, AXP, BA, CAT, CSCO, DIS, DWDP, GS, HD, IBM, INTC, JPM, MSFT, UTX, V				
Dec 2012	CAT, CSCO, GE, HD, IBM, INTC, MMM, MSFT, UTX, V	AXP, BA, DIS, DWDP, GS, JPM, TRV	CVX, JNJ, KO, MRK, PFE, PG, UNH, VZ, XOM			
Jun 2013	AXP, CAT, CVX, DWDP, GE, GS, IBM, JPM, TRV, XOM	DIS, JNJ, KO, MRK, PFE, PG	BA, HD, INTC, MMM, MSFT, UTX, V			
Dec 2013	JNJ, KO, PFE, PG	AXP, BA, GS, JPM, UTX	CAT, CVX, DIS, DWDP, GE, MMM, TRV, XOM			
Jun 2014	AXP, GE, GS, JPM, V	DIS, NKE	DWDP, MMM, TRV	BA, CAT, CVX, JNJ, UTX, XOM		
Dec 2014	JNJ, MRK, PFE, UNH	BA, CSCO, INTC, MMM, MSFT, TRV, UTX	AXP, DIS, DWDP, GS, HD, JPM, V	CAT, CVX, GE, XOM		
Jun 2015	BA, CSCO, IBM, JNJ, KO, MMM, MRK, PFE, PG, TRV, UNH	AXP, DIS, GS, UTX, V	HD, NKE, WMT	CAT, CVX, GE, JPM, XOM		
Dec 2015	AXP, DIS, JNJ, JPM, PFE, TRV, UNH	AAPL, BA, CSCO, KO, MCD, MMM, MRK, MSFT, PG, UTX, V, VZ	HD, NKE, WMT	CAT, CVX, GE, GS, IBM, INTC, XOM		
Jun 2016	AXP, BA, CAT, CVX, DWDP, GE, GS, IBM, JPM, MMM, UTX, XOM	AAPL, CSCO, DIS, HD, INTC, MSFT, NKE, V	MRK, PFE, UNH	JNJ, KO, MCD, PG, TRV, VZ		
Dec 2016	CSCO, INTC, MSFT, V	JNJ, MRK, PFE	KO, PG	GE, HD, IBM, MMM, NKE, TRV, UTX	AXP, BA, CAT, GS, JPM	CVX, XOM
Jun 2017	AXP, GS, JPM	MRK, PFE	BA, MMM, UTX	INTC, MSFT, V	CVX, XOM	
Dec 2017	AXP, GS, JPM	INTC, MSFT, V	CVX, XOM			

Table C.8 (cont): Communities disclosed by gradual forgetting, semestral snapshots from 1997 through 2017.

Year	Communities					
	1	2	3	4	5	6
Jun 1997	AXP, DWDP, GE, JPM, MMM	CSCO, INTC, MSFT	JNJ, KO, MRK, PFE, PG	CVX, XOM		
Dec 1997	BA, CSCO, GE, IBM, INTC, KO, MCD, MSFT	JNJ, MRK, PFE, TRV	AXP, CAT, HD, JPM, UTX, WMT	CVX, DIS, DWDP, PG, XOM		
Jun 1998	AXP, JPM	CSCO, INTC, MSFT	GE, JNJ, KO, MRK, PG, WMT	BA, UTX	CVX, XOM	
Dec 1998	AAPL, CSCO, IBM, INTC, MSFT, UTX	DIS, HD, JPM, MCD	AXP, DWDP, MMM, TRV	GE, JNJ, KO, MRK, PFE, PG, WMT	CVX, XOM	
Jun 1999	CAT, DWDP	AXP, IBM, JPM	GE, JNJ, MRK, PFE	CSCO, INTC, MSFT	HD, WMT	CVX, XOM
Dec 1999	AXP, GE, GS, HD, JPM	INTC, MSFT	JNJ, MRK, PFE	CVX, XOM		
Jun 2000	CAT, DWDP, MMM	AAPL, CSCO, IBM, INTC, MSFT	JNJ, MRK, PFE	AXP, GE, HD, JPM, WMT	CVX, XOM	
Dec 2000	CSCO, INTC, MSFT	AXP, GE, JPM	JNJ, MRK	CVX, XOM		
Jun 2001	BA, CAT, DWDP, MMM, UTX	AAPL, CSCO, IBM, INTC, MSFT	AXP, DIS, GE, GS, JPM	MRK, PFE	HD, WMT	CVX, XOM
Dec 2001	AAPL, CAT, CSCO, GS, IBM, INTC, JPM, MMM, MSFT, WMT	MRK, PFE	AXP, BA, DIS, DWDP, GE, HD, NKE, TRV, UTX	CVX, XOM		
Jun 2002	AAPL, CSCO, IBM, INTC, MSFT	BA, CAT, DWDP, MMM, UTX	KO, PG	DIS, VZ	HD, PFE, WMT	AXP, CVX, GE, GS, JPM, XOM
Dec 2002	AAPL, CSCO, DIS, GS, IBM, INTC, JPM, MSFT, VZ	AXP, BA, CAT, CVX, DWDP, GE, MRK, NKE, PFE, TRV, UTX, XOM	HD, JNJ, KO, MMM, PG, WMT			
Jun 2003	AAPL, CAT, CSCO, GE, GS, HD, IBM, INTC, MMM, MSFT, NKE, WMT	AXP, BA, CVX, DIS, DWDP, JNJ, JPM, KO, MRK, PFE, PG, TRV, UNH, UTX, VZ, XOM				
Dec 2003	JNJ, MRK, PFE	AAPL, CAT, CSCO, GS, IBM, INTC, MSFT, UTX, WMT	AXP, CVX, DWDP, GE, JPM, XOM			
Jun 2004	AXP, DIS, GS, HD, JPM, WMT	CSCO, INTC, MSFT	JNJ, MRK, PFE	BA, UTX	CAT, CVX, GE, MMM, XOM	
Dec 2004	CSCO, DWDP, GE, GS, JPM	AXP, DIS, VZ	HD, WMT	CVX, XOM		
Jun 2005	DWDP, GS, HD, IBM, JPM, MMM, NKE, UTX	AXP, CSCO, DIS, GE, INTC, JNJ, MSFT	MRK, PFE	TRV, VZ	CVX, XOM	
Dec 2005	AXP, GS, JPM	CAT, GE, KO, MMM, UTX	HD, WMT	CVX, XOM		
Jun 2006	GE, JNJ	AXP, CAT, DWDP, GS, JPM, MMM	CSCO, KO, MRK, PFE, PG, VZ	CVX, XOM		
Dec 2006	GE, IBM, INTC	AXP, TRV	GS, JPM, UTX, WMT	CVX, XOM		
Jun 2007	CSCO, INTC	HD, JNJ, JPM, KO, PFE, TRV, WMT	AXP, BA, DWDP, GS, UTX, VZ	CAT, CVX, DIS, GE, MSFT, PG, XOM		
Dec 2007	AAPL, CSCO, DIS, IBM, INTC, MMM, MSFT, PFE	AXP, GE, GS, HD, JNJ, JPM, MRK, NKE, TRV, WMT	CVX, KO, PG, VZ, XOM	BA, CAT, DWDP, MCD, UTX		
Jun 2008	CSCO, DIS, IBM, INTC, MCD, MSFT	JNJ, KO, MMM, PFE, PG, TRV, VZ	AXP, GE, GS, HD, JPM, NKE, WMT	BA, CAT, CVX, DWDP, UTX, XOM		
Dec 2008	AAPL, AXP, CAT, CSCO, DWDP, GE, GS, HD, IBM, INTC, JPM, TRV, V	BA, CVX, DIS, JNJ, KO, MCD, MMM, MRK, MSFT, NKE, PFE, PG, UNH, UTX, VZ, WMT, XOM				
Jun 2009	CVX, DIS, JNJ, KO, MCD, MRK, PFE, PG, TRV, UNH, XOM	AAPL, AXP, BA, CAT, CSCO, DWDP, GE, GS, HD, IBM, INTC, JPM, MMM, MSFT, NKE, UTX, V, VZ				
Dec 2009	AAPL, AXP, CSCO, GS, IBM, INTC, JPM, TRV	CVX, JNJ, MRK, PFE, PG, VZ, XOM	BA, CAT, DIS, DWDP, GE, HD, MMM, MSFT, UTX			
Jun 2010	CVX, IBM, JNJ, KO, MCD, MSFT, PG, TRV, UTX, VZ, WMT, XOM	AAPL, AXP, BA, CAT, CSCO, DIS, DWDP, GE, GS, HD, INTC, JPM, MMM, MRK, NKE, PFE, V				
Dec 2010	AXP, CSCO, DIS, GE, GS, IBM, INTC, JPM, MSFT, TRV	CVX, JNJ, MCD, MRK, PFE, PG, UNH, XOM	AAPL, BA, CAT, DWDP, HD, KO, MMM, NKE, UTX, VZ, WMT			
Jun 2011	BA, HD, IBM, INTC, KO, MSFT, UTX, WMT	AXP, DIS, GE, GS, JPM, VZ	CAT, CVX, DWDP, JNJ, MMM, TRV, XOM			
Dec 2011	AAPL, AXP, BA, CAT, CSCO, CVX, DIS, DWDP, GE, GS, HD, IBM, INTC, JNJ, JPM, KO, MCD, MMM, MRK, MSFT, NKE, PFE, PG, TRV, UNH, UTX, V, VZ, WMT, XOM					

Continued on next page

Table C.9: Communities disclosed by sliding windows, semestral snapshots from 1997 through 2017.

Continued from previous page

Year	Communities					
	1	2	3	4	5	6
Jun 2012	AAPL, BA, V	AXP, CAT, DIS, DWDP, GS, HD, JPM, UTX	CSCO, CVX, IBM, INTC, JNJ, KO, MSFT, XOM	GE, MMM, MRK, PFE, TRV		
Dec 2012	JNJ, KO, MCD, MRK, PFE, PG, TRV, VZ	BA, CAT, CSCO, CVX, DIS, DWDP, IBM, INTC, MMM, MSFT, UTX, XOM	AXP, GE, GS, HD, JPM, V			
Jun 2013	AXP, BA, CAT, CVX, DWDP, GS, JPM, MMM, TRV, UTX, V, XOM	GE, IBM	DIS, HD, JNJ, KO, PFE, PG			
Dec 2013	GE, JNJ, KO, PFE, PG, WMT	AXP, BA, DIS, DWDP, GS, JPM, MMM, TRV, UTX	CVX, XOM			
Jun 2014	BA, CAT, UTX	AXP, DIS, GS, JPM, NKE, V	CVX, DWDP, GE, JNJ, MMM, TRV, WMT, XOM			
Dec 2014	AXP, DWDP, HD, NKE, PG, V, VZ, WMT	CSCO, INTC, MMM, MSFT	DIS, GS, JNJ, MRK, PFE, UNH	BA, CAT, CVX, GE, JPM, TRV, UTX, XOM		
Jun 2015	CSCO, IBM, JNJ, KO, MCD, MMM, MRK, MSFT, NKE, PG	DIS, GS, PFE, TRV, UTX, V, VZ	AAPL, BA, HD, UNH, WMT	CAT, CVX, JPM, XOM		
Dec 2015	AXP, BA, DIS, GS, HD, JNJ, JPM, NKE, PFE, TRV, UNH, V, WMT	AAPL, CAT, CSCO, CVX, DWDP, GE, IBM, INTC, KO, MCD, MMM, MRK, MSFT, PG, UTX, VZ, XOM				
Jun 2016	BA, CAT, DIS, DWDP, GE, IBM, MMM, UTX	AAPL, GS, HD, INTC, JPM, MCD, MRK, MSFT, NKE, PFE, UNH, V	KO, PG, TRV, VZ	CSCO, CVX, JNJ, XOM		
Dec 2016	JNJ, MRK, PFE	KO, PG, VZ	AXP, CAT, GS, JPM	CSCO, IBM, INTC, MSFT, V	BA, CVX, GE, HD, MMM, UTX, XOM	
Jun 2017	AXP, GS, JPM	INTC, MSFT, V	CVX, XOM			
Dec 2017	AAPL, INTC, MSFT, V	CVX, XOM				

Table C.9 (cont): Communities disclosed by sliding windows, semestral snapshots from 1997 through 2017.

Appendix D

Frequent item sets and association rules in communities

stock-set	Support
CSCO,INTC,MSFT	0.33
AXP,GE,JPM	0.33
CSCO,IBM,INTC,MSFT	0.24
DWDP,MMM	0.22
CAT,DWDP,MMM	0.16
BA,DWDP	0.16
AXP,GE,JPM,TRV	0.16
AXP,DIS,GE,JPM	0.16
AXP,GE,HD,JPM,WMT	0.16

Table D.1: Frequent stock-sets in communities disclosed by landmark window, annual snapshots from 1997 through 2017.

Rule		Support	Confidence	Lift
AXP,GE,JPM,WMT	⇒ HD	0.16	1.00	6.30
AXP,GE,HD,JPM	⇒ WMT	0.16	1.00	6.30
CAT,DWDP	⇒ MMM	0.16	1.00	4.50
DWDP,MMM	⇒ CAT	0.16	0.71	4.50
BA	⇒ DWDP	0.16	1.00	4.20
DWDP	⇒ BA	0.16	0.67	4.20
CAT,MMM	⇒ DWDP	0.16	1.00	4.20
CSCO,IBM,MSFT	⇒ INTC	0.24	1.00	3.00
CSCO,IBM,INTC	⇒ MSFT	0.24	1.00	3.00
IBM,INTC,MSFT	⇒ CSCO	0.24	1.00	3.00
CSCO,INTC,MSFT	⇒ IBM	0.24	0.71	3.00
GE,JPM,TRV	⇒ AXP	0.16	1.00	3.00
AXP,JPM,TRV	⇒ GE	0.16	1.00	3.00
AXP,GE,TRV	⇒ JPM	0.16	1.00	3.00
DIS,GE,JPM	⇒ AXP	0.16	1.00	3.00
AXP,DIS,JPM	⇒ GE	0.16	1.00	3.00
AXP,DIS,GE	⇒ JPM	0.16	1.00	3.00
GE,HD,JPM,WMT	⇒ AXP	0.16	1.00	3.00
AXP,HD,JPM,WMT	⇒ GE	0.16	1.00	3.00
AXP,GE,HD,WMT	⇒ JPM	0.16	1.00	3.00

Table D.2: Association rules in communities disclosed by the landmark window, annual snapshots from 1997 through 2017.

stock-set	Support
CSCO,INTC	0.30
AXP,JPM	0.30
MRK,PFE	0.28
JNJ,MRK,PFE	0.26
INTC,MSFT	0.25
CSCO,INTC,MSFT	0.23
CSCO,IBM,INTC	0.19
AXP,GS,JPM	0.19
AXP,GE	0.19
BA,MMM	0.18
AXP,DWDP	0.18
BA,UTX	0.18
AXP,GE,JPM	0.18
KO,PG	0.16
JNJ,KO,MRK,PFE	0.16
AXP,CAT	0.16
CAT,GE	0.16
CAT,DWDP	0.16
AXP,DWDP,JPM	0.16
AXP,DIS	0.16

Table D.3: Frequent stock-sets in communities disclosed by gradual forgetting, annual snapshots from 1997 through 2017.

Rule		Support	Confidence	Lift
PG	⇒ KO	0.16	1.00	5.70
KO	⇒ PG	0.16	0.90	5.70
KO,MRK,PFE	⇒ JNJ	0.16	1.00	3.80
JNJ,KO,PFE	⇒ MRK	0.16	1.00	3.56
JNJ,MRK,PFE	⇒ KO	0.16	0.60	3.42
MMM	⇒ BA	0.18	0.77	3.37
BA	⇒ MMM	0.18	0.77	3.37
IBM,INTC	⇒ CSCO	0.19	1.00	3.35
CSCO,INTC	⇒ IBM	0.19	0.65	3.35
AXP,JPM	⇒ GS	0.19	0.65	3.35
JNJ,KO,MRK	⇒ PFE	0.16	1.00	3.35
CAT	⇒ DWDP	0.16	0.75	3.29
DWDP	⇒ CAT	0.16	0.69	3.29
GS,JPM	⇒ AXP	0.19	1.00	3.17
AXP,GS	⇒ JPM	0.19	1.00	3.17
DWDP,JPM	⇒ AXP	0.16	1.00	3.17
GE,JPM	⇒ AXP	0.18	1.00	3.17
BA	⇒ UTX	0.18	0.77	3.13
UTX	⇒ BA	0.18	0.71	3.13
INTC,MSFT	⇒ CSCO	0.23	0.93	3.11
CSCO,IBM	⇒ INTC	0.19	1.00	3.00
CSCO,MSFT	⇒ INTC	0.23	1.00	3.00
AXP,GE	⇒ JPM	0.18	0.91	2.88
AXP,DWDP	⇒ JPM	0.16	0.90	2.85
CAT	⇒ GE	0.16	0.75	2.67
GE	⇒ CAT	0.16	0.56	2.67
CSCO,INTC	⇒ MSFT	0.23	0.76	2.56
CAT	⇒ AXP	0.16	0.75	2.38
AXP	⇒ CAT	0.16	0.50	2.38
AXP,JPM	⇒ DWDP	0.16	0.53	2.32
AXP,JPM	⇒ GE	0.18	0.59	2.10
DIS	⇒ AXP	0.16	0.64	2.04
AXP	⇒ DIS	0.16	0.50	2.04

Table D.4: Association rules in communities disclosed by gradual forgetting, annual snapshots from 1997 through 2017.

stock-set	Support
GS,JPM	0.27
CSCO,INTC	0.25
AXP,JPM	0.25
MRK,PFE	0.24
INTC,MSFT	0.24
CSCO,MSFT	0.24
JNJ,PFE	0.22
CSCO,INTC,MSFT	0.22
AXP,GS	0.22
IBM,INTC	0.20
AXP,GS,JPM	0.20
JNJ,MRK,PFE	0.19
CSCO,IBM,INTC	0.19
CVX,XOM	0.17
AAPL,CSCO	0.17
CSCO,IBM,INTC,MSFT	0.17
GE,JPM	0.17
JNJ,KO,PFE	0.15
AAPL,CSCO,INTC	0.15
AAPL,CSCO,MSFT	0.15
BA,UTX	0.15
GE,GS	0.15
DIS,GS	0.15

Table D.5: Frequent stock-sets in communities disclosed by sliding windows, annual snapshots from 1997 through 2017.

Rule		Support	Confidence	Lift
XOM	⇒ CVX	0.17	0.91	4.88
CVX	⇒ XOM	0.17	0.91	4.88
KO,PFE	⇒ JNJ	0.15	1.00	4.54
JNJ,PFE	⇒ KO	0.15	0.69	4.54
CSCO,MSFT	⇒ AAPL	0.15	0.64	3.79
JNJ,KO	⇒ PFE	0.15	1.00	3.69
JNJ,MRK	⇒ PFE	0.19	1.00	3.69
MRK,PFE	⇒ JNJ	0.19	0.79	3.57
JNJ,PFE	⇒ MRK	0.19	0.85	3.57
CSCO,INTC	⇒ AAPL	0.15	0.60	3.54
CSCO,INTC,MSFT	⇒ IBM	0.17	0.77	3.49
AAPL,INTC	⇒ CSCO	0.15	1.00	3.47
AAPL,MSFT	⇒ CSCO	0.15	1.00	3.47
IBM,INTC,MSFT	⇒ CSCO	0.17	1.00	3.47
CSCO,IBM,MSFT	⇒ INTC	0.17	1.00	3.47
CSCO,IBM,INTC	⇒ MSFT	0.17	0.91	3.16
UTX	⇒ BA	0.15	0.69	3.14
BA	⇒ UTX	0.15	0.69	3.14
AAPL,CSCO	⇒ INTC	0.15	0.90	3.12
AAPL,CSCO	⇒ MSFT	0.15	0.90	3.12
AXP,GS	⇒ JPM	0.20	0.92	2.72
AXP,JPM	⇒ GS	0.20	0.80	2.62
GE	⇒ GS	0.15	0.69	2.27
GS	⇒ GE	0.15	0.50	2.27
GE	⇒ JPM	0.17	0.77	2.27
JPM	⇒ GE	0.17	0.50	2.27
GS,JPM	⇒ AXP	0.20	0.75	2.21
DIS	⇒ GS	0.15	0.60	1.97
GS	⇒ DIS	0.15	0.50	1.97

Table D.6: Association rules in communities disclosed by sliding windows, annual snapshots from 1997 through 2017.

stock-set	Support
CSCO,INTC,MSFT	0.33
AXP,GE,JPM	0.33
CSCO,IBM,INTC,MSFT	0.24
DWDP,MMM	0.21
BA,DWDP	0.18
BA,CAT,DWDP,MMM,UTX	0.15
AXP,GE,JPM,TRV	0.15
AXP,DIS,GE,JPM	0.15
AXP,GE,HD,JPM,WMT	0.15

Table D.7: Frequent stock-sets in communities disclosed by landmark window, bi-ennual snapshots from 1997 through 2017.

Rule		Support	Confidence	Lift
BA,DWDP,MMM,UTX	⇒ CAT	0.15	1.00	6.60
BA,CAT,DWDP,MMM	⇒ UTX	0.15	1.00	6.60
AXP,GE,JPM,WMT	⇒ HD	0.15	1.00	6.60
AXP,GE,HD,JPM	⇒ WMT	0.15	1.00	6.60
CAT,DWDP,MMM,UTX	⇒ BA	0.15	1.00	5.50
BA,CAT,DWDP,UTX	⇒ MMM	0.15	1.00	4.71
BA,CAT,MMM,UTX	⇒ DWDP	0.15	1.00	4.13
GE,JPM,TRV	⇒ AXP	0.15	1.00	3.00
AXP,JPM,TRV	⇒ GE	0.15	1.00	3.00
AXP,GE,TRV	⇒ JPM	0.15	1.00	3.00
CSCO,IBM,MSFT	⇒ INTC	0.24	1.00	3.00
CSCO,IBM,INTC	⇒ MSFT	0.24	1.00	3.00
IBM,INTC,MSFT	⇒ CSCO	0.24	1.00	3.00
CSCO,INTC,MSFT	⇒ IBM	0.24	0.73	3.00
DIS,GE,JPM	⇒ AXP	0.15	1.00	3.00
AXP,DIS,JPM	⇒ GE	0.15	1.00	3.00
AXP,DIS,GE	⇒ JPM	0.15	1.00	3.00
GE,HD,JPM,WMT	⇒ AXP	0.15	1.00	3.00
AXP,HD,JPM,WMT	⇒ GE	0.15	1.00	3.00
AXP,GE,HD,WMT	⇒ JPM	0.15	1.00	3.00

Table D.8: Association rules in communities disclosed by the landmark window, biennual snapshots from 1997 through 2017.

stock-set	Support
AXP,JPM	0.35
INTC,MSFT	0.32
CSCO,INTC	0.32
CSCO,INTC,MSFT	0.29
AXP,GE,JPM	0.29
CSCO,IBM,INTC	0.26
CSCO,IBM,INTC,MSFT	0.23
JNJ,MRK,PFE	0.19
BA,GE	0.19
BA,UTX	0.16
DWDP,MMM	0.16
AXP,CAT,GS,JPM	0.16
BA,CAT,DWDP	0.16
AXP,DWDP,GE,JPM	0.16
AXP,BA,GE,JPM	0.16

Table D.9: Frequent stock-sets in communities disclosed by gradual forgetting, biennual snapshots from 1997 through 2017.

Rule		Support	Confidence	Lift
AXP,CAT,JPM	⇒ GS	0.16	1.00	6.20
JNJ,MRK	⇒ PFE	0.19	1.00	5.17
JNJ,PFE	⇒ MRK	0.19	1.00	5.17
MRK,PFE	⇒ JNJ	0.19	1.00	5.17
BA,DWDP	⇒ CAT	0.16	1.00	5.17
AXP,GS,JPM	⇒ CAT	0.16	1.00	5.17
CAT,DWDP	⇒ BA	0.16	1.00	3.88
BA,CAT	⇒ DWDP	0.16	1.00	3.88
IBM,INTC,MSFT	⇒ CSCO	0.23	1.00	3.10
AXP,DWDP,JPM	⇒ GE	0.16	1.00	3.10
AXP,BA,JPM	⇒ GE	0.16	1.00	3.10
CSCO,INTC,MSFT	⇒ IBM	0.23	0.78	3.01
CSCO,IBM,MSFT	⇒ INTC	0.23	1.00	2.82
CAT,GS,JPM	⇒ AXP	0.16	1.00	2.82
AXP,CAT,GS	⇒ JPM	0.16	1.00	2.82
DWDP,GE,JPM	⇒ AXP	0.16	1.00	2.82
AXP,DWDP,GE	⇒ JPM	0.16	1.00	2.82
BA,GE,JPM	⇒ AXP	0.16	1.00	2.82
AXP,BA,GE	⇒ JPM	0.16	1.00	2.82
UTX	⇒ BA	0.16	0.71	2.77
BA	⇒ UTX	0.16	0.63	2.77
MMM	⇒ DWDP	0.16	0.71	2.77
DWDP	⇒ MMM	0.16	0.63	2.77
CSCO,IBM,INTC	⇒ MSFT	0.23	0.88	2.71
AXP,GE,JPM	⇒ DWDP	0.16	0.56	2.15
AXP,GE,JPM	⇒ BA	0.16	0.56	2.15

Table D.10: Association rules in communities disclosed by gradual forgetting, biennual snapshots from 1997 through 2017.

stock-set	Support
CSCO,INTC	0.26
MRK,PFE	0.26
GS,JPM	0.26
CSCO,INTC,MSFT	0.23
AXP,GE,JPM	0.23
KO,PG	0.19
JNJ,MRK,PFE	0.19
DWDP,JPM	0.19
HD,WMT	0.16
KO,XOM	0.16
JNJ,KO,MRK,PFE,PG	0.16
DWDP,GS,JPM	0.16
AXP,DWDP,GE,JPM	0.16
AXP,GE,GS,JPM	0.16

Table D.11: Frequent stock-sets in communities disclosed by sliding windows, biennual snapshots from 1997 through 2017.

Rule		Support	Confidence	Lift
KO,MRK,PFE,PG	⇒ JNJ	0.16	1.00	5.17
JNJ,KO,MRK,PFE	⇒ PG	0.16	1.00	5.17
XOM	⇒ KO	0.16	1.00	4.43
KO	⇒ XOM	0.16	0.71	4.43
AXP,DWDP,JPM	⇒ GE	0.16	1.00	4.43
AXP,GS,JPM	⇒ GE	0.16	1.00	4.43
JNJ,MRK,PFE,PG	⇒ KO	0.16	1.00	4.43
INTC,MSFT	⇒ CSCO	0.23	1.00	3.88
CSCO,INTC	⇒ MSFT	0.23	0.88	3.88
DWDP,GE,JPM	⇒ AXP	0.16	1.00	3.88
GE,GS,JPM	⇒ AXP	0.16	1.00	3.88
JNJ,KO,PFE,PG	⇒ MRK	0.16	1.00	3.88
JNJ,KO,MRK,PG	⇒ PFE	0.16	1.00	3.88
WMT	⇒ HD	0.16	0.83	3.69
HD	⇒ WMT	0.16	0.71	3.69
CSCO,MSFT	⇒ INTC	0.23	1.00	3.44
DWDP,JPM	⇒ GS	0.16	0.83	3.23
AXP,GE,JPM	⇒ DWDP	0.16	0.71	3.16
DWDP,GS	⇒ JPM	0.16	1.00	3.10
AXP,DWDP,GE	⇒ JPM	0.16	1.00	3.10
AXP,GE,GS	⇒ JPM	0.16	1.00	3.10
GS,JPM	⇒ DWDP	0.16	0.63	2.77
AXP,GE,JPM	⇒ GS	0.16	0.71	2.77

Table D.12: Association rules in communities disclosed by sliding windows, biennual snapshots from 1997 through 2017.

stock-set	Support
CSCO,INTC,MSFT	0.33
AXP,JPM	0.33
AXP,GE,JPM	0.32
DWDP,MMM	0.23
CSCO,IBM,INTC,MSFT	0.22
CAT,DWDP,MMM	0.15
BA,DWDP	0.15
AXP,GE,JPM,TRV	0.15
DIS,GE	0.15

Table D.13: Frequent stock-sets in communities disclosed by landmark window, semestral snapshots from 1997 through 2017.

Rule		Support	Confidence	Lift
CAT,DWDP	⇒ MMM	0.15	1.00	4.34
DWDP,MMM	⇒ CAT	0.15	0.66	4.34
BA	⇒ DWDP	0.15	1.00	4.20
DWDP	⇒ BA	0.15	0.63	4.20
CAT,MMM	⇒ DWDP	0.15	1.00	4.20
DIS	GE⇒	0.15	1.00	3.07
AXP,JPM,TRV	⇒ GE	0.15	1.00	3.07
IBM,INTC,MSFT	⇒ CSCO	0.22	1.00	3.00
CSCO,IBM,MSFT	⇒ INTC	0.22	1.00	3.00
CSCO,IBM,INTC	⇒ MSFT	0.22	1.00	3.00
CSCO,INTC,MSFT	⇒ IBM	0.22	0.67	3.00
GE,JPM,TRV	⇒ AXP	0.15	1.00	3.00
AXP,GE,TRV	⇒ JPM	0.15	1.00	3.00

Table D.14: Association rules in communities disclosed by the landmark window, semestral snapshots from 1997 through 2017.

stock-set	Support
AXP,JPM	0.28
INTC,MSFT	0.24
CSCO,INTC	0.23
GS,JPM	0.23
CSCO,MSFT	0.22
CSCO,INTC,MSFT	0.21
AXP,GS	0.21
MRK,PFE	0.20
AXP,GS,JPM	0.20
JNJ,PFE	0.19
JNJ,MRK	0.19
JNJ,MRK,PFE	0.17
GE,JPM	0.17
CSCO,IBM	0.16
IBM,INTC	0.16
AXP,GE,JPM	0.16
CVX,XOM	0.15
KO,PG	0.15
CSCO,IBM,INTC	0.15
IBM,INTC,MSFT	0.15
BA,UTX	0.15

Table D.15: Frequent stock-sets in communities disclosed by gradual forgetting, semestral snapshots from 1997 through 2017.

Rule		Support	Confidence	Lift
CVX	⇒ XOM	0.15	1.00	6.67
XOM	⇒ CVX	0.15	1.00	6.67
PG	⇒ KO	0.15	0.90	5.14
KO	⇒ PG	0.15	0.86	5.14
BA	⇒ UTX	0.15	0.82	3.78
UTX	⇒ BA	0.15	0.69	3.78
JNJ,PFE	⇒ MRK	0.17	0.87	3.73
MRK,PFE	⇒ JNJ	0.17	0.83	3.70
IBM,INTC	⇒ MSFT	0.15	0.95	3.67
IBM,INTC	⇒ CSCO	0.15	0.95	3.67
IBM,MSFT	⇒ INTC	0.15	1.00	3.64
JNJ,MRK	⇒ PFE	0.17	0.87	3.60
CSCO,MSFT	⇒ INTC	0.21	0.96	3.50
CSCO,INTC	⇒ MSFT	0.21	0.89	3.46
CSCO,IBM	⇒ INTC	0.15	0.95	3.44
CSCO,INTC	⇒ IBM	0.15	0.64	3.35
INTC,MSFT	⇒ CSCO	0.21	0.86	3.34
INTC,MSFT	⇒ IBM	0.15	0.62	3.24
AXP,GE	⇒ JPM	0.16	1.00	3.16
GE,JPM	⇒ AXP	0.16	0.95	3.08
AXP,GS	⇒ JPM	0.20	0.96	3.03
AXP,JPM	⇒ GS	0.20	0.73	2.91
GS,JPM	⇒ AXP	0.20	0.86	2.78
AXP,JPM	⇒ GE	0.16	0.58	2.23

Table D.16: Association rules in communities disclosed by gradual forgetting, semestral snapshots from 1997 through 2017.

stock-set	Support
AXP,JPM	0.25
GS,JPM	0.24
INTC,MSFT	0.24
CSCO,INTC	0.23
CSCO,MSFT	0.20
AXP,GS	0.20
CSCO,INTC,MSFT	0.19
AXP,GS,JPM	0.19
MRK,PFE	0.18
JNJ,PFE	0.18
IBM,INTC	0.18
JNJ,MRK	0.17
IBM,MSFT	0.17
CSCO,IBM	0.17
CVX,XOM	0.16
KO,PG	0.16
CSCO,IBM,INTC	0.16
DWDP,MMM	0.16
CAT,DWDP	0.15
CSCO,IBM,MSFT	0.15
IBM,INTC,MSFT	0.15

Table D.17: Frequent stock-sets in communities disclosed by sliding windows, semestral snapshots from 1997 through 2017.

Rule			Support	Confidence	Lift
XOM	⇒	CVX	0.16	1.00	6.26
CVX	⇒	XOM	0.16	1.00	6.26
PG	⇒	KO	0.16	0.86	4.11
KO	⇒	PG	0.16	0.76	4.11
IBM,INTC	⇒	CSCO	0.16	0.90	3.59
IBM,MSFT	⇒	CSCO	0.15	0.90	3.57
CSCO,MSFT	⇒	INTC	0.19	0.96	3.46
CSCO,IBM	⇒	INTC	0.16	0.95	3.43
CSCO,IBM	⇒	MSFT	0.15	0.90	3.35
MRK	⇒	PFE	0.18	0.81	3.34
PFE	⇒	MRK	0.18	0.76	3.34
CSCO,MSFT	⇒	IBM	0.15	0.75	3.31
INTC,MSFT	⇒	CSCO	0.19	0.82	3.26
IBM,MSFT	⇒	INTC	0.15	0.90	3.25
IBM,INTC	⇒	MSFT	0.15	0.86	3.19
CSCO,INTC	⇒	MSFT	0.19	0.85	3.17
CSCO,INTC	⇒	IBM	0.16	0.70	3.10
AXP,GS	⇒	JPM	0.19	0.96	3.00
CAT	⇒	DWDP	0.15	0.72	2.95
DWDP	⇒	CAT	0.15	0.62	2.95
INTC,MSFT	⇒	IBM	0.15	0.64	2.83
MMM	⇒	DWDP	0.16	0.68	2.78
DWDP	⇒	MMM	0.16	0.66	2.78
JNJ	⇒	MRK	0.17	0.63	2.75
MRK	⇒	JNJ	0.17	0.74	2.75
PFE	⇒	JNJ	0.18	0.72	2.69
JNJ	⇒	PFE	0.18	0.66	2.69
AXP,JPM	⇒	GS	0.19	0.77	2.68
GS,JPM	⇒	AXP	0.19	0.79	2.48

Table D.18: Association rules in communities disclosed by sliding windows, semestral snapshots from 1997 through 2017.