FEDERAL UNIVERSITY OF TECHNOLOGY - PARANÁ – UTFPR
GRADUATE PROGRAM IN APPLIED COMPUTING (PPGCA)

ADEMIR CRISTIANO GABARDO

A HEURISTIC TO DETECT COMMUNITY STRUCTURES IN DYNAMIC COMPLEX NETWORKS

DISSERTATION

CURITIBA

2014

ADEMIR CRISTIANO GABARDO

A HEURISTIC TO DETECT COMMUNITY STRUCTURES IN DYNAMIC COMPLEX NETWORKS

Dissertation presented to the Graduate Program in Applied Computing (PPGCA) at Federal University of Technology - Paraná – UTFPR as partial requirement to obtain a “Master of Science” degree – Area: Computer Engineering.

Dissertation Advisor: Heitor Silvério Lopes. PhD.

CURITIBA

2014

Dados Internacionais de Catalogação na Publicação

G112h 2014

Gabardo, Ademir Cristiano
A heuristic to detect community structures in dynamic complex networks / Ademir Cristiano Gabardo.-- 2014.
114 f.: il.; 30 cm

Texto em inglês
Dissertação (Mestrado) - Universidade Tecnológica Federal do Paraná. Programa de Pós-Graduação em Computação Aplicada, Curitiba, 2014
Bibliografia: f. 89-95

1. Redes sociais. 2. Redes complexas. 3. Mineração de dados (Computação). 4. Teoria dos grafos. 5. Algoritmos heurísticos. 6. Análise por conglomerados. 7. Detecção de comunidades. 8. Computação - Dissertações. I. Lopes, Heitor Silvério, orient. II. Universidade Tecnológica Federal do Paraná - Programa de Pós-graduação em Computação Aplicada. III. Título.

CDD 22 -- 621.39

Biblioteca Central da UTFPR, Câmpus Curitiba

Ministério da Educação
Universidade Tecnológica Federal do Paraná
Câmpus Curitiba
Diretoria de Graduação e Educação Profissional
Departamento Acadêmico de Informática – DAINF
Programa de Pós-Graduação em Computação Aplicada - PPGCA
Mestrado Profissional

ATA DA DEFESA DE DISSERTAÇÃO DE MESTRADO 18

DISSERTAÇÃO PARA OBTENÇÃO DO GRAU DE MESTRE EM **COMPUTAÇÃO APLICADA**

No dia 25 de agosto de 2014, às 10:00 horas, reuniu-se na Sala C-301 - bloco C - 3º andar do Câmpus Curitiba, a banca examinadora composta pelos professores doutores:

Prof. Gustavo Alberto Giménez Lugo, Dr. (Presidente)	UTFPR - CT
Prof. Heitor Silvério Lopes, Dr.	UTFPR - CT
Prof. Fabrício Martins Lopes, Dr.	UTFPR - CP
Prof. Murilo Vicente Gonçalves da Silva, Dr.	UTFPR - CT

sob Presidência de **Gustavo Alberto Giménez Lugo** para examinar a dissertação do candidato **ADEMIR CRISTIANO GABARDO**, intitulada**:** "A Heuristic to Detect Community Structures in Dynamic Complex Netwoks"**.** Após a apresentação, o candidato foi arguido pelos examinadores e foi dada a palavra aos presentes para formularem perguntas ao candidato. Os examinadores reunidos deliberaram pela **APROVAÇÃO** da dissertação.

O candidato foi informado que a concessão do referido grau, na área de concentração Engenharia de Sistemas Computacionais, está condicionada à (i) satisfação dos requisitos solicitados pela Banca Examinadora e lavrados na documentação entregue ao Candidato; (ii) entrega da dissertação em conformidade com as normas exigidas pela UTFPR; e (iii) entrega da documentação necessária para elaboração do Diploma. A Banca Examinadora determina um **prazo de 30 dias** para o cumprimento dos requisitos (desconsiderar esse parágrafo caso a dissertação seja reprovada).

_____

_____

Nada mais havendo a tratar, a sessão foi encerrada às 12:00, dela sendo lavrada a presente ata que segue assinada pela Banca Examinadora e pelo Candidato.

Prof. **Gustavo Alberto Giménez Lugo, Dr.**
presidente - (UTFPR - CT)

Prof. **Heitor Silvério Lopes, Dr.**
(UTFPR - CT)

Prof. **Fabrício Martins Lopes, Dr.**
(UTFPR - CP)

Prof. **Murilo Vicente Gonçalves da Silva**
(UTFPR - CT)

Candidato: _____

DECLARAÇÃO PARA A OBTENÇÃO DO GRAU DE MESTRE

A coordenação do Programa declara que foram cumpridos todos os requisitos exigidos pelo Programa de Pós-Graduação para a obtenção do grau de mestre.

Curitiba, ____de _____de 20____.

"A Ata de Defesa original está arquivada na Secretaria do PPGCA".

This work is dedicated to my parents who encouraged me since very young to appreciate the academic world.

ACKNOWLEDGEMENTS

Gratitude is a noble virtue. There are so many people to thank that there is not enough space here to name it all. I would like to extend my thanks very respectfully to the Great Architect of the Universe who has bestowed His blessing on us. Likewise, I thank my teacher and mentor Heitor Silverio Lopes.

I also like to thank the Court of Auditors of the State of Paraná (TCE/PR) for the scholarship which they awarded for this work to be accomplished. I am also grateful for the collaboration of the Court of Auditors of the State of Paraná (TCE/PR) for yielding data for analysis in this project. I would also like to express my thanks to colleagues from the laboratory of bioinformatics at the Federal Technological University of Paraná (UTFPR) Chidambaram, Hugo, André, César, Manassés, Jonas and Ismael for all their help.

I also thank the Graduate Program in Applied Computing (PPGCA) and their teachers for the valuable lessons and constant support, especially the examining board, teachers Heitor S. Lopes, Gustavo G. Lugo and Murilo V. G. Silva, who have contributed with valuable suggestions for this dissertation. I also extend a special thanks to Professor Fabricio M. Lopes for his availability to evaluate and contribute to this board of review.

I thank my faithful collaborator Gláucio Porcides Czekailo, who contributed to this work by helping to create the tools needed for this project.

I'm also very grateful to my wife Juliana Gabardo, who was sometimes unable to understand the meaning of the word silence but was always there to talk about vertices and edges and how these colorful full-of-lines drawings will help to make a better world.

In addition, I wish to thank my family who supported and encouraged me in this journey.

If I have ever made any valuable discoveries, it has been owing more to patient attention, than to any other talent. *Isaac Newton*

ABSTRACT

Gabardo, Ademir. . 115 f. Dissertation – Graduate Program in Applied Computing (PPGCA), Federal University of Technology - Paraná – UTFPR. Curitiba, 2014.

Complex networks are ubiquitous; billions of people are connected through social networks; there is an equally large number of telecommunication users and devices generating implicit complex networks. Furthermore, several structures can be represented as complex networks in nature, genetic data, social behavior, financial transactions and many other structures.

Most of these complex networks present communities in their structure. Unveiling these communities is highly relevant in many fields of study. However, depending on several factors, the discover of these communities can be computationally intensive. Several algorithms for detecting communities in complex networks have been introduced over time. We will approach some of them. Our goal in this work is to identify or create an understandable and applicable heuristic to detect communities in complex networks, with a focus on time repetitions and strength measures.

This work proposes a semi-supervised clustering approach as a modification of the traditional K-means algorithm submitting each dimension of data to a weight in order to obtain a weighted clustering method.

As a first case study, databases of companies that have participated in public bids in Paraná state, will be analyzed to detect communities that can suggest structures such as cartels.

As a second case study, the same methodology will be used to analyze datasets of microarray data for gene expressions, representing the correlation of the genes through a complex network, applying community detection algorithms in order to witness such correlations between genes.

Keywords: Social Networks, Complex Networks, Graphs, Data Mining, Clustering, Algorithms, Community detection.

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

G	– Graph.
V	– Set of vertices of a graph G.
E	– Set of edges of a graph G.
v	– A vertice of a graph G.
e	– An edge of a graph G.
$deg(v)$	– Degree of a vertice v.
$(d)G$	– Network Diameter of G.
$w(e)$	– Weight of an edge e.
$w(x,y)$	– Weight of an edge connection x to y.
A	– Adjacency Matrix.
$A(_i j)$	– Element of line i and column j from an Adjacency Matrix.
S_r	– Success rate.
C	– Community of a graph.
C_i	– Cluster belonging to the cluster set C.
$dist(u,v)$	– Distance between vertices u and v.
ρ	– Spearman Rank Correlation Coefficient.
τ	– Kendall Rank Correlation Coefficient.

CONTENTS

1 INTRODUCTION

The study of complex networks pervades a wide range of fields, from neurobiology to statistical physics. The theory of complex networks is addressed in the field of mathematics known as graph theory, evolving from purely mathematical models to faithful representations of complex systems present in nature, chemical processes, human behavior, food chains, and so forth.

Nowadays, complex networks are ubiquitous; billions of people are connected through social networks and there is an equally large number of telecommunications users forming implicit complex networks. The vast amount of social data available creates a scenario in which it is possible to trace several aspects of human life and behavior. Social network analysis has been used to portray social organization, alliances, political preferences and influence, among many other aspects that permeate society.

Most of the studied networks include groups or communities, which are clusters of vertexes highly similar or in which the vertices are strongly connected. These characteristics have been widely studied in several fields.

Community structures in networks are fundamental to understanding the structural and functional properties of a large network and also to denote which entities exist in networks and how they are kept together (NEWMAN, 2003b). In this sense, several algorithms have been created, with strengths and drawbacks, but still, some of them are not capable of dealing with large datasets, while others are unable to handle real-world complex networks such as the internet or small world networks (MILGRAM, 1967).

1.1 MOTIVATION

Complex networks can be represented by means of graphs, which are a mathematical abstraction of vertices in the network and their connections. Although complex network analysis is often referred to as a snapshot of the network, many of the complex networks under

study are dynamic. The relationships between vertices and from vertices to the community change dynamically (SANTORO et al., 2011).

Despite the dynamic characteristics of the complex networks, most of the techniques are mainly created for static networks and fail to capture the evolution of phenomena, their dynamical properties and the temporal dimension, focusing instead on structural or statistical aspects of the systems (CASTEIGTS et al., 2012). It is higly relevant to consider dynamic complex networks where connections can change over time.

1.1.1 GENERAL GOAL

The development of a heuristic to detect communities in dynamic complex networks.

1.1.2 SPECIFIC GOALS

- The development and use of a heuristic method for identifying communities in complex networks with temporal repetitions through clustering algorithms and techniques for complex network analysis.
- The creation of a variation of K-means algorithm using a weighted approach in order to obtain a supervised clustering algorithm.
- To apply the resulting heuristic to identify communities in different networks in order to validate the solution.
- The creation or identification of a similarity measure capable of depicting homogeneous communities in large networks.
- The creation or identification of a heuristic to trace these communities through a temporal analysis in search of repetitions.
- As a first case study: Applying the proposed heuristic method to a public bids database practiced in Paraná State to identify the formation of cartels.
- As a second case study: to apply the methodology developed to analyze gene expression data by means of complex networks in order to evaluate the correlation among genes.

2 GRAPHS, COMPLEX NETWORKS AND COMMUNITY DETECTION ALGORITHMS

This section briefly examines the historical background of complex networks and graphs. It also looks at the main principles of complex networks and graphs, metrics, algorithms and strategies for detecting communities in complex networks.

2.1 HISTORICAL BACKGROUND

Theories of complex networks have been addressed by a field of mathematics known as graph theory. The first studies date back to Euler's solution of the Königsberg bridges problem (ALBERT; BARABÁSI, 2002; ALEXANDERSON, 2006) illustrated in Figure 1.

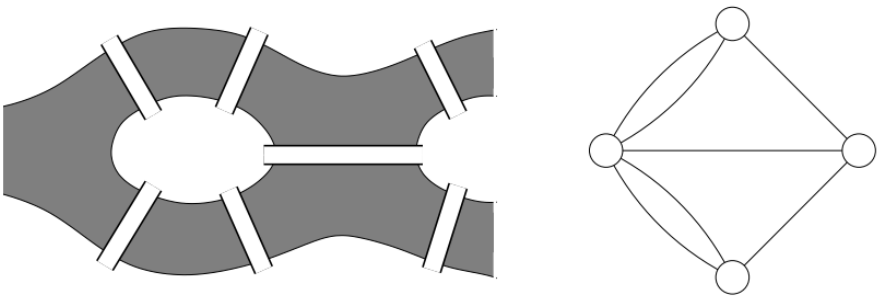


Figure 1: Left: a simplified depiction of the pattern of the rivers and bridges in the Königsberg bridge problem. Right: the corresponding network of vertices and edges (NEWMAN; BARABASI; WATTS, 2006).

Evolving from purely mathematical models of graphs from the Random Graph of the Erdős and Rényi (1960), to the small-world from Watts and Strogatz (1998), complex network analysis has encompassed graph theory and gone further to represent real world networks.

One of most popularly known social network experiments was conducted by the psychologist Stanley Milgram, known as the 'Small-World Experiment' (MILGRAM, 1967). The idea consists of a chain of 'friends of a friends' connecting any two people in a maximum of six steps. In this context, ordinary people can reach whoever target they want in the world if

they know the six correct (or maybe fewer) connections.

Even though not directly related to the graphs theory and complex networks or to detecting communities in graphs, Milgram's experiment has attracted attention to social network analysis and inspired many other researchers. As an area with a vast field of applications it is easy to find work on to social network analysis, an example can be addressed to Frank (1996), who studied cohesive subgroups of professionals and the influence of the group.

Most recently, Newman (2003) have conducted experiments with a collaboration network of scientists at the Santa Fé Institute to denote how the members interact and how they work as a team. He also conducted experiments with a food web of marine organisms. These are some examples of how community detection can be applied to denote a wide range of interactions between subjects.

Social network analysis is widely used nowadays to help understand how to conduct marketing (MANGOLD; FAULDS, 2009) and to measure political influence (MAYFIELD, 2005).

These are some examples of how graph theory, complex networks and social network analysis have evolved to a complex scenario where large amounts of data can be used to: solve or minimize routing problems, study biological networks, reveal social behavior and understand massive communication.

2.2 GRAPH

A graph is represented as a set of vertices connected by edges. Graphs are mathematical abstract representations of complex networks. A graph G can be defined as an ordered pair of vertices $G = (V,E)$ where,

- V is a set of vertices.
- E is a set of edges.

Figure 1 on the right shows a graph with four vertices and seven edges. The size of G is the number n of vertices in V and the order of G is the number L of edges in E.

2.2.1 VERTICES

A vertex is the fundamental unit of a graph. Vertices are fundamental and indivisible parts of a graph. Usually vertices are featureless and represented by an ID and a label, although

they may have supplementary features depending on the meaning of his application.

Figure 2 shows a graph consisting of vertices A, B, C, D, E and F and seven edges connecting them.

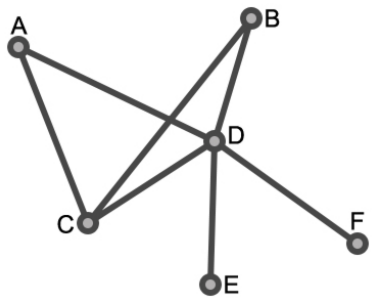


Figure 2: A Graph with six vertices.

Some examples of elements that can be represented as vertices in graphs: Individuals in a social network, genes in a network of gene expression or cities on a map.

2.2.2 EDGES

The simplest definition of an edge is a connection between two vertices. Edges are represented by the symbol e, and the set of edges in a graph is represented by the symbol E.

A definition for a non-weighted edge can be written as: $e \in E(G) = v_i, v_j$, where e is the edge itself, E is the set of all edges in graph G, v_i and v_j are vertices of G. For an edge $\{v_i, v_j\}$, a shorter notation is $v_i v_j$.

2.2.2.1 DIRECTION OF THE CONNECTIONS

A Graph could be either directed, undirected or mixed. In directed graphs an edge $e\{v_i, v_j\}$ could have a different value from the edge $e\{v_j, v_i\}$. Examples of directed graphs are: a traffic route into a highway system with one-way roads; sending a message from a sender to a recipient. etc.

For a non-directed graph the value for an edge $e\{v_i, v_j\}$ is the same for the edge $e\{v_j, v_i\}$. If $e\{v_i, v_j\} \in G$, $e\{v_j, v_i\} \in G$. Examples of non-directed graphs are: A highway system that connects points and allows traffic in both directions, protein chains or any other networks where is not possible to determine the origin and the destination of the information.

Graphs also can present both, undirected edges and directed edges in the same graph. In this case, the graph is classified as mixed.

2.2.2.2 CONNECTIONS WEIGHT

Edges in a graph may or may not have a weight. Edge weights define wheter a graph is weighted or not. In non-weighted graphs, all connections have the same value, usually set as 1 for an existing connection between two vertices. Otherwise, if there is no connection between two vertices, the edge could simply be omitted, or the value set as 0.

In weighted graphs, the edges can have values other than 0 or 1, and the value for the weight can be any continuous value. The weight of an edge can represent cost, distance or strength. For instance, taking a routing system represented by means of a graph the weights of the edges could be the cost of reaching one point from another. Another point in question is that it is possible to use the number of times that two individuals exchange messages as a metric to define the strength of the connection between them in a social network.

A weighted edge $e \in E(G)$ is usually represented as $\{v_i, v_j, w\}$, where w is the weight value.

2.2.2.3 VERTEX AND GRAPH DEGREE

Vertex degree is the counting of how many connections the vertex has (STEPHENSON; ZELEN, 1989). It shows direct contact between vertices (FREEMAN, 1977). Vertex degree is also known as local degree. The degree of a vertex v is denoted $\deg(v)$. Given a graph $G = (V;E)$, the degree can be computed by the Equation 1:

$$\sum_{v \in V} deg(v) = 2|E|, \tag{1}$$

where E is the number of edges in G. Minimum degree, also known as the minimum vertex degree, of a graph G is the smallest vertex degree in G. The maximum degree, also known as maximum vertex degree, of a graph G is the largest vertex degree in G (PEMMARAJU; SKIENA, 2003).

2.3 METRICS IN GRAPHS

In this section, we will examine some but not all the metrics of graphs and complex networks with a focus on which can be used as metrics in order to detect communities. Distance metrics, such as the shortest paths and Average Path Length, which is the number of steps

along the shortest paths for all possible pairs of network nodes (ALBERT; BARABÁSI, 2002) Betweenness Centrality, Closeness, Degree and Eigenvector, which are metrics used to show the relevance of a vertex in a network (PFEFFER; CARLEY, 2012). Moreover, some additional metrics of graphs and complex networks will be covered for the purpose of mathematical and topological comprehension of complex networks.

2.3.1 BETWEENNESS CENTRALITY

Betweenness Centrality is a metric in graphs used to define densely connected regions of a network. It takes in to account the fraction of the number of shortest paths that pass through a vertex over all pair of vertices (FREEMAN, 1977). Using Betweeness Centrality, it is possible to compute the influence that a distinct vertex has over the network. Girvan and Newman (NEWMAN, 2005) used Betweenness Centrality as a metric to detect communities in graphs by progressively removing the edges with higher Betweenness Centrality. At each step, all edge Betweenness must be recalculated, resulting in a drawback: the highly computational cost of the algorithm (COSTA; RODRIGUES; TRAVIESO G. ANDVILLAS BOAS, 2005).

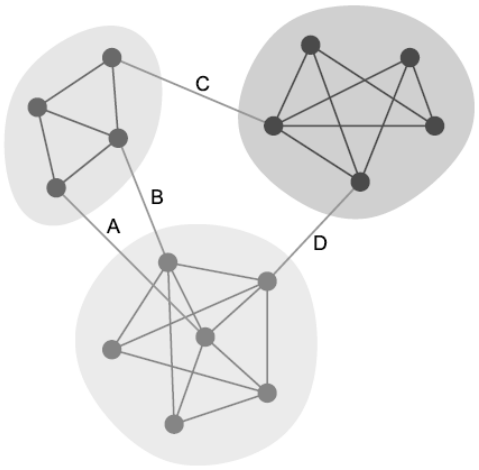


Figure 3: Vertices A,B,C and D connect distinct subgroups in the network. These vertices have high Betweenness Centrality (NEWMAN, 2012).

The Betweenness Centrality $BC(v)$ of a vertex $v \in V$ is the sum of all pairs of vertices $v_i; v_j \in V$, of the fraction of shortest paths between v_i and v_j that pass through v as shown in Equation 2:

$$BC(v) = \sum_{v_i \neq v_j \neq v} \frac{\sigma_{v_i v_j}(v)}{\sigma_{v_i v_j}}, \tag{2}$$

where $\sigma_{v_i v_j}(v)$ denotes the total number of shortest paths between v_i and v_j that pass through vertex v, and $\sigma_{v_i v_j}$ denotes the total number of shortest paths between v_i and v_j.

A simple example of this concept is shown in Figure 3 Vertices A,B,C and D connect distinct subgroups in the network. These vertices have high Betweenness Centrality. It is easy to see that by removing the edges A, B, C and D, the three groups in this figure are separated, revealing the presence of communities.

2.3.2 CLOSENESS CENTRALITY

Closeness centrality is a metric that evaluates how far a vertex is in relation to all other vertices in a graph (FREEMAN, 1977; STEPHENSON; ZELEN, 1989). Intuitively, it is easy to note that the most central a vertex is, the lower the total distance is from that vertex to all the other vertices. The closeness centrality can be computed using the Equation 3:

$$c(i) = \sum_{j} \left[d_{ij} \right]^{-1}, \tag{3}$$

where i is the focal vertex, j is another vertex in the graph, and d_{ij} is the shortest path between these two vertices.

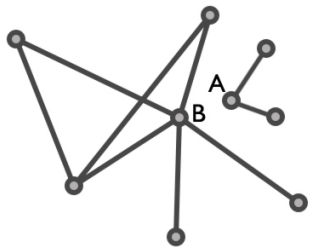


Figure 4: An example where Closeness Centrality is infinite, with no path from A to B.

Taking the graph in Figure 4 as an example, there is a well-know issue with closeness centrality: the distance between vertex A and vertex B is infinite as no path exists between them.

As long as at least one vertex is unreachable by the others, the sum of distances to all other vertices is infinite. This peculiarity means that closeness centrality is usually applied to the largest component.

A vertex in the center of a graph is closest to all the other vertices relative to a vertex positioned at the periphery of the graph. Eventually, the vertices located at the periphery are closer to some other vertices, but possibly many others are more distant compared to a vertex

in the center of the graph (WASSERMAN, 1994).

2.3.3 CONNECTED COMPONENTS

The key feature of a connected component is that from any vertex of the component there is at least one path to all the other vertices of the same component. A connected component, or component in an undirected graph, is also called a sub-graph (ERDŐS; RÉNYI, 1960).

It is possible to discover the connected components of a graph by using either breadth-first search or depth-first search. The search starts at any vertex v until the entire connected component containing v is found, restarting a new search whenever it reaches a vertex that has not been included in a previously found connected component (HOPCROFT; TARJAN, 1973). This procedure is illustrated by pseudocode shown in Algorithm 1.

Algorithm 1: Depth-first Search Algorithm for finding connected components.

```
dfs(vertex u);
foreach vertex v connected to u do
    visited[v] = true;
    dfs(v);
end
foreach vertex u do
    if u is not visited then
        visited[u] = true;
        component ++;
    end
    dfs(vertex u);
end
```

2.3.4 CLUSTERING COEFFICIENT

Clustering coefficient is a metric used to evaluate the existence of communities in a graph by computing the tendency of the vertices to cluster together. There are two Clustering Coefficients: the global clustering coefficient and the local clustering coefficient.

The global clustering coefficient is based on triplets of nodes and measures the number of closed triplets or triangles. The local clustering coefficient is relative to the number of connections to a particular vertex, the proportion between the number of connections to a vertex

and the total number of possible connections between the vertex and its neighbors (NEWMAN, 2003c). Local undirected Clustering Coefficient formula is show in Equation 4:

$$C_i = \frac{2|\{e_{jk} : v_i, v_j \in N_i, e_{jk} \in E\}|}{K_i(K_i - 1)}, \tag{4}$$

where C_i is the local clustering coefficient, K_i is the number of neighbors of a vertex, v_i and v_j represent the vertices in the graph from i to j, and N_i represents the neighbourhood of a vertex that is defined by its immediately connected neighbours.

2.3.5 DEGREE DISTRIBUTION

The degree of a vertex is related to the number of connections that this vertex has. Real networks such as the Internet, Social networks, and Scientific Collaboration Networks, usually have some features related to degree distribution. Vertices with small degrees are the most frequent, although high degree vertices do exist, albeit in fewer numbers. The fraction of highly connected nodes decreases, but is not zero.

Many real-world networks have a degree distribution that follows a power law or has a long tail (BARABÁSI; ALBERT, 1999). These characteristics provide a glimpse of the topology of the network. The degree distribution of a graph is usually presented by means of a histogram of the network, which lists the distinct degrees in the network.

By computing the frequency of each degree, we form the degree distribution $P_{deg}(k)$, as defined in Equation 5:

$$P_{deg}(k) = \nu, \tag{5}$$

where ν is the frequency of that particular degree, or simply the fraction of nodes in the graph with degree k.

Figure 5 shows a random graph which follows the Erdős and Rényi model on the left and its respective chart with degree distribution on the right. The size of the graph is 100, the average degree is 1.9, and the clustering coefficient is 0.004.

Figure 6 shows a network which follows the Barabási and Albert model and its

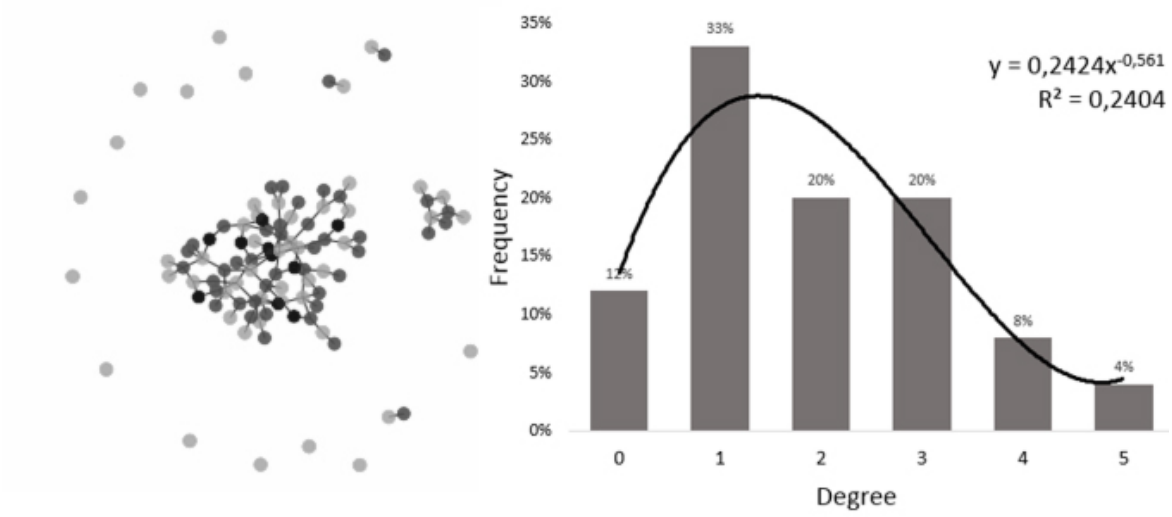


Figure 5: On the left, an Erdős and Rényi Random network. On the right, the chart with the degree distribution for this network.

respective chart with degree distribution on the right. The size of the graph is 102, the average degree is 3.9, and the clustering coefficient is 0.102, with a degree distribution that follows a power low.

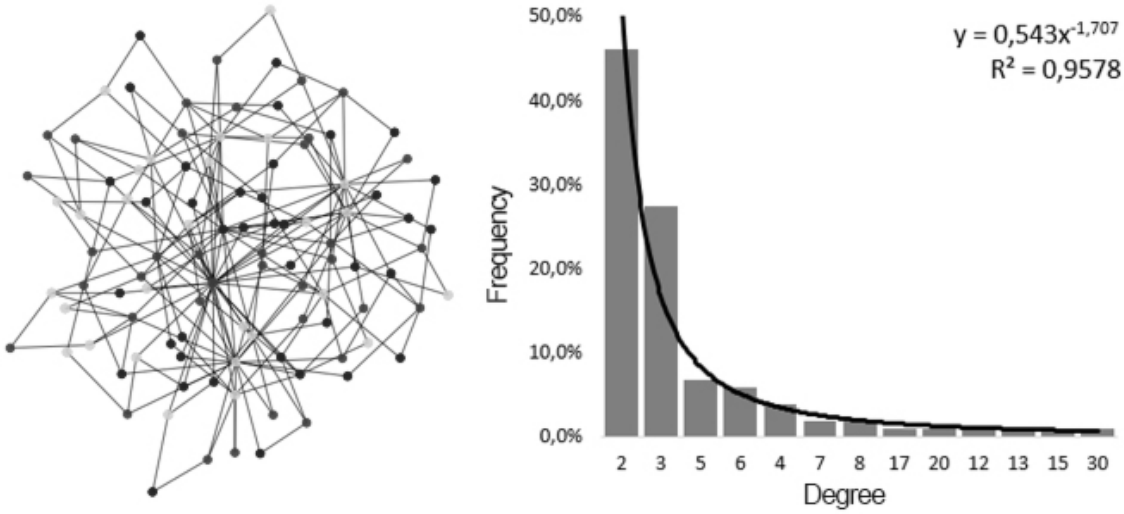


Figure 6: On the left, a Barabási and Albert network. On the right, the chart with the degree distribution for this network.

2.3.6 DISTANCE BETWEEN VERTICES

Although simple, this is an important concept in graph theory. The vertex distance is a mathematical measure of distance from one vertex to another. The distance $d(v_i, v_j)$ between two vertices v_i and v_j of a finite graph is the minimum length of connection between these two vertices. Distance also can be a dimension of cost.

The geodesic distance between two vertices in a graph is the length of any shortest path between them. Given a graph G, the distance $d(v_i, v_j)$ between two vertices v_i and v_j is the length of the shortest path from v_i to v_j, taking into account all possible paths in G from v_i to v_j. If there is no path from v_i to v_j, then $d(v_i, v_j)$ is infinite. Furthermore, the distance from any node to itself is zero.

2.3.7 MODULARITY

Modularity is a quality index for clustering. It is a metric which aims to demonstrate the division of a network into modules (NEWMAN, 2003; NEWMAN; GIRVAN, 2004). A module presents dense connections between the vertices within the modules and sparse connections between nodes in different modules.

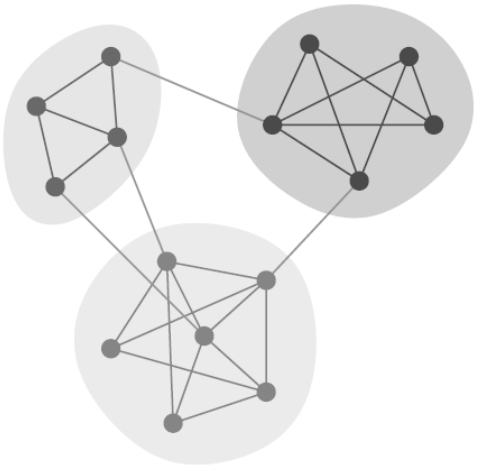


Figure 7: Example of a network showing community structure. This network is divided into three groups, with most connections within groups and few connections between groups (NEWMAN, 2012).

The modularity of a graph $G(V,E)$ can be computed by Equation 6:

$$Q = \sum_{u} \left\{ \frac{e_{uv}}{2m} - \left(\frac{a_u}{2m} \right)^2 \right\} \tag{6}$$

where e_{uv} is the total number of edges between cluster u and v, a_u is the total number of edges that are attached to vertices in cluster u and m is the total number of edges in the whole graph. $\frac{a_u}{2m}$ is the expected fraction of edges of u, which can be obtained when we assume the graph to be a random graph. As a result, well clustered graphs will show high modularity, since the value of e_{uv} is higher than in a random graph (SHIOKAWA; FUJIWARA; ONIZUKA,

2013).

Figure 7 shows a small network with three communities. Modularity measures aim to reveal the communities. Modularity is also a measure of the quality of a particular division of a network.

2.3.8 NETWORK DIAMETER

The network diameter (d), also know as the geodesic distance, is the length of the shortest path between the vertices of a graph that are farthest apart. In a disconnected graph the diameter is infinite (WEST et al., 2001).

A network diameter definition is: The length $max_{u,v}d(u,v)$ of the longest shortest path between any pair of vertices (u,v) in G, where $d(u,v)$ is the graph distance.

2.4 COMPLEX NETWORKS

Complex networks are suitable for representing complex systems (AMARAL; OTTINO, 2004). Some examples of these systems are: Ecosystems, the internet, social networks, spread of diseases, routes of roads, etc..

A complex network is a graph with non-trivial topological features, a complex network that shows features not present in simple graphs and classical networks. Features that characterize complex networks are a heavy tail in the degree distribution, a high clustering coefficient, high assortativity or disassortativity, community structure, and hierarchical structure (ALBERT; BARABÁSI, 2002).

A central aspect to the study of networks is discovering, characterizing and modeling the structure of the network. The study of the topology of complex networks leads to understanding phenomena such as the presence of cohesive groups in communities and networks.

A Complex Network Model was developed by Barabási and Albert, with characteristics of a scale-free network showing degree distribution that follows a power law (long tail) (BARABÁSI; ALBERT, 1999). A detailed version of this model is shown in the Section 2.6.2.

Watts and Strogatz also developed a complex network model known as Small World Network, with a high clustering coefficient (WATTS; STROGATZ, 1998). This model is show in detail in Section 2.6.3.

In addition to the models of complex networks developed by Barabási and Albert and Watts and Strogatz in this study, we will address real-world networks via two distinct case studies that aim to observe the phenomena described by the mathematical models of complex networks and the detection of communities in these complex networks.

2.5 REPRESENTING COMPLEX NETWORKS AS GRAPHS

2.5.1 ADJACENCY MATRIX

The Adjacency Matrix $A = [x_a b]$ for G is a matrix with n rows and n columns and entries given by: $x_a b = 1$ if (a,b) is an edge in G, otherwise it will be equal to 0.

Figure 8 illustrates a graph of size 3 and order 2. The adjacency matrix shown in Equation 7 illustrates the connections between the vertices of the graph shown in Figure 8.

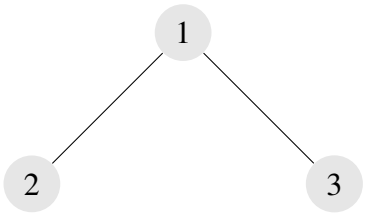


Figure 8: A small graph of size 3 and order 2.

$$A = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix} \tag{7}$$

The value that references the vertex itself (self loops) is set as zero. In weighted graphs, values other than 0 and 1 can be assumed as the weight of the connection.

2.5.2 ADJACENCY LIST

An adjacency matrix $A(ij)$ has i rows and j columns, where $i = j$. The size of dimensions i and j corresponds to the size of a graph G. As a result, the number of elements of the adjacency matrix corresponds to the square of the number of vertices in a graph G.

For large graphs, this representation may not be the most appropriate. Even unconnected vertices are represented in the adjacency matrix, consuming computational storage and processing resources.

To overcome this issue, an adjacency list can be used. In this kind of representation, row i of the matrix contains the vertices adjacent to vertex i. Each vertex i has a variable $d[i]$ that keeps the degree of vertex i. Only non-zero values are represented in the list, and self loops are usually ignored, resulting in a more compact representation, which is desirable for complex networks with large amounts of data.

The Adjacency List Al_i shown in Equation 8 exemplifies the connections between the vertices of the graph shown in Figure 8.

$$Al_i = \{\{1,2\},\{1,3\}\} \tag{8}$$

2.6 REAL WORLD NETWORKS AND NETWORK MODELS

Real World networks have properties such as those observed in the small-world networks. Real World networks show a scale-free degree distribution with clustered local neighborhoods and low average-shortest path. Such characteristics are derived from preferential attachment and connectivity distribution that decays as a power law (AMARAL et al., 2000). It is possible to evaluate whether a complex network corresponds to a real world network through the calculation of its diameter as a function of network size (WATTS, 1999).

To enable the evaluation of models and algorithms, it is very important to be able to produce graphs and complex networks in accordance with the already known topology graphs (BATAGELJ; BRANDES, 2005). In this section the three most well-known network models implemented in this work are presented.

2.6.1 ERDŐS AND RÉNYI MODEL

The Erdős and Rényi Model (ERDŐS; RÉNYI, 1960) is a model of graphs in which the connections are made for each pair of nodes with equal probability, independent from other edges. It is a completely random organization of a network. This model is widely used to test network properties and metrics.

In this work, we have implemented a random graph generator which follows the Erdős and Rényi model and receives two values as parameters. The parameters are the number of vertices n and the probability p. Based on probability, the number of edges incident to each vertex is set.

Figure 9 shows a network sample generated by the Erdős and Rényi Model produced

by the Random Network Generator Algorithm presented in Algorithm 2.

Algorithm 2: Random Network Generator Algorithm (BATAGELJ; BRANDES, 2005).

Input: number of vertices n, edge probability $0 > p > 1$
Result: $G = (\{0,...,n-1\}, E) \in G(n,p)$
initialization;
$E \leftarrow 0$; // Initialize the number of edges.
$v \leftarrow 1$; // Initialize the number of vertices.
while *(*$v < n$*)* **do**
  draw $r \in [0,1)$ uniformly at random;
  $w \leftarrow w + 1 + [log(1-r)/log(1-p)]$; // Calculates the number of Edges in the graph.
  while $w \leq v$ *and* $v < n$ **do**
    $w \leftarrow w - v; v \leftarrow v + 1$; // Update the number of vertices and edges in the graph.
  end
  if $v < n$ **then**
    $E \leftarrow E \cup \{v, w\}$
  end
end

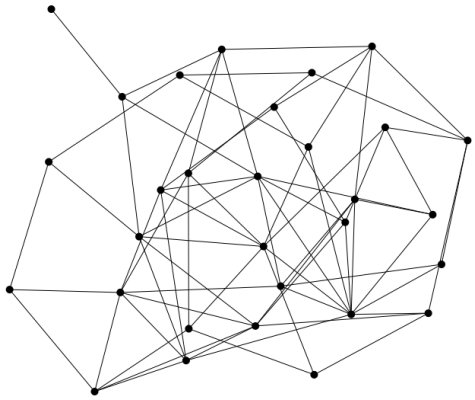


Figure 9: Erdős and Rényi Model network sample.

2.6.2 BARABÁSI AND ALBERT MODEL

Differing from the completely random model of Erdős and Rényi, the Barabási and Albert (BARABÁSI; ALBERT; BONABEAU, 2003; BARABÁSI; ALBERT, 1999) (BA) model generates Scale-free random networks in which the connectivity probability follows a

power-law degree distribution. The algorithm is named for its inventors Albert-László Barabási and Réka Albert.

The model proposed by Barabási and Albert follows the idea of the rich-get-richer, which states that a vertex with a high degree has a higher probability of receiving new connections than vertices with a lower degree. Many of the complex networks in the real world follow this kind of distribution, using a preferential attachment mechanism, such as the internet, human-made systems or social interaction. Figure 10 shows a Barabási and Albert scale-free model network sample created using our implementation of the Barabási and Albert network generator.

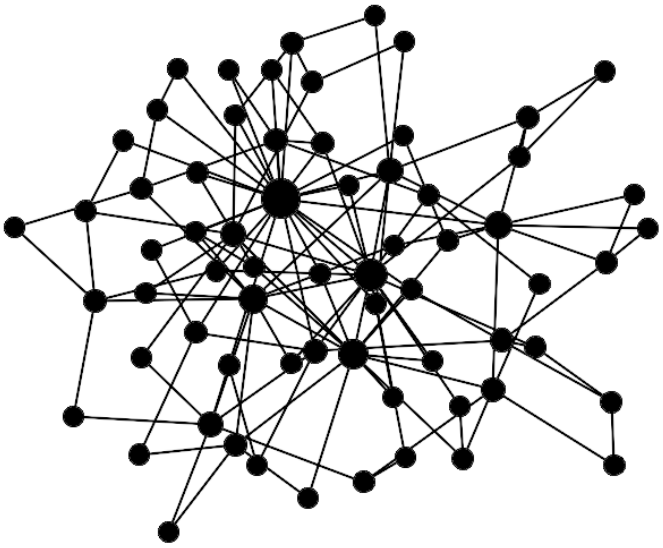


Figure 10: Barabási and Albert scale-free model network sample.

In order to generate instances of Barabási and Albert network we used a generator that receives a single parameter, which is the number of links per step. The network shown in Figure 10 was created with our Barabási and Albert network generator, as shown in Algorithm 3.

Algorithm 3: Barabási and Albert Model Network Generator Algorithm (BATAGELJ; BRANDES, 2005).

Input : Positive integer $n > 1$ and minimum degree $d \geq 1$.

Output: Scale-free network with n vertices.

$G \leftarrow \overline{K_n}$ `/* vertex set is` $V = \{0, 1, \ldots, n-1\}$ `*/`

$M \leftarrow$ list of length $2nd$

for $v = 0, 1, \ldots, n-1$ **do**

  for $i = 0, 1, \ldots, d-1$ **do**

    $M[2(vd+i)] \leftarrow v$

    $r \leftarrow$ draw uniformly at random from $\{0, 1, \ldots, 2(vd+i)\}$

    $M[2(vd+i)+1] \leftarrow M[r]$

  end

end

add edge $(M[2i], M[2i+1])$ to G for $i = 0, 1, \ldots, nd-1$

return G

2.6.3 WATTS AND STROGATZ MODEL

The Watts and Strogatz (WATTS; STROGATZ, 1998) small-world model is a complex network model designed to generate complex networks with small world properties, including short average path lengths and a high clustering coefficient. These networks are characterized by the presence of community structures. Figure 11 shows a Watts and Strogatz Small World network sample.

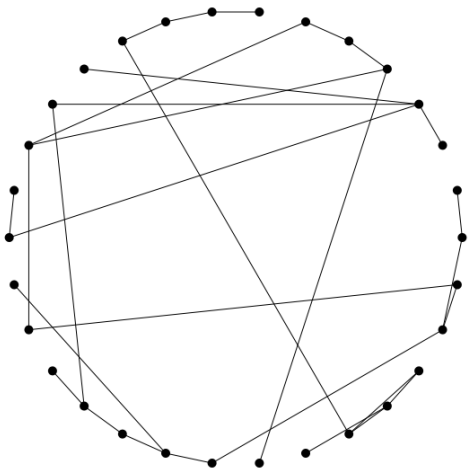


Figure 11: Watts and Strogatz Small World network sample.

We also implemented a generator for Watts and Strogatz networks in order to obtain

graphs with small world characteristics. The three complex network models were used in order to produce graphs with well-known topology and use the metrics from these networks in comparison with the metrics computed in the Case Study I and II. The network shown in Figure 11 was created with our Watts and Strogatz network generator whose pseudo-code is presented in Algorithm 4.

Algorithm 4: Watts and Strogatz Generator Algorithm.

Input : Positive integer n denoting the number of vertices. Positive even integer k for the degree of each vertex, where $n \gg k \gg \ln n \gg 1$. In particular, k should satisfy $0 < k < n/2$. Rewiring probability $0 < p \leq 1$.

Output: A Watts-Strogatz network with n vertices.

$M \leftarrow nk$ `/* sum of all vertex degrees = twice number of edges */`

$r \leftarrow$ draw uniformly at random from interval $(0,1)$

$v \leftarrow 1 + \lfloor \ln(1-r)/\ln(1-p) \rfloor$

$E \leftarrow$ contiguous edge list of k-circulant graph with n vertices

while $v \leq M$ **do**

  $u \leftarrow$ draw uniformly at random from $[0,1,\ldots,n-1]$

  if $v-1$ is even **then**

    while $E[v] = u$ or $(u, E[v]) \in E$ **do**

      $u \leftarrow$ draw uniformly at random from $[0,1,\ldots,n-1]$

    end

  else

    while $E[v-2] = u$ or $(E[v-2], u) \in E$ **do**

      $u \leftarrow$ draw uniformly at random from $[0,1,\ldots,n-1]$

    end

  end

  $E[v-1] \leftarrow u$

  $r \leftarrow$ draw uniformly at random from interval $(0,1)$

  $v \leftarrow v + 1 + \lfloor \ln(1-r)/\ln(1-p) \rfloor$

end

$G \leftarrow \overline{K_n}$

add edges in E to G

return G

2.7 SIMILARITY MEASURES

Similarity measures aim to assess how similar or dissimilar two objects are. The term “similarity” should be understood as mathematical similarity (SPEARMAN, 2010). The concepts of similarity and distance are the roots for clustering algorithms grouping individuals or network vertices.

Given the set of objects, the goal of clustering is to assign them to groups, based on their mutual similarity. In social networks, we can define clusters as a collection of individuals with dense friendship patterns internally and sparse friendships externally. Vertices assigned to the same cluster should be highly similar; vertices assigned to different clusters should be highly dissimilar.

Communities can be also considered as entities with their own autonomy in the graph (FORTUNATO, 2010). A graph cluster can also be a subgraph, which may or may not be connected to other clusters or to the graph itself. In the sense of a complex network, a community is a group of vertices with similarity between them. This measure depends on which attribute is used to build the edges of the network.

The similarity measures may be based on one or more dimensions of data. Figure 12 shows five distinct clusters of objects in a three-dimensional space organized according to their similarity.

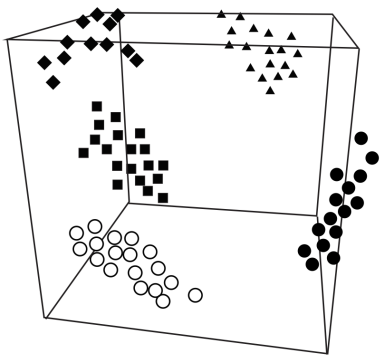


Figure 12: Five distinct clusters of objects in a three-dimensional space organized according to their similarity.

In order to evaluate how similar or dissimilar two or more objects are, it is possible to use correlation measures such as: Spearman’s Rank Correlation, Kendall’s Rank Correlation and Pearson’s Correlation Coefficient, which are covered in detail in appendices C.1, C.2 and C.3, respectively.

In addition to correlation methods, clustering algorithms such as K-means may also

use distance measures such as Manhattan or Chebychev distance between objects as a criterion for grouping them.

2.8 COMMUNITIES IN COMPLEX NETWORKS

Communities in complex networks can be characterized in several ways depending on their subject and the context in which they are analyzed. In terms of similarity, communities can be groups of objects with a high similarity between community members and a high degree of dissimilarity with objects that are outside the community.

In social network analysis, in addition to similarity measures, it is necessary to evaluate interactions between individuals.

In other networks, such as telecommunications networks, the identification of communities is based on which vertices are interconnected, as well as their paths and weights of connections.

These may also be cases where both items of information are relevant, and it is necessary to evaluate both similarity measures and the network topology.

"Studies of communities in networks go back at least to the 1970s" (NEWMAN, 2012). Community detection algorithms are strongly attached to the theories of Graph Partitioning and Hierarchical Clustering. Many techniques have been employed for clustering data, e.g.: Graph Degree Linkage, Hierarchical Clustering Algorithms, Nearest Neighbor Clustering and Partition Algorithms.

At the macroscopic level, there are global properties, such as network distance, graph diameter, the longest path and the shortest path. At the microscopic level, there are properties related to the vertices, mainly degree distribution and the clustering coefficient.

Another approach to detecting communities in complex networks is the graph partition method. A partition is a network divided into clusters, where each vertex belongs to one cluster at least (FORTUNATO, 2010). A definition for graph partitioning is: given a graph consisting of vertices and edges, divide the vertices into sets of equal size, so that the number of edges between these parts is minimized. This technique is also know as Edgecut (BONDY; MURTY, 1976).

We can also cite the hierarchical clustering algorithms, which use an adjacency matrix to calculate the relation between vertices in a complex network (PORTER; ONNELA; MUCHA, 2009), as well as several other approaches.

The problem of detecting communities in complex networks can be formulated as follow: Given a network, directed or undirected, non-weighted or weighted, generate the following solution: a reasonable decomposition of the graph into sub-graphs, where in some sense the vertices in each sub-graph have more similarity to each other than with outsiders.

2.9 COMMUNITY DETECTION ALGORITHMS

In this section, algorithms for clustering and detecting communities in complex networks, as well as the main characteristics, strengths and drawbacks of each of these methods will be discussed.

2.9.1 K-MEANS

The K-Means clustering method is numerical, unsupervised, non-deterministic and iterative. The K-means aims to divide M points with N dimensions into K clusters (HARTIGAN; WONG, 1979). K-means is a simple and fast clustering algorithm that provides satisfactory results for many applications.

K-means is a well-known and widely used algorithm to cluster similar subjects in a set, an algorithm which enables the partitioning of a network into a predefined number of groups (HUANG, 1998). K-means allows the use of Euclidean distance to evaluate the similarity between a set of elements by signaling the nearest K centroid of each element in the dataset.

K-Means iteratively allocates the partitions of a dataset into K clusters, locally minimizing the distance between the vertices to the centroids (MACQUEEN, 1967). We can formulate the function which minimizes the within-cluster sum of squares as shown in Equation 9:

$$C = \sum_{j=1}^{k} \sum_{i=1}^{n} \|x_i^j - c_j\|^2, \tag{9}$$

where $\|x_i^j - c_j\|^2$ represents the distance between an element to a centroid. K-means usually computes the distance between the elements and the centroids using the Euclidean distance, as shown in Equation 10:

$$Ed = \sqrt{\sum_{i=1}^{n}(p_i - q_i)^2}, \tag{10}$$

where p_i and q_i are two distinct points in Euclidean space.

The operation of the K-means algorithm is outlined in the pseudocode shown in Algorithm 5.

Algorithm 5: Pseudo code for the K-means algorithm

```
initialize(K:centroids) ;
place K centroids in random locations;
while centroids positions changes do
    Assign the dataset elements to the nearest centroid;
    Recalculate centroid positions based on values of elements attached to the
    centroid.;
end
```

In addition to the usual Euclidean distance, other metrics can be applied to compute the distances with the K-means algorithm, such as Manhattan distance (NIEDERMEIER; SANDERS, 1996) or Chebyshev distance (SOUZA; CARVALHO, 2004). The K-means algorithm usually converges within few interactions. Among the strengths of the algorithm are its simplicity and low computational cost. However, there also are well known drawbacks, such as its sensitivity to outliers and to the initialization. To work with noisy data and not clearly partitioned data or values that are too distinct, it is desirable to consider some of its variants.

Several variants of the K-means algorithms were developed for various purposes, from improving performance as to meeting certain drawbacks resulting from the original algorithm. Among these variants we can underline X-means, which aims to find automatically the number of partitions K (KANUNGO et al., 2002), Fuzzy C-means (FCM) which allows the overlapping of communities (PAL; BEZDEK, 1995), weighted K-means which aims to submit each dimension of data to a weighted fashion to evaluate similarity (HUANG et al., 2005; AMORIM; KOMISARCZUK, 2012), and also the version of Bradley; Fayyad and Reina, which aims to deal with large amounts of data (FAYYAD; BRADLEY; REINA, 2001).

In the K-means algorithm, every graph vertex is of equal importance in locating the centroid of the cluster. This characteristic makes K-means very sensitive to outliers, i.e., vertices that have values dramatically far from centroid and tend attract the centroid towards

themselves. The algorithm is also sensitive to the initialization of the centroid, especially with very heterogeneous cluster sizes and noisy data. To overcome these drawbacks, some variations of the algorithm have been proposed. These variations will be outlined in following subsections.

Some of the drawbacks for the K-means clustering method are identified as follows:

- Peripheral vertices must be assigned to a community, even if their connection to it is weak.
- Each vertex is attributed only to one cluster, which could be an unrealistic scenario.
- Results are dependent on initialization; accuracy can be compromised.
- The number of desired clusters must be known in advance.
- Clusters are sensitive to noisy data and outliers.
- Lack of accuracy.

2.9.2 FUZZY C-MEANS

In 1965, the mathematician and computer engineer Lotfi Zadeh introduced fuzzy sets in order to come closer to the physical world. Fuzzy logic, derived from fuzzy sets, admits ranges of values between the crisp true or false Boolean values (ZADEH, 1965). Fuzzy clustering methods followed the creation of fuzzy sets (BEZDEK; EHRLICH; FULL, 1984). These methods allow vertices to be assigned to different clusters in different degrees consisting of partial memberships.

The proposition is that vertices with a high degree of similarity are closer to a cluster than vertices with a low or close to zero degree of similarity to that cluster. Consequently, every vertex in the network belongs to all clusters with a distinct degree of membership.

When a vertex coincides with the center of the cluster, the maximum degree of membership is assigned to that vertex. Degrees of membership vary from zero to one (BEZDEK; EHRLICH; FULL, 1984). It is possible to blur the limits of the clusters by using a fuzzification constant.

The FCM clustering is obtained by minimizing an objective function, as shown in Equation 11:

$$J = \sum_{i=1}^{n} \sum_{k=1}^{c} \mu_{ik}^{m} |p_i - v_k|^2, \tag{11}$$

where J is the objective function, n is the number of objects, c is the number of clusters, μ is the fuzzy membership value from the table, m is a fuzziness factor (a value > 1), p_i is the ith object, v_k is the centroid of the k_th cluster and $|p_i - v_k|$ is the Euclidean Distance between p_i and v_k.

The fluxogram of steps to perform the algorithm is presented in Figure 13.

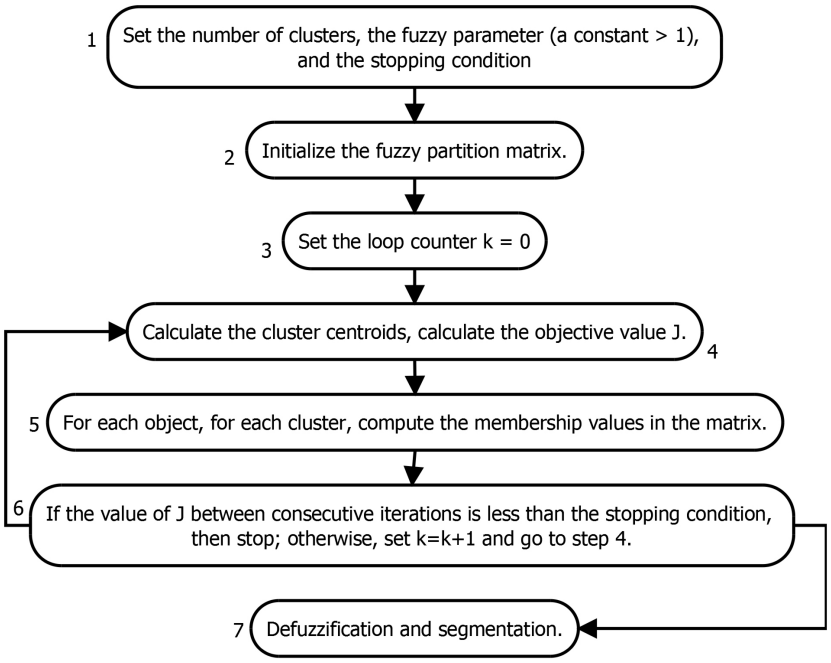


Figure 13: The Fuzzy C-means clustering algorithm flowchart.

Figure 14 (a) shows a sample of points with unclustered data. Figure 14 (b) shows the very same data points clustered using the FCM algorithm into three distinct clusters.

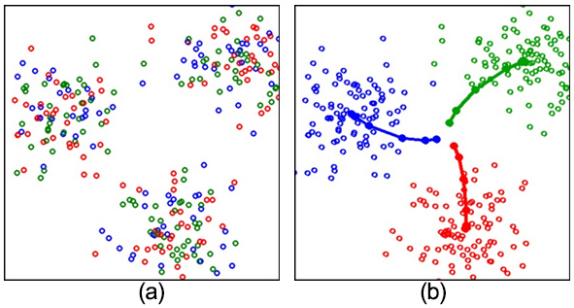


Figure 14: Unclustered data sample (left). Clustered data with three clusters (right).

Unlike from traditional K-means, in which the only expected parameter is the K number of desired clusters, in order to use FCM the following parameters must be provided:

the number of clusters, C, the 'fuzziness' exponent, m, the termination tolerance, and the norm-inducing matrix. The fuzzy partition matrix, U, must also be initialized. The number of clusters, C, similar to regular crisp K-means, represents the number of the partitions desired. Normally, this is an empirical value. In several situations, fuzzy clustering could be more natural than hard clustering (CANNON; DAVE; BEZDEK, 1986). Fuzzy C-means also known as FCM algorithm, it is a powerful unsupervised method for data analysis. It produces better results than the traditional K-means approach, avoiding the local minima. Fuzzy C-means is also less sensitive to noisy data that the regular K-means (WU; YANG, 2002).

2.9.3 WEIGHTED K-MEANS

There are several variations for weighted K-means (HUANG et al., 2005; MODHA; SPANGLER, 2003; WU, 2008).

The analysis of data from interactions among data samples commonly occurs in different dimensions. It is reasonable to assume that in certain cases some dimensions may be more relevant (MODHA; SPANGLER, 2003). However, even weak correlations can still have a significant value for data analysis. Thus, it is convenient to have an algorithm capable of measuring attributes under different weights (HUANG et al., 2005).

Weighted K-means is also called Minkowski Weighted K-means, with the algorithm automatically calculating feature weights for each cluster and using the Minkowski metric. Unlike the Euclidean distance, Minkowski space also has also one '*timelike*' dimension (AMORIM; MIRKIN, 2012). Weighted K-means will output the K centroids to a given set of n data points considering weights when computing the centroids.

Distinct characteristics should have distinct weights, which can constitute a problem for very similar dimensions. The weights must be non-negative values between zero and one. As a result, the dimensions are distributed uniformly across the clusters. Dimensions assigned a smaller weight will be agglutinated near the centroids, while dimensions with a larger weight (AMORIM; MIRKIN, 2012) will be distant from the centroids.

This balanced characteristic of weighted K-means results in more homogeneous divisions, since none of the dimensions will lead the partitions in a specific direction.

We propose a different approach to the K-means that provides a weight for each dimension of data in a supervised fashion. With distinct weights assigned to each dimension of data, it is possible to change the balance of the equation and obtain an overall modified result. If an attribute has less importance, we will counterbalance the others, assigning more importance

to them. The modified version of the weighted K-means with a Euclidean distance subject to weights is shown in Equation 12:

$$wEd = \sqrt{\frac{(p_1 - q_1)^2}{w_1} + \frac{(p_2 - q_2)^2}{w_2} + ... + \frac{(p_n - q_n)^2}{w_n}} \tag{12}$$

Equation 13 shows the Euclidean distance submited to weights in sigma notation.

$$wEd = \sqrt{\sum_{k=1}^{n} \frac{(p_k - q_k)^2}{w_k}} \tag{13}$$

The weights will influence the clustering results only if distinct values have been set for at least two dimensions of data; otherwise the algorithm will behave like the traditional K-means.

If there is sufficient change in at least one centroid to reassign at least one vertex to a different cluster, both centroids (which lose the vertex and receive the vertex) will be recalculated. This procedure changes centroid positions and may cause new changes to other vertices. The process is repeated until the centroids are mathematically in the center of each cluster.

Therefore, it is possible to balance the dimensions without losses. Even the weak relations can be added to the network with a small weight. This approach is a complementary way to find communities in graphs with wide dimensions of data, if a distinct dimension has more or less relevance to the network. This approach minimizes distortions caused by outliers.

2.9.4 GIRVAN AND NEWMAN ALGORITHM

The Girvan and Newman algorithm (GIRVAN; NEWMAN, 2002) is an iterative process which aims to detect communities in networks. The algorithm explores the concept in which network nodes are joined together in tightly knit groups, based on the node centrality explored in section 2.3.1. Figure 15 shows in bold an edge with high betweenness connecting two distinct groups.

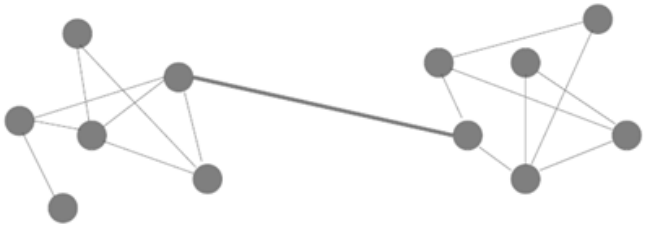


Figure 15: A Small network with two clusters connected by an edge with high betweenness (FORTUNATO, 2010).

The Girvan and Newman algorithm detects communities, focusing on betweenness, by removing edges with the largest centrality, as introduced by Freeman (1979). The betweenness of a vertex v in a graph $G(V,E)$ with V vertices is computed as shown in Equation 14:

$$C_b(v) = \sum_{s \neq v \neq t \in V} \frac{\sigma_s t(v)}{\sigma_s t}, \tag{14}$$

where σ_{st} is the total number of shortest paths from vertex s to vertex t and $\sigma_s t(v)$ is the number of those paths that pass through v.

The fluxogram of the Girvan and Newman algorithm is presented in Figure 16.

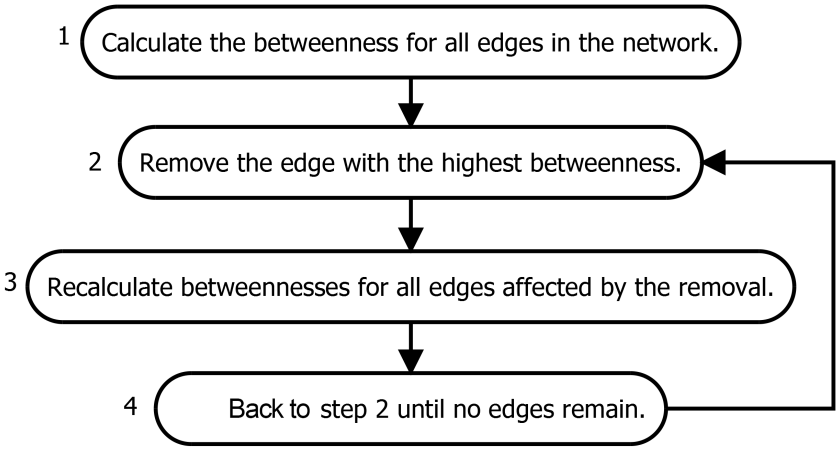


Figure 16: The Girvan and Newman clustering algorithm flowchart

A modification of the Girvan and Newman algorithm was proposed by (WILKINSON; HUBERMAN, 2004). We can intuitively note that removing the central edge shown in Figure 15 will detach the two evident clusters presented in this small network.

New partitions can be added depending on recalculations while the algorithm is executed. Partitions are nested and the process needs to be repeated until the number of

components meet the maximum number of partitions desired. This will eventually increase the number of weak components, which are the cohesive subgroups or communities from a partition of the original data.

The pseudo-code shown in Algorithm 6 illustrates the method proposed by Girvan and Newman based on betweenness.

Algorithm 6: Pseudo-code of the Girvan and Newman clustering algorithm based in betweenness.

Data: A Graph G
Result: A clustering solution of G
initialization
$E \leftarrow E(G)$ `// The set of all edges contained in G.`
while $E \neq \emptyset$ **do**
    `// While the set of edges is not empty.`
    foreach $e \in E$ **do**
        `// For every edge in the set of edges.`
        Compute $C_b(e)$ `// Compute the betweenness for the edge.`
    end
    $e' \leftarrow max_{e \in E} C_B(e)$ `// Updates the betweenness.`
    $E \leftarrow E - \{e'\}$ `// Remove the edge from the set of edges.`
end

2.9.5 THE LOUVAIN METHOD BY BLONDEL ET AL

Blondel et al (2008) proposed a method for extracting the community structure of large networks, a heuristic method based in modularity optimization.

Modularity is a quality index for clusterings. The objective of modularity is to evaluate the division of a network into modules (NEWMAN; GIRVAN, 2004). Modules will have dense connections between the vertices within the modules and sparse connections between vertices what belong to different modules. The modularity of a graph can be computed using Equation 15:

$$Q = \frac{1}{2m} \sum_{i,j \in V} \left[A_{ij} - \frac{k_i k_j}{2m} \right] \delta(c_i, c_j), \tag{15}$$

where A_{ij} are elements of the adjacency matrix of $G(E,V)$, and k_i is the out-degree of node i, and $m = |E|$ and $\delta(c_i, c_j)$ is equal to 1 if i and j belong to the same community, being equal to 0 otherwise (BRANDES et al., 2008).

High modularity serves as an indicator of the presence of communities within a complex network. Partition modularity is a scalar value between -1 and 1 that measures the density of links within communities compared to links between communities.

The algorithm proposed by Blondel et al. (2008) follows the steps related below:

- Each vertex is attached to a different community; thus, the initial partition will have an equal number of communities and vertices;
- Then each vertex i will consider the neighbors j of i and we evaluate the gain of modularity that would take place by moving i from its community and placing it in the community of j;
- The vertex i will be placed in the community where it maximizes modularity, and only if this gain is positive;
- If moving a vertex does not improve the modularity, the vertex will remain in its own community;
- The process is repeated for all nodes until no improvement can be achieved;

The gain in modularity δQ is obtained by moving a vertex i into a community C, which can be computed by Equation 16:

$$\delta Q = \left[\frac{\sum_{in} + K_{i,in}}{2m} - \left(\frac{\sum_{tot} + K_i}{2m} \right)^2 \right] - \left[\frac{\sum_{in}}{2m} - \left(\frac{\sum_{tot}}{2m} \right)^2 - \left(\frac{k_i}{2m} \right)^2 \right], \tag{16}$$

where $\sum_{in}$ is the sum of the weights of all edges in C. $\sum_{tot}$ is the sum of the weights of the links incident to nodes in C. k_i is the sum of the weights of the links incident to node i, k_i. It is the sum of the weights of the links from i to nodes in C, and m is the sum of the weights of all the links in the network (BLONDEL et al., 2008). The δQ serves as a measure of quality for

communities, low values means that the community structure it is not relevant, higher values indicates evidence of relevant community structure.

The pseudo-code shown in Algorithm 7 describes the method proposed by Blondel et al based on modularity.

Algorithm 7: Pseudo-code of the Louvain method proposed by Blondel et al.

Data: A Graph G

Result: The clustering of G

initialization;

$C \leftarrow 0$;

; // The set of clusters is initialized as zero.

foreach $i \in V(G)$ **do**

; // For each vertex i in the graph G

$C \leftarrow C \cup \{i\}$; // Vertex i is joined to the cluster.

$a_i \leftarrow \frac{d(i)}{2|E(G)|}$; // The a_i matrix is updated with the degree of vertex i added.

end

foreach $i, j \in E(G)$ **do**

; // For each edge in G

if *i is connected to j* **then**

$\delta Q_{ij} \leftarrow \frac{1}{2|E(G)|} - \frac{d(i)d(j)}{|E(G)|^2}$; // If i is connected to j modularity for ij is updated.

else

$\delta Q_{ij} \leftarrow 0$; // If i is not connected to j modularity for ij is set to 0.

end

end

while $|C| > 1$ **do**

Select the higher δQ_{ij} ; // Select the highest modularity for two specific vertices.

Merge the clusters i and j ; // Merge the cluster of i with the cluster of j.

Update δQ_{ij} and the Matrix a_i ; // Recomputed a_i modularity after merge.

Update row and column j and remove row and column i from the matrix a_i ;

end

Table 1 shows comparisons between the algorithms presented in this section with their respective computational complexity.

Table 1: Comparative table for clustering and community detection algorithms features and computational complexity.

Algorithm	Purpose	Metric Used	Complexity
K-means	Clustering	Euclidean distance	$O(3NK)$
Fuzzy C-means	Clustering	Euclidean distance	$O(\frac{n^2}{s})$
W-kmeans	Clustering	Euclidean distance	$O(3NK)$
The Girvan and Newman Algorithm	Community detection	Betweenness	$O(n^3)$
The Louvain Method	Community detection	Modularity	$nlog(n)$

3 MATERIALS AND METHODS

This chapter discusses the methodology used to detect communities in complex networks. Techniques of clustering and community detection algorithms are employed to identify the communities in complex networks.

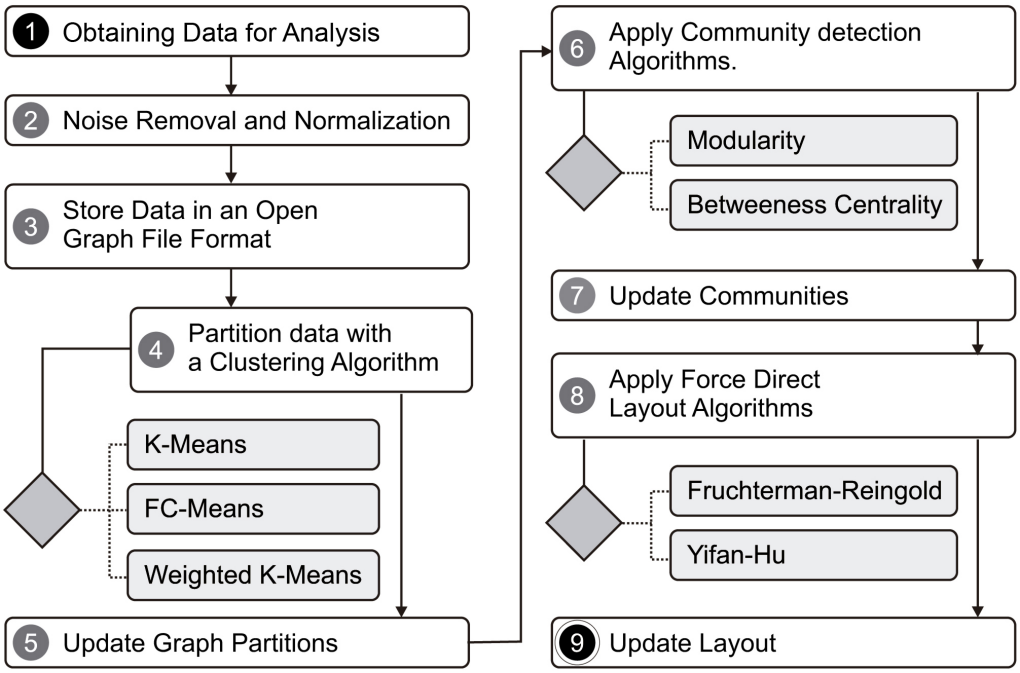


Figure 17: Methodology Workflow

3.1 METHODOLOGY WORKFLOW

The workflow is organized in accordance with the steps presented in Figure 17 detailed as follows:

1. **Obtaining data for analysis.** In Case Study I - Detecting Cartels in public bids, we worked with data provided from Court of auditors of the Paraná State (TCE-PR), the extraction of supplementary data from third-part databases such as the Brazilian Institute of Geography and Statistics (IBGE), public governmental websites, public records from

the Secretariat of the Federal Revenue of Brazil. This process consisted of obtaining all information needed and organizing it as an open-source relational database.

In Case Study II - Representing gene expression data as graphs, the datasets were formed by large matrices of microarrays for several samples, which led to large datasets with millions of records. The goal for this step was establish a methodology capable of reading and representing datasets as graphs and complex networks.

2. **Noise removal and data normalization.** This step was performed to remove noisy data, null values, incomplete records which can lead to misinterpretation, malformed data records and truncated values. This step created the expectation of being able to to apply foreign keys to the tables of a relational database to ensure consistency of data.

3. **Storage of pre-processed data in an open graph file format.** In order to manipulate and apply distinct algorithms of clustering, the pre-processed data was stored in an open graph file format such as Graph Modelling Language (GML) or XML-based graph file format (GraphML), shown in detail in Appendix A.

4. **Application of clustering algorithms.** This was the procedure for the application and experimentation of several clustering algorithms with distinct features are performed.

5. **Updating graph partitions.** Updates of the graph partitions according to the results obtained in the clustering process.

6. **Application of community detection algorithms.** The most important step in the process is at this stage, with one or more algorithms to detect communities in graphs.

7. **Updating communities.** Updating the graph communities according to results obtained in the community detection process.

8. **Performing the layout algorithms.** Graph Layout algorithms aimed to present the graph in a more aesthetic and pleasant way. This was especially desirable for Case Study I when the post processed data was presented to the general public.

9. **Updating the layout.** Updates of the graph layout according to the results obtained from the graph layout algorithms.

3.2 EMPLOYING A SIMILARITY MEASURE

As already addressed in detail in section 2.7, similarity measures are one way to assess how similar or how different two objects of a complex network are. Vertices assigned to same cluster should be considered highly similar; vertices assigned to distinct clusters should be highly dissimilar. The term assortativity is used to describe the phenomena of Social Network Analysis, which also states that a network vertex prefer to be attached to others that are similar in some way (NEWMAN, 2003a).

To evaluate the similarity between vertices in a Social Network we used a weighted variation of the K-means algorithm outlined in section 2.9.3, where N dimensions of data were measured in an Euclidean Space. Thus, similar elements were close to each other, and non-similar elements were distant from each other.

The number of attributes submitted to the Euclidean space depends on the number of variables that are part of the problem. However, distinct dimensions can have distinct degrees of importance according to the context.

Similarity measures are the basis for comparisons between elements of social networks formed by the companies participating in public bids objects of the Case Study I. Particularly in this Case Study, the dimensions of data meant, allowing a semi-supervised clustering approach.

However, in some situations the relevant data of each dimension were unknown, or simply all sizes of data were of equal importance, which most accurately reflected the data from Case Study II. For this particular Case Study used correlation measures such as Spearman's Rank Correlation, Kendall's Rank Correlation and Pearson's Correlation Coefficient, which are further outlined in Appendix C.

3.3 THE DYNAMIC ASPECT OF COMPLEX NETWORKS

Most of the tools and techniques used for Social Network Analysis focus on the topological aspects of static networks. Most often ignore the dynamic aspects of these networks (SANTORO et al., 2011).

However, Social Networks have dynamic aspects that provide an inadequate the analysis of a snapshot of the network. Members often change their connections, starting to participate in new groups that are not involved in their activities. New members can join the network and members can leave the network (CASTEIGTS et al., 2012).

In other complex networks, this phenomena can also be observed. Food chains can be modified by the introduction of new predators, chemical reactions could be observed from an initial state to several interactions to become to a final state. Gene Data Expression could be changed due to interaction with pathogenic organisms or diseases, and so forth.

The dynamic aspect of the network can vary according to the context of the network. This may can be related to time itself, e.g.; hours, days, weeks or other metrics. It is also can be related to events that occur in the network. The following graph events represent atomic changes in graphs:

- Creation (or joining) of a vertex;
- Removal (or disjoint) of a vertex;
- Creation of an edge;
- Removal of an edge;
- Weight increase for an edge;
- Weight decrease for an edge.

To overcome this issue all interactions between two distinct vertices over time as a metric of strength for the connection between two vertices were taken into account. The proposed methodology is described in Sub-section 3.3.1.

3.3.1 A MEASURE OF STRENGTH FOR CONNECTIONS IN DYNAMIC COMPLEX NETWORKS

Besides the similarity of its members, connections between community members are usually stronger than the connections to members outside the community (GRANOVETTER, 1973; PORTER; ONNELA; MUCHA, 2009).

Considering the dynamic aspects of complex networks, a metric was proposed that enables the measurement of the strength of the connections over time for connections. Correlations of strength and intensity in complex networks have been studied over time with different purposes (GRANOVETTER, 1973; PARSHANI; BULDYREV; HAVLIN, 2010; XIANG; NEVILLE; ROGATI, 2010).

Given an undirected weighted graph $G(V,E,w)$, equation 17 is a simple formula for calculating the weight of a connection between the vertices (x,y) over time,

$$w(x,y) = \frac{\sum_{i=1}^{t} w(x_i, y_i)}{t}, \tag{17}$$

where $w(x,y)$ is the weight between edges x and y, t is the number of changes in whole graph and not only for a specific vertex. Consequently, the number of interactions of each vertex will be considered for calculating the edge weights. Figure 18 illustrates the process of measuring strength over time.

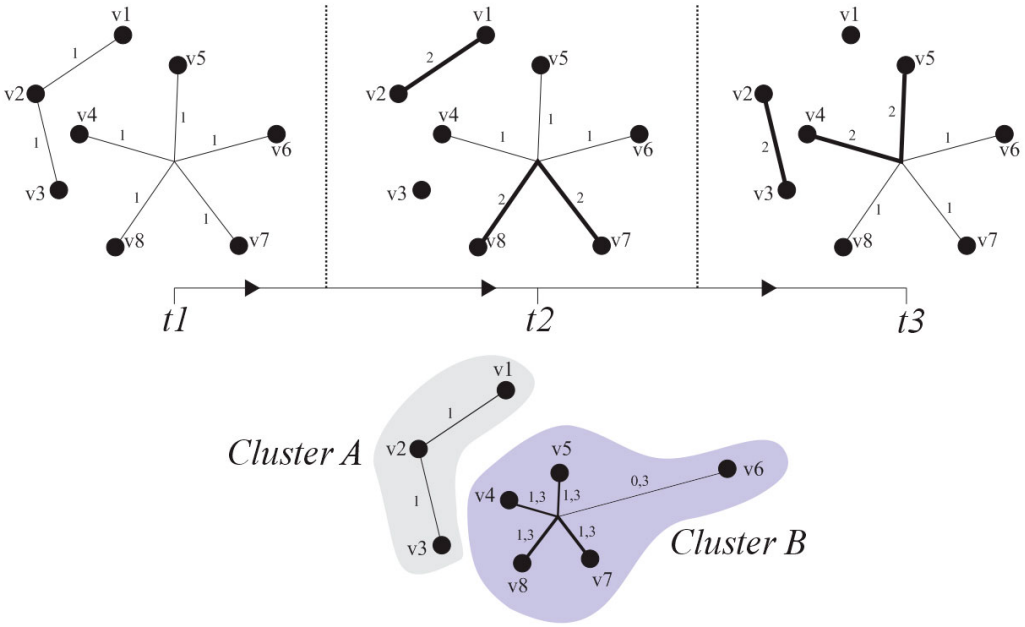


Figure 18: Clusters in a graph based on strength measures over repetitions in time.

The variable t can be set as other metrics, number of interactions, time-sample, recurrence, etc. As a result, communities that maintain their links between members unchanged over time will be more evident. Communities which receive or lose members frequently will be less evident. By applying the concept of repetition as a measure of strength it is possible to observe in Figure 18 the result by considering three different moments in time. After processing, the results can be expressed in the forms of Cluster A and Cluster B.

4 EXPERIMENTS AND RESULTS

4.1 CASE STUDY I - DETECTING CARTELS IN PUBLIC BIDS

The Brazilian government keeps its data from public bids with the support of computer programs such as databases and digitalized documents which are searchable and indexable. In 2009, the Federal Integrated System of Financial Administration (SIAFI) registered one billion of financial transactions in twenty four thousand administrative units (SILVA; RALHA, 2011).

In this work are computed data from the public bids of Paraná state as an agreement between the Federal University of Technology of Paraná (UTFPR) and The Court of Auditors of the State of Paraná (TCE/PR) in an effort to establish a methodology for detecting possible cartels and other non-compliances in public bids.

4.1.1 PUBLIC BIDS

Public procurement can be briefly described as processes whereby public institutions undertake the procurement of goods, products and services. Bidding processes are subject to regulatory instructions, rules, laws and the whole apparatus that aims to prevent individuals or corporations from obtaining illicit commercial and financial advantages as a result of trading goods, products or services with public agencies.

In order to inhibit abusive and predatory practices, in Brazil law Nº. 12,529 at November 30, 2011 (BRASIL, 2011) was passed various devices providing for the prevention and repression of offenses against the economy using a number of legal devices.

4.1.2 CARTELS IN PUBLIC BIDS

One way to defraud free competition is the formation of groups of companies that coordinate their operation in violation of the principles of the fair market.

The most common forms of cartel activity include: the turnover of winners and the

combination of pricing and proposals by fraudulent companies for the sole purpose of meeting the minimum legal requirements in terms of numbers that will not effectively provide the product or service subject to bidding.

The operation of cartels substantially hinders efforts to employ the country's resources fairly, thus, hindering the development of the nation. Companies take unfair advantage, by conspiring among themselves, to cheat the competitive nature of the bidding.

The classification of the type of offense to the economic public order is given from some concepts, such impeding free competition and free enterprise, domination of the relevant goods or services, arbitrary settings of profit levels and abusive exercise of dominance (CARVALHO, 2005).

For this reason it is necessary to observe what is provided in Chapter II of Law 12.529 (BRASIL, 2011) regarding infractions. Article 36 states that:

> Article 36. It is a violation of economic order, regardless of the fault, to conduct any actions of whatever kind, for the pursue of or resulting in the following effects, although they may be not achieved:
>
> I - limit, restrain or in any way impede open competition or free enterprise;
> II - dominate the market of relevant goods or services;
> III - arbitrarily increase profits; and
> IV - abusively exercise a dominant position.

Due to these issues, it is evident that there is a need to identify cartel formations. With this information, competent organs will be able to prevent such practices and ensure compliance with current legislation in the country.

The Office of Fair Trading, a governmental body from United Kingdom (OFT Office of Fair Trading, 2013) defines a cartel as: "an agreement between businesses not to compete with each other. The agreement is usually secret, verbal and often informal." Furthermore, "Typically, cartel members may agree on:"

- Prices;
- Output Levels;
- Discounts;
- Credit Terms;
- Which Customers They Will Supply;

- Which Areas They Will Supply;
- Who Should Win a Contract (Bid Rigging).

4.1.3 COHESIVE GROUPS

For a cartel to be effective it is necessary to maintain a cohesive group of companies (HUCK; NORMANN; OECHSSLER, 2001). The groups who share characteristics in public bids and are directly related need to be small in order to keep themselves under control and profitable (LEVENSTEIN; SUSLOW, 2006).

In terms of complex networks, cohesive groups or cohesive subgroups can be described thus: “A community or cluster, or cohesive subgroup is a subset of individuals among whom there are relatively strong, direct, intense ties” (FORTUNATO; LATORA; MARCHIORI, 2004).

Both concepts encompass the problem of Case Study I, detecting a small strongly connected cohesive group of companies acting together for a common purpose. That assumption supports the problem in a methodological approach.

4.1.4 HYPOTHESIS

Cartels in public biddings are a way to infringe free competition. It can be argued that groups of companies acting as cartels systematically operate to gain an advantage over the other contestants who are not part of the cartel.

In order to verify wheter members of a group are taking advantage by acting as a community, we proposed an index successfully capable of measuring the success rate of the cartel participants. The proposed success rate is illustrated in Equation 18 and can be defined as the ratio between the number of participations in public bids and the number of victories,

$$I_p = \frac{V_p}{P_p}, \tag{18}$$

where I_p is a value between 0 and 1 representing the index, V_p represents the number of victories for the participant $_p$, and P_p is the number of participations for participant $_p$.

We observed empirically that the companies with a highest rate of winning bids are

also the most connected, i.e., vertices with a higher degree in the graph.

However, comparing the proportion of wins of a company does not provide a complete metric. For example, a company with 1 share and 1 victory will have a success rate of 100%. Yet, a company with 10 stakes and 7 wins will have a rate of 70%, although the relevance of the second company to be greater in the context of the problem.

To overcome this problem we used a weighted approach to compute the number of wins, number of participations in bids and the total number of bids, as defined by the Equation 19,

$$wI_p = \frac{(\frac{V_p}{P_p})}{T}, \tag{19}$$

where wI_p is a value between 0 and 1 representing the success rate for a company $_p$ and T is the total number of participations for all participants in public bids. Consequently, it is possible to compute a ranking in terms of sucessfull, average or failing companies by means of a weighted index based on the number of wins and number of participations in public bids.

In order to simplify the observation of this index, the variables were discretized according to the values shown in Table 2.

Table 2: Discretizated values for sucess rate

Value	Discretizated Values
>= 70%	High success rate
30% ><70%	Average success rate
<= 30%	Low success rate

The success rate index allows an evaluation of which companies are most influential in the network and can trace their success or failure winning public bids. Moreover, it is possible to evaluate wheter network hubs are directly connected to high success vertices or with very low success vertices. This provide us a glimpse of companies taking advantage of 'ghost' competitors in public bids.

4.1.5 OBTAINING DATA FOR ANALYSIS

The information regarding public bids is public, but it is not available to the general public as a structured data. The datasets from public bids in the State of Paraná, Brazil, from

2005 to 2012 were provided to due a cooperative effort between The Court of Auditors from Paraná State (TCE/PR) and the Federal University of Technology - Paraná (UTFPR).

Public bids in Brazil can occur at three levels of government: local, state and federal. Our study focuses on the state level. The first step was to transfer the information from an MS-SQL Server Database to an open-source database. The MySQL Database and coma separated values (.csv) files, were chosen as the standard for this work.

Five dimensions of data were used to build a similarity measure between companies, which are listed as follows:

1. **Public bids occurred in Paraná State** - This feature provides information regarding the time and subject for the public bids which are generating the complex networks.
2. **Participation in public bids by companies** - This dimension of data provides the information about the connections between companies within the complex network.
3. **Companies which won public bids** - Determine which are the major winners in quantitative terms it is relevant to evaluate similarity within the communities and if these communities are experimenting advantages in relation to the complex network average success rates.
4. **Companies which lost public bids** - This feature evaluates the existence of companies which are participating to favor other companies, successively participating with overpriced offers or quitting the competition.
5. **The correlation between who won and who lost** - This dimension of data evaluates if major winners has strong connections with frequent losers, this can be an indication that frequent losers are participating to favor the winners.

4.1.6 REPRESENTING PUBLIC BIDS BY MEANS OF COMPLEX NETWORKS

Employing the five dimensions of data presented in section 4.1.5, the complex network produced by the participation of the companies in public bids in Paraná State have the following structure:

- Each company represents a vertex of the graph;
- If two or more companies are participating in the same public bid, these connections are used to generate the edges;

- The process is repeated for each company present in each public bid.

If two distinct companies i and j are participating in the same bid, the value for the edge ij in the Adjacency Matrix is set as 1; otherwise is set as 0. Self loops are always set as 0. Table 3 shows the connections for public bids from 1 to 4 and companies from A to D.

Table 3: Matrix A_0 from public bids 1 to 4 for companies A to D.

	Bid 1	**Bid 2**	**Bid 3**	**Bid 4**
Company A	0	1	1	1
Company B	1	1	1	1
Company C	0	0	0	1
Company D	1	1	1	0

The dynamic aspect is approached by considering every interaction between two distinct companies. The weight of the edge is the sum of all interactions between the two vertices. In order to obtain an undirected weighted graph $G(V,E,w)$, where w means the weight of the edges, the values of the Matrix A_0 are submitted to Equation 20,

$$w(x,y) = \sum_{n=1}^{n} A(j_i, j_{i'}), \tag{20}$$

where i is the correspondent line for the vertex x in the matrix and i' is the correspondent line for vertex y and n corresponds to the number of columns in the matrix. As a result, the adjacency matrix A shown in Table 4 was generated.

Table 4: Resulting adjacency matrix A for a weighted graph $G(V,E,w)$.

	Company D	**Company C**	**Company B**	**Company A**
Company A	2	1	3	0
Company B	3	1	0	
Company C	0	0		
Company D	0			

As already presented in Section 4.1.4, in this case study, unveiling the winner of the public bid is highly relevant. The complex network theories already show that similar subjects tends to cluster together (SPEARMAN, 2010; GRANOVETTER, 1973; GIRVAN; NEWMAN, 2002), while very dissimilar subjects tends to be split.

Knowing the major winners and losers aims to answer practical questions such as:

- Do the winners have an advantage over companies that always lose?
- Are the winners clustered together?
- Will a company have a better success rate the more it participates?
- How many companies there in of a cartel?

Levenstein and Suslow (2006) point out that the number of companies in a cartel must be small to enable the collusive organization of the cartels.

We built a success rate simply by computing the average number of victories which corresponds to the total number of victories over the total number of participations in public bids. This can be mathematically formulated as shown in Equation 21:

$$Sr = \frac{\sum_{i=1}^{\rho} \vartheta}{\rho}, \tag{21}$$

where Sr is the success rate, ρ is the total number of participations and ϑ is the number of victories. This results in a value between 0 and 1, which corresponds to the percentage of victories for each public bid participant.

In order to produce clear partitions this variable Sr was discretized according to the criteria shown in Table 2. The criteria to use the values presented in Table 2 are based on the theory that the existence of cartels is based on some companies taking advantage over other companies. The network partition was set to provide the three elements related to this hypothesis, that are: companies within a cartel have higher success rates than the average. Companies that are favoring the cartel have lower success rates than the average. And companies that are not operating or cooperating with a cartel will show regular average.

In this particular case study the aesthetic and visualization aspects of graphs were considered. We targeted the general public that may not be familiar with the mathematical characteristics of complex networks. To represent the graphs, three distinct dimensions for vertices and edges were adopted. The three dimensions are listed as follows:

- Vertex degree is analogous to the size of the vertex. Vertices with a high degree are bigger than vertices with a low degree.
- The strength of connections is represented by edge weights; thus, higher weights are noticed as bold connections: otherwise, weak connections are observed as thin lines.
- To easily spot winners or losers we used grey-scale values; black for winners, grey for average and white for losers.

Although this representation was driven by very simple concepts, the final result provided an opportunity to visualize the network in a pleasant aesthetic way, in which the vertices are more influential in the network and how they are connected to other vertices. Figure 19 shows a simple example of this representation.

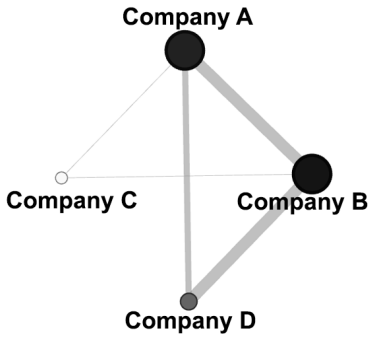


Figure 19: The resulting non directed weighted complex network

The aesthetic aspects added to the networks have no implication in the network topology or other features. The only purpose is to produce a visualization which any audience can interpret easily.

4.1.7 EXPERIMENTS PERFORMED

For the first experiment, 101 public bids were selected encompassing 117 distinct companies. In this dataset, there are only public bids from construction and engineering services in the city of Curitiba - PR - Brazil for the year of 2011, as a sample of data. These summarized data are shown in Table 5.

Figure 20 shows a complex network from this partial dataset. The connections between the companies, however, are relative to the entire period of time available in the databases. If two distinct vertices are not connected, this means that, they are never connected. If two distinct

Table 5: Public Bids for Construction and Engineering Services in Curitiba in 2011.

Number of public bids	101
Companies that participate in public bids	117
Companies that won at least one public bid	97
Companies that did not win at least one public bid	20

companies have co-participated frequently, the edge between these companies will be more evident.

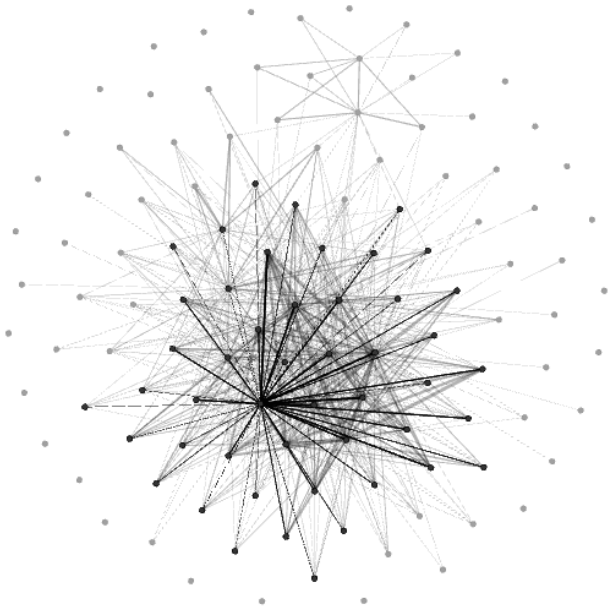


Figure 20: Partial group of companies that participate in public bids for Construction and Engineering Services at Curitiba - PR - Brazil in 2011.

The chart shown in Figure 21 displays the degree distribution for the complex network in Figure 20. The network follows a power-law degree-distribution.

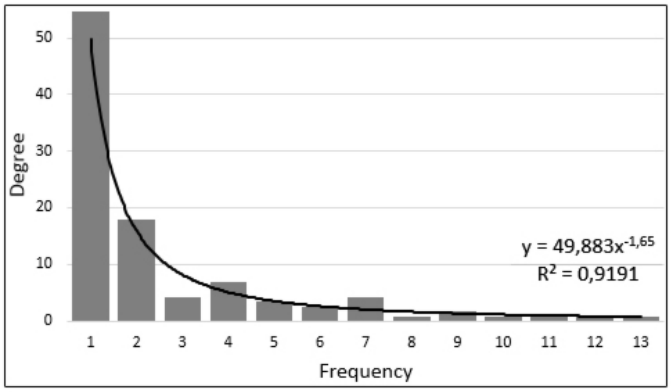


Figure 21: Degree distribution for the complex network in Figure 20.

The topology of the network observed in this first experiment provided a clue that community structures were present in the networks. The degree distribution and the presence of hubs showed the existence of communities among the elements of the networks.

The focus of Case Study I was the public bids conducted in the State of Paraná. The State of Paraná is a unit of the Federation of Brazilian States composed of 399 municipalities. This state is also divided geographically into 10 mesoregions, [1] as shown in Figure 22.

	Mesoregion	Most populous municipality
a	Metropolitana de Curitiba	Curitiba
b	Sudeste Paranaense	Irati
c	Centro Oriental Paranaense	Ponta Grossa
d	Centro-Sul Paranaense	Guarapuava
e	Norte Pioneiro Paranaense	Cornélio Procópio
f	Norte Central Paranaense	Londrina
g	Centro cidental Paranaense	Campo Mourão
h	Sudoeste Paranaense	Francisco Beltrão
i	Oeste Paranaense	Cascavel
j	Noroeste Paranaense	Umuarama

Figure 22: Mesoregions of Paraná State (Laboratório de Cartografia Tátil Escolar - UFSC, 2012).

To reduce processing time we divided the dataset according to the meso-regions shown in Figure 22.

Given the positive results obtained in the first experimentation, we proceeded to

[1] http://www.ipardes.gov.br/pdf/mapas/base_fisica/relacao_mun_micros_mesos_parana.pdf

replicate the same methodology for an entire meso-region. The complex network shown in Figure 23 illustrates the complete group of companies which participated in public bids for construction and engineering services in Curitiba - PR - Brazil in 2011 and their relationships.

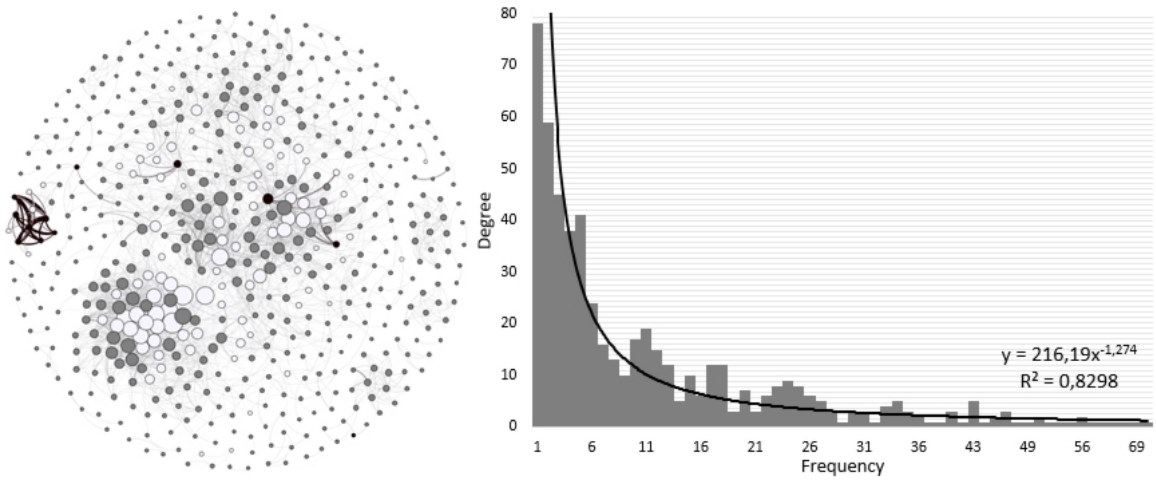


Figure 23: Complete group of companies which participated in public bids for Construction and Engineering Services in Curitiba - PR - Brazil in 2011 (left). Degree distribution for this network (right).

The complex network shown in Figure 23 has three partitions detected by the weighted K-means clustering algorithm. These partitions were obtained using the success rate as a measure of similarity. To store the partitions, an attribute was added to the vertices that identifies to which partition it belongs. At a later stage, we used the Louvain Method to detect the communities in the network. The complex network in Figure 23 has the following statistics, as presented in Table 6.

Table 6: Statistics for the Complex Network presented in Figure 23.

Size	544
Order	3129
Average Degree	11.50
Diameter	11
Modularity	0.718
Clustering Coefficient	0.521
Connected Components	35

The attribute which indicates to which group each vertex belongs was then updated and a graph layout algorithm applied.

4.1.8 RESULTS

Since the communities were evident in the graph, we isolated a very strongly connected community as a sample in order to analyze the participation and victories of each company for that particular community. Figure 24 shows a strongly attached group, suggesting a community or a cartel. This group of six companies is labeled as: A, B, C, D, E and F, with each element representing a distinct company. Table 8 shows the summarized data for this group.

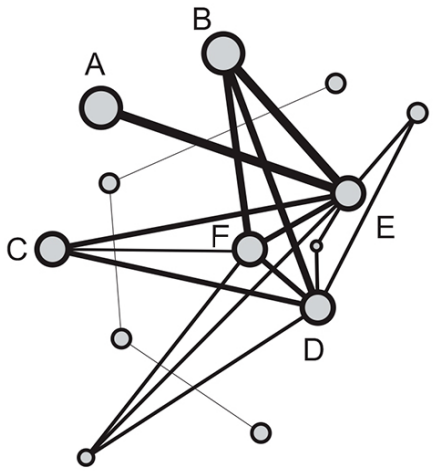


Figure 24: Indication of a community

It is very important point out that a mere connection between companies does not characterize a cartel. It is necessary to conduct an extensive evaluation of the public bids in which these companies participated to confirm or refute the hypothesis of a cartel. We highlighted as evidence of cartel formation: the number of companies ranging from 4 to 6 in the cartel; the increasing success rate in public bids that the group participates compared with its members where competing alone, and the repetition of the presence of the same group over time.

Table 7 shows the success rate for these companies labeled as A, B, C, D, E and F, referred to as Group-1, highlighted in Figure 24. Only B,C and D won public bids for 'Construction and engineering services' at Curitiba in 2011.

Table 8 shows the summarised data for companies A to F for all coincident participation in public bids from 2005 to 2011.

The triad corresponding to the three winners in this portion of data represented by the companies B, C and D had 26 repetitions over time.

For this group the winnings in public bids added up to 28 for 26 distinct public bids. There were 7 bids that none of the companies from this group won and there were 19 public

Table 7: Success rate of Group-1

Company	Participation	Number of Victories	Success Rate
A	2	0	0%
B	**7**	**2**	**29%**
C	**8**	**2**	**25%**
D	**2**	**1**	**50%**
E	5	0	0%
F	5	0	0%
Average	4.83	0.83	17%
Std. Dev.	2.48	0.98	0.21

Table 8: Summarised data from 2005 to 2011 for Group-1

Company	Number of participations	Number of Victories	Success Rate
A	39	15	38%
B	**236**	**73**	**31%**
C	**264**	**79**	**30%**
D	**134**	**48**	**36%**
E	281	92	33%
F	187	52	28%
Means	190.17	59.83	33%
Std. Dev.	91.45	27.52	4%

bids in which at least one member of the group won.

The average success rate for the companies in Group-1 participating detached from the group was 33%, as shown in Table 8. The average success rate for the companies of group-1 acting as a group is 77%. This represents an increase of 44% in the success rate.

The main evidence of the analogous behavior of a cartel identified through our methodology was:

- Increased rate of use in relation to individual participants;
- Presence of hubs in communities;
- Repetition of the same community over time;
- High similarity between community members (winners bind to winners and losers bind to losers);
- Companies with many participations in public bids proportionally tend to win more than companies with few holdings in public tenders.

These procedures were repeated for the ten mesoregions shown in Figure 22. Similar

results were observed for each mesoregion. The results were presented to the Court of Auditors of the State of Paraná (TCE/PR) through a specific report with several confirmations for the results presented.

4.2 CASE STUDY II - REPRESENTING MICROARRAY DATA AS GRAPHS

In the last decade, a massive volume of complex network data has been produced with surprising and unexpected results. The identification of essential principles common to complex networks are among these studies. The life-sciences and biology communities are active actors behind this revolution, using complex network analysis to help understand biological phenomena, diseases, epidemics and genetics (BOCCALETTI, 2009).

Several biological conditions including cancer and Alzheimer's disease have been studied through gene expression data. Techniques such as filtering data, cluster analysis and representation by means of complex networks can be employed to discover relationships between genes related to diseases and biological conditions (FINOCCHIARO et al., 2007).

The identification of regulatory gene networks are significant in order to understand diseases and biological conditions (LOPES; JR; COSTA, 2011). By using microarray data, it is possible to evaluate the correlation between tens of thousands of genes (MOSCATO; BERRETTA; MENDES, 2005). In Figure 25, some clustered data from microarray gene expression data are shown.

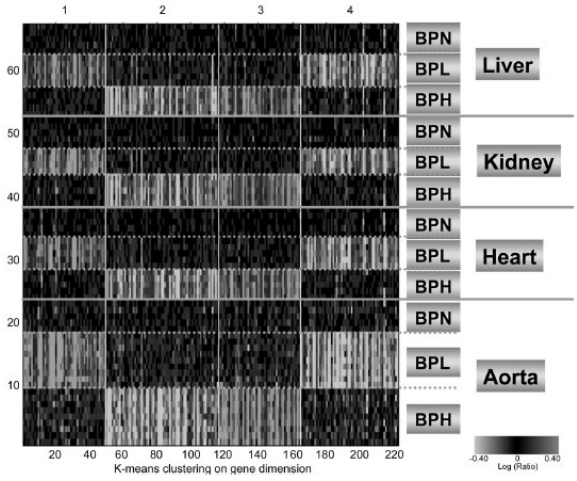


Figure 25: Heatmap of gene expression data from mice samples used to investigate the correlation between gene expression and arterial hypotension and arterial hypertension. - (PUIG et al., 2010).

Through DNA microarray analysis, it is possible to study distinct genes, groups of genes or even an entire genome. Gene expression data are usually represented by a matrix that

contains the genes identification as well their corresponding values of expression for a small number of samples.

In Case Study II, we analyzed microarray data to investigate the group of genes correlated to Alzheimer's disease with the following goals.

- Evaluate wheter the methodology presented and used in case study I is suitable for a distinct problem.
- Identify the group of genes with major changes for each stage of Alzheimer's disease.
- Identify the presence of communities of genes for each stage of Alzheimer's disease and how they as the disease progresses.

4.2.1 SIMILARITY MEASURES IN MICROARRAY DATA

Similarity measures in gene expression data play a key role in representing this information by means of complex networks (ALLOCCO; KOHANE; BUTTE, 2004). It is crucial to establish a similarity measure which allows distinct genes to be correlated according to a grouping criterion. This procedure enables the revelation of relationships among these groups of genes.

Similarity measures were used to compare subjects in Complex Networks for several distinct purposes. For instance, Blondel et al. (2004), used a measure of similarity between graph vertices applied to "synonym extraction" and web searching. Comparisons between two graphs were studied by Melnik, Garcia-Molina and Rahm (2002).

In this work, we are particularly interested in two dimensions: the macro-scale, where similarity comparisons will be made between graphs: and the micro-scale, where comparisons are drawn between vertices.

At the macro-scale, the intention was to establish comparisons between gene expression data and verify previously classified groups of genes found in the complex networks. The hypothesis was to identify communities of closely related genes present in one of the classes which are not present in other classes, thereby defining which set of genes is related to the progression of the pathology.

On the micro-scale, comparisons of genes were represented by means of graph vertices and their relationships are evaluated. Community detection algorithms were employed and some metrics for complex networks computed, such as: similarity, connectivity, clustering coefficients, etc.

4.2.2 EXPERIMENTS PERFORMED

In this case study we conducted experiments with a microarray dataset concerning the expression profiling of brain hippocampi from 22 postmortem subjects with Alzheimer's Disease (AD) with varying stages of severity, with 7, 8, and 7 subjects diagnosed with incipient, moderate and severe respectively. The results provide insight into mechanisms underlying the early pathogenesis of AD.

Alzheimer's is a form of dementia that affects ten percent of the population over 65 years old and almost fifty percent of the population over 85 years old (MOSCATO et al., 2005). It is a progressive disease which worsens as it evolves. It is treatable but can not be cured. The pathologic characteristics are the degeneration of the nerve cells, presence of neuritic plaques and neurofibrillary tangles. It was first described by German psychiatrist and neuropathologist Alois Alzheimer in 1906, after whom it was named (MCKHANN et al., 1984).

The dataset was obtained from the Gene Expression Omnibus (GEO), [2] a public functional genomics data repository from the National Center for Biotechnology Information (NCBI) [3]. The dataset is composed of 31 records (patient samples), comprising 22,283 probes, with 14,093 distinct genes.

The first step was to download the dataset, extract the headers and establish a proper identification of which case is related to each stage of Alzheimer's disease. GEO databases have the following structure: in the header section there is information about, the number of samples collected, number of genes, what classes are present as well as to which class each of the samples contained in the dataset belongs, plus the origin of samples, etc.. In the data table section, the values of expression for each gene of each sample are represented by means of a matrix $n \times n$, as shown in Table 9.

Table 9: Sample data structure for gene expression data.

	Sample 1	**Sample 2**	**Sample N**
Gene 1	1.3	2.9	1.9
Gene 2	2.1	1.8	0.1
Gene N	1.7	1.4	1.5

The same methodology used in the Case Study I, presented in the workflow shown in Figure 17 in Section 3.1 was adopted here to represent gene expression data from microarrays by means of complex networks.

[2] http://www.ncbi.nlm.nih.gov/geo/
[3] http://www.ncbi.nlm.nih.gov/

The second step was data preparation and normalization. We performed the normalization of the values by substituting the absolute gene expression values for the base-10 logarithm of the values, thus avoiding the sensibility to outliers and obtaining a normalized dataset. This procedure is usual when dealing with microarray data because the data can be in a large range of values.

In the third step, the dataset was divided according to the number of classes. Each class resulted in a distinct graph file. This approach allows the comparison of several metrics for each class.

Defining a criterion for the relationship between the vertices of a complex network is one of the features that most influences the final topology of the network. The criterion defining whether a relationship exists or not between two vertices may vary according to the network subject. For complex networks derived from gene expression data, one approach is to evaluate the correlation strength between two genes. If two genes show a strong correlation of their expression values, they receive an edge connecting them; otherwise there is no connection.

In the fourth step a similarity measure for evaluating the correlation between the vertices of the graph is established. Usually Pearson, Spearman or Kendall correlation metrics are employed. These metrics produce values between -1 and 1. Values close to -1 or 1 indicate strong, negative or positive correlations, respectively. Most tools suited for the analysis of complex networks as well as most of the algorithms for detecting community in complex networks do not support negative values for edge weights, so that the most common approach is to consider the absolute value of the correlation, converting negative values to positive. An edge between two vertices of the graph (representing genes) is set if the correlation between the expression of the corresponding genes is above a given threshold.

Steps five to eight outlined in Section 3.1 aim to identify communities inside the complex networks. Given the large dimension of the networks, the algorithm proposed by Blondel et al (BLONDEL et al., 2008) which was further discussed in Section 2.9.5, presented a suitable performance to analyze complex networks constructed with microarray data.

The final step in our methodology deals with the presentation of complex networks by graph layout algorithms. For large graphs, with thousands or more vertices, the visual representation provides little or no relevant information concerning the structure and topology of the graph. For the Case Study in question, this step has little or no relevance.

4.2.3 RESULTS

In this section we present results obtained from the analysis of the graphs generated by the microarray datasets used in Case Study II. It was stated that two distinct genes are connected by an edge if the correlation of the gene expressions throughout the cases is strong enough. Here, the meaning of strength is related to values that exceed a predefined threshold, as mentioned before.

A threshold equal to zero produces a fully connected graph, a considerable amount of noise and most connections with no meaning at all. Threshold value close to 1 produce less connected complex networks with more relevant connections. There is no pre-established threshold value. The proper adjustment of the threshold value depends on the subject and the problem modeling. Several authors (TROYANSKAYA et al., 2001; MEI et al., 2002; MCCLINTICK; EDENBERG, 2006) have established threshold value empirically according to the experimentation results.

The plots shown in Figure 26 gather information relative to the average degree of the complex networks for each stage of Alzheimer's disease, with the threshold ranging from 0 to 1. The continuous line represents the data itself and the dotted line represent the trend-line with the respective r^2 value. The threshold value used to produce the networks in this section is 0.85. This value was empirically set based on several results obtained during the course of the experiments.

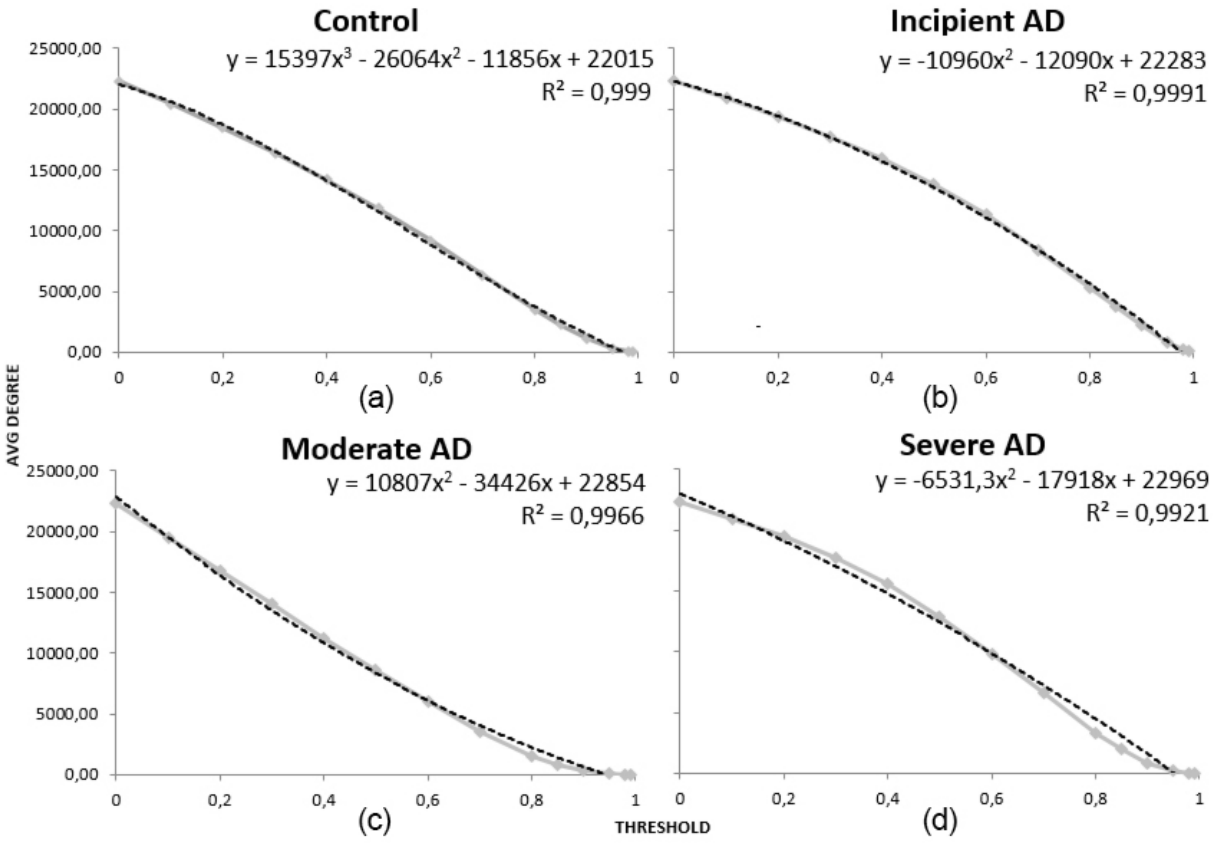


Figure 26: Average degree for each Alzheimer's disease stage with threshold ranging from 0 to 1. (a) Control group, (b) Incipient stage of AD, (c) Moderate stage of AD and (d) Severe stage of AD.

The four complex networks representing each stage of Alzheimer's disease are shown in Figure 27. This graphic representation was created using Gephi (BASTIAN; HEYMANN; JACOMY, 2009), a software for complex network visualization.

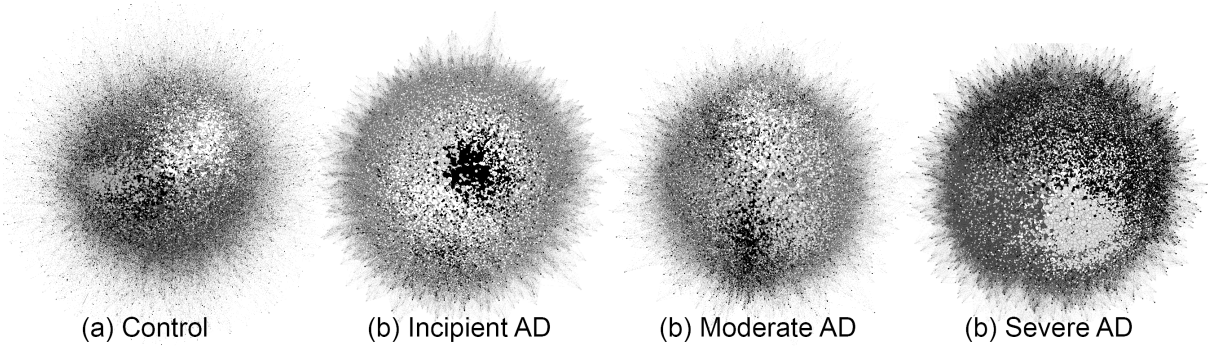


Figure 27: (a) complex network relative to the control group, (b) complex network relative to the Incipient stage of AD, (c) complex network relative to the Moderate stage of AD (d) complex network relative to the Severe stage of AD.

For networks with such complexity, no visual inference can be made that provide clues

about the distinctions between the four graphs. Accordingly, it is necessary to gather metrics and statistics that provide evidence of changes in the network for each of the classes presented. Table 10 contains statistics for the four complex networks shown in Figure 27.

Table 10: Metrics of complex networks for the classes shown in the graphs of Figure 27.

	Vertices	**Edges**	**AVG Degree**	**Modularity**	**Clustering Coefficient**	**# Communities**
Control	22215	1554337	140.11	0.579	0.307	12
Incipient AD	22215	8202414	738.45	0.501	0.396	9
Moderate AD	22215	2414388	217.36	0.586	0.332	11
Severe AD	22215	5949661	535.64	0.505	0.398	8

Table 10 shows the differences in the complex networks topology for each stage of Alzheimer's disease. In the Incipient AD and Severe AD classes we observed the most significant changes, with the highest average degrees in accordance with the highest number of edges. The number of communities is based on the network modularity and is computed using the Louvain method for community detection in large networks (BLONDEL et al., 2008) outlined in Section 2.9.5.

The average clustering coefficient was computed by using the algorithm of Latapy (2008). Another metric used to evaluate complex networks is the degree distribution network. Figure 28 contains the degree distribution for each complex network shown in Figure 27.

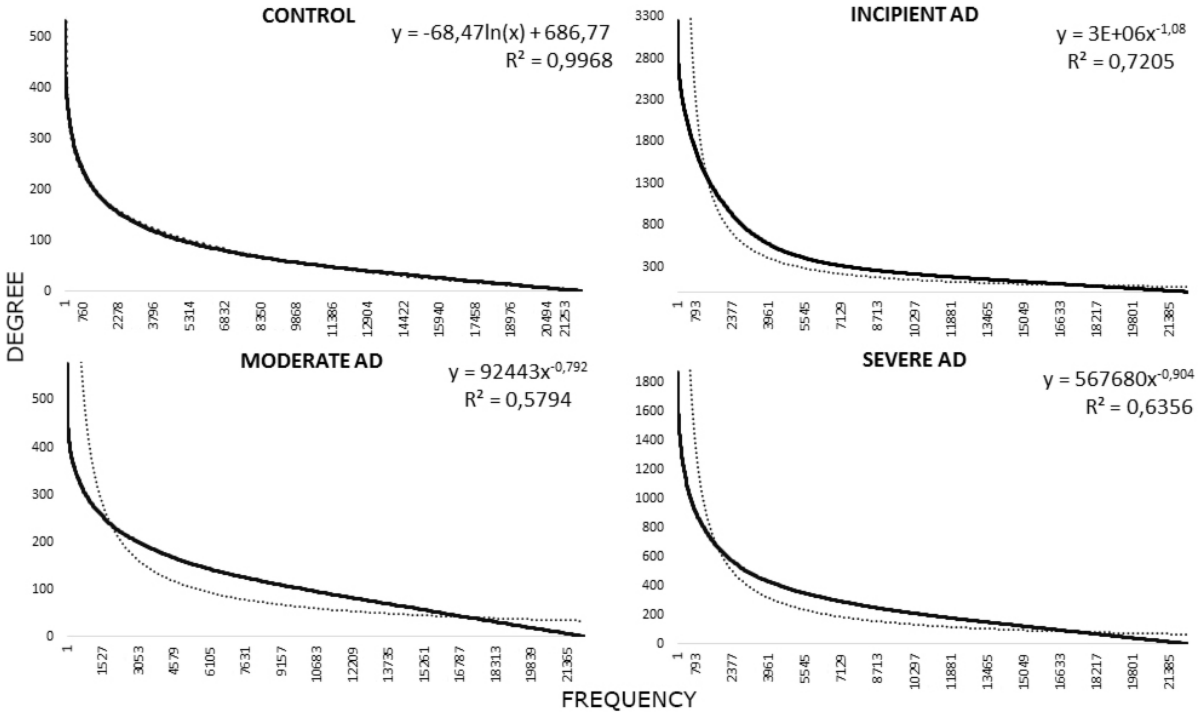


Figure 28: Degree Distribution for the classes Control, Incipient, Moderate and Severe from the Alzheimer's disease dataset.

In figure 28, the continuous lines represent the data itself while the dashed lines represent the trend-lines with respective R^2 values.

By submitting the complex networks to the algorithm for community detection proposed by Blondel et al. (2004) we identified communities for each stage of Alzheimer's disease as well for the control group. The number and dimensions of the identified communities are represented in Table 11.

Table 11: Comparative table of the size of communities for each Alzheimer's disease stage.

Control			Incipient			Moderate			Severe		
#	Size	Vertices	#	Size	Vertices	#	Size	Vertices	#	Size	Vertices
1	14.95%	2107	1	36.05%	5081	1	44.98%	6339	1	23.66%	3334
2	14.83%	2090	2	15.39%	2169	2	7.47%	1053	2	16.41%	2313
3	14.16%	1996	3	12.62%	1779	3	7.22%	1018	3	13.84%	1950
4	13.23%	1865	4	10.20%	1437	4	6.94%	978	4	11.35%	1600
5	10.27%	1447	5	9.13%	1287	5	6.93%	977	5	9.52%	1342
6	9.10%	1282	6	8.33%	1174	6	6.86%	967	6	8.60%	1212
7	8.39%	1182	7	8.27%	1165	7	6.60%	930	7	8.51%	1199
8	7.21%	1016	-	-	-	8	6.56%	925	8	8.12%	1144
9	6.72%	947	-	-	-	9	6.40%	902	-	-	-

Although interesting, the observation of these metrics does not provide a clear answer to the problem related to this dataset, that is, the identification of the group of genes related to the development or evolution of this pathology.

A possible approach to identifying the subset of genes related to a specific pathology is to identify the genes that have undergone major changes according to the progression of the disease. It is consistent to say that isolating the connections that are common to the control class from the other classes will filter out a relevant amount of data.

In order to evaluate the proportion of the complex network directly related to Alzheimer's disease, the group of edges and consequently vertices that are present in the Control class were removed from the analysis.

The Venn-diagram shown in Figure 29 illustrates the four distinct classes present in the dataset and the respective intersections.

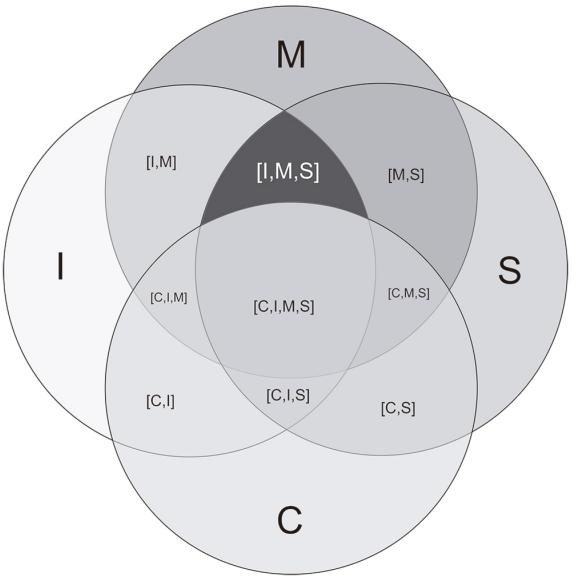


Figure 29: Venn diagram of the classes Identified as; C for Control, I for Incipient, M for Moderate and S for Severe, and the corresponding intersections.

By removing the intersection between the classes Incipient AD, Moderate AD and Severe AD which intersect with the Control class represented in the Venn-Diagram in Figure 29 by the intersections [C,I], [C,S], [C,I,S], [C,I,M], [C,M,S] and [C,I,M,S] three new complex networks were obtained. The three new complex networks are shown in Figure 30.

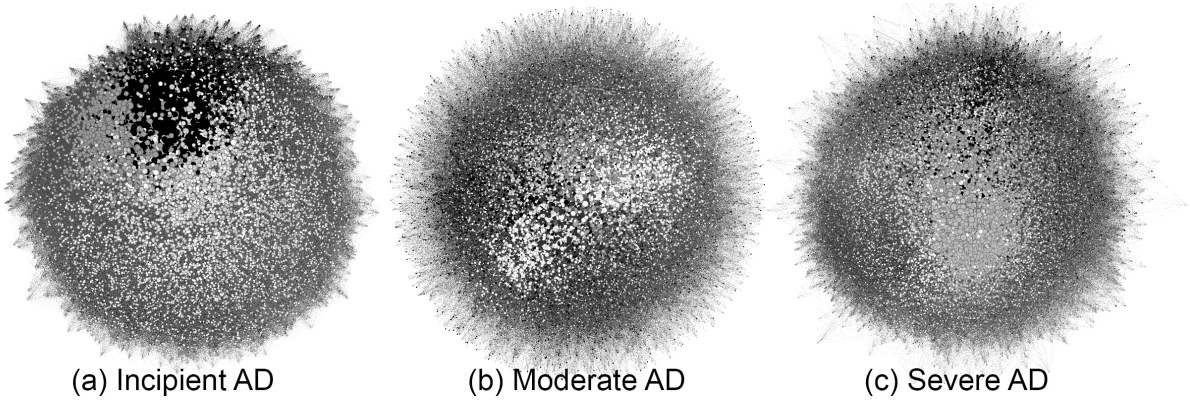


Figure 30: Complex networks from (a) Incipient AD, (b) Moderate AD and (c) Severe AD without in-common edges with the Control group.

Table 12 contains metrics and statistics for the complex networks resulting from the subtraction of the edges that are common to the Control group with the Incipient, Moderate and Severe stages of the Alzheimer's disease.

Table 12: Metrics of complex networks shown in Figure 30.

	Size	Order	Average Degree	Modularity	Clustering Coefficient	# Comunities
Incipient AD	14093	3085273	437.84	0.287	0.335	5
Moderate AD	14093	770046	109.28	0.408	0.241	11
Severe AD	14093	2175637	155.79	0.361	0.306	9

Computing the degree distribution for each of complex network generated, it is possible to observe that these complex networks also follow a power-law degree distribution as shown in Figure 31. The solid line represents the degree-distribution and the dotted line represents a power-law trend-line with corresponding r^2 value.

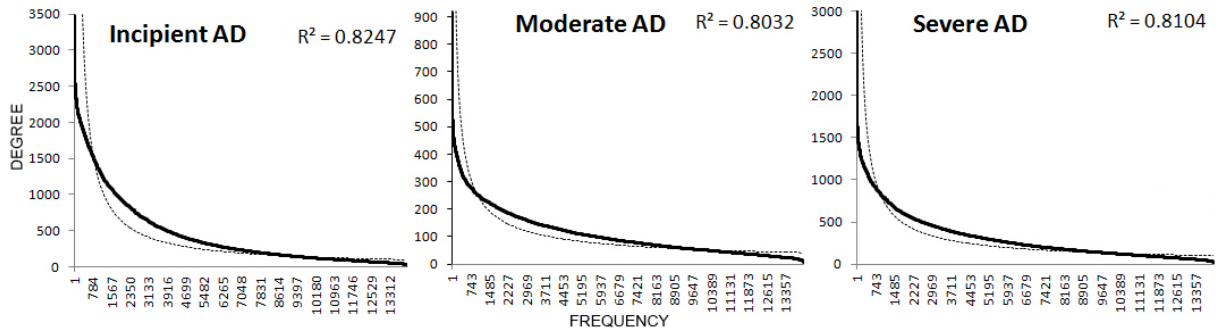


Figure 31: Degree distribution for the complex networks shown in Figure 30.

By observing the degree distribution of these three complex networks it is very clear that there are few vertices, under 10%, with a very high degree, while the majority, over 90% with a lower degree. Taking the connectivity of the vertices as a metric of importance in the network, the top 100 most connected vertices for each Alzheimer's disease stage were selected. Figure 32 shows the three complex networks relative to these groups of genes.

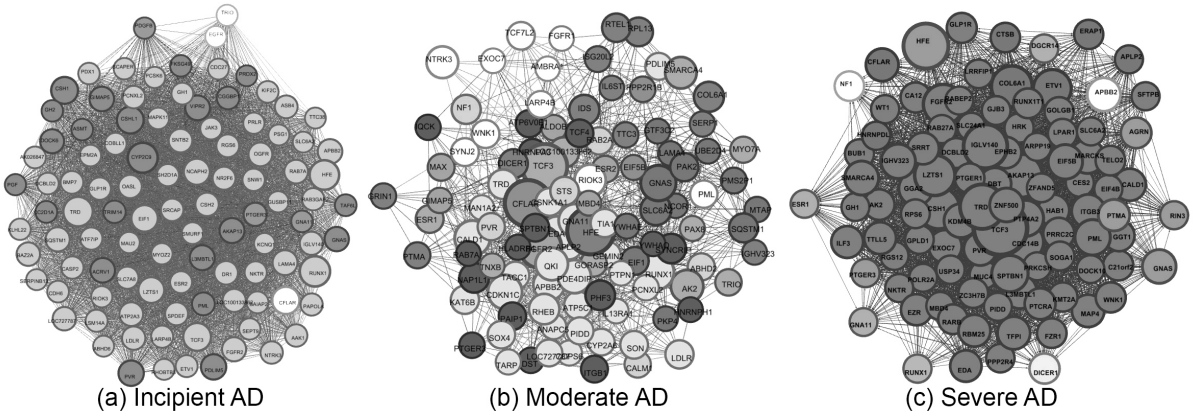


Figure 32: (a) Top-100 most connected genes for the group with Incipient AD, (b) top-100 most connected genes for the group with Moderate AD and (c) top-100 most connected genes for the group with Severe AD.

The repetition of a particular connection was used as a metric of strength for the connections. The stages Incipient, Moderate and Severe were considered for computing the weight of the edges, simply adding the repetitions, the connections in the control group were previously filtered and did not need to be computed at this stage. This procedure resulted in edge weights ranging from 1 to 3. The resulting complex network is shown in Figure 33.

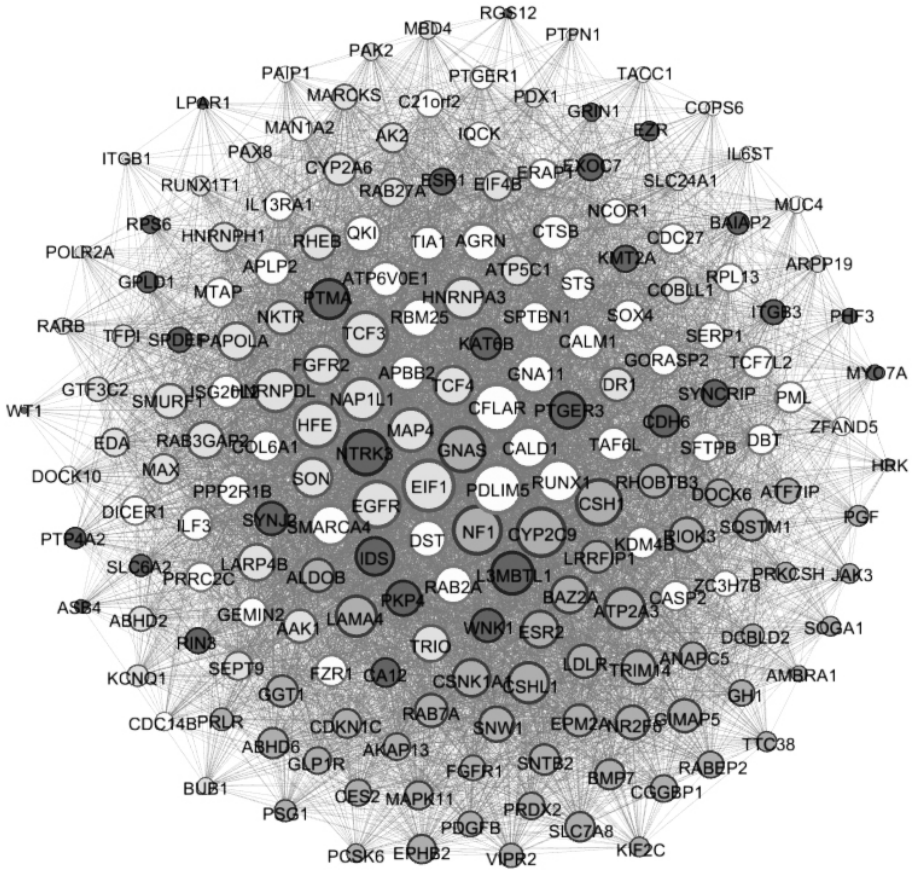


Figure 33: Complex network from the interaction between the top-100 most connected genes in all stages of Alzheimer's disease

Taking this complex network as a result of the combination of all the genes present in each stage of Alzheimer's disease, the communities present among these genes were computed according the modularity of the genes by the Blondel et al. (2004) algorithm showing 4 communities of genes listed as A, B, C and D. These genes and their respective communities are listed in Table 15 in the appendix D. The proportion of these communities relative to the entire network is listed in Table 13.

Table 13: Communities detected in the complex network for the combination of the top-100 genes for all stages of Alzheimer's disease.

Comunity A	Comunity B	Comunity C	Comunity D
55 Genes	31 Genes	47 Genes	55 Genes
29%	16%	25%	29%

As an alternative approach, data relative to the intersection [I,M,S] identified in the Venn Diagram of Figure 29 were selected as result of the intersection between the groups labeled as I for incipient Alzheimer's disease, M for Moderate Alzheimer's disease and S for severe Alzheimer's disease. The connections within this intersection and those that are not present in the Control class were selected in order to include the genes which were present in all stages of Alzheimer's disease and which are not present in the Control group. These genes and their respective connections are illustrated by the complex network shown in Figure 34.

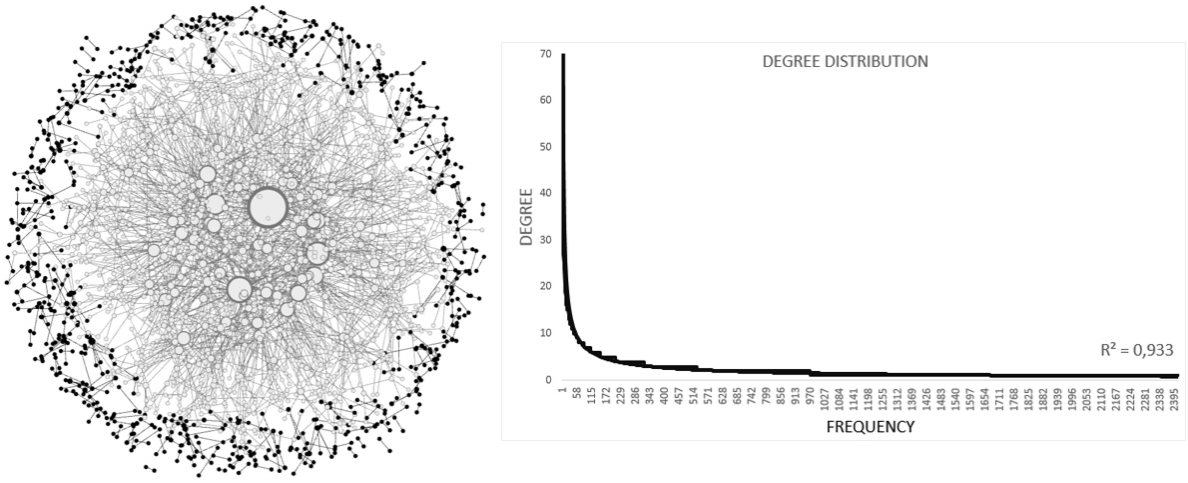


Figure 34: Complex network relative to the group of genes present in all stages of Alzheimer's disease and not present in the control group and its respective degree distribution chart.

By computing the clustering coefficient for the complex network presented in Figure 34 a total of 298 connected components were identified. The distribution of components constitutes a large component equivalent to 70.39% of the complex network and 297 small components with a portion of 0.05% or less.

By computing the modularity for the complex network presented in Figure 34 a total of 294 modularity classes were computed, with a large class encompassing 70.4% of the complex network and with 293 small classes with a size equal or less than 0.37% of the network.

For a simplified representation, the largest components are presented in light color and all other components are presented as black in Figure 34.

In order to obtain a reasonable small-set of genes, the same methodology of selecting the top-100 most connected genes was used. These genes are illustrated in the complex network presented in Figure 35.

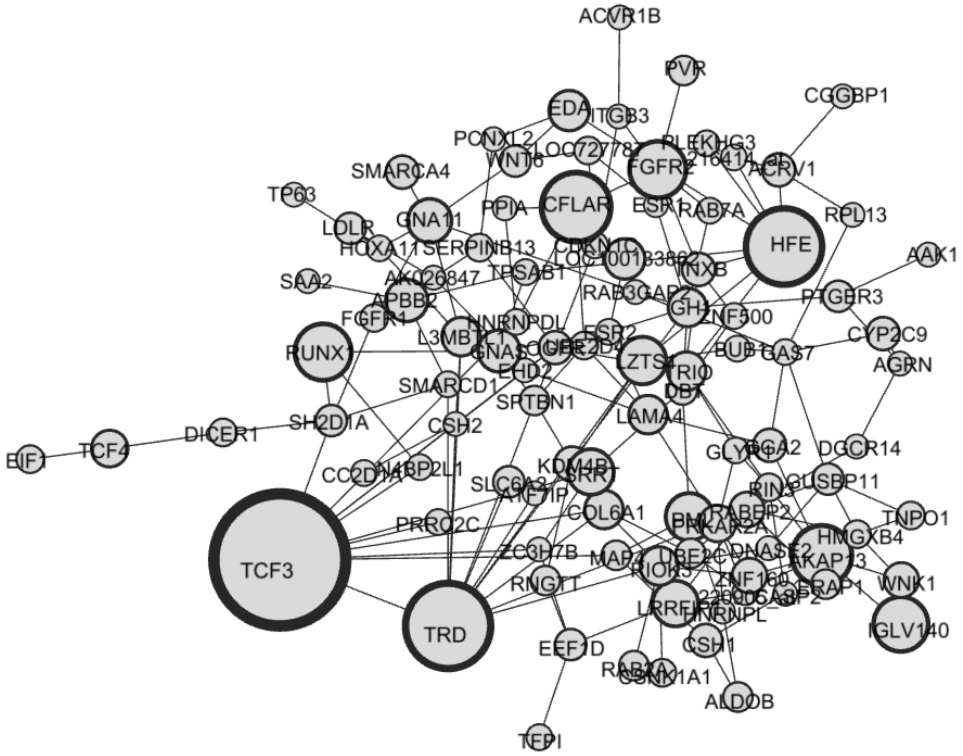


Figure 35: Top-100 most connected genes in complex network from intersection of Incipient AD, Moderate AD and Severe AD which are not part of the Control Group.

This group of genes is totally contained in the large component that overlaps the large modularity class presented in Figure 34.

4.2.4 EVALUATION OF RESULTS

Geneticists have conducted several studies in order to identify which genes are associated with human diseases with substantial progress. Once identified a gene or a group of genes associated with a specific disease, it is possible to develop new drugs and treatments. The fast evolution of research related to the human genome combined with the considerable attention given to Alzheimer's disease in recent decades has led to important advances regarding the identification of which genes are directly linked or correlated to the development of Alzheimer's disease.

The vast literature on this topic points to discoveries encompassing from a small group of genes composed by four or five genes to datasets with thousands of genes correlated to Alzheimer's disease.

The top-100 most connected genes presented in Table 14 in Appendix D for the group of subjects identified with Incipient AD has 74 genes listed in The Comparative Toxicogenomics

Database (CTD) [4] (DAVIS et al., 2013). Class Moderate AD has 88 genes listed and class Severe AD has 80 genes listed. The average of matching in genes present in Figure 32 and the dataset The Comparative Toxicogenomics Database (CTD) is 80,66%.

Furthermore, in the group present in the complex network shown in Figure 13 there are twelve genes present in all stages of Alzheimer's disease, as follows: *APBB2*, *CFLAR*, *FGFR2*, *GNA11*, *GNAS*, *HFE*, *PML*, *PTGER3*, *PVR*, *RUNX1*, *SLC6A2* and *TCF3*.

In all the analyse we observed that, the phase with the highest genetic changes was the early stage of Alzheimer's disease, which is consistent with the results observed by Blalock et al. (2004). "These studies reveal that widespread changes in genomic regulation of multiple cellular pathways are major correlates of incipient AD".

Harold et al. (2009) conducted a study that identified variants at *CLU* or *APOJ* associated with Alzheimer's disease. These associations were replicated in stage 2 (2,023 cases and 2,340 controls), producing compelling evidence for association with Alzheimer's disease. Genes *CLU*, *CR1* and *PICALM* were also highlighted as compelling evidence for association with Alzheimer's disease by Harold et al. (2009), both of which are present in complex network shown in Figure 34.

Genes *CELF1*, *FERMT2*, *INPP5D*, *MEF2C*, also present in the complex network shown in Figure 34, were identified as susceptibility loci for late-onset Alzheimer's disease by Lambert et al. (2013).

Genes *HFE* hemochromatosis, *APBB2* amyloid beta (A4) precursor protein-binding, family B, member 2, *BLMH* bleomycin hydrolase, *MPO* myeloperoxidase, and *NOS3* nitric oxide synthase 3 (endothelial cell) are also positively identified in the group of genes in the complex network shown in Figure 34. According to Bird (2014), these genes are also associated whit Alzheimer's disease.

The genes presented in the complex network shown in Figure 34 include 16 genes listed in the previously cited studies, and with 1764 of 2411 which corresponds to 73% of matching with genes listed in The Comparative Toxicogenomics Database (CTD) (DAVIS et al., 2013).

The results obtained by the alternative approach, which consists of selecting the genes which are present at the Incipient, Moderate and Severe stages of Azheimer's disease and are not present in the Control group relative to the complex network presented in Figure 35. This complex network has a majority of the 73% of its genes listed in The Comparative

[4] http://ctdbase.org/downloads

Toxicogenomics Database (CTD).

All 73% of the genes positively matched with the (CTD) are within the large component which also coincides with the largest modularity class. Likewise, the majority of the genes presented in (HAROLD et al., 2009; LAMBERT et al., 2013; BIRD, 2014) that are positively matched with this complex network are outside the large component. The list of top-100 most connected genes in complex networks from intersection of Incipient AD, Moderate AD and Severe AD which are not part of the Control Group is presented in Table 16 in appendix D.

5 CONCLUSIONS AND FUTURE WORK

In this dissertation two distinct case-studies were addressed in order to illustrate a methodology by means of complex network for representing datasets and detecting communities among these data.

5.1 CASE STUDY I - DETECTING CARTELS IN PUBLIC BIDS

In Case Study I, the representation of public bids by means of complex networks allowed the discovery of companies acting as a tight group of members analogous to cartels. Metrics of strength based on repetitions over time were employed, as well as algorithms for detecting communities in order to unveil these groups of companies. The problem was addressed by representing the relationships between companies similar to a Social Network, allowing the employment of Social Network Analysis tools and algorithms to answer the questions stated in the problem.

Case Study I was conducted under an agreement with the Court of Auditors of Paraná State, Brazil that provided data from the public bids in Paraná State. In return, a report with the findings relative to this case-study was provided to the Court of Auditors of Paraná State which had several positive results that corroborated the accuracy of the methodology employed here.

The following answers were obtained for the questions addressed to the Case Study I.

- **Do the winners have an advantage over companies that always lose?** As per the information presented in Section 4.1.8 we were able to observe strong connections between major winners and companies with very low success rates, thus, demonstrating evidence of the presence of ghost companies in public bids in the State of Paraná.
- Are the winners clustered together? It is possible to observe that similar subjects tends to be clustered. The experimentations and results shown that companies with higher success rates tend to be grouped. Companies with lower success rates also appears clustered within the complex networks. The companies with average success rate are present in

both communities and only a few number of very low success rate are connected to successful companies.

- **Will a company have a better success rate the more it participates?** The number of participations show a correlation with the success rates, we were able to observe that as more a company participates, higher the success rate.

- **How many companies there in of a cartel?** According the literature cartels are small and cohesive groups of companies acting together to obtain advantages in public bids or auctions preventing the free concurrence. The number of companies within a cartel it is estimated between three and six. We were able to observe groups of companies with strong connections ranging with the same number of companies. However, sometimes this number is slightly larger.

5.2 CASE STUDY II - REPRESENTING MICRO ARRAY DATA AS GRAPHS

In Case Study II, the main goal was to evaluate the methodology proposed in this dissertation through a distinct problem and dataset. Microarray gene expression data related to Alzheimer's disease was represented by means of complex networks. Correlation measurements were used to establish network connections.

The result of this procedure was large complex networks composed of thousands of vertices and millions of edges, which constituted a computational challenge itself.

Community detection algorithms were also employed in order to unveil the presence of groups of genes for each stage of the Alzheimer's disease. The comparison of the results found with previously published works corroborates the findings obtained by the methodology presented in this dissertation.

The following answers were obtained for the questions addressed to the Case Study II.

- **The methodology presented and used in case study I is suitable for a distinct problem?** The same methodology of case Study I was used for the Case Study II with good results. For the Case Study II the graphical representation of the complex networks it is not relevant due the network dimensions, thus, the optional steps eight and nine are not performed.

- **It is possible to identify the group of genes with major changes for each stage of Alzheimer's disease?** The results found in the Case Study II were compared with the

The Comparative Toxicogenomics Database (CTD) with an average matching of 80.66 percent. The groups of genes identified with the methodology suggests strong evidence to be correlated with the Alzheimer's disease.

- **It is possible to identify the presence of communities of genes for each stage of Alzheimer's disease and how they as the disease progresses?** Two distinct approaches were performed to evaluate the progression of Alzheimer's disease and in both case is possible correlate the progression of each phase to a distinct group of genes.

5.3 PRODUCTS

During the course of this study academic material, software and technical report were produced as a result of this research. In this section are listed the major productions related to this work.

5.3.1 SOFTWARE

The software produced during this study consists in a graphical interface and a set of algorithms capable to deal with the complex networks presented in this dissertation. The software comprise the three artificial network generators for random network, Barabási and Albert model and Erdős and Rényi network model.

The software is able to deal with open graph file formats such as .gml and .graphml. Further information about the Open Graph File Format is available in Appendix A.

For visualization of the complex networks the GraphStream Java library [1] was used. For correlation and statistics, The Apache Commons Mathematics Library [2] was used, aside with, The Gephi Toolkit [3] for community detection algorithms.

5.3.2 TECH REPORT

A tech report entitled 'Relatório de Atividades Relativo ao Convênio TCE/PR e UTFPR Referente ao Estudo de Caso de Identificação de Cartéis por Meio de Redes Complexas' (Gabardo et al., 2014) was produced and delivered to the Court of Auditors of Paraná State as result of the cooperative effort accomplished during the Case Study I.

[1] http://graphstream-project.org/
[2] http://commons.apache.org/proper/commons-math/
[3] http://gephi.github.io/toolkit/

5.3.3 PUBLICATIONS

The following publications were produced during the course of this study:

- **Scientific Journal** - Gabardo, A. C., & Pérez, M. (2013). Classificação de dados relativos á cirurgia de câncer de mama, um comparativo entre solução por Redes Neurais e Fuzzy. *REAVI-Revista Eletrônica do Alto Vale do Itajaí*, 2(2), 50-59.
- **Book Chapter** - Gabardo, A. C., & Lopes, H. S. (2014). Clustering Methods for Detecting Communities in Networks. In M. Khosrow-Pour (Ed.), *Encyclopedia of Information Science and Technology, Third Edition* (pp. 3507-3516). Hershey, PA: Information Science Reference. doi:10.4018/978-1-4666-5888-2.ch344
- **Conference Presentation** - Gabardo, A. C., & Lopes, H. S. (2014). Using Social Network Analysis to Unveil Cartels in Public Bids. in *Proceedings of The First European Network Intelligence Conference*.

5.4 CONCLUSION

Both case studies show that complex networks are a powerful tool for representing complex data relationships. Several techniques can be employed to reveal communities and groups of subjects with a high degree of similarity. The information related to community discovery in complex networks seems to be a reliable methodology for several fields of study, pervading a wide range of sciences, from Social Network Analysis to biological data, and beyond to other possible applications.

5.5 FUTURE WORK

Future work should include experiments with other biological datasets, with special interest in gene expression data. The methodology and findings obtained in this study will be gathered in scientific publications at an appropriate time.

Another direction for future research could be focused on the discovery of strength metrics for the relationships between vertices for relationships and the dynamic aspects of complex networks.

REFERENCES

ABDI, H. The Kendall rank correlation coefficient. **Encyclopedia of Measurement and Statistics**, p. 508–510, 2007.

ALBERT, R.; BARABÁSI, A.-L. Statistical mechanics of complex networks. **Reviews of modern physics**, v. 74, n. 1, p. 47–97, 2002.

ALEXANDERSON, G. About the cover: Euler and Königsberg's bridges: A historical view. **Bulletin of the american mathematical society**, v. 43, n. 4, p. 567–573, 2006.

ALLOCCO, D. J.; KOHANE, I. S.; BUTTE, A. J. Quantifying the relationship between co-expression, co-regulation and gene function. **BMC bioinformatics**, v. 5, n. 1, p. 18, 2004.

AMARAL, L. A.; OTTINO, J. M. Complex networks. **The European Physical Journal B-Condensed Matter and Complex Systems**, v. 38, n. 2, p. 147–162, 2004.

AMARAL, L. A. N. et al. Classes of small-world networks. **Proceedings of the National Academy of Sciences**, v. 97, n. 21, p. 11149–11152, 2000.

AMORIM, R. C. de; KOMISARCZUK, P. On initializations for the minkowski weighted K-means. In: **Advances in Intelligent Data Analysis XI**. [S.l.: s.n.], 2012. v. 7619, p. 45–55.

AMORIM, R. C. de; MIRKIN, B. Minkowski metric, feature weighting and anomalous cluster initializing in k-means clustering. **Pattern Recognition**, v. 45, n. 3, p. 1061–1075, 2012.

BARABÁSI, A.-L.; ALBERT, R. Emergence of scaling in random networks. **science**, v. 286, n. 5439, p. 509–512, 1999.

BARABÁSI, A.-L.; ALBERT, R.; BONABEAU, E. Scale-free. **Scientific American**, 2003.

BASTIAN, M.; HEYMANN, S.; JACOMY, M. Gephi: An open source software for exploring and manipulating networks. In: ADAR, E. et al. (Ed.). **ICWSM**. The AAAI Press, 2009. Disponível em: <http://dblp.uni-trier.de/db/conf/icwsm/icwsm2009.html>.

BATAGELJ, V.; BRANDES, U. Efficient generation of large random networks. **Physical Review E**, v. 71, n. 3, p. 036113, 2005.

BEZDEK, J.; EHRLICH, R.; FULL, W. FCM: The fuzzy c-means clustering algorithm. **Computers & Geosciences**, v. 10, n. 2-3, p. 191–203, 1984.

BIRD, T. D. **Alzheimer's disease - Conditions Phenotypes**. 2014. `http://www.ncbi.nlm.nih.gov/gtr/conditions/C0002395`. Accessed: 2014-06-20.

BLALOCK, E. M. et al. Incipient Alzheimer's disease: microarray correlation analyses reveal major transcriptional and tumor suppressor responses. **Proceedings of the National Academy of Sciences of the United States of America**, v. 101, n. 7, p. 2173–2178, 2004.

BLONDEL, V. D. et al. A measure of similarity between graph vertices: Applications to synonym extraction and web searching. **Society for Industrial and Applied Mathematics Review**, v. 46, n. 4, p. 647–666, 2004.

BLONDEL, V. D. et al. Fast unfolding of communities in large networks. **Journal of Statistical Mechanics: Theory and Experiment**, v. 2008, n. 10, p. P10008, 2008.

BOCCALETTI, S. **Handbook on Biological Networks**. Hackensack, USA: World Scientific, 2009.

BONDY, J. A.; MURTY, U. S. R. **Graph Theory with Applications**. [S.l.]: Elsevier Science Publishing Co., Inc - New York, USA, 1976.

BRANDES, U. et al. On modularity clustering. **IEEE Transactions on Knowledge and Data Engineering**, v. 20, n. 2, p. 172–188, 2008.

BRANDES, U. et al. Graphml progress report structural layer proposal. In: **Graph Drawing**. New York: Springer, 2002. p. 501–512.

BRANDES, U.; PICH, C. Graphml transformation. In: **Graph Drawing**. New York: Springer, 2005. p. 89–99.

BRASIL. **Lei n^{o} 12.529, de 30 de novembro de 2011. Estrutura o Sistema Brasileiro de Defesa da Concorrência; dispõe sobre a prevenção e repressão às infrações contra a ordem econômica; altera a Lei no 8.137, de 27 de dezembro de 1990, o Decreto-Lei no 3.689, de 3 de outubro de 1941 - Código de Processo Penal, e a Lei no 7.347, de 24 de julho de 1985; revoga dispositivos da Lei no 8.884, de 11 de junho de 1994, e a Lei no 9.781, de 19 de janeiro de 1999; e dá outras providências. Diário Oficial [da República Federativa do Brasil], Brasília, p.1, 01 dez. 2011. Seção 1, pt1.** Novembro 2011.

CANNON, R. L.; DAVE, J. V.; BEZDEK, J. C. Efficient implementation of the fuzzy c-means clustering algorithms. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, n. 2, p. 248–255, 1986.

CARVALHO, J. dos S. F. **Manual de Direito Administrativo**. [S.l.]: Lúmen Júris - Curitiba, 2005.

CASTEIGTS, A. et al. Time-varying graphs and dynamic networks. **International Journal of Parallel, Emergent and Distributed Systems**, v. 27, n. 5, p. 387–408, 2012.

CHEN, P. Y.; POPOVICH, P. M. **Correlation: Parametric and nonparametric measures**. Thousand Oaks, USA: Sage, 2002.

COSTA, L. D. F.; RODRIGUES, F. A.; TRAVIESO G. ANDVILLAS BOAS, P. R. Characterization of complex networks: A survey of measurements. **Advances in Physics**, v. 56, n. 1, p. 167–242, 2005.

DAVIS, A. P. et al. **The Comparative Toxicogenomics Database**. Jan 2013. Disponível em: <http://ctdbase.org/downloads>.

ERDŐS, P.; RÉNYI, A. On the evolution of random graphs. **Publication of the Mathematical Institute of the Hungarian Academy of Sciences**, v. 5, p. 17–61, 1960.

FAYYAD, U.; BRADLEY, P. S.; REINA, C. **Scalable system for expectation maximization clustering of large databases**. [S.l.]: Google Patents, jul. 17 2001. US Patent 6,263,337.

FINOCCHIARO, G. et al. Graph-based identification of cancer signaling pathways from published gene expression signatures using publime. **Nucleic Acids Research**, v. 35, n. 7, p. 2343–2355, 2007.

FORTUNATO, S. Community detection in graphs. **Physics Reports**, v. 486, n. 3-5, p. 75–174, 2010.

FORTUNATO, S.; LATORA, V.; MARCHIORI, M. Method to find community structures based on information centrality. **Physical Review E**, v. 70, n. 5, p. 056104, 2004.

FRANK, K. A. Mapping interactions within and between cohesive subgroups. **Social Networks**, v. 18, n. 2, p. 93–119, 1996.

FREEMAN, L. Centrality in social networks conceptual clarification. **Social Networks**, v. 1, n. 3, p. 215–239, 1979.

FREEMAN, L. C. A set of measures of centrality based on betweenness. **Sociometry**, v. 40, n. 1, p. 35–41, 1977.

FRUCHTERMAN, T. M.; REINGOLD, E. M. Graph drawing by force-directed placement. **Software: Practice and experience**, v. 21, n. 11, p. 1129–1164, 1991.

GIRVAN, M.; NEWMAN, M. E. J. Community structure in social and biological networks. **Proceedings of the National Academy of Sciences of the United States of America**, v. 99, n. 12, p. 7821–7826, June 2002.

GRANOVETTER, M. The strength of weak ties. **American Journal of Sociology**, v. 78, n. 6, p. l, 1973.

HAROLD, D. et al. Genome-wide association study identifies variants at CLU and PICALM associated with Alzheimer's disease. **Nature Genetics**, v. 41, n. 10, p. 1088–1093, 2009.

HARTIGAN, J. A.; WONG, M. A. Algorithm AS 136: A k-means clustering algorithm. **Applied Statistics**, v. 28, n. 1, p. 100–108, 1979.

HOPCROFT, J. E.; TARJAN, R. E. Dividing a graph into triconnected components. **SIAM Journal on Computing**, v. 2, n. 3, p. 135–158, 1973.

HU, Y. Efficient, high-quality force-directed graph drawing. **Mathematica Journal**, v. 10, n. 1, p. 37–71, 2005.

HUANG, J. Z. et al. Automated variable weighting in k-means type clustering. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 27, n. 5, p. 657–668, 2005.

HUANG, Z. Extensions to the k-means algorithm for clustering large data sets with categorical values. **Data Mining and Knowledge Discovery**, v. 2, n. 3, p. 283–304, 1998.

HUCK, S.; NORMANN, H.-T.; OECHSSLER, J. **Two are Few and Four are Many: Number Effects in Experimental Oligopolies**. Bonn, Germany: Bonn Graduate School of Economics, 2001.

IMAN, R. L.; CONOVER, W. A distribution-free approach to inducing rank correlation among input variables. **Communications in Statistics-Simulation and Computation**, v. 11, n. 3, p. 311–334, 1982.

KANUNGO, T. et al. An efficient k-means clustering algorithm: Analysis and implementation. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 24, n. 7, p. 881–892, 2002.

KENDALL, M. G. A new measure of rank correlation. **Biometrika: a journal for the statistical study of biological problems**, v. 30, n. 1/2, p. 81–93, 1938.

KENDALL, M. G. Rank correlation methods. Griffin, Oxford, England, 1948.

KOBOUROV, S. G.; WAMPLER, K. Non-euclidean spring embedders. **IEEE Transactions on Visualization and Computer Graphics**, v. 11, n. 6, p. 757–767, 2005.

Laboratório de Cartografia Tátil Escolar - UFSC. **Mapa Mesoregiões Paraná**. 2012. `http://www.labtate.ufsc.br/images/PR_mapa_tatil_regioes.pdf`. Accessed: 2014-05-27.

LAMBERT, J.-C. et al. Meta-analysis of 74,046 individuals identifies 11 new susceptibility loci for Alzheimer's disease. **Nature Genetics**, v. 45, n. 12, p. 1452–1458, 2013.

LATAPY, M. Main-memory triangle computations for very large (sparse (power-law)) graphs. **Theoretical Computer Science**, v. 407, n. 1, p. 458–473, 2008.

LEVENSTEIN, M. C.; SUSLOW, V. Y. What determines cartel success? **Journal of Economic Literature**, v. 44, n. 1, p. 43–95, 2006.

LOPES, F. M.; JR, R. M. C.; COSTA, L. D. F. Gene expression complex networks: synthesis, identification, and analysis. **Journal of Computational Biology**, v. 18, n. 10, p. 1353–1367, 2011.

MACQUEEN, J. B. Some methods for classification and analysis of multivariate observations. In: CAM, L. M. L.; NEYMAN, J. (Ed.). **Procedures of the Fifth Berkeley Symposium on Mathematical Statisticsand Probability**. Berkeley, USA: University of California Press, 1967. v. 1, p. 281–297.

MANGOLD, W. G.; FAULDS, D. J. Social media: The new hybrid element of the promotion mix. **Business Horizons**, v. 52, n. 4, p. 357–365, 2009.

MATLAB. **version 7.10.0 (R2010a)**. [S.l.]: The MathWorks Inc., 2010.

MAYFIELD, R. Extreme democracy. In: LEBKOWSKY, J.; RATCLIFFE, M. (Ed.). [S.l.: s.n.], 2005. cap. Social Network Dynamics and Participatory Politics, p. 116–132.

MCCLINTICK, J. N.; EDENBERG, H. J. Effects of filtering by present call on analysis of microarray experiments. **BMC Bioinformatics**, v. 7, n. 1, p. 49, 2006.

MCKHANN, G. et al. Clinical diagnosis of Alzheimer's disease report of the NINCDS-ADRDA Work Group under the auspices of department of health and human services task force on Alzheimer's disease. **Neurology**, v. 34, n. 7, p. 939–939, 1984.

MEI, R. et al. Analysis of high density expression microarrays with signed-rank call algorithms. **Bioinformatics**, v. 18, n. 12, p. 1593–1599, 2002.

MELNIK, S.; GARCIA-MOLINA, H.; RAHM, E. Similarity flooding: A versatile graph matching algorithm and its application to schema matching. In: **18th International Conference on Data Engineering (ICDE 2002)**. [S.l.: s.n.], 2002.

MILGRAM, S. The small world problem. **Psychology Today**, v. 67, n. 1, p. 61–67, 1967.

MODHA, D. S.; SPANGLER, W. S. Feature weighting in k-means clustering. **Machine Learning**, v. 52, n. 3, p. 217–237, 2003.

MOSCATO, P.; BERRETTA, R.; MENDES, A. A new memetic algorithm for ordering datasets: applications in microarray analysis. In: **Proc. MIC2005–The 6th Metaheuristics International Conference**. [S.l.: s.n.], 2005. p. 695–700.

MOSCATO, P. et al. Genes related with alzheimer's disease: A comparison of evolutionary search, statistical and integer programming approaches. In: **Applications of Evolutionary Computing**. [S.l.: s.n.], 2005. p. 84–94.

NEWMAN, M. Fast algorithm for detecting community structure in networks. **Physical Review E**, v. 69, n. 6, p. 066133, 2003.

NEWMAN, M. A measure of betweenness centrality based on random walks. **Social Networks**, v. 27, n. 1, p. 39–54, 2005.

NEWMAN, M. Communities, modules and large-scale structure in networks. **Nature Physics**, v. 8, n. 1, p. 25–31, 2012.

NEWMAN, M.; BARABASI, A.-L.; WATTS, D. J. **The Structure and Dynamics of Networks**. Princeton, NJ: Princeton University Press, 2006.

NEWMAN, M. E. Mixing patterns in networks. **Physical Review E**, v. 67, n. 2, p. 026126, 2003.

NEWMAN, M. E. The structure and function of complex networks. **Society for Industrial and Applied Mathematics Review**, v. 45, n. 2, p. 167–256, 2003.

NEWMAN, M. E. J. The structure and function of complex networks. **SIAM Review**, v. 45, n. 2, p. 167–256, 2003.

NEWMAN, M. E. J.; GIRVAN, M. Finding and evaluating community structure in networks. **Physical Review E**, v. 69, n. 2, p. 026113, 2004.

NIEDERMEIER, R.; SANDERS, P. **On the Manhattan Distance Between Points on Space Filling Mesh Indexings**. Tuebingen, DE: Univ., Fak. für Informatik, 1996.

OFT Office of Fair Trading. **Quick Guide to Cartels and Leniency for Individuals**. 2013.

PAL, N. R.; BEZDEK, J. C. On cluster validity for the fuzzy c-means model. **IEEE Transactions on Fuzzy Systems**, v. 3, n. 3, p. 370–379, 1995.

PARSHANI, R.; BULDYREV, S. V.; HAVLIN, S. Interdependent networks: Reducing the coupling strength leads to a change from a first to second order percolation transition. **Physical Review Letters**, v. 105, n. 4, p. 048701, 2010.

PEARSON, K. Notes on regression and inheritance in the case of two parents. **Proceedings of the Royal Society of London**, v. 58, p. 240–242, 1895.

PEMMARAJU, S. V.; SKIENA, S. S. **Computational Discrete Mathematics**. Cambridge, UK: Cambridge University Press, 2003.

PFEFFER, J.; CARLEY, K. M. K-centralities: local approximations of global measures based onshortest paths. In: **Proceedings of the 21st International Conference Companion on World Wide Web**. New York, NY, USA: ACM, 2012. p. 1043–1050.

PORTER, M. A.; ONNELA, J.-P.; MUCHA, P. J. Communities in networks. **Notices of the AMS**, v. 56, n. 9, p. 1082–1097, 2009.

PUIG, O. et al. Transcriptome profiling and network analysis of genetically hypertensive mice identifies potential pharmacological targets of hypertension. **Physiological Genomics**, v. 42, n. 1, p. 24–32, 2010.

SANTORO, N. et al. Time-varying graphs and social network analysis: Temporal indicators and metrics. **AISB Social Networks and Multiagent Systems Symposium**, v. 3, p. 32–38, 2011.

SHIOKAWA, H.; FUJIWARA, Y.; ONIZUKA, M. Fast algorithm for modularity-based graph clustering. In: **Twenty-Seventh AAAI Conference on Artificial Intelligence**. [S.l.: s.n.], 2013. p. 1170–1176.

SILVA, C. V. S.; RALHA, C. G. Detection of cartel formation in government biddings using data mining agents. **Revista Eletrônica de Sistemas de Informação**, v. 10, n. 1, p. 1–8, 2011.

SOUZA, R. de; CARVALHO, F. D. Dynamic clustering of interval data based on adaptive chebyshev distances. **Electronics Letters**, v. 40, n. 11, p. 658–660, 2004.

SPEARMAN, C. The proof and measurement of association between two things. **International Journal of Epidemiology**, v. 39, n. 5, p. 1137–1150, 2010.

STEPHENSON, K.; ZELEN, M. Rethinking centrality: Methods and examples. **Social Networks**, v. 11, n. 1, p. 1–37, 1989.

TROYANSKAYA, O. et al. Missing value estimation methods for dna microarrays. **Bioinformatics**, v. 17, n. 6, p. 520–525, 2001.

WASSERMAN, S. **Social network analysis: Methods and applications**. Cambridge, UK: Cambridge University Press, 1994.

WATTS, D. J. **Small worlds: the dynamics of networks between order and randomness**. Princeton, USA: Princeton University Press, 1999.

WATTS, D. J.; STROGATZ, S. H. Collective dynamics of ”small-world” networks. **Nature**, v. 393, n. 6684, p. 440–442, 1998.

WEST, D. B. et al. **Introduction to Graph Theory**. Upper Saddle River, NJ: Prentice hall Upper Saddle River, 2001.

WILKINSON, D. M.; HUBERMAN, B. A. A method for finding communities of related genes. **Proceedings of The National Academy of Sciences**, v. 101, n. suppl 1, p. 5241–5248, 2004.

WU, F.-X. Genetic weighted k-means algorithm for clustering large-scale gene expression data. **BMC Bioinformatics**, v. 9, n. Suppl 6, p. S12, 2008.

WU, K.-L.; YANG, M.-S. Alternative c-means clustering algorithms. **Pattern recognition**, v. 35, n. 10, p. 2267–2278, 2002.

XIANG, R.; NEVILLE, J.; ROGATI, M. Modeling relationship strength in online social networks. In: ACM. **Proceedings of The 19th International Conference on World Wide Web**. [S.l.], 2010. p. 981–990.

ZADEH, L. A. Fuzzy sets. **Information and Control**, v. 8, n. 3, p. 338–353, 1965.

Appendices

A THE GRAPHML FILE FORMAT

The interest in Graph Theory, Complex Networks, and Social Network Analysis it is increasing arguably. It is also increasing the number of software's related to graphs and complex networks for several purposes. This diversity also leads to a wide variation in the way of representing graphs and complex networks.

Some of the possible representations are:

- Adjacency Matrix - Let $G = (V, E)$ be a graph with n vertices. The adjacency matrix for G it is a $n \times n$ bi-dimensional matrix, denoted by A, where $A(i, j) = 1$ if the edge (i, j) is present in G. And $A(i, j) = 0$ if the edge (i, j) is not present in G.

- Adjacency List - In this representation, the row i of the matrix contains the vertices adjacent to vertex i. Each vertex i has a variable $d[i]$ that keeps the degree of vertex i. Only non zero values are represented in the list, self loops are usually ignored, resulting in a more compact representation, which is desirable for complex networks with large amount of data.

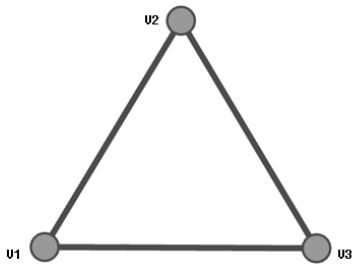
- Incident Matrix - Let $G = (V, E)$ be a graph with n vertices and m edges. The incidence matrix for G is a two-dimensional $n \times m$ matrix, denoted by A, where $A(i, j) = 1$ if j is focused in the node i in G.

Besides representing the vertices itself and the connections between vertices, complex networks could have various dimensions of data as attributes. For example, if we take a social network as a database possibly there will be data dimensions such as gender, age, geographical location, ethnicity, etc. As for a network that represents a food chain there may be data dimensions as phylum, order, class, etc. Such data may be categorical, textual, numeric, etc. In addition, there is still the possibility of assigning the vertices visual elements such as size, color, position, etc.

```
<?xml version="1.0" encoding="UTF-8"?><graphml xmlns="http://graphml.graphdrawing.org/xmlns">
        <key attr.name="label" attr.type="string" for="node" id="label"/>
        <key attr.name="Edge Label" attr.type="string" for="edge" id="edgelabel"/>
        <key attr.name="weight" attr.type="double" for="edge" id="weight"/>
        <key attr.name="Edge Id" attr.type="string" for="edge" id="edgeid"/>
        <graph edgedefault="undirected">
                <node id="51">
                        <data key="label">V1</data>
                </node>
                <node id="52">
                        <data key="label">V2</data>
                </node>
                <node id="53">
                        <data key="label">V3</data>
                </node>

                <edge source="51" target="52">
                        <data key="weight">1.0</data>
                </edge>
                <edge source="51" target="53">
                        <data key="weight">1.0</data>
                </edge>
                <edge source="52" target="53">
                        <data key="weight">1.0</data>
                </edge>
        </graph>
</graphml>
```

(a)



(b)

Figure 36: A GraphML source code and the respective graph.

To enable the construction of graphs with such features many file formats have been developed, such as *.csv* (comma-separated values), *.net* from the Pajek software [1], *.gml* a widely used graph exchange format (non-XML) suitable for Gephi software [2], *.vna* from the NetDraw software [3] among several others.

GraphML file format was used in this work to represent the complex networks presented in Case Study I, this file format is suitable for mostly the graph visualization softwares and graph manipulation software's. GraphML is an XML-based file format for graphs (BRANDES et al., 2002; BRANDES; PICH, 2005) with a language core to describe the structural properties of a graph and a flexible extension mechanism to add application-specific data. The GraphML [4] format main features include:

- Directed, Undirected, and Mixed Graphs.
- Hhypergraphs.
- Hierarchical Graphs.
- Graphical Representations.
- References to external Data.
- Application-specific Attribute Data.
- Light-weight Parsers.

[1] http://pajek.imfm.si
[2] https://gephi.org
[3] https://sites.google.com/site/netdrawsoftware
[4] http://graphml.graphdrawing.org/

A GraphML file consists of an XML syntax file with a graph, within a sequence of nodes and edges. Each node element should have a distinct id attribute, and each edge element has source and target attributes that identify the endpoints of an edge by having the same value as the id attributes of those endpoints. Figure 36a illustrates a source of a GraphML file and Figure 36b illustrates his respective graph.

All generated graphs for the experiments of the case study I presented in this work were modeled according to this pattern file. It is important to note that certain types of complex networks are composed of millions of vertices and edges and that the use of the structural elements of XML will result in a significant increase in file size, by consequence, increasing its complexity and requiring more time to process.

B GRAPH LAYOUT ALGORITHMS

Graph layout is an important field of graph and complex network studies. Graph drawing algorithms are algorithms which aim to present graphs in a more pleasant and comprehensible way. The complexity of this kind of algorithm depends in which kind of graph must be drawn. In this work we focus in 2D graphs, however, is possible to apply some of those algorithms to 3D structures such as trees and complex structures.

On Figure 37 is shown three layout examples, figure 37-A shows the random layout, 37-B shows a Fruchterman-Reingold layout which is based in force-directed layout and 37-C shows the same graph organized under the Yifan-Hu graph layout which is also based in force-directed layout.

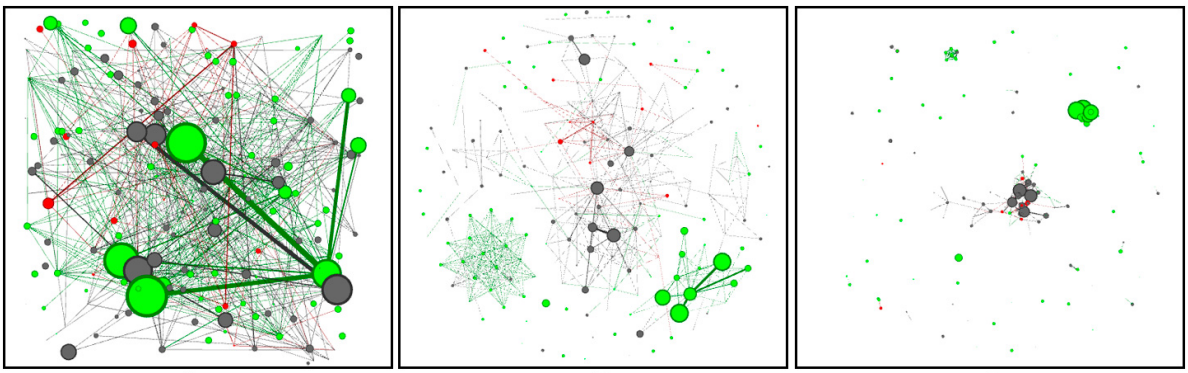


Figure 37: Graphs organized under the force-direct layout algorithms

The purpose of the graph layout algorithms is positioning the vertices of a graph into a space in which vertices and edge are a representation of its values equally (FRUCHTERMAN; REINGOLD, 1991). Some layout algorithms are able to depict the proximity of vertex and even the agglutination of communities, which is the focus of this dissertation. For these reasons, we focus in force directed layouts, which produces most accurate results for 2D undirected graphs.

The concepts adopted by Frutchermand and Reingold are based in nuclear bonds in analogy to electric binds between protons.

These algorithms consider strongly connected elements tend to be close, acting with a force of attraction. Those vertices who have weak connections, or have no connections must be apart, acting with a force of repulsion. This concepts lead to two very simple principles (FRUCHTERMAN; REINGOLD, 1991).

1. Vertices connected by an edge should be drawn near each other.

2. Vertices should not be drawn too close to each other.

The pseudo-code shown in algorithm 8 describes the basic operation of a graph layout force-directed algorithm.

Algorithm 8: Pseudo-code for a force-directed algorithm.

```
Input: Graph G;
Output: Straight-line drawing of G;
Initialize Positions: place vertices of G in random locations;
for i = 1toM do
    calculate the force acting in each vertex;
    move the vertex * (force in vertex);
end
draw a filled circle for each vertex;
draw a straight-line segment for each edge;
```

The optimal distance between vertices k defined as

$$k = C\sqrt{\frac{a}{n}} \tag{22}$$

Where k is the distance between two vertices, a is the area and n is the number of vertices.

The attractive forces function is defined as $f_a(d) = d^2/k$, and the repulsive forces function is defined as $f_r(d) = -k^2/d,$. The algorithm 9 (KOBOUROV; WAMPLER, 2005) shows an implementation of Frutchermand & Reingold algorithm.

Algorithm 9: Frutchermand & Reingold algorithm

area $\leftarrow W * L$; // frame: width W and length L
initialize G = (V, E) ; // place vertices at random
K $\leftarrow \sqrt{area/|V|}$; // compute optimal pairwise distance
function $f_r(x) = k^2/x$; // compute repulsive force
for i = *1 to iterations* **do**
  for ***each*** $v \in V$ **do**
    $v.disp := 0$; // initialize displacement vector
    for $u \in V$ **do**
      if $u \neq v$ **then**
        $\triangle \leftarrow v.pos - u.pos$; // distance between u and v
        $v.disp \leftarrow v.disp + (\triangle/|\triangle|) * f_r(|\triangle|)$; // displacement
      end
  end
  function $f_a(x) = x^2/K$; // compute attractive force
  for ***each*** $e \in E$ **do**
    $\triangle \leftarrow e.v.pos - e.u.pos$; // e is ordered vertex pair .v and .u
    $e.v.disp \leftarrow e.v.disp - (\triangle/|\triangle|) * f_a(|\triangle|)$;
    $e.u.disp \leftarrow e.u.disp - (\triangle/|\triangle|) * f_a(|\triangle|)$;
  end
  for ***each*** $v \in V$; // limit max displacement to frame; use temp. t to scale
  do
    $v.pos \leftarrow v.pos + (v.disp/|v.disp|) * min(v.disp, t)$;;
    $v.pos.x \leftarrow min(W/2, max(-W/2, v.pos.x))$;;
    $v.pos.y \leftarrow min(L/2, max(-L/2, v.pos.y))$;
  end
  $t \leftarrow cool(t)$; // reduce temperature for next iteration
end

The drawbacks of Frutchermand-Reingold algorithm are the computational cost; either sometimes the algorithm does not converge, resulting in an unstable graph.

A desired characteristic of graph layout algorithms is the capacity of organize the graph minimizing edge crossing. This allows a clean and intuitive visualization.

Another force-directed layout algorithm implemented in this work is the Yifan-Hu Multilevel layout. The algorithm combines high performance with multilevel coarsening to reduce the complexity. One of the goals for the Yifan-Hu algorithm is visualizing cluster relationships as maps, offering an intuitive way to represent communities and denote the relationships between them. It also desired to maintain readability and aesthetics.

A difference between the Frutchermand-Reingold and Yifan-Hu is that, in Yifan-Hu, each vertex is updated as soon the forces of the vertex is calculated, instead wait for the whole system to be updated.

The pseudocode for the iterative force-directed layout algorithm is shown in algorithm 10 (HU, 2005).

Algorithm 10: Iterative force-directed layout algorithm

function MultilevelLayout(G, x, tol);

- Coarsest graph layout;

  if $(n^{i+1} < MinSizeorn^{i+1}/n^i > \rho)$ **then**

    x^i = random initial layout;

    x^i = ForceDirectedAlgorithm(G^i, x^i, tol);

    return x^i ;

  end

- The coarsening phase:;

  setup the $n^i \times n^{i+1}$ prolongation matrix P^i ;

  $G^i + 1 = P^{iT} G^i P^i$;

  $x^{i+1} = MultilevelLayout(G^{i+1}, tol)$;

- The prolongation and refinement phase:;

  Prolongate to get initial Layout: $x^i = P^i x^{i+1}$;

  Refinement: $x^i = ForceDirectAlgorithm(G^i, x^i, tol)$;

  Return x^i;

In the Algorithm 10 $n^i = |V^i|$ is the number of vertices in the graph, x^i is the coordinate vector for the vertices in V^i. Prolongation operation for G_{i+1} to G_i is also represented by a matrix P^i, of dimension $n^i \times n^{i+1}$. *MinSize* is defined as 2 and ρ is defined as 0.75.

C CORRELATION MEASURES

Correlation measures are an effective way to evaluate the relationship between two sets of values. Their similarity or dissimilarity. Used in various areas of knowledge such as psychology, sociology and the statistical correlation measures aimed at providing an assessment of proximity or similarity between two sets of values.

Correlation analysis between two or more variables provides a number that summarizes the degree of linear relationship between variables.

In this section, three measures of correlation often used to assess the relationship between variables are outlined.

C.1 SPEARMAN RANK CORRELATION

Spearman Rank Correlation is a non-parametric test that is used to measure the degree of association between two variables (SPEARMAN, 2010). Often denoted by the Greek letter ρ (rho) or as ρ. This is a nonparametric method that uses only the ranks, and not make any assumptions. Essentially all it does is calculate the Pearson correlation coefficient of the ranks. Rank is the position in which a number within a list occupies when the data is sorted in ascending order.

The Spearman correlation coefficient is less sensitive than Pearson's correlation coefficient to very far from expected values, being more suitable for application to datasets which present outliers (IMAN; CONOVER, 1982).

The Spearman ρ coefficient varies between -1 and 1. As much near are the correlation from the extremes the greater is the association between the variables. Negative correlation values means that the correlation variables vary in the opposite direction.

A formula for calculating the Spearman coefficient ρ is presented in Equation 23,

$$\rho = 1 - \frac{6 \sum_{i=1}^{n} d_i^2}{n^3 - n}, \tag{23}$$

where n is the number of pairs (x_i, y_i), and $d_i = (x_i \text{ posts from the values of } x) - (y_i \text{ posts from the } y \text{ values})$ (CHEN; POPOVICH, 2002).

Figure C.1 shows three data samples for which the Spearman correlation was calculated.

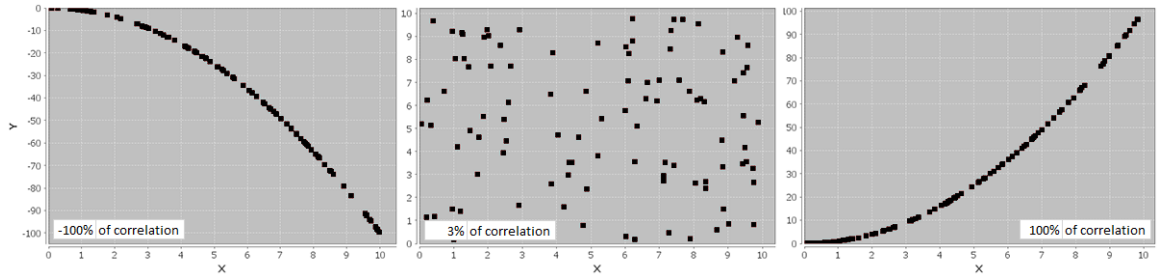


Figure 38: On the left, a dataset with negative Spearman correlation, at center a dataset with low Spearman correlation and on the right a dataset with high Spearman correlation.

C.2 KENDALL RANK CORRELATION

Kendall Rank Correlation Coefficient is a non-parametric test that measures the strength of dependence between two variables developed by Maurice Kendall in 1938 (KENDALL, 1948, 1938).

> Kendall coefficient of correlation is obtained by normalizing the symmetric difference such that it will take values between -1 and +1 with -1 corresponding to the largest possible distance (obtained when one order is the exact reverse of the other order) and +1 corresponding to the smallest possible distance (equal to 0, obtained when both orders are identical) (ABDI, 2007).

The Figure 39 shows four examples of Kendall's Rank Correlation Coefficient for a series of values.

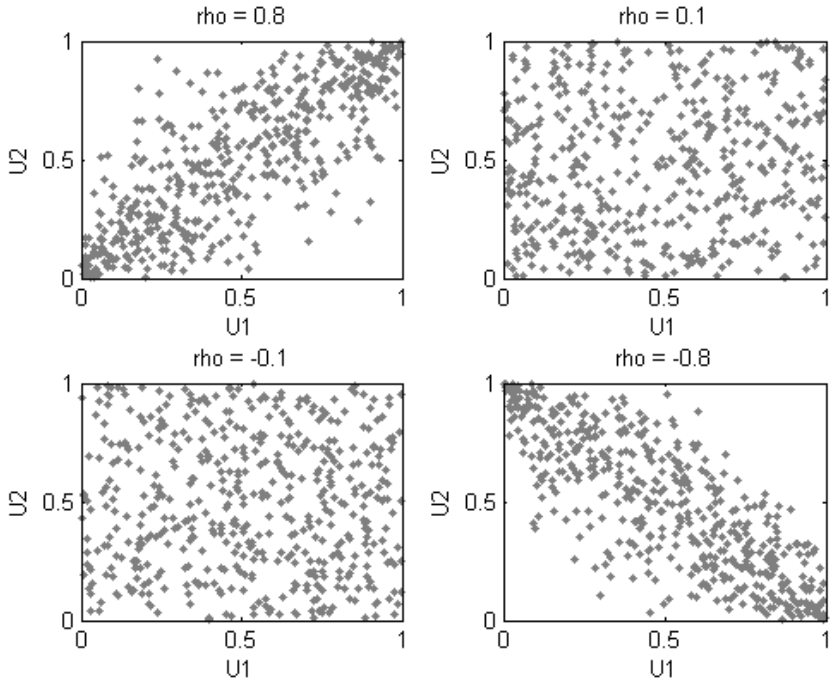


Figure 39: Kendall's Correlation examples for a series of values (MATLAB, 2010).

Kendall Rank Correlation Coefficient between two random variables with n observations is defined as shown in Equation 24

$$\tau = \frac{nc - nd}{\frac{1}{2}n(n-1)} \tag{24}$$

Where n is the number of observations, nc is the number of concordant pairs and nd is the number of discordant pairs.

- If the agreement between the two rankings is perfect the coefficient has value 1.
- If the disagreement between the two rankings is perfect the coefficient has value -1.
- If X and Y are independent, then is expect that the coefficient is approximately zero.

C.3 PEARSON'S CORRELATION COEFFICIENT

Pearson's Correlation Coefficient is widely used in statistics to measure the degree of the relationship between linear related variables. Given a series of values for X and Y Pearson's Correlation Coefficient will measure the degree of strength of the linear relationship between these two variables. The resultant value will range between -1 and 1 (PEARSON, 1895).

> If coefficient values are 1 or -1, there will be perfect linear relationship between the variables. Positive sign with coefficient value shows positive (direct, or supportive), while negative sign with coefficient value show negative (indirect, opposite) relationship between the variables. The Zero value implies the absence of a linear relation and it shows that variables are independent.

Figure 40 shows three examples of Pearson's Correlation Coefficient with values ranging between 4 and -4.

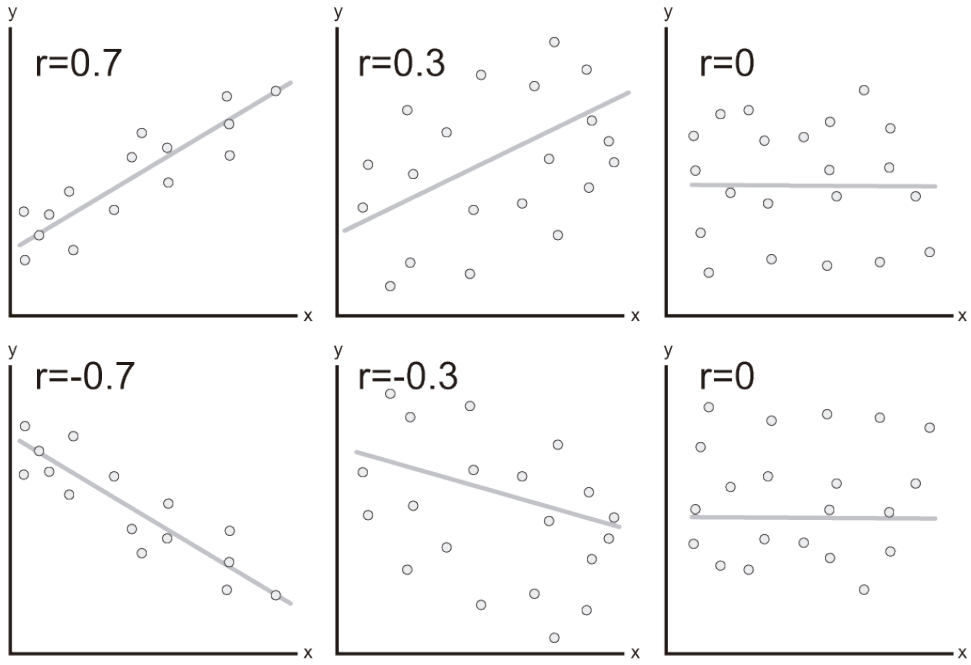


Figure 40: Examples of Pearson's Correlation Coefficient (PEARSON, 1895).

One possible approach to represent gene expression data through graphs it is to use the value of the Pearson's correlation between two vertices as a weight for the edge connecting them.

In this approach must take into consideration that graphs cannot receive negative values as weights for the edges and the values generated by Pearson's correlation vary between -1 and 1.

D TOP-100 GENES RELATED TO ALZHEIMER'S DISEASE

Table 14: Top-100 Genes Related to each stage of Alzheimer's Disease.

GENE	I-AD	M-AD	S-AD	GENE	I-AD	M-AD	S-AD	GENE	I-AD	M-AD	S-AD	GENE	I-AD	M-AD	S-AD	GENE	I-AD	M-AD	S-AD
APBB2	YES	YES	YES	RIOK3	YES	YES		DR1	YES			MAPK11	YES			RIN3			YES
CFLAR	YES	YES	YES	SMARCA4		YES	YES	DST		YES		MARCKS			YES	RPL13		YES	
FGFR2	YES	YES	YES	SPTBN1		YES	YES	EGFR	YES			MAU2	YES			RPS6			YES
GNA11	YES	YES	YES	SQSTM1	YES	YES		EIF4B			YES	MAX		YES		RTEL1		YES	
GNAS	YES	YES	YES	TRD	YES		YES	EPHB2			YES	MTAP		YES		RUNX1T1			YES
HFE	YES	YES	YES	AAK1	YES			EPM2A	YES			MUC4			YES	SCAPER	YES		
PML	YES	YES	YES	ABHD2		YES		ERAP1			YES	MYO7A		YES		SEPT9	YES		
PTGER3	YES	YES	YES	ABHD6	YES			EZR			YES	MYOZ2	YES			SERP1		YES	
PVR	YES	YES	YES	ACRV1	YES			FGFR1		YES		NAP1L1		YES		SERPINB13	YES		
RUNX1	YES	YES	YES	AGRN			YES	FKSG49	YES			NCAPH2	YES			SFTPB			YES
SLC6A2	YES	YES	YES	AK026847	YES			FZR1			YES	NCOR1		YES		SH2D1A	YES		
TCF3	YES	YES	YES	ALDOB		YES		GEMIN2		YES		NR2F6	YES			SLC24A1			YES
AK2		YES	YES	AMBRA1		YES		GGA2			YES	OASL	YES			SLC7A8	YES		
AKAP13	YES		YES	ANAPC5		YES		GGT1			YES	OGFR	YES			SMURF1	YES		
APLP2		YES	YES	ARPP19			YES	GH2	YES			PAIP1		YES		SNTB2	YES		
CALD1		YES	YES	ASB4	YES			GJB3			YES	PAK2		YES		SNW1	YES		
COL6A1		YES	YES	ASMT	YES			GOLGB1			YES	PAPOLA	YES			SOGA1			YES
CSH1	YES		YES	ATF7IP	YES			GORASP2		YES		PAX8		YES		SON		YES	
DCBLD2	YES		YES	ATP2A3	YES			GPLD1			YES	PCSK6	YES			SOX4		YES	
DICER1		YES	YES	ATP5C1		YES		GRIN1		YES		PDGFB	YES			SPDEF	YES		
EDA		YES	YES	ATP6V0E1		YES		GTF3C2		YES		PDX1	YES			SRCAP	YES		
EIF1	YES	YES		BAIAP2	YES			GUSBP11	YES			PGF	YES			SRRT			YES
EIF5B		YES	YES	BAZ2A	YES			HAB1			YES	PHF3		YES		STS		YES	
ESR1		YES	YES	BMP7	YES			HLADRB4		YES		PKP4		YES		SYNCRIP		YES	
ESR2	YES	YES		BUB1			YES	HNRNPA3		YES		PMS2P1		YES		SYNJ2		YES	
ETV1	YES		YES	C21orf2			YES	HNRNPDL			YES	POLR2A			YES	TACC1		YES	
EXOC7		YES	YES	CA12			YES	HNRNPH1		YES		PPP2R1B		YES		TAF6L	YES		
GH1	YES		YES	CALM1		YES		HRK			YES	PPP2R4			YES	TARP		YES	
GIMAP5	YES	YES		CASP2	YES			IDS		YES		PRDX2	YES			TCF4		YES	
GLP1R	YES		YES	CC2D1A	YES			IL13RA1		YES		PRKCSH			YES	TCF7L2		YES	
IGHV323		YES	YES	CDC14B			YES	IL6ST		YES		PRLR	YES			TELO2			YES
IGLV140	YES		YES	CDC27	YES			ILF3			YES	PRRC2C			YES	TFPI			YES
L3MBTL1	YES		YES	CDH6	YES			IQCK		YES		PSG1	YES			TIA1		YES	
LAMA4	YES	YES		CDKN1C		YES		ISG20L2		YES		PTCRA			YES	TRIM14	YES		
LARP4B	YES	YES		CES2			YES	ITGB1		YES		PTGER1			YES	TRIO	YES		
LDLR	YES	YES		CGGBP1	YES			ITGB3			YES	PTP4A2			YES	TTC38	YES		
LOC100133862	YES	YES		COBLL1	YES			JAK3	YES			PTPN1		YES		TTLL5			YES
LOC727787	YES	YES		COPS6		YES		KAT6B		YES		QKI		YES		USP34			YES
LZTS1	YES		YES	CSH2	YES			KCNQ1	YES			RAB27A			YES	VIPR2	YES		
MBD4		YES	YES	CSHL1	YES			KDM4B			YES	RAB2A		YES		WNK1			YES
NF1		YES	YES	CSNK1A1		YES		KIF2C	YES			RAB3GAP2	YES			WT1			YES
NKTR	YES		YES	CTSB			YES	KLHL22	YES			RABEP2			YES	ZC3H7B			YES
NTRK3	YES	YES		CYP2A6		YES		KMT2A			YES	RARB			YES	ZFAND5			YES
PCNXL2	YES	YES		CYP2C9	YES			LPAR1			YES	RBM25			YES	ZNF500			YES
PDLIM5	YES	YES		DBT			YES	LRRFIP1			YES	RGS12			YES	DOCK6	YES		
PIDD		YES	YES	DGCR14			YES	LSM14A	YES			RGS6	YES			MAP4			YES
PTMA		YES	YES	DOCK10			YES	MAN1A2		YES		RHEB		YES		RHOBTB3	YES		
RAB7A	YES	YES		(I-AD = Incipient Alzheimer's Disease, M-AD = Moderate Alzheimer's Disease, S-AD = Severe Alzheimer's Disease)															

Table 15: Communities identified from the interaction between the top-100 most connected genes for each stage of Alzheimer's disease.

COMUNITY A		COMUNITY B		COMUNITY C		COMUNITY D	
CALD1	CFLAR	EXOC7	WT1	AAK1	FGFR2	ATP2A3	PSG1
DST	COL6A1	KMT2A	LPAR1	AK2	HNRNPA3	CDKN1C	ABHD6
IQCK	CTSB	L3MBTL1	MYO7A	BUB1	MBD4	CES2	AKAP13
KDM4B	DBT	PHF3	SYNJ2	EIF1	PAK2	CSH1	BMP7
PAIP1	GORASP2	SYNCRIP	BAIAP2	HFE	PDX1	CSNK1A1	DOCK6
RAB2A	IL6ST	ESR1	PKP4	MAP4	RARB	CYP2C9	FGFR1
SMARCA4	ISG20L2	EZR	PTMA	NAP1L1	EDA	EPHB2	GH1
SPTBN1	MAN1A2	SPDEF	ITGB3	PAPOLA	TRIO	EPM2A	GLP1R
TAF6L	MUC4	WNK1	GRIN1	SMURF1	KCNQ1	ESR2	LRRFIP1
TCF7L2	PDLIM5	CA12	RGS12	SON	LARP4B	GIMAP5	RIOK3
ZC3H7B	PML	PTGER3	IDS	TCF4	NKTR	GNAS	ATF7IP
APBB2	PRRC2C	SLC6A2	KAT6B	TFPI	MAX	JAK3	SNTB2
CALM1	PTGER1	CDH6	PTP4A2	ABHD2	ARPP19	KIF2C	SQSTM1
CASP2	RPL13	NTRK3	GPLD1	ATP5C1	COBLL1	LAMA4	SOGA1
DICER1	SFTPB	RIN3	ASB4	EIF4B	HNRNPDL	LDLR	ANAPC5
GEMIN2	TIA1	RPS6	-	ITGB1	HNRNPH1	NF1	DCBLD2
ILF3	NCOR1	-	-	PAX8	RUNX1T1	PCSK6	PDGFB
QKI	MTAP	-	-	PTPN1	GTF3C2	RABEP2	PRDX2
RBM25	CDC14B	-	-	RHEB	POLR2A	RHOBTB3	AMBRA1
RUNX1	ERAP1	-	-	TCF3	RAB27A	SLC7A8	NR2F6
SOX4	GNA11	-	-	DR1	RAB3GAP2	SNW1	RAB7A
STS	PPP2R1B	-	-	EGFR	SEPT9	TRIM14	CGGBP1
AGRN	FZR1	-	-	MARCKS	SLC24A1	TTC38	PRKCSH
APLP2	IL13RA1	-	-	CYP2A6	-	ALDOB	BAZ2A
ATP6V0E1	ZFAND5	-	-	-	-	CSHL1	PGF
C21orf2	COPS6	-	-	-	-	GGT1	HRK
CDC27	DOCK10	-	-	-	-	MAPK11	VIPR2
TACC1	SERP1	-	-	-	-	PRLR	-

Table 16: Top-100 most connected genes in complex network from intersection of Incipient AD, Moderate AD and Severe AD which are not part of the Control Group.

Gene	Match	Gene	Match	Gene	Match	Gene	Match
TCF3	YES	TCF4	YES	RNGTT	-	HNRNPDL	YES
TRD	-	KDM4B	YES	SPTBN1	YES	HOXA11	YES
HFE	YES	RABEP2	YES	UBE2C	YES	IGHV323	-
CFLAR	YES	WNK1	YES	CSNK1A1	YES	N4BP2L1	YES
AKAP13	YES	ZNF160	-	DICER1	YES	PLEKHG3	YES
FGFR2	YES	ACRV1	-	EIF1	YES	PRRC2C	YES
RUNX1	YES	CSH1	YES	ETV1	-	RAB7A	YES
IGLV140	-	CYP2C9	YES	FGFR1	YES	SMARCD1	YES
LZTS1	-	GGA2	-	HMGXB4	-	TFPI	YES
PML	YES	OGFR	-	LOC727787	-	TPSAB1	YES
LRRFIP1	YES	SMARCA4	YES	RIN3	YES	ZNF500	-
SRRT	-	TNXB	YES	SERPINB13	-	AAK1	YES
GNA11	YES	EEF1D	YES	TNPO1	YES	ACVR1B	-
APBB2	YES	GUSBP11	-	UBE2D4	-	AGRN	YES
GNAS	YES	LDLR	YES	216414_at	-	AK026847	-
EDA	YES	PTGER3	YES	220905_at	-	ATF7IP	YES
LOC100133862	-	SH2D1A	-	BUB1	YES	CASP2	YES
COL6A1	YES	SLC6A2	YES	CC2D1A	-	CGGBP1	YES
GH1	YES	WNT6	YES	CDKN1C	YES	CSH2	-
L3MBTL1	YES	ALDOB	YES	ESR2	YES	DGCR14	-
LAMA4	YES	DBT	YES	GAS7	YES	DNASE2	YES
RIOK3	YES	ERAP1	YES	GLYR1	YES	EHD2	YES
TRIO	YES	MAP4	YES	HNRNPA3	YES	PVR	-
PRKAR2A	YES	NTRK3	YES	HNRNPC	YES	RAB2A	YES
ESR1	YES	HNRNPL	YES	ITGB3	YES	GJB3	-

A gene is labeled with ‘YES’ value in the match column if that particular gene is listed as related to Alzheimer disease in The Comparative Toxicogenomics Database (CTD) (DAVIS et al., 2013).